

9.2

*IBM MQ Geliştirme Uygulamaları  
Başvurusu*

**IBM**

**Not**

Bu bilgileri ve desteklediđi ürünü kullanmadan önce, [“Özel notlar” sayfa 2165](#) bölümündeki bilgileri okuyun.

This edition applies to version 9 release 2 of IBM® MQ and to all subsequent releases and modifications until otherwise indicated in new editions.

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

© Copyright International Business Machines Corporation 2007, 2024.

# İçindekiler

<b>Uygulama başvurusu geliştirilmesi.....</b>	<b>7</b>
MQI uygulamaları başvurusu.....	7
Kod örnekleri.....	8
Değişmezler.....	61
MQI ' da kullanılan veri tipleri.....	235
İşlev çağrıları.....	615
Nesnelerin öznitelikleri.....	785
Dönüş kodları.....	859
MQI seçeneklerinin geçerliliğini denetleme kuralları.....	860
Kuyruğa alınan yayınlama/abone olma komutu iletileri.....	863
Makine kodlamaları.....	884
Rapor seçenekleri ve ileti işaretleri.....	887
Veri dönüştürme çıkışı.....	891
MQRFH2 öğeleri olarak belirtilen özellikler.....	914
Kod sayfası dönüşümü.....	922
64 bit altyapılarda kodlama standartları.....	976
IBM i Application Programming Reference (ILE/RPG).....	980
IBM üzerindeki veri tipi açıklamaları.....	981
Function calls on IBM i.....	1230
IBM üzerindeki nesnelerin öznitelikleri.....	1345
Uygulamalar.....	1390
IBM i için dönüş kodları (ILE RPG).....	1403
IBM i için MQI seçeneklerinin geçerliliğini denetleme kuralları (ILE RPG).....	1404
Machine encodings on IBM i.....	1407
IBM i ' ta rapor seçenekleri ve ileti işaretleri.....	1410
IBM üzerinde veri dönüştürme.....	1413
IBM üzerinde dönüştürme işlemi.....	1414
IBM üzerindeki işleme kuralları.....	1415
Conversion of report messages on IBM i.....	1418
MQDXP (Data-conversion exit parameter) on IBM i.....	1419
MQXCNCV (Convert characters) on IBM i.....	1424
IBM üzerinde MQCONVX (Veri dönüştürme çıkışı).....	1429
Kullanıcı çıkışları, API çıkışları ve kurulabilir hizmetler başvurusu.....	1432
MQIEP yapısı.....	1433
Veri dönüştürme çıkış başvurusu.....	1436
MQ_PUBLISH_EXIT-Yayınlama çıkışı.....	1440
Kanal-çıkış çağrıları ve veri yapıları.....	1448
Küme iş yükü çıkış çağrısı ve veri yapıları.....	1511
API çıkış başvurusu.....	1535
Kurulabilir hizmetler arabirimi başvuru bilgileri.....	1595
Installable services interface reference information on IBM i.....	1658
IBM MQ .NET sınıfları ve arabirimleri.....	1697
MQAsyncStatus.NET sınıfı.....	1697
MQAuthenticationInformationRecord.NET sınıfı.....	1698
MQDestination.NET sınıfı.....	1699
MQEnvironment.NET sınıfı.....	1702
MQException.NET sınıfı.....	1704
MQGetMessageOptions.NET sınıfı.....	1705
MQManagedObject.NET sınıfı.....	1708
MQMessage.NET sınıfı.....	1710
MQProcess.NET sınıfı.....	1722
MQPropertyDescriptor.NET sınıfı.....	1724

MQPutMessageOptions.NET sınıfı.....	1726
MQQueue.NET sınıfı.....	1728
MQQueueManager.NET sınıfı.....	1736
MQSubscription.NET sınıfı.....	1748
MQTopic.NET sınıfı.....	1749
IMQObjectTrigger.NET arabirimi.....	1755
MQC.NET arabirimi.....	1756
.NET uygulamaları için karakter kümesi tanıtıcıları.....	1756
IBM MQ C++ sınıfları.....	1758
C++ ve MQI çapraz başvurusu.....	1760
ImqAuthenticationKayıt C++ sınıfı.....	1775
ImqBinary C++ sınıfı.....	1777
ImqCache C++ sınıfı.....	1779
ImqChannel C++ sınıfı.....	1782
ImqCICSBridgeHeader C++ sınıfı.....	1787
ImqDeadLetterHeader C++ sınıfı.....	1793
ImqDistributionListe C++ sınıfı.....	1795
ImqError C++ sınıfı.....	1797
ImqGetMessageOptions C++ sınıfı.....	1798
ImqHeader C++ sınıfı.....	1801
ImqIMSBridgeHeader C++ sınıfı.....	1803
ImqItem C++ sınıfı.....	1806
ImqMessage C++ sınıfı.....	1807
ImqMessageTracker C++ sınıfı.....	1814
ImqNamelist C++ sınıfı.....	1816
ImqObject C++ sınıfı.....	1818
ImqProcess C++ sınıfı.....	1823
ImqPutMessageOptions C++ sınıfı.....	1825
ImqQueue C++ sınıfı.....	1827
ImqQueueManager C++ sınıfı.....	1837
ImqReferenceHeader C++ sınıfı.....	1853
ImqString C++ sınıfı.....	1856
ImqTrigger C++ sınıfı.....	1861
ImqWorkÜstbilgi C++ sınıfı.....	1864
IBM MQ classes for JMS nesnelerinin özellikleri.....	1866
IBM MQ classes for JMS nesnelere ilişkin özellikler arasındaki bağımlılıklar.....	1869
UYGULAMA ADI.....	1871
ASYNCEXCEPTION.....	1871
BROKERCCDURSUBQ.....	1872
BROKERCCSUQ.....	1873
BROKERCONQ.....	1873
BROKERDURSUQ.....	1874
BROKERPUBQ.....	1874
BROKERPUBQMGR.....	1874
BROKERQMGR.....	1875
BROKERSUBQ.....	1875
BROKERVER.....	1876
CCDTURL.....	1876
CCSID.....	1877
Kanal.....	1877
Temizle.....	1878
CLEANUPINT.....	1878
CONNECTIONNAMELIST.....	1879
İSTEMCILERİYÜZLER.....	1879
CLIENTRECONNECTTIMEOUT.....	1880
CLIENTID.....	1881
ÇALIŞTIRMA.....	1881
KARMAŞIK.....	1882



MSG.....	1882
HAYIR.....	1882
CONNTAG.....	1883
AÇIKLAMA.....	1884
YÖN YÖNTEMÇİ.....	1884
Kodlama.....	1885
Son kullanma tarihi.....	1886
FAILIFQUIESCE.....	1886
Anasistem adı.....	1887
YEREL ADRES.....	1887
MAPNAMESTILI.....	1888
MAXBUFFSIZE.....	1889
MDREAD.....	1889
MDWRITE.....	1890
MDMSGCTX.....	1890
MSGBATCHSZ.....	1891
MSGONT.....	1891
MSGRECALOG.....	1892
MSGSEÇEK.....	1892
ÇOK YA.....	1893
EN İYİ BILDIRIM.....	1893
KALDIRIM.....	1894
Kalıcılık.....	1894
GİRİŞ TARİHİ.....	1895
PORT.....	1896
Öncelik.....	1896
SÜREÇ.....	1897
SAĞLAMA Sü.....	1897
PROXYHOSTNAME.....	1899
TANI TİM.....	1900
PUBKACT.....	1900
PUTASINCALLI.....	1901
YÖNETİM.....	1901
kuyruk.....	1902
OKUNABİLENİZİN.....	1902
READAHEADCLOSEPOLICY.....	1903
RECEIVECCSTnt.....	1903
ALINMASI.....	1904
RECEIVEI.....	1904
YENİDEN ÇIKIŞ.....	1905
YENİDEN KALDIRMA.....	1905
REPLYTOSTYLE.....	1906
SIFIRA.....	1906
SEÇİLEN.....	1907
GÜNCELLEMİYİ.....	1907
SENDCHECKCOUNT.....	1908
SENDEXIT.....	1908
SENDEXITINIT.....	1909
FARKLILIKLI.....	1909
SPARSESUBS.....	1910
SSLCIPHERSUIT.....	1911
SLSCRL.....	1911
SSLFIPSREQUIRED.....	1911
SLAYICI ADI.....	1912
SSLRESTCOUNT.....	1912
DEYİM YENİDEN.....	1913
ALT DÜZEN.....	1913
SYNCPOINTALLGETS.....	1914

HEDEF İSTEMCI.....	1914
TARGCLIENTEŞLEŞTİRME.....	1915
TEMPMODEL.....	1915
TEMPQOPREFIX.....	1916
TEMPTOPICFIX.....	1916
Konu.....	1917
Aktarım.....	1917
JOKER BIÇIMI.....	1918
ENCODING özelliği.....	1918
TLS properties of JMS objects.....	1919
IBM MQ Message Service Client (XMS) for .NET başvuru.....	1920
.NET Arabirimler.....	1920
XMS nesnelerinin özellikleri.....	2001
Managed File Transfer uygulama başvurularını geliştirme.....	2066
fteCreateAktarımı programını başlatmak için kullanma örnekleri.....	2066
<b>fteAnt</b> : run Ant tasks in MFT.....	2068
Özelleştirme başvurusu içinMFT kullanıcı çıkışları.....	2093
MFT Agent komut kuyruğuna yerleştirebileceğiniz iletiler için ileti biçimleri.....	2133
Messaging REST API başvurusu.....	2133
REST API Kaynak.....	2133
<b>Özel notlar.....</b>	<b>2165</b>
Programlama arabirimi bilgileri.....	2166
Ticari Markalar.....	2166

# Uygulama başvurusu geliştirilmesi

---

IBM MQ uygulamalarınızı geliştirmenize yardımcı olması için bu bölümde sağlanan bağlantıları kullanın.

- [“MQI uygulamaları başvurusu” sayfa 7](#)
- [IBM i “IBM i Application Programming Reference \(ILE/RPG\)” sayfa 980](#)
- [IBM i “IBM üzerinde veri dönüştürme” sayfa 1413](#)
- [“Kullanıcı çıkışları, API çıkışları ve kurulabilir hizmetler başvurusu” sayfa 1432](#)
- [“IBM MQ .NET sınıfları ve arabirimleri” sayfa 1697](#)
- [“IBM MQ C++ sınıfları” sayfa 1758](#)
- [“IBM MQ classes for JMS nesnelerinin özellikleri” sayfa 1866](#)
- [“Messaging REST API başvurusu” sayfa 2133](#)

## İlgili görevler

[Uygulamaların geliştirilmesi](#)

## İlgili başvurular

[Java kitaplıkları için IBM MQ sınıfları](#)

[JMS için IBM MQ sınıfları](#)

# MQI uygulamaları başvurusu

---

Bu kısımda, Message Queue Interface (MQI) uygulamalarınızı geliştirmenize yardımcı olması için sağlanan bağlantıları kullanın.

- [“Kod örnekleri” sayfa 8](#)
- [“Değişmezler” sayfa 61](#)
- [“MQI ' da kullanılan veri tipleri” sayfa 235](#)
- [“İşlev çağrıları” sayfa 615](#)
- [“Nesnelerin öznitelikleri” sayfa 785](#)
- [“Dönüş kodları” sayfa 859](#)
- [“MQI seçeneklerinin geçerliliğini denetleme kuralları” sayfa 860](#)
- [“Makine kodlamaları” sayfa 884](#)
- [“Rapor seçenekleri ve ileti işaretleri” sayfa 887](#)
- [“Veri dönüştürme çıkışı” sayfa 891](#)
- [“MQRFH2 öğeleri olarak belirtilen özellikler” sayfa 914](#)
- [“Kod sayfası dönüşümü” sayfa 922](#)

## İlgili kavramlar

[“Kullanıcı çıkışları, API çıkışları ve kurulabilir hizmetler başvurusu” sayfa 1432](#)

Kullanıcı çıkışlarınızı, API çıkışlarınızı ve kurulabilir hizmet uygulamalarınızı geliştirmenize yardımcı olması için bu bölümdeki bilgileri kullanın:

## İlgili görevler

[Uygulamaların geliştirilmesi](#)

## İlgili başvurular

[“IBM MQ .NET sınıfları ve arabirimleri” sayfa 1697](#)

IBM MQ .NET sınıfları ve arabirimleri alfabetik olarak sıralanır. Özellikler, yöntemler ve oluşturucular açıklanır.

“IBM MQ C++ sınıfları” sayfa 1758

IBM MQ C++ sınıfları, IBM MQ Message Queue Interface (MQI) 'ı sarsalıyor. Bu sınıfların tümünü kapsayan tek bir C++ üstbilgi dosyası ( **imqi.hpp**) vardır.

[Java kitaplıkları için IBM MQ sınıfları](#)

[JMS için IBM MQ Sınıfları](#)

## Kod örnekleri

İş gereksinimlerinizi karşılayacak görevleri gerçekleştirmek için bu kısımdaki başvuru bilgilerini kullanın.

### C dili örnekleri

Bu konu derlemi çoğunlukla IBM MQ for z/OS örnek uygulamalarından alınır. Bunlar, belirtilenler dışında, tüm platformlar için geçerli olur.

#### ***Kuyruk yöneticisiyle bağlantı kuruluyor***

Bu örnek, bir programı z/OS toplu kipte bir kuyruk yöneticisine bağlamak için MQCONN çağrısının nasıl kullanılacağını gösterir.

This extract is taken from the Browse sample application (program CSQ4BCA1) supplied with IBM MQ for z/OS. Diğer platformlardaki örnek uygulamaların adları ve yerleri için bkz. [Örnek yordamsal programlar \(z/OS dışındaki platformlar\)](#).

```
#include <cmqc.h>
...
static char Parm1[MQ_Q_MGR_NAME_LENGTH] ;

int main(int argc, char *argv[] )
{
    /*
    /*      Variables for MQ calls
    /*
    MQHCONN Hconn;      /* Connection handle
    MQLONG  CompCode;   /* Completion code
    MQLONG  Reason;     /* Qualifying reason

    /* Copy the queue manager name, passed in the
    /* parm field, to Parm1
    strncpy(Parm1,argv[1],MQ_Q_MGR_NAME_LENGTH);

    /*
    /* Connect to the specified queue manager.
    /* Test the output of the connect call. If the
    /* call fails, print an error message showing the
    /* completion code and reason code, then leave the
    /* program.
    /*
    MQCONN(Parm1,
           &Hconn,
           &CompCode,
           &Reason);
    if ((CompCode != MQCC_OK) | (Reason != MQRC_NONE))
    {
        sprintf(pBuff, MESSAGE_4_E,
                ERROR_IN_MQCONN, CompCode, Reason);
        PrintLine(pBuff);
        RetCode = CSQ4_ERROR;
        goto AbnormalExit2;
    }
    ...
}
```

#### ***Kuyruk yöneticisinden bağlantı kesiliyor***

This example demonstrates how to use the MQDISC call to disconnect a program from a queue manager in z/OS batch.

Bu kod özetinde kullanılan değişkenler, “Kuyruk yöneticisiyle bağlantı kuruluyor” sayfa 8’inde ayarlananlar. This extract is taken from the Browse sample application (program CSQ4BCA1) supplied

with IBM MQ for z/OS. Diğer platformlardaki örnek uygulamaların adları ve yerleri için bkz. Örnek yordamsal programlar ( z/OS dışındaki platformlar).

```
:
/*
/* Disconnect from the queue manager. Test the
/* output of the disconnect call. If the call
/* fails, print an error message showing the
/* completion code and reason code.
/*
MQDISC(&Hconn,
      &CompCode,
      &Reason);
if ((CompCode != MQCC_OK) || (Reason != MQRC_NONE))
{
  sprintf(pBuff, MESSAGE_4_E,
          ERROR_IN_MQDISC, CompCode, Reason);
  PrintLine(pBuff);
  RetCode = CSQ4_ERROR;
}
:
```

### ***Dinamik kuyruk yaratılması***

Bu örnek, dinamik bir kuyruk yaratmak için MQOPEN çağrısının nasıl kullanılacağını gösterir.

Bu alma, IBM MQ for z/OS ile sağlanan Mail Manager örnek uygulamasından (program CSQ4TCD1) alınır. Diğer platformlardaki örnek uygulamaların adları ve yerleri için bkz. Örnek yordamsal programlar ( z/OS dışındaki platformlar).

```
:
MQLONG HCONN = 0; /* Connection handle */
MQHOBJ HOBJ; /* MailQ Object handle */
MQHOBJ HobjTempQ; /* TempQ Object Handle */
MQLONG CompCode; /* Completion code */
MQLONG Reason; /* Qualifying reason */
MQOD ObjDesc = {MQOD_DEFAULT};
MQLONG OpenOptions; /* Options control MQOPEN */

/*-----*/
/* Initialize the Object Descriptor (MQOD)
/* control block. (The remaining fields
/* are already initialized.)
/*-----*/
strncpy( ObjDesc.ObjectName,
        SYSTEM_REPLY_MODEL,
        MQ_Q_NAME_LENGTH );
strncpy( ObjDesc.DynamicQName,
        SYSTEM_REPLY_INITIAL,
        MQ_Q_NAME_LENGTH );
OpenOptions = MQOO_INPUT_AS_Q_DEF;
/*-----*/
/* Open the model queue and, therefore,
/* create and open a temporary dynamic
/* queue
/*-----*/
MQOPEN( HCONN,
        &ObjDesc,
        OpenOptions,
        &HobjTempQ,
        &CompCode,
        &Reason );
if ( CompCode == MQCC_OK ) {
}
else {
  /*-----*/
  /* Build an error message to report the
  /* failure of the opening of the model
  /* queue
  /*-----*/
  MQMErrorHandling( "OPEN TEMPQ", CompCode,
                  Reason );
  ErrorFound = TRUE;
}
}
```

```

return ErrorFound;
}
...

```

### **Varolan bir kuyruğun açılması**

Bu örnek, önceden tanımlanmış bir kuyruğu açmak için MQOPEN çağrısını nasıl kullanacağını gösterir.

This extract is taken from the Browse sample application (program CSQ4BCA1) supplied with IBM MQ for z/OS. Diğer platformlardaki örnek uygulamaların adları ve yerleri için bkz. [Örnek yordamsal programlar \(z/OS dışındaki platformlar\)](#).

```

#include <cmqc.h>
...
static char Parm1[MQ_Q_MGR_NAME_LENGTH];
...
int main(int argc, char *argv[] )
{
/*
/* Variables for MQ calls */
/*
MQHCONN Hconn ; /* Connection handle */
MQLONG CompCode; /* Completion code */
MQLONG Reason; /* Qualifying reason */
MQOD ObjDesc = { MQOD_DEFAULT };
/* Object descriptor */
MQLONG OpenOptions; /* Options that control */
/* the MQOPEN call */
MQHOBJ Hobj; /* Object handle */
...
/* Copy the queue name, passed in the parm field, */
/* to Parm2 strncpy(Parm2,argv[2], */
/* MQ_Q_NAME_LENGTH); */
...
/* Initialize the object descriptor (MQOD) control */
/* block. (The initialization default sets StrucId, */
/* Version, ObjectType, ObjectQMgrName, */
/* DynamicQName, and AlternateUserid fields) */
/*
strncpy(ObjDesc.ObjectName,Parm2,MQ_Q_NAME_LENGTH);
...
/* Initialize the other fields required for the open */
/* call (Hobj is set by the MQCONN call). */
/*
OpenOptions = MQOO_BROWSE;
...
/* Open the queue. */
/* Test the output of the open call. If the call */
/* fails, print an error message showing the */
/* completion code and reason code, then bypass */
/* processing, disconnect and leave the program. */
/*
MQOPEN(Hconn,
      &ObjDesc,
      OpenOptions,
      &Hobj,
      &CompCode,
      &Reason);

if ((CompCode != MQCC_OK) || (Reason != MQRC_NONE))
{
printf(pBuff, MESSAGE_4_E,
      ERROR_IN_MQOPEN, CompCode, Reason);
PrintLine(pBuff);
RetCode = CSQ4_ERROR;
goto AbnormalExit1; /* disconnect processing */
}
...
} /* end of main */

```

### **Kuyruğun Kapatılması**

Bu örnek, bir kuyruğu kapatmak için MQCLOSE çağrısının nasıl kullanılacağını gösterir.

This extract is taken from the Browse sample application (program CSQ4BCA1) supplied with IBM MQ for z/OS. Diğer platformlardaki örnek uygulamaların adları ve yerleri için bkz. [Örnek yordamsal programlar \(z/OS dışındaki platformlar\)](#).

```

:
/*                                     */
/* Close the queue.                    */
/* Test the output of the close call   */
/* fails, print an error message showing the */
/* completion code and reason code.    */
/*                                     */
MQCLOSE(Hconn,
        &Hobj,
        MQCO_NONE,
        &CompCode,
        &Reason);
if ((CompCode != MQCC_OK) || (Reason != MQRC_NONE))
{
    sprintf(pBuff, MESSAGE_4_E,
            ERROR_IN_MQCLOSE, CompCode, Reason);
    PrintLine(pBuff);
    RetCode = CSQ4_ERROR;
}
:

```

### **MQPUT kullanarak ileti koyma**

Bu örnek, bir iletiyi kuyruğa koymak için MQPUT çağrısının nasıl kullanılacağını gösterir.

Bu alma, IBM MQ ile verilen örnek uygulamalardan alınmıyor. Örnek uygulamaların adları ve yerleri için bkz. [Örnek yordamsal programlar \(platforms dışındaki platformlar \(z/OS\) !\[\]\(9dfdaff1d86ba3c1f8353b4d1b61b8c5\_img.jpg\) ve IBM MQ for z/OS için örnek programlar](#) .

```

:
qput()
{
    MQMD    MsgDesc;
    MQPMO   PutMsgOpts;
    MQLONG  CompCode;
    MQLONG  Reason;
    MQHCONN Hconn;
    MQHOBJ  Hobj;
    char message_buffer[] = "MY MESSAGE";
    /*-----*/
    /* Set up PMO structure.          */
    /*-----*/
    memset(&PutMsgOpts, '\0', sizeof(PutMsgOpts));
    memcpy(PutMsgOpts.StrucId, MQPMO_STRUC_ID,
           sizeof(PutMsgOpts.StrucId));
    PutMsgOpts.Version = MQPMO_VERSION_1;
    PutMsgOpts.Options = MQPMO_SYNCPOINT;

    /*-----*/
    /* Set up MD structure.           */
    /*-----*/
    memset(&MsgDesc, '\0', sizeof(MsgDesc));
    memcpy(MsgDesc.StrucId, MQMD_STRUC_ID,
           sizeof(MsgDesc.StrucId));
    MsgDesc.Version      = MQMD_VERSION_1;
    MsgDesc.Expiry       = MQEI_UNLIMITED;
    MsgDesc.Report       = MQRO_NONE;
    MsgDesc.MsgType      = MQMT_DATAGRAM;
    MsgDesc.Priority     = 1;
    MsgDesc.Persistence = MQPER_PERSISTENT;
    memset(MsgDesc.ReplyToQ,
           '\0',
           sizeof(MsgDesc.ReplyToQ));
    /*-----*/
    /* Put the message.                */
    /*-----*/
    MQPUT(Hconn, Hobj, &MsgDesc, &PutMsgOpts,
           sizeof(message_buffer), message_buffer,
           &CompCode, &Reason);
}
:

```

```

/*-----*/
/* Check completion and reason codes. */
/*-----*/
switch (CompCode)
{
    case MQCC_OK:
        break;
    case MQCC_FAILED:
        switch (Reason)
        {
            case MQRC_Q_FULL:
            case MQRC_MSG_TOO_BIG_FOR_Q:
                break;
            default:
                break; /* Perform error processing */
        }
        break;
    default:
        break; /* Perform error processing */
}
}

```

### ***MQPUT1 kullanılarak ileti konması***

Bu örnek, bir kuyruğu açmak için MQPUT1 çağrısının nasıl kullanılacağını gösterir, kuyruğa tek bir ileti koyar ve kuyruğu kapatır.

This extract is taken from the Credit Check sample application (program CSQ4CCB5) supplied with IBM MQ for z/OS. Diğer platformlardaki örnek uygulamaların adları ve yerleri için bkz. [Örnek yordamsal programlar \(z/OS dışındaki platformlar\)](#).

```

:
MQLONG  Hconn; /* Connection handle */
MQHOBJ  Hobj_CheckQ; /* Object handle */
MQLONG  CompCode; /* Completion code */
MQLONG  Reason; /* Qualifying reason */
MQOD    ObjDesc = {MQOD_DEFAULT}; /* Object descriptor */
MQMD    MsgDesc = {MQMD_DEFAULT}; /* Message descriptor */
MQLONG  OpenOptions; /* Control the MQOPEN call */
MQGMO   GetMsgOpts = {MQGMO_DEFAULT}; /* Get Message Options */
MQLONG  MsgBuffLen; /* Length of message buffer */
CSQ4BCAQ MsgBuffer; /* Message structure */
MQLONG  DataLen; /* Length of message */
MQPMO   PutMsgOpts = {MQPMO_DEFAULT}; /* Put Message Options */
CSQ4BQRM PutBuffer; /* Message structure */
MQLONG  PutBuffLen = sizeof(PutBuffer); /* Length of message buffer */
:

```

```

void Process_Query(void)
{
    /*
    /* Build the reply message
    /*
    :
    /*
    /* Set the object descriptor, message descriptor and
    /* put message options to the values required to
    /* create the reply message.
    /*
    strncpy(ObjDesc.ObjectName, MsgDesc.ReplyToQ,
            MQ_Q_NAME_LENGTH);
    strncpy(ObjDesc.ObjectQMGrName, MsgDesc.ReplyToQMGr,
            MQ_Q_MGR_NAME_LENGTH);
    MsgDesc.MsgType = MQMT_REPLY;
    MsgDesc.Report = MQRO_NONE;
    memset(MsgDesc.ReplyToQ, ' ', MQ_Q_NAME_LENGTH);
    memset(MsgDesc.ReplyToQMGr, ' ', MQ_Q_MGR_NAME_LENGTH);

```



```

memcpy(MsgDesc.MsgId, MQMI_NONE, sizeof(MsgDesc.MsgId));
PutMsgOpts.Options = MQPMO_SYNCPOINT +
                    MQPMO_PASS_IDENTITY_CONTEXT;
PutMsgOpts.Context = Hobj_CheckQ;
PutBufLen = sizeof(PutBuffer);
MQPUT1(Hconn,
        &ObjDesc,
        &MsgDesc,
        &PutMsgOpts,
        PutBufLen,
        &PutBuffer,
        &CompCode,
        &Reason);

if (CompCode != MQCC_OK)
{
    strncpy(TS_Operation, "MQPUT1",
            sizeof(TS_Operation));
    strncpy(TS_ObjName, ObjDesc.ObjectName,
            MQ_Q_NAME_LENGTH);
    Record_Call_Error();
    Forward_Msg_To_DLQ();
}
return;
}
:

```

## İleti alınıyor

Bu örnek, bir iletiyi kuyruktan kaldırmak için MQGET çağrısının nasıl kullanılacağını gösterir.

This extract is taken from the Browse sample application (program CSQ4BCA1) supplied with IBM MQ for z/OS. Diğer platformlardaki örnek uygulamaların adları ve yerleri için bkz. [Örnek yordamsal programlar \(z/OS dışındaki platformlar\)](#).

```

#include "cmqc.h"
:
#define BUFFERLENGTH 80
:
int main(int argc, char *argv[] )
{
    /*                                     */
    /*   Variables for MQ calls           */
    /*                                     */
    MQHCONN Hconn ;                      /* Connection handle */
    MQLONG  CompCode;                    /* Completion code   */
    MQLONG  Reason;                      /* Qualifying reason */
    MQHOBJ  Hobj;                        /* Object handle     */
    MQMD    MsgDesc = { MQMD_DEFAULT }; /* Message descriptor */
    MQLONG  DataLength ;                 /* Length of the message */
    MQCHAR  Buffer[BUFFERLENGTH+1];      /* Area for message data */
    MQGMO   GetMsgOpts = { MQGMO_DEFAULT }; /* Options which control */
    MQLONG  BufferLength = BUFFERLENGTH ; /* Length of buffer   */
    :
    /*   No need to change the message descriptor */
    /*   (MQMD) control block because initialization */
    /*   default sets all the fields.             */
    /*   Initialize the get message options (MQGMO) */
    /*   control block (the copy file initializes all */
    /*   the other fields).                       */
    /*   GetMsgOpts.Options = MQGMO_NO_WAIT      + */
    /*                               MQGMO_BROWSE_FIRST + */
    /*                               MQGMO_ACCEPT_TRUNCATED_MSG; */
    /*   Get the first message.                 */
    /*   Test for the output of the call is carried out */
    /*   in the 'for' loop.                    */
    /*   MQGET(Hconn,                          */
    /*         Hobj,                            */
    /*         &MsgDesc,

```

```

        &GetMsgOpts,
        BufferLength,
        Buffer,
        &DataLength,
        &CompCode,
        &Reason);

/*
/* Process the message and get the next message,
/* until no messages remaining.
...
/* If the call fails for any other reason,
/* print an error message showing the completion
/* code and reason code.
/*
/*
if ( (CompCode == MQCC_FAILED) &&
     (Reason == MQRC_NO_MSG_AVAILABLE) )
{
...
}
else
{
    sprintf(pBuff, MESSAGE_4_E,
            ERROR_IN_MQGET, CompCode, Reason);
    PrintLine(pBuff);
    RetCode = CSQ4_ERROR;
}
...
} /* end of main */

```

### ***Bekleme seçeneğini kullanarak ileti alınması***

Bu örnek, MQGET çağrısının bekleme seçeneğinin nasıl kullanılacağını gösterir.

Bu kod kesilen iletileri kabul eder. This extract is taken from the Credit Check sample application (program CSQ4CCB5) supplied with IBM MQ for z/OS. Diğer platformlardaki örnek uygulamaların adları ve yerleri için bkz. [Örnek yordamsal programlar \(z/OS dışındaki platformlar\)](#).

```

:
MQLONG  Hconn;           /* Connection handle      */
MQHOBJ  Hobj_CheckQ;    /* Object handle          */
MQLONG  CompCode;       /* Completion code        */
MQLONG  Reason;         /* Qualifying reason      */
MQOD    ObjDesc = {MQOD_DEFAULT}; /* Object descriptor      */
MQMD    MsgDesc = {MQMD_DEFAULT}; /* Message descriptor     */
MQLONG  OpenOptions;    /* Control the MQOPEN call */
MQGMO   GetMsgOpts = {MQGMO_DEFAULT}; /* Get Message Options   */
MQLONG  MsgBufLen;      /* Length of message buffer */
CSQ4BCAQ MsgBuffer;     /* Message structure      */
MQLONG  DataLen;        /* Length of message      */

```

```

:
void main(void)
{
:
/*
/* Initialize options and open the queue for input
/*
:
/*
/* Get and process messages
/*
/*
GetMsgOpts.Options = MQGMO_WAIT +
                    MQGMO_ACCEPT_TRUNCATED_MSG +
                    MQGMO_SYNCPOINT;
GetMsgOpts.WaitInterval = WAIT_INTERVAL;
MsgBufLen = sizeof(MsgBuffer);
memcpy(MsgDesc.MsgId, MQMI_NONE,
        sizeof(MsgDesc.MsgId));
memcpy(MsgDesc.CorrelId, MQCI_NONE,

```

```

        sizeof(MsgDesc.CorrelId));
/*
/* Make the first MQGET call outside the loop
/*
MQGET(Hconn,
      Hobj_CheckQ,
      &MsgDesc,
      &GetMsgOpts,
      MsgBufLen,
      &MsgBuffer,
      &DataLen,
      &CompCode,
      &Reason);
:
/*
/* Test the output of the MQGET call. If the call
/* failed, send an error message showing the
/* completion code and reason code, unless the
/* reason code is NO_MSG_AVAILABLE.
/*
/*
if (Reason != MQRC_NO_MSG_AVAILABLE)
{
    strncpy(TS_Operation, "MQGET", sizeof(TS_Operation));
    strncpy(TS_ObjName, ObjDesc.ObjectName,
            MQ_Q_NAME_LENGTH);
    Record_Call_Error();
}
:

```

## Sinyal kullanarak ileti alınıyor

Sinyalizasyon yalnızca IBM MQ for z/OS ile kullanılabilir.

Bu örnek, bir sinyal ayarlamak için MQGET çağrısının nasıl kullanılacağını gösterir. Böylece, kuyruğa uygun bir ileti geldiğinde size bildirim gönderilir. Bu alma, IBM MQ ile verilen örnek uygulamalardan alınmıştır.

```

:
get_set_signal()
{
    MQMD    MsgDesc;
    MQGMO   GetMsgOpts;
    MQLONG  CompCode;
    MQLONG  Reason;
    MQHCONN Hconn;
    MQHOBJ  Hobj;
    MQLONG  BufferLength;
    MQLONG  DataLength;
    char message_buffer[100];
    long int q_ecb, work_ecb;
    short int signal_sw, endloop;
    long int mask = 255;

    /*-----*/
    /* Set up GMO structure.
    /*-----*/
    memset(&GetMsgOpts, '\0', sizeof(GetMsgOpts));
    memcpy(GetMsgOpts.StrucId, MQGMO_STRUC_ID,
           sizeof(GetMsgOpts.StrucId));
    GetMsgOpts.Version      = MQGMO_VERSION_1;
    GetMsgOpts.WaitInterval = 1000;
    GetMsgOpts.Options      = MQGMO_SET_SIGNAL +
                              MQGMO_BROWSE_FIRST;

    q_ecb = 0;
    GetMsgOpts.Signal1 = &q_ecb;
    /*-----*/
    /* Set up MD structure.
    /*-----*/
    memset(&MsgDesc, '\0', sizeof(MsgDesc));
    memcpy(MsgDesc.StrucId, MQMD_STRUC_ID,
           sizeof(MsgDesc.StrucId));
    MsgDesc.Version = MQMD_VERSION_1;
    MsgDesc.Report  = MQRO_NONE;
    memcpy(MsgDesc.MsgId, MQMI_NONE,
           sizeof(MsgDesc.MsgId));
    memcpy(MsgDesc.CorrelId, MQCI_NONE,
           sizeof(MsgDesc.CorrelId));
}

```

```

/*-----*/
/* Issue the MQGET call. */
/*-----*/
BufferLength = sizeof(message_buffer);
signal_sw = 0;

MQGET(Hconn, Hobj, &MsgDesc, &GetMsgOpts,
      BufferLength, message_buffer, &DataLength,
      &CompCode, &Reason);
/*-----*/
/* Check completion and reason codes. */
/*-----*/
switch (CompCode)
{
  case (MQCC_OK):          /* Message retrieved */
    break;
  case (MQCC_WARNING):
    switch (Reason)
    {
      case (MQRC_SIGNAL_REQUEST_ACCEPTED):
        signal_sw = 1;
        break;
      default:
        break; /* Perform error processing */
    }
    break;
  case (MQCC_FAILED):
    switch (Reason)
    {
      case (MQRC_Q_MGR_NOT_AVAILABLE):
      case (MQRC_CONNECTION_BROKEN):
      case (MQRC_Q_MGR_STOPPING):
        break;
      default:
        break; /* Perform error processing. */
    }
    break;
  default:
    break; /* Perform error processing. */
}
/*-----*/
/* If the SET_SIGNAL was accepted, set up a loop to */
/* check whether a message has arrived at one second */
/* intervals. The loop ends if a message arrives or */
/* the wait interval specified in the MQGMO */
/* structure has expired. */
/* */
/* If a message arrives on the queue, another MQGET */
/* must be issued to retrieve the message. If other */
/* MQM calls have been made in the intervening */
/* period, this may necessitate reinitializing the */
/* MQMD and MQGMO structures. */
/* In this code, no intervening calls */
/* have been made, so the only change required to */
/* the structures is to specify MQGMO_NO_WAIT, */
/* since we now know the message is there. */
/* */
/* This code uses the EXEC CICS DELAY command to */
/* suspend the program for a second. A batch program */
/* may achieve the same effect by calling an */
/* assembler language subroutine which issues a */
/* z/OS STIMER macro. */
/*-----*/

```

```

if (signal_sw == 1)
{
  endloop = 0;
  do
  {
    EXEC CICS DELAY FOR HOURS(0) MINUTES(0) SECONDS(1);
    work_ecb = q_ecb & mask;
    switch (work_ecb)
    {
      case (MQEC_MSG_ARRIVED):
        endloop = 1;
        mqgmo_options = MQGMO_NO_WAIT;
        MQGET(Hconn, Hobj, &MsgDesc, &GetMsgOpts,
              BufferLength, message_buffer,

```

```

        &DataLength, &CompCode, &Reason);
    if (CompCode != MQCC_OK)
        ; /* Perform error processing. */
    break;
    case (MQEC_WAIT_INTERVAL_EXPIRED):
    case (MQEC_WAIT_CANCELED):
        endloop = 1;
        break;
    default:
        break;
    }
} while (endloop == 0);
}
return;
}

```

### ***Bir nesnenin özniteliklerinin sorulmasına neden oluyor***

Bu örnek, bir kuyruğun öznitelikleriyle ilgili olarak sorgulamak için MQINQ çağrısının nasıl kullanılacağını gösterir.

Bu alma işlemi, IBM MQ for z/OS ile sağlanan Kuyruk Öznitelikleri örnek uygulamasından (CSQ4CCC1programı) alınır. Diğer platformlardaki örnek uygulamaların adları ve yerleri için bkz. [Örnek yordamsal programlar \(z/OS dışındaki platformlar\)](#).

```

#include <cmqc.h> /* MQ API header file */
:
#define NUMBEROFSELECTORS 2

const MQHCONN Hconn = MQHC_DEF_HCONN;
:
static void InquireGetAndPut(char *Message,
                             PMQHOBJ pHobj,
                             char *Object)
{
/* Declare local variables */
/*
MQLONG SelectorCount = NUMBEROFSELECTORS; /* Number of selectors */
MQLONG IntAttrCount = NUMBEROFSELECTORS; /* Number of int attrs */
MQLONG CharAttrLength = 0; /* Length of char attribute buffer */
MQCHAR *CharAttrs; /* Character attribute buffer */
MQLONG SelectorsTable[NUMBEROFSELECTORS]; /* attribute selectors */
MQLONG IntAttrsTable[NUMBEROFSELECTORS]; /* integer attributes */
MQLONG CompCode; /* Completion code */
MQLONG Reason; /* Qualifying reason */
/*
/* Open the queue. If successful, do the inquire */
/* call. */
/*
/* Initialize the variables for the inquire */
/* call: */
/* - Set SelectorsTable to the attributes whose */
/* status is */
/* required */
/* - All other variables are already set */
/*
SelectorsTable[0] = MQIA_INHIBIT_GET;
SelectorsTable[1] = MQIA_INHIBIT_PUT;
/*
/* Issue the inquire call */
/* Test the output of the inquire call. If the */
/* call failed, display an error message */
/* showing the completion code and reason code, */
/* otherwise display the status of the */
/* INHIBIT-GET and INHIBIT-PUT attributes */
/*
MQINQ(Hconn,
      *pHobj,

```

```

        SelectorCount,
        SelectorTable,
        IntAttrCount,
        IntAttrsTable,
        CharAttrLength,
        CharAttrs,
        &CompCode,
        &Reason);
if (CompCode != MQCC_OK)
{
    sprintf(Message, MESSAGE_4_E,
            ERROR_IN_MQINQ, CompCode, Reason);
    SetMsg(Message);
}
else
{
    /* Process the changes */
} /* end if CompCode */

```

## Kuyruğun Özniteliklerinin Ayarlanmasını

Bu örnek, bir kuyruğun özniteliklerini değiştirmek için MQSET çağrısının nasıl kullanılacağını gösterir.

Bu alma işlemi, IBM MQ for z/OS ile sağlanan Kuyruk Öznitelikleri örnek uygulamasından (CSQ4CCC1programı) alınır. Diğer platformlardaki örnek uygulamaların adları ve yerleri için bkz. [Örnek yordamsal programlar \(z/OS dışındaki platformlar\)](#).

```

#include <cmqc.h>      /* MQ API header file      */
:
#define NUMBEROFSELECTORS 2

const MQHCONN Hconn = MQHC_DEF_HCONN;

static void InhibitGetAndPut(char *Message,
                             PMQHOBJ pHobj,
                             char *Object)
{
    /*
    /*   Declare local variables
    /*
    /*
    MQLONG SelectorCount = NUMBEROFSELECTORS;
    /* Number of selectors */
    MQLONG IntAttrCount = NUMBEROFSELECTORS;
    /* Number of int attrs */
    MQLONG CharAttrLength = 0;
    /* Length of char attribute buffer */
    MQCHAR *CharAttrs ;
    /* Character attribute buffer */
    MQLONG SelectorTable[NUMBEROFSELECTORS];
    /* attribute selectors */
    MQLONG IntAttrsTable[NUMBEROFSELECTORS];
    /* integer attributes */
    MQLONG CompCode;
    /* Completion code */
    MQLONG Reason;
    /* Qualifying reason */
    :
    /*
    /*   Open the queue.  If successful, do the
    /*   inquire call.
    /*
    /*
    :
    /*
    /*   Initialize the variables for the set call:
    /*   - Set SelectorTable to the attributes to be
    /*   set
    /*   - Set IntAttrsTable to the required status
    /*   - All other variables are already set
    /*
    SelectorTable[0] = MQIA_INHIBIT_GET;
    SelectorTable[1] = MQIA_INHIBIT_PUT;
    IntAttrsTable[0] = MQQA_GET_INHIBITED;
    IntAttrsTable[1] = MQQA_PUT_INHIBITED;
    :

    /*
    /*   Issue the set call.
    /*   Test the output of the set call.  If the
    */

```

```

/*      call fails, display an error message      */
/*      showing the completion code and reason    */
/*      code; otherwise move INHIBITED to the    */
/*      relevant screen map fields                */
/*      */
MQSET(Hconn,
      *pHobj,
      SelectorCount,
      SelectorTable,
      IntAttrCount,
      IntAttrsTable,
      CharAttrLength,
      CharAttrs,
      &CompCode,
      &Reason);
if (CompCode != MQCC_OK)
{
    sprintf(Message, MESSAGE_4_E,
            ERROR_IN_MQSET, CompCode, Reason);
    SetMsg(Message);
}
else
{
    /* Process the changes */
} /* end if CompCode */

```

### ***MQSTAT ile durum bilgileri alınıyor***

Bu örnek, zamanuysuz bir MQPUT ' nin nasıl verileceğini ve MQSTAT ile durum bilgilerinin nasıl alınacağını gösterir.

Bu alma Calling MQSTAT örnek uygulamasından alınır (program amqsapt0 ) IBM MQ for Windows sistemleriyle birlikte sağlanır. Diğer altyapılardaki örnek uygulamaların adları ve yerleri için bkz. [Örnek yordam programları \( z/OS dışında platformlar\)](#).

```

/*****
/*
/* Program name: AMQSAPT0
/*
/* Description: Sample C program that asynchronously puts messages
/*              to a message queue (example using MQPUT & MQSTAT).
/*
/* Licensed Materials - Property of IBM
/*
/* 63H9336
/* (c) Copyright IBM Corp. 2006, 2024. All Rights Reserved.
/*
/* US Government Users Restricted Rights - Use, duplication or
/* disclosure restricted by GSA ADP Schedule Contract with
/* IBM Corp.
/*
/*****
/*
/* Function:
/*
/* AMQSAPT0 is a sample C program to put messages on a message
/* queue with asynchronous response option, querying the success
/* of the put operations with MQSTAT.
/*
/* -- messages are sent to the queue named by the parameter
/*
/* -- gets lines from StdIn, and adds each to target
/* queue, taking each line of text as the content
/* of a datagram message; the sample stops when a null
/* line (or EOF) is read.
/* New-line characters are removed.
/* If a line is longer than 99 characters it is broken up
/* into 99-character pieces. Each piece becomes the
/* content of a datagram message.
/* If the length of a line is a multiple of 99 plus 1, for
/* example, 199, the last piece will only contain a
/* new-line character so will terminate the input.
/*
/* -- writes a message for each MQI reason other than
/* MQRC_NONE; stops if there is a MQI completion code
/* of MQCC_FAILED
/*
/* -- summarizes the overall success of the put operations
/*

```

```

/*      through a call to MQSTAT to query MQSTAT_TYPE_ASYNC_ERROR*/
/*
/*      Program logic:
/*      MQOPEN target queue for OUTPUT
/*      while end of input file not reached,
/*      . read next line of text
/*      . MQPUT datagram message with text line as data
/*      MQCLOSE target queue
/*      MQSTAT connection
/*
/*
/*
/*****
/*
/*      AMQSAPTO has the following parameters
/*      required:
/*      optional:
/*      (1) The name of the target queue
/*      (2) Queue manager name
/*      (3) The open options
/*      (4) The close options
/*      (5) The name of the target queue manager
/*      (6) The name of the dynamic queue
/*
/*****
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
/* includes for MQI */
#include <cmqc.h>

int main(int argc, char **argv)
{
/* Declare file and character for sample input
FILE *fp;

/* Declare MQI structures needed
MQOD    od = {MQOD_DEFAULT}; /* Object Descriptor
MQMD    md = {MQMD_DEFAULT}; /* Message Descriptor
MQPMO   pmo = {MQPMO_DEFAULT}; /* put message options
MQSTS   sts = {MQSTS_DEFAULT}; /* status information
/** note, sample uses defaults where it can **/
MQHCONN Hcon; /* connection handle
MQHOBJ  Hobj; /* object handle
MQLONG  O_options; /* MQOPEN options
MQLONG  C_options; /* MQCLOSE options
MQLONG  CompCode; /* completion code
MQLONG  OpenCode; /* MQOPEN completion code
MQLONG  Reason; /* reason code
MQLONG  CReason; /* reason code for MQCONN
MQLONG  messlen; /* message length
char    buffer[100]; /* message buffer
char    QMName[50]; /* queue manager name

printf("Sample AMQSAPTO start\n");
if (argc < 2)
{
printf("Required parameter missing - queue name\n");
exit(99);
}

/*****
/*
/*      Connect to queue manager
/*
/*****
QMName[0] = 0; /* default */
if (argc > 2)
strcpy(QMName, argv[2]);
MQCONN(QMName, /* queue manager
        &Hcon, /* connection handle
        &Compcode, /* completion code
        &Reason); /* reason code
/* report reason and stop if it failed */
if (CompCode == MQCC_FAILED)
{
printf("MQCONN ended with reason code %d\n", CReason);
exit( (int)CReason );
}

/*****
/*
/*      Use parameter as the name of the target queue

```



```

/*
/*****
strncpy(od.ObjectName, argv[1], (size_t)MQ_Q_NAME_LENGTH);
printf("target queue is %s\n", od.ObjectName);

if (argc > 5)
{
    strncpy(od.ObjectQMgrName, argv[5], (size_t) MQ_Q_MGR_NAME_LENGTH);
    printf("target queue manager is %s\n", od.ObjectQMgrName);
}

if (argc > 6)
{
    strncpy(od.DynamicQName, argv[6], (size_t) MQ_Q_NAME_LENGTH);
    printf("dynamic queue name is %s\n", od.DynamicQName);
}

/*****
/*
/*   Open the target message queue for output
/*
/*
/*****
if (argc > 3)
{
    O_options = atoi( argv[3] );
    printf("open options are %d\n", O_options);
}
else
{
    O_options = MQOO_OUTPUT           /* open queue for output
    | MQOO_FAIL_IF_QUIESCING /* but not if MQM stopping
    ;                               /* = 0x2010 = 8208 decimal
}

MQOPEN(Hcon,           /* connection handle
        &od,           /* object descriptor for queue
        O_options,     /* open options
        &Hobj,         /* object handle
        &OpenCode,     /* MQOPEN completion code
        &Reason);     /* reason code

/* report reason, if any; stop if failed
if (Reason != MQRC_NONE)
{
    printf("MQOPEN ended with reason code %d\n", Reason);
}

if (OpenCode == MQCC_FAILED)
{
    printf("unable to open queue for output\n");
}

/*****
/*
/*   Read lines from the file and put them to the message queue
/*
/*   Loop until null line or end of file, or there is a failure
/*
/*
/*****
CompCode = OpenCode; /* use MQOPEN result for initial test
fp = stdin;

memcpy(md.Format, /* character string format
        MQFMT_STRING, (size_t)MQ_FORMAT_LENGTH);

/*****
/* These options specify that put operation should occur
/* asynchronously and the application will check the success
/* using MQSTAT at a later time.
/*
/*****
md.Persistence = MQPER_NOT_PERSISTENT;
pmo.Options |= MQPMO_ASYNC_RESPONSE;

/*****
/* These options cause the MsgId and CorrelId to be replaced, so
/* that there is no need to reset them before each MQPUT
/*
/*****
pmo.Options |= MQPMO_NEW_MSG_ID;
pmo.Options |= MQPMO_NEW_CORREL_ID;

while (CompCode != MQCC_FAILED)
{
    if (fgets(buffer, sizeof(buffer), fp) != NULL)

```

```

{
    messlen = (MQLONG)strlen(buffer); /* length without null */
    if (buffer[messlen-1] == '\n') /* last char is a new-line */
    {
        buffer[messlen-1] = '\0'; /* replace new-line with null */
        --messlen; /* reduce buffer length */
    }
}
else messlen = 0; /* treat EOF same as null line */

/*****
/*
/* Put each buffer to the message queue */
/*
/*****
if (messlen > 0)
{
    MQPUT(Hcon, /* connection handle */
        Hobj, /* object handle */
        &md, /* message descriptor */
        &pmo, /* default options (datagram) */
        messlen, /* message length */
        buffer, /* message buffer */
        &CompCode, /* completion code */
        &Reason); /* reason code */

    /* report reason, if any */
    if (Reason != MQRC_NONE)
    {
        printf("MQPUT ended with reason code %d\n", Reason);
    }
}
else /* satisfy end condition when empty line is read */
    CompCode = MQCC_FAILED;
}

/*****
/*
/* Close the target queue (if it was opened) */
/*
/*****
if (OpenCode != MQCC_FAILED)
{
    if (argc > 4)
    {
        C_options = atoi( argv[4] );
        printf("close options are %d\n", C_options);
    }
    else
    {
        C_options = MQCO_NONE; /* no close options */
    }

    MQCLOSE(Hcon, /* connection handle */
        &Hobj, /* object handle */
        C_options,
        &CompCode, /* completion code */
        &Reason); /* reason code */

    /* report reason, if any */
    if (Reason != MQRC_NONE)
    {
        printf("MQCLOSE ended with reason code %d\n", Reason);
    }
}

/*****
/*
/* Query how many asynchronous puts succeeded */
/*
/*****
MQSTAT(&Hcon, /* connection handle */
    MQSTAT_TYPE_ASYNC_ERROR, /* status type */
    &Sts, /* MQSTS structure */
    &CompCode, /* completion code */
    &Reason); /* reason code */

/* report reason, if any */
if (Reason != MQRC_NONE)
{
    printf("MQSTAT ended with reason code %d\n", Reason);
}
}

```

```

else
{
/* Display results */
printf("Succeeded putting %d messages\n",
      sts.PutSuccessCount);
printf("%d messages were put with a warning\n",
      sts.PutWarningCount);
printf("Failed to put %d messages\n",
      sts.PutFailureCount);

if(sts.CompCode == MQCC_WARNING)
{
printf("The first warning that occurred had reason code %d\n",
      sts.Reason);
}
else if(sts.CompCode == MQCC_FAILED)
{
printf("The first error that occurred had reason code %d\n",
      sts.Reason);
}
}
}

/*****
/*
/* Disconnect from MQM if not already connected
/*
/*
/*****
if (CReason != MQRC_ALREADY_CONNECTED)
{
MQDISC(&Hcon,          /* connection handle
      &CompCode,       /* completion code
      &Reason);        /* reason code

/* report reason, if any
if (Reason != MQRC_NONE)
{
printf("MQDISC ended with reason code %d\n", Reason);
}
}

/*****
/*
/* END OF AMQSAPT0
/*
/*
/*****
printf("Sample AMQSAPT0 end\n");
return(0);
}

```

## COBOL örnekleri

Bu konu derlemi, IBM MQ for z/OS örnek uygulamalarından alınır. Bunlar, belirtilenler dışında, tüm platformlar için geçerli olur.

### ***Kuyruk yöneticisiyle bağlantı kuruluyor***

Bu örnek, bir programı z/OS toplu kipte bir kuyruk yöneticisine bağlamak için MQCONN çağrısının nasıl kullanılacağını gösterir.

This extract is taken from the Browse sample application (program CSQ4BVA1) supplied with IBM MQ for z/OS. Diğer platformlardaki örnek uygulamaların adları ve yerleri için bkz. [Örnek yordamsal programlar \(z/OS dışındaki platformlar\)](#).

```

* -----*
WORKING-STORAGE SECTION.
* -----*
* W02 - Data fields derived from the PARM field
01 W02-MQM PIC X(48) VALUE SPACES.
* W03 - MQM API fields
01 W03-HCONN PIC S9(9) BINARY.
01 W03-COMPCODE PIC S9(9) BINARY.
01 W03-REASON PIC S9(9) BINARY.
*
* MQV contains constants (for filling in the control
* blocks)
* and return codes (for testing the result of a call)

```

```

*
01 W05-MQM-CONSTANTS.
COPY CMQV SUPPRESS.
:
* Separate into the relevant fields any data passed
* in the PARM statement
*
UNSTRING PARM-STRING DELIMITED BY ALL ','
          INTO W02-MQM
          W02-OBJECT.
:
* Connect to the specified queue manager.
*
CALL 'MQCONN' USING W02-MQM
                  W03-HCONN
                  W03-COMPCODE
                  W03-REASON.
:
*
* Test the output of the connect call. If the call
* fails, print an error message showing the
* completion code and reason code.
*
IF (W03-COMPCODE NOT = MQCC-OK) THEN
:
END-IF.
:

```

### ***Kuyruk yöneticisinden bağlantı kesiliyor***

This example demonstrates how to use the MQDISC call to disconnect a program from a queue manager in z/OS batch.

Bu kod özetinde kullanılan değişkenler, [“Kuyruk yöneticisiyle bağlantı kuruluyor”](#) sayfa 23içinde ayarlanırlar. This extract is taken from the Browse sample application (program CSQ4BVA1) supplied with IBM MQ for z/OS. Diğer platformlardaki örnek uygulamaların adları ve yerleri için bkz. [Örnek yordamsal programlar \( z/OS dışındaki platformlar\)](#).

```

:
*
* Disconnect from the queue manager
*
CALL 'MQDISC' USING W03-HCONN
                  W03-COMPCODE
                  W03-REASON.
:
*
* Test the output of the disconnect call. If the
* call fails, print an error message showing the
* completion code and reason code.
*
IF (W03-COMPCODE NOT = MQCC-OK) THEN
:
END-IF.
:

```

### ***Dinamik kuyruk yaratılması***

Bu örnek, dinamik bir kuyruk yaratmak için MQOPEN çağrısının nasıl kullanılacağını gösterir.

This extract is taken from the Credit Check sample application (program CSQ4CVB1) supplied with IBM MQ for z/OS. Diğer platformlardaki örnek uygulamaların adları ve yerleri için bkz. [Örnek yordamsal programlar \( z/OS dışındaki platformlar\)](#).

```

:
* -----*
WORKING-STORAGE SECTION.
* -----*
*
* W02 - Queues processed in this program
*
01 W02-MODEL-QNAME PIC X(48) VALUE
   'CSQ4SAMP.B1.MODEL'
01 W02-NAME-PREFIX PIC X(48) VALUE
   'CSQ4SAMP.B1.*'
01 W02-TEMPORARY-Q PIC X(48).

```

```

*
*   W03 - MQM API fields
*
01  W03-HCONN      PIC S9(9) BINARY VALUE ZERO.
01  W03-OPTIONS   PIC S9(9) BINARY.
01  W03-HOBJ      PIC S9(9) BINARY.
01  W03-COMPCODE  PIC S9(9) BINARY.
01  W03-REASON    PIC S9(9) BINARY.
*
*   API control blocks
*
01  MQM-OBJECT-DESCRIPTOR.
    COPY CMQODV.
*
*   CMQV contains constants (for setting or testing
*   field values) and return codes (for testing the
*   result of a call)
*
01  MQM-CONSTANTS.
    COPY CMQV SUPPRESS.
* -----*
*   PROCEDURE DIVISION.
* -----*
:
* -----*
*   OPEN-TEMP-RESPONSE-QUEUE SECTION.
* -----*

```

```

*
*   This section creates a temporary dynamic queue
*   using a model queue
*
* -----*
*
*   Change three fields in the Object Descriptor (MQOD)
*   control block. (MQODV initializes the other fields)
*
    MOVE MQOT-Q          TO MQOD-OBJECTTYPE.
    MOVE W02-MODEL-QNAME TO MQOD-OBJECTNAME.
    MOVE W02-NAME-PREFIX TO MQOD-DYNAMICQNAME.
*
    COMPUTE W03-OPTIONS = MQOD-INPUT-EXCLUSIVE.
*
    CALL 'MQOPEN' USING W03-HCONN
                        MQOD
                        W03-OPTIONS
                        W03-HOBJ-MODEL
                        W03-COMPCODE
                        W03-REASON.
*
    IF W03-COMPCODE NOT = MQCC-OK
        MOVE 'MQOPEN' TO M01-MSG4-OPERATION
        MOVE W03-COMPCODE TO M01-MSG4-COMPCODE
        MOVE W03-REASON TO M01-MSG4-REASON
        MOVE M01-MESSAGE-4 TO M00-MESSAGE
    ELSE
        MOVE MQOD-OBJECTNAME TO W02-TEMPORARY-Q
    END-IF.
*
    OPEN-TEMP-RESPONSE-QUEUE-EXIT.
*
*   Return to performing section.
*
    EXIT.
    EJECT
*

```

### ***Varolan bir kuyruğun açılması***

Bu örnek, varolan bir kuyruğu açmak için MQOPEN çağrısının nasıl kullanılacağını gösterir.

This extract is taken from the Browse sample application (program CSQ4BVA1) supplied with IBM MQ for z/OS. Diğer platformlardaki örnek uygulamaların adları ve yerleri için bkz. [Örnek yordamsal programlar \(z/OS dışındaki platformlar\)](#).

:

```

* -----*
WORKING-STORAGE SECTION.
* -----*
*
*   W01 - Fields derived from the command area input
*
01 W01-OBJECT          PIC X(48).
*
*   W02 - MQM API fields
*
01 W02-HCONN          PIC S9(9) BINARY VALUE ZERO.
01 W02-OPTIONS        PIC S9(9) BINARY.
01 W02-HOBJ           PIC S9(9) BINARY.
01 W02-COMPCODE       PIC S9(9) BINARY.
01 W02-REASON         PIC S9(9) BINARY.
*
*   CMQODV defines the object descriptor (MQOD)
*
01 MQM-OBJECT-DESCRIPTOR.
   COPY CMQODV.
*
* CMQV contains constants (for setting or testing
* field values) and return codes (for testing the
* result of a call)
*
01 MQM-CONSTANTS.
   COPY CMQV SUPPRESS.
* -----*
E-OPEN-QUEUE SECTION.
* -----*
*
* This section opens the queue
*
* Initialize the Object Descriptor (MQOD) control
* block
* (The copy file initializes the remaining fields.)
*
MOVE MQOT-Q           TO MQOD-OBJECTTYPE.
MOVE W01-OBJECT       TO MQOD-OBJECTNAME.
*
* Initialize W02-OPTIONS to open the queue for both
* inquiring about and setting attributes
*
COMPUTE W02-OPTIONS = MQ00-INQUIRE + MQ00-SET.

```

```

*
* Open the queue
*
CALL 'MQOPEN' USING W02-HCONN
                   MQOD
                   W02-OPTIONS
                   W02-HOBJ
                   W02-COMPCODE
                   W02-REASON.
*
* Test the output from the open
*
* If the completion code is not OK, display a
* separate error message for each of the following
* errors:
*
* Q-MGR-NOT-AVAILABLE - MQM is not available
* CONNECTION-BROKEN   - MQM is no longer connected to CICS
* UNKNOWN-OBJECT-NAME - The queue does not exist
* NOT-AUTHORIZED      - The user is not authorized to open
*                     the queue
*
* For any other error, display an error message
* showing the completion and reason codes
*
IF W02-COMPCODE NOT = MQCC-OK
   EVALUATE TRUE
*
   WHEN W02-REASON = MQRC-Q-MGR-NOT-AVAILABLE
      MOVE M01-MESSAGE-6 TO M00-MESSAGE
*
   WHEN W02-REASON = MQRC-CONNECTION-BROKEN
      MOVE M01-MESSAGE-6 TO M00-MESSAGE
*
   WHEN W02-REASON = MQRC-UNKNOWN-OBJECT-NAME

```

```

        MOVE M01-MESSAGE-2 TO M00-MESSAGE
*
    WHEN W02-REASON = M0RC-NOT-AUTHORIZED
        MOVE M01-MESSAGE-3 TO M00-MESSAGE
*
    WHEN OTHER
        MOVE 'M0OPEN'          TO M01-MSG4-OPERATION
        MOVE W02-COMPCODE     TO M01-MSG4-COMPCODE
        MOVE W02-REASON       TO M01-MSG4-REASON
        MOVE M01-MESSAGE-4 TO M00-MESSAGE
    END-EVALUATE
    END-IF.
E-EXIT.
*
*   Return to performing section
*
    EXIT.
    EJECT

```

### ***Kuyruğun Kapatılması***

Bu örnek, MQCLOSE çağrısının nasıl kullanılacağını gösterir.

Bu kod özetinde kullanılan değişkenler, “[Kuyruk yöneticisiyle bağlantı kuruluyor](#)” sayfa 23’ünde ayarlananlar. This extract is taken from the Browse sample application (program CSQ4BVA1) supplied with IBM MQ for z/OS. Diğer platformlardaki örnek uygulamaların adları ve yerleri için bkz. [Örnek yordamsal programlar \( z/OS dışındaki platformlar\)](#).

```

:
*
*   Close the queue
*
    MOVE MQCO-NONE TO W03-OPTIONS.
*
    CALL 'MQCLOSE' USING W03-HCONN
                        W03-HOBJ
                        W03-OPTIONS
                        W03-COMPCODE
                        W03-REASON.
*
*   Test the output of the MQCLOSE call.  If the call
*   fails, print an error message showing the
*   completion code and reason code.
*
    IF (W03-COMPCODE NOT = MQCC-OK) THEN
        MOVE 'CLOSE'          TO W04-MSG4-TYPE
        MOVE W03-COMPCODE     TO W04-MSG4-COMPCODE
        MOVE W03-REASON       TO W04-MSG4-REASON
        MOVE W04-MESSAGE-4 TO W00-PRINT-DATA
        PERFORM PRINT-LINE
        MOVE W06-CSQ4-ERROR TO W00-RETURN-CODE
    END-IF.
*

```

### ***MQPUT kullanarak ileti koyma***

Bu örnek, bağlamı kullanarak MQPUT çağrısının nasıl kullanılacağını gösterir.

This extract is taken from the Credit Check sample application (program CSQ4CVB1) supplied with IBM MQ for z/OS. Diğer platformlardaki örnek uygulamaların adları ve yerleri için bkz. [Örnek yordamsal programlar \( z/OS dışındaki platformlar\)](#).

```

:
* -----*
*   WORKING-STORAGE SECTION.
* -----*
*
*   W02 - Queues processed in this program
*
01  W02-TEMPORARY-Q          PIC X(48).
*
*   W03 - MQM API fields
*
01  W03-HCONN                PIC S9(9) BINARY VALUE ZERO.
01  W03-HOBJ-INQUIRY        PIC S9(9) BINARY.

```

```

01 W03-OPTIONS          PIC S9(9) BINARY.
01 W03-BUFFLEN          PIC S9(9) BINARY.
01 W03-COMPCODE         PIC S9(9) BINARY.
01 W03-REASON           PIC S9(9) BINARY.
*
01 W03-PUT-BUFFER.
*
05 W03-CSQ4BIIM.
COPY CSQ4VB1.
*
* API control blocks
*
01 MQM-MESSAGE-DESCRIPTOR.
COPY CMQMDV.
01 MQM-PUT-MESSAGE-OPTIONS.
COPY CMQPMOV.
*
* MQV contains constants (for filling in the
* control blocks) and return codes (for testing
* the result of a call).
*
01 MQM-CONSTANTS.
COPY CMQV SUPPRESS.
* -----*
PROCEDURE DIVISION.
* -----*
:
* Open queue and build message.
:

```

```

*
* Set the message descriptor and put-message options to
* the values required to create the message.
* Set the length of the message.
*
MOVE MQMT-REQUEST          TO MQMD-MSGTYPE.
MOVE MQCI-NONE             TO MQMD-CORRELID.
MOVE MQMI-NONE             TO MQMD-MSGID.
MOVE W02-TEMPORARY-Q      TO MQMD-REPLYTOQ.
MOVE SPACES                TO MQMD-REPLYTOQMGR.
MOVE 5                     TO MQMD-PRIORITY.
MOVE MQPER-NOT-PERSISTENT TO MQMD-PERSISTENCE.
COMPUTE MQPMO-OPTIONS     = MQPMO-NO-SYNCPOINT +
                          MQPMO-DEFAULT-CONTEXT.
MOVE LENGTH OF CSQ4BIIM-MSG TO W03-BUFFLEN.
*
CALL 'MQPUT' USING W03-HCONN
                  W03-HOBJ-INQUIRY
                  MQMD
                  MQPMO
                  W03-BUFFLEN
                  W03-PUT-BUFFER
                  W03-COMPCODE
                  W03-REASON.
IF W03-COMPCODE NOT = MQCC-OK
:
END-IF.

```

### ***MQPUT1 kullanılarak ileti konması***

Bu örnek, MQPUT1 çağrısının nasıl kullanılacağını gösterir.

This extract is taken from the Credit Check sample application (program CSQ4CVB5) supplied with IBM MQ for z/OS. Diğer platformlardaki örnek uygulamaların adları ve yerleri için bkz. [Örnek yordamsal programlar \(z/OS dışındaki platformlar\)](#).

```

:
* -----*
WORKING-STORAGE SECTION.
* -----*
*
* W03 - MQM API fields
*
01 W03-HCONN          PIC S9(9) BINARY VALUE ZERO.
01 W03-OPTIONS       PIC S9(9) BINARY.
01 W03-COMPCODE       PIC S9(9) BINARY.

```



```

01 W03-REASON          PIC S9(9) BINARY.
01 W03-BUFFLEN        PIC S9(9) BINARY.
*
01 W03-PUT-BUFFER.
   05 W03-CSQ4BQRM.
   COPY CSQ4VB4.

```

```

*
*   API control blocks
*
01 MQM-OBJECT-DESCRIPTOR.
   COPY CMQODV.
01 MQM-MESSAGE-DESCRIPTOR.
   COPY CMQMDV.
01 MQM-PUT-MESSAGE-OPTIONS.
   COPY CMQPMOV.
*
* CMQV contains constants (for filling in the
* control blocks) and return codes (for testing
* the result of a call).
*
01 MQM-MQV.
   COPY CMQV SUPPRESS.
* -----*
PROCEDURE DIVISION.
* -----*
:
*   Get the request message.
:
* -----*
PROCESS-QUERY SECTION.
* -----*
:
*   Build the reply message.
:
*
* Set the object descriptor, message descriptor and
* put-message options to the values required to create
* the message.
* Set the length of the message.
*
MOVE MQMD-REPLYTOQ      TO MQOD-OBJECTNAME.
MOVE MQMD-REPLYTOQMGR  TO MQOD-OBJECTQMGRNAME.
MOVE MQMT-REPLY        TO MQMD-MSGTYPE.
MOVE SPACES            TO MQMD-REPLYTOQ.
MOVE SPACES            TO MQMD-REPLYTOQMGR.
MOVE LOW-VALUES        TO MQMD-MSGID.
COMPUTE MQPMO-OPTIONS = MQPMO-SYNCPOINT +
                        MQPMO-PASS-IDENTITY-CONTEXT.
MOVE W03-HOBJ-CHECKQ   TO MQPMO-CONTEXT.
MOVE LENGTH OF CSQ4BQRM-MSG TO W03-BUFFLEN.
*
   CALL 'MQPUT1' USING W03-HCONN
                        MQOD
                        MQMD
                        MQPMO
                        W03-BUFFLEN
                        W03-PUT-BUFFER
                        W03-COMPCODE
                        W03-REASON.
   IF W03-COMPCODE NOT = MQCC-OK
      MOVE 'MQPUT1'      TO M02-OPERATION
      MOVE MQOD-OBJECTNAME TO M02-OBJECTNAME
      PERFORM RECORD-CALL-ERROR
      PERFORM FORWARD-MSG-TO-DLQ
   END-IF.
*

```

## ***İleti alınıyor***

Bu örnek, bir iletiyi kuyruktan kaldırmak için MQGET çağrısının nasıl kullanılacağını gösterir.

This extract is taken from the Credit Check sample application (program CSQ4CVB1) supplied with IBM MQ for z/OS. Diğer platformlardaki örnek uygulamaların adları ve yerleri için bkz. Örnek yordamsal programlar ( z/OS dışındaki platformlar).

```

:
* -----*
WORKING-STORAGE SECTION.
* -----*
*
*   W03 - MQM API fields
*
01 W03-HCONN          PIC S9(9) BINARY VALUE ZERO.
01 W03-HOBJ-RESPONSE PIC S9(9) BINARY.
01 W03-OPTIONS       PIC S9(9) BINARY.
01 W03-BUFFLEN       PIC S9(9) BINARY.
01 W03-DATALEN       PIC S9(9) BINARY.
01 W03-COMPCODE      PIC S9(9) BINARY.
01 W03-REASON        PIC S9(9) BINARY.
*
01 W03-GET-BUFFER.
   05 W03-CSQ4BAM.
   COPY CSQ4VB2.
*
*   API control blocks
*
01 MQM-MESSAGE-DESCRIPTOR.
   COPY CMQMDV.
01 MQM-GET-MESSAGE-OPTIONS.
   COPY CMQGMV.
*
*   MQV contains constants (for filling in the
*   control blocks) and return codes (for testing
*   the result of a call).
*
01 MQM-CONSTANTS.
   COPY CMQV SUPPRESS.
* -----*
A-MAIN SECTION.
* -----*
:
*   Open response queue.
:
* -----*
PROCESS-RESPONSE-SCREEN SECTION.
* -----*
*
*   This section gets a message from the response queue.
*
*   When a correct response is received, it is
*   transferred to the map for display; otherwise
*   an error message is built.
*
* -----*

```

```

*
*   Set get-message options
*
*   COMPUTE MQGMO-OPTIONS = MQGMO-SYNCPOINT +
*                           MQGMO-ACCEPT-TRUNCATED-MSG +
*                           MQGMO-NO-WAIT.
*
*   Set msgid and correlid in MQMD to nulls so that any
*   message will qualify.
*   Set length to available buffer length.
*
*   MOVE MQMI-NONE TO MQMD-MSGID.
*   MOVE MQCI-NONE TO MQMD-CORRELID.
*   MOVE LENGTH OF W03-GET-BUFFER TO W03-BUFFLEN.
*
*   CALL 'MQGET' USING W03-HCONN
*                       W03-HOBJ-RESPONSE
*                       MQMD
*                       MQGMO
*                       W03-BUFFLEN
*                       W03-GET-BUFFER
*                       W03-DATALEN
*                       W03-COMPCODE

```

```

                                W03-REASON.
EVALUATE TRUE
  WHEN W03-COMPCODE NOT = MQCC-FAILED
  :
*   Process the message
  :
  WHEN (W03-COMPCODE = MQCC-FAILED AND
        W03-REASON = MQRC-NO-MSG-AVAILABLE)
    MOVE M01-MESSAGE-9 TO M00-MESSAGE
    PERFORM CLEAR-RESPONSE-SCREEN
*
  WHEN OTHER
    MOVE 'MQGET '      TO M01-MSG4-OPERATION
    MOVE W03-COMPCODE TO M01-MSG4-COMPCODE
    MOVE W03-REASON   TO M01-MSG4-REASON
    MOVE M01-MESSAGE-4 TO M00-MESSAGE
    PERFORM CLEAR-RESPONSE-SCREEN
END-EVALUATE.

```

### ***Bekleme seçeneğini kullanarak ileti alınması***

Bu örnek, bekleme seçeneği ile MQGET çağrısının nasıl kullanılacağını ve kesilmiş iletilerin nasıl kabul edileceğini gösterir.

This extract is taken from the Credit Check sample application (program CSQ4CVB5) supplied with IBM MQ for z/OS. Diğer platformlardaki örnek uygulamaların adları ve yerleri için bkz. [Örnek yordamsal programlar \( z/OS dışındaki platformlar\)](#).

```

:
* -----*
WORKING-STORAGE SECTION.
* -----*
*
*   W00 - General work fields
*
01 W00-WAIT-INTERVAL   PIC S9(09) BINARY VALUE 30000.
*
*   W03 - MQM API fields
*
01 W03-HCONN          PIC S9(9) BINARY VALUE ZERO.
01 W03-OPTIONS        PIC S9(9) BINARY.
01 W03-HOBJ-CHECKQ    PIC S9(9) BINARY.
01 W03-COMPCODE        PIC S9(9) BINARY.
01 W03-REASON          PIC S9(9) BINARY.
01 W03-DATALEN        PIC S9(9) BINARY.
01 W03-BUFFLEN         PIC S9(9) BINARY.
*
01 W03-MSG-BUFFER.
   05 W03-CSQ4BCAQ.
   COPY CSQ4VB3.
*
*   API control blocks
*
01 MQM-MESSAGE-DESCRIPTOR.
   COPY CMQMDV.
01 MQM-GET-MESSAGE-OPTIONS.
   COPY CMQGMV.
*
*   CMQV contains constants (for filling in the
*   control blocks) and return codes (for testing
*   the result of a call).
*
01 MQM-MQV.
   COPY CMQV SUPPRESS.
* -----*
PROCEDURE DIVISION.
* -----*
:
*   Open input queue.
:

```

```

*
*   Get and process messages.
*
COMPUTE MQGMO-OPTIONS = MQGMO-WAIT +
                        MQGMO-ACCEPT-TRUNCATED-MSG +

```

```

MQGMO-SYNCPOINT.
MOVE LENGTH OF W03-MSG-BUFFER TO W03-BUFFLEN.
MOVE W00-WAIT-INTERVAL TO MQGMO-WAITINTERVAL.
MOVE MQMI-NONE TO MQMD-MSGID.
MOVE MQCI-NONE TO MQMD-CORRELID.
*
*   Make the first MQGET call outside the loop.
*
CALL 'MQGET' USING W03-HCONN
                  W03-HOBJ-CHECKQ
                  MQMD
                  MQGMO
                  W03-BUFFLEN
                  W03-MSG-BUFFER
                  W03-DATALEN
                  W03-COMPCODE
                  W03-REASON.
*
*   Test the output of the MQGET call using the
*   PERFORM loop that follows.
*
*   Perform whilst no failure occurs
*   - process this message
*   - reset the call parameters
*   - get another message
*   End-perform
*
*
*   Test the output of the MQGET call.  If the call
*   fails, send an error message showing the
*   completion code and reason code, unless the
*   completion code is NO-MSG-AVAILABLE.
*
IF (W03-COMPCODE NOT = MQCC-FAILED) OR
(W03-REASON NOT = MQRC-NO-MSG-AVAILABLE)
  MOVE 'MQGET '          TO M02-OPERATION
  MOVE MQ0D-OBJECTNAME  TO M02-OBJECTNAME
  PERFORM RECORD-CALL-ERROR
END-IF.
:

```

### ***Sinyal kullanarak ileti alınıyor***

Bu örnek, MQGET çağrısının sinyalizasyon ile nasıl kullanılacağını gösterir. This extract is taken from the Credit Check sample application (program CSQ4CVB2) supplied with IBM MQ for z/OS.

*Sinyalizasyon yalnızca IBM MQ for z/OS ile kullanılabilir.*

```

:
* -----*
WORKING-STORAGE SECTION.
* -----*
*
*   W00 - General work fields
*
01 W00-WAIT-INTERVAL    PIC S9(09) BINARY VALUE 30000.
*
*   W03 - MQM API fields
*
01 W03-HCONN           PIC S9(9) BINARY VALUE ZERO.
01 W03-HOBJ-REPLYQ    PIC S9(9) BINARY.
01 W03-COMPCODE       PIC S9(9) BINARY.
01 W03-REASON         PIC S9(9) BINARY.
01 W03-DATALEN        PIC S9(9) BINARY.
01 W03-BUFFLEN        PIC S9(9) BINARY.
:
01 W03-GET-BUFFER.
05 W03-CSQ4BQRM.
COPY CSQ4VB4.
*
05 W03-CSQ4BIIM REDEFINES W03-CSQ4BQRM.
COPY CSQ4VB1.
*
05 W03-CSQ4BPGM REDEFINES W03-CSQ4BIIM.
COPY CSQ4VB5.
:

```

```

*   API control blocks
*
01  MQM-MESSAGE-DESCRIPTOR.
    COPY CMQMDV.
01  MQM-GET-MESSAGE-OPTIONS.
    COPY CMQGMV.
    :
*   MQV contains constants (for filling in the
*   control blocks) and return codes (for testing
*   the result of a call).
*
01  MQM-MQV.
    COPY CMQV SUPPRESS.
* -----*
LINKAGE SECTION.
* -----*
01  L01-ECB-ADDR-LIST.
     05  L01-ECB-ADDR1          POINTER.
     05  L01-ECB-ADDR2          POINTER.

*
01  L02-ECBS.
     05  L02-INQUIRY-ECB1       PIC S9(09) BINARY.
     05  L02-REPLY-ECB2        PIC S9(09) BINARY.
01  REDEFINES L02-ECBS.
     05                                     PIC X(02).
     05  L02-INQUIRY-ECB1-CC    PIC S9(04) BINARY.
     05                                     PIC X(02).
     05  L02-REPLY-ECB2-CC     PIC S9(04) BINARY.

*
* -----*
PROCEDURE DIVISION.
* -----*
:
* Initialize variables, open queues, set signal on
* inquiry queue.
:
* -----*
PROCESS-SIGNAL-ACCEPTED SECTION.
* -----*
* This section gets a message with signal.  If a
* message is received, process it.  If the signal
* is set or is already set, the program goes into
* an operating system wait.
* Otherwise an error is reported and call error set.
* -----*
*
PERFORM REPLYQ-GETSIGNAL.
*
EVALUATE TRUE
  WHEN (W03-COMPCODE = MQCC-OK AND
        W03-REASON = MQRC-NONE)
    PERFORM PROCESS-REPLYQ-MESSAGE
*
  WHEN (W03-COMPCODE = MQCC-WARNING AND
        W03-REASON = MQRC-SIGNAL-REQUEST-ACCEPTED)
    OR
    (W03-COMPCODE = MQCC-FAILED AND
     W03-REASON = MQRC-SIGNAL-OUTSTANDING)
    PERFORM EXTERNAL-WAIT
*
  WHEN OTHER
    MOVE 'MQGET SIGNAL' TO M02-OPERATION
    MOVE MQOD-OBJECTNAME TO M02-OBJECTNAME
    PERFORM RECORD-CALL-ERROR
    MOVE W06-CALL-ERROR TO W06-CALL-STATUS
END-EVALUATE.
*
PROCESS-SIGNAL-ACCEPTED-EXIT.
* Return to performing section
EXIT.
EJECT
*
* -----*
EXTERNAL-WAIT SECTION.
* -----*
* This section performs an external CICS wait on two
*

```

```

* ECBs until at least one is posted. It then calls      *
* the sections to handle the posted ECB.                *
* -----*
EXEC CICS WAIT EXTERNAL
      ECBLIST(W04-ECB-ADDR-LIST-PTR)
      NUMEVENTS(2)
END-EXEC.
*
* At least one ECB must have been posted to get to this
* point. Test which ECB has been posted and perform
* the appropriate section.
*
      IF L02-INQUIRY-ECB1 NOT = 0
          PERFORM TEST-INQUIRYQ-ECB
      ELSE
          PERFORM TEST-REPLYQ-ECB
      END-IF.
*
EXTERNAL-WAIT-EXIT.
*
* Return to performing section.
*
      EXIT.
      EJECT
      :
* -----*
REPLYQ-GETSIGNAL SECTION.
* -----*
* This section performs an MQGET call (in syncpoint with *
* signal) on the reply queue. The signal field in the *
* MQGMO is set to the address of the ECB.             *
* Response handling is done by the performing section. *
* -----*
*
      COMPUTE MQGMO-OPTIONS          = MQGMO-SYNCPOINT +
          MQGMO-SET-SIGNAL.
      MOVE W00-WAIT-INTERVAL        TO MQGMO-WAITINTERVAL.
      MOVE LENGTH OF W03-GET-BUFFER TO W03-BUFFLEN.
*
      MOVE ZEROS                    TO L02-REPLY-ECB2.
      SET MQGMO-SIGNAL1 TO ADDRESS OF L02-REPLY-ECB2.
*
*
* Set msgid and correlid to nulls so that any message
* will qualify.
*
      MOVE MQMI-NONE TO MQMD-MSGID.
      MOVE MQCI-NONE TO MQMD-CORRELID.
*
      CALL 'MQGET' USING W03-HCONN
          W03-HOBJ-REPLYQ
          MQMD
          MQGMO
          W03-BUFFLEN
          W03-GET-BUFFER
          W03-DATALEN
          W03-COMPCODE
          W03-REASON.
*
REPLYQ-GETSIGNAL-EXIT.
*
* Return to performing section.
*
      EXIT.
      EJECT
*
      :

```

### ***Bir nesnenin özniteliklerinin sorulmasına neden oluyor***

Bu örnek, bir kuyruğun öznitelikleriyle ilgili olarak sorgulamak için MQINQ çağrısının nasıl kullanılacağını gösterir.

This extract is taken from the Queue Attributes sample application (program CSQ4CVC1) supplied with IBM MQ for z/OS. Diğer platformlardaki örnek uygulamaların adları ve yerleri için bkz. Örnek yordamsal programlar ( z/OS dışındaki platformlar).

```

:
* -----*
WORKING-STORAGE SECTION.
* -----*
*
*   W02 - MQM API fields
*
01 W02-SELECTORCOUNT    PIC S9(9) BINARY VALUE 2.
01 W02-INTATTRCOUNT    PIC S9(9) BINARY VALUE 2.
01 W02-CHARATTRLENGTH   PIC S9(9) BINARY VALUE ZERO.
01 W02-CHARATTRS        PIC X    VALUE LOW-VALUES.
01 W02-HCONN             PIC S9(9) BINARY VALUE ZERO.
01 W02-HOBJ              PIC S9(9) BINARY.
01 W02-COMPCODE          PIC S9(9) BINARY.
01 W02-REASON            PIC S9(9) BINARY.
01 W02-SELECTORS-TABLE.
   05 W02-SELECTORS      PIC S9(9) BINARY OCCURS 2 TIMES
01 W02-INTATTRS-TABLE.
   05 W02-INTATTRS      PIC S9(9) BINARY OCCURS 2 TIMES
*
*   CMQODV defines the object descriptor (MQOD).
*
01 MQM-OBJECT-DESCRIPTOR.
   COPY CMQODV.
*
*   CMQV contains constants (for setting or testing field
*   values) and return codes (for testing the result of a
*   call).
*
01 MQM-CONSTANTS.
   COPY CMQV SUPPRESS.
* -----*
PROCEDURE DIVISION.
* -----*
*
*   Get the queue name and open the queue.
*
:
*
*   Initialize the variables for the inquiry call:
*   - Set W02-SELECTORS-TABLE to the attributes whose
*   status is required
*   - All other variables are already set
*
MOVE MQIA-INHIBIT-GET TO W02-SELECTORS(1).
MOVE MQIA-INHIBIT-PUT TO W02-SELECTORS(2).

*
*   Inquire about the attributes.
*
CALL 'MQINQ' USING W02-HCONN,
                  W02-HOBJ,
                  W02-SELECTORCOUNT,
                  W02-SELECTORS-TABLE,
                  W02-INTATTRCOUNT,
                  W02-INTATTRS-TABLE,
                  W02-CHARATTRLENGTH,
                  W02-CHARATTRS,
                  W02-COMPCODE,
                  W02-REASON.
*
*   Test the output from the inquiry:
*
*   - If the completion code is not OK, display an error
*   message showing the completion and reason codes
*
*   - Otherwise, move the correct attribute status into
*   the relevant screen map fields
*
IF W02-COMPCODE NOT = MQCC-OK
  MOVE 'MQINQ'          TO M01-MSG4-OPERATION
  MOVE W02-COMPCODE     TO M01-MSG4-COMPCODE
  MOVE W02-REASON      TO M01-MSG4-REASON

```

```

        MOVE M01-MESSAGE-4 TO M00-MESSAGE
*
*   ELSE
*       Process the changes.
*       :
*       END-IF.
*       :

```

## Kuyruğun Özniteliklerinin Ayarlanmasını

Bu örnek, bir kuyruğun özniteliklerini değiştirmek için MQSET çağrısının nasıl kullanılacağını gösterir.

This extract is taken from the Queue Attributes sample application (program CSQ4CVC1) supplied with IBM MQ for z/OS. Diğer platformlardaki örnek uygulamaların adları ve yerleri için bkz. [Örnek yordamsal programlar \( z/OS dışındaki platformlar\)](#)

```

:
* -----*
*   WORKING-STORAGE SECTION.
* -----*
*
*   W02 - MQM API fields
*
*01 W02-SELECTORCOUNT    PIC S9(9) BINARY VALUE 2.
*01 W02-INTATTRCOUNT    PIC S9(9) BINARY VALUE 2.
*01 W02-CHARATTRLENGTH   PIC S9(9) BINARY VALUE ZERO.
*01 W02-CHARATTRS        PIC X      VALUE LOW-VALUES.
*01 W02-HCONN            PIC S9(9) BINARY VALUE ZERO.
*01 W02-HOBJ             PIC S9(9) BINARY.
*01 W02-COMPCODE         PIC S9(9) BINARY.
*01 W02-REASON           PIC S9(9) BINARY.
*01 W02-SELECTORS-TABLE.
*   05 W02-SELECTORS     PIC S9(9) BINARY OCCURS 2 TIMES.
*01 W02-INTATTRS-TABLE.
*   05 W02-INTATTRS     PIC S9(9) BINARY OCCURS 2 TIMES.
*
*   CMQODV defines the object descriptor (MQOD).
*
*01 MQM-OBJECT-DESCRIPTOR.
*   COPY CMQODV.
*
*   CMQV contains constants (for setting or testing
*   field values) and return codes (for testing the
*   result of a call).
*
*01 MQM-CONSTANTS.
*   COPY CMQV SUPPRESS.
* -----*
*   PROCEDURE DIVISION.
* -----*

```

```

*
*   Get the queue name and open the queue.
*
*   :
*
*   Initialize the variables required for the set call:
*   - Set W02-SELECTORS-TABLE to the attributes to be set
*   - Set W02-INTATTRS-TABLE to the required status
*   - All other variables are already set
*
*   MOVE MQIA-INHIBIT-GET    TO W02-SELECTORS(1).
*   MOVE MQIA-INHIBIT-PUT    TO W02-SELECTORS(2).
*   MOVE MQQA-GET-INHIBITED TO W02-INTATTRS(1).
*   MOVE MQQA-PUT-INHIBITED TO W02-INTATTRS(2).
*
*   Set the attributes.
*
*   CALL 'MQSET' USING W02-HCONN,
*                     W02-HOBJ,
*                     W02-SELECTORCOUNT,
*                     W02-SELECTORS-TABLE,
*                     W02-INTATTRCOUNT,
*                     W02-INTATTRS-TABLE,
*                     W02-CHARATTRLENGTH,

```



```

                                W02-CHARATTRS,
                                W02-COMPCODE,
                                W02-REASON.
*
* Test the output from the call:
*
* - If the completion code is not OK, display an error
*   message showing the completion and reason codes
*
* - Otherwise, move 'INHIBITED' into the relevant
*   screen map fields
*
IF W02-COMPCODE NOT = MQCC-OK
    MOVE 'MQSET'          TO M01-MSG4-OPERATION
    MOVE W02-COMPCODE     TO M01-MSG4-COMPCODE
    MOVE W02-REASON      TO M01-MSG4-REASON
    MOVE M01-MESSAGE-4   TO M00-MESSAGE
ELSE
*
*   Process the changes.
*
:
END-IF.

```

## System/390 çevirici-dil örnekleri

Bu konu derlemi çoğunlukla IBM MQ for z/OS örnek uygulamalarından alınır.

### *Kuyruk yöneticisiyle bağlantı kuruluyor*

Bu örnek, bir programı z/OS toplu kipte bir kuyruk yöneticisine bağlamak için MQCONN çağrısının nasıl kullanılacağını gösterir.

Bu alma, IBM MQ for z/OS ile sağlanan Göz At örnek programından (CSQ4BAA1) alınır.

```

:
WORKAREA DSECT
*
PARMLIST CALL ,(0,0,0,0,0,0,0,0,0,0,0,0),VL,MF=L
*
COMPCODE DS    F           Completion code
REASON   DS    F           Reason code
HCONN   DS    F           Connection handle
          ORG
PARMADDR DS    F           Address of parm field
PARMLEN  DS    H           Length of parm field
*
MQMNAME  DS    CL48        Queue manager name
*
*****
* SECTION NAME : MAINPARM *
*****
MAINPARM DS    0H
          MVI   MQMNAME,X'40'
          MVC   MQMNAME+1(L'MQMNAME-1),MQMNAME
*
* Space out first byte and initialize
*
*
* Code to address and verify parameters passed omitted
*
*
PARM1MVE DS    0H
          SR    R1,R3           Length of data
          LA   R4,MQMNAME       Address for target
          BCTR R1,R0           Reduce for execute
          EX   R1,MOVEPARM       Move the data
*
*****
* EXECUTES *
*****
MOVEPARM MVC   0(*-*,R4),0(R3)
*
EJECT

```

```

*****
* SECTION NAME : MAINCONN *
*****
*
*
MAINCONN DS 0H
XC HCONN,HCONN Null connection handle
*
CALL MQCONN, X
(MQMNAME, X
HCONN, X
COMPCODE, X
REASON), X
MF=(E,PARMLIST),VL
*
LA R0,MQCC_OK Expected compcode
C R0,COMPCODE As expected?
BER R6 Yes .. return to caller
*
MVC INF4_TYP,=CL10'CONNECT '
BAL R7,ERRCODE Translate error
LA R0,8 Set exit code
ST R0,EXITCODE to 8
B ENDPROG End the program
*

```

### ***Kuyruk yöneticisinden bağlantı kesiliyor***

This example demonstrates how to use the MQDISC call to disconnect a program from a queue manager in z/OS batch.

Bu alma, IBM MQ ile verilen örnek uygulamalardan alınmıyor.

```

:
*
* ISSUE MQI DISC REQUEST USING REENTRANT FORM
* OF CALL MACRO
*
* HCONN WAS SET BY A PREVIOUS MQCONN REQUEST
* R5 = WORK REGISTER
*
DISC DS 0H
CALL MQDISC, X
(HCONN, X
COMPCODE, X
REASON), X
VL,MF=(E,CALLST)
*
LA R5,MQCC_OK
C R5,COMPCODE
BNE BADCALL
:

```

```

BADCALL DS 0H
:
*
* CONSTANTS
*
* CMQA
*
* WORKING STORAGE (RE-ENTRANT)
*
WEG3 DSECT
*
CALLST CALL ,(0,0,0,0,0,0,0,0,0,0,0,0),VL,MF=L
*
HCONN DS F
COMPCODE DS F
REASON DS F
*
*
LEG3 EQU *-WKEG3
END

```

## Dinamik kuyruk yaratılması

Bu örnek, dinamik bir kuyruk yaratmak için MQOPEN çağrısının nasıl kullanılacağını gösterir.

Bu alma, IBM MQ ile verilen örnek uygulamalardan alınmıyor.

```
:
*
* R5 = WORK REGISTER.
*
OPEN DS 0H
*
MVC WOD_AREA,MQOD_AREA INITIALIZE WORKING VERSION OF
* MQOD WITH DEFAULTS
MVC WOD_OBJECTNAME,MOD_Q COPY IN THE MODEL Q NAME
MVC WOD_DYNAMICQNAME,DYN_Q COPY IN THE DYNAMIC Q NAME
L R5,=AL4(MQOD_OUTPUT) OPEN FOR OUTPUT AND
A R5,=AL4(MQOD_INQUIRE) INQUIRE
ST R5,OPTIONS

*
* ISSUE MQI OPEN REQUEST USING REENTRANT
* FORM OF CALL MACRO
*
CALL MQOPEN, X
(HCONN, X
WOD, X
OPTIONS, X
HOBJ, X
COMPCODE, X
REASON),VL,MF=(E,CALLST)
*
LA R5,MQCC_OK CHECK THE COMPLETION CODE
C R5,COMPCODE FROM THE REQUEST AND BRANCH
BNE BADCALL TO ERROR ROUTINE IF NOT MQCC_OK
*
MVC TEMP_Q,WOD_OBJECTNAME SAVE NAME OF TEMPORARY Q
CREATED BY OPEN OF MODEL Q
*
*
*
BADCALL DS 0H
*
*
*
* CONSTANTS:
*
MOD_Q DC CL48'QUERY.REPLY.MODEL' MODEL QUEUE NAME
DYN_Q DC CL48'QUERY.TEMPQ.*' DYNAMIC QUEUE NAME
*
CMQODA DSECT=NO,LIST=YES CONSTANT VERSION OF MQOD
CMQA MQI VALUE EQUATES
*
* WORKING STORAGE
*
DFHEISTG
HCONN DS F CONNECTION HANDLE
OPTIONS DS F OPEN OPTIONS
HOBJ DS F OBJECT HANDLE
COMPCODE DS F MQI COMPLETION CODE
REASON DS F MQI REASON CODE
TEMP_Q DS CL(MQ_Q_NAME_LENGTH) SAVED QNAME AFTER OPEN
*
WOD CMQODA DSECT=NO,LIST=YES WORKING VERSION OF MQOD
*
CALLST CALL ,(0,0,0,0,0,0,0,0,0,0),VL,MF=L LIST FORM
OF CALL
MACRO
*
*
*
*
END
```

## Varolan bir kuyruğun açılması

Bu örnek, önceden tanımlanmış bir kuyruğu açmak için MQOPEN çağrısını nasıl kullanacağını gösterir.

Bu, iki seçeneğin nasıl belirtileceğini gösterir. Bu alma, IBM MQ ile verilen örnek uygulamalardan alınmıyor.

```

:
*
*   R5 = WORK REGISTER.
*
OPEN   DS   0H
*
*       MVC WOD_AREA,MQOD_AREA  INITIALIZE WORKING VERSION OF
*                               MQOD WITH DEFAULTS
*       MVC WOD_OBJECTNAME,Q_NAME  SPECIFY Q NAME TO OPEN
*       LA  R5,MQOO_INPUT_EXCLUSIVE OPEN FOR MQGET CALLS
*
*       ST   R5,OPTIONS
*
* ISSUE MQI OPEN REQUEST USING REENTRANT FORM
* OF CALL MACRO
*
*       CALL MQOPEN,                X
*           (HCONN,                 X
*            WOD,                   X
*            OPTIONS,               X
*            HOBJ,                 X
*            COMPCODE,             X
*            REASON),VL,MF=(E,CALLST)
*
*       LA  R5,MQCC_OK              CHECK THE COMPLETION CODE
*       C   R5,COMPCODE             FROM THE REQUEST AND BRANCH
*       BNE BADCALL                TO ERROR ROUTINE IF NOT MQCC_OK
*
*       :
BADCALL DS   0H
:
*
*   CONSTANTS:
*
Q_NAME  DC   CL48'REQUEST.QUEUE'  NAME OF QUEUE TO OPEN
*
*       CMQODA DSECT=NO,LIST=YES  CONSTANT VERSION OF MQOD
*       CMQA                                MQI VALUE EQUATES
*
*   WORKING STORAGE
*
*       DFHEISTG
HCONN   DS F                       CONNECTION HANDLE
OPTIONS DS F                       OPEN OPTIONS
HOBJ    DS F                       OBJECT HANDLE
COMPCODE DS F                     MQI COMPLETION CODE
REASON  DS F                       MQI REASON CODE
*
WOD     CMQODA DSECT=NO,LIST=YES  WORKING VERSION OF MQOD
*
CALLLST CALL , (0,0,0,0,0,0,0,0,0,0,0),VL,MF=L  LIST FORM
*                                           OF CALL
*                                           MACRO
*
*       :
*       END

```

### ***Kuyruğun Kapatılması***

Bu örnek, bir kuyruğu kapatmak için MQCLOSE çağrısının nasıl kullanılacağını gösterir.

Bu alma, IBM MQ ile verilen örnek uygulamalardan alınmıyor.

```

:
*
* ISSUE MQI CLOSE REQUEST USING REENTRANT FROM OF
* CALL MACRO
*
*       HCONN WAS SET BY A PREVIOUS MQCONN REQUEST
*       HOBJ  WAS SET BY A PREVIOUS MQOPEN REQUEST
*       R5 = WORK REGISTER
*
*       CLOSE DS   0H
*       LA   R5,MQCO_NONE          NO SPECIAL CLOSE OPTIONS
*       ST   R5,OPTIONS            ARE REQUIRED.

```

```

*          CALL MQCLOSE,                X
          (HCONN,                       X
           HOBJ,                         X
           OPTIONS,                      X
           COMPCODE,                    X
           REASON),                     X
          VL,MF=(E,CALLLST)

*
          LA  R5,MQCC_OK
          C   R5,COMPCODE
          BNE BADCALL

*
          :
BADCALL   DS   0H
          :

*          CONSTANTS
*
          CMQA
*
          WORKING STORAGE (REENTRANT)
*
WEG4      DSECT
*
CALLLST   CALL , (0,0,0,0,0,0,0,0,0,0,0,0),VL,MF=L
*
HCONN    DS   F
HOBJ     DS   F
OPTIONS  DS   F
COMPCODE DS   F
REASON   DS   F
*
*
LEG4     EQU  *-WKEG4
          END

```

### ***MQPUT kullanarak ileti koyma***

Bu örnek, bir iletiyi kuyruğa koymak için MQPUT çağrısının nasıl kullanılacağını gösterir.

Bu alma, IBM MQ ile verilen örnek uygulamalardan alınmıştır.

```

:
*          CONNECT TO QUEUE MANAGER
*
CONN     DS   0H
:
*
*          OPEN A QUEUE
*
OPEN     DS   0H
:
*
*          R4,R5,R6,R7 = WORK REGISTER.
*
PUT      DS   0H
          LA  R4,MQMD                SET UP ADDRESSES AND
          LA  R5,MQMD_LENGTH        LENGTH FOR USE BY MVCL
          LA  R6,WMD                INSTRUCTION, AS MQMD IS
          LA  R7,WMD_LENGTH        OVER 256 BYES LONG.
          MVCL R6,R4                INITIALIZE WORKING VERSION
*                                     OF MESSAGE DESCRIPTOR
*
          MVC WPMO_AREA,MQPMO_AREA INITIALIZE WORKING MQPMO
*
          LA  R5,BUFFER_LEN        RETRIEVE THE BUFFER LENGTH
          ST  R5,BUFFLEN           AND SAVE IT FOR MQM USE
*
          MVC BUFFER,TEST_MSG      SET THE MESSAGE TO BE PUT
*
*          ISSUE MQI PUT REQUEST USING REENTRANT FORM
*          OF CALL MACRO
*
*          HCONN WAS SET BY PREVIOUS MQCONN REQUEST
*          HOBJ WAS SET BY PREVIOUS MQOPEN REQUEST
*
          CALL MQPUT,                X
          (HCONN,                    X

```

```

HOBJ, X
WMD, X
WPMO, X
BUFFLEN, X
BUFFER, X
COMPCODE, X
REASON),VL,MF=(E,CALLLST)
*
LA R5,MQCC_OK
C R5,COMPCODE
BNE BADCALL
*
:
BADCALL DS 0H
:

```

```

*
*   CONSTANTS
*
CMQMDA DSECT=NO,LIST=YES,PERSISTENCE=MQPER_PERSISTENT
CMQPMOA DSECT=NO,LIST=YES
CMQA
TEST_MSG DC CL80'THIS IS A TEST MESSAGE'
*
*   WORKING STORAGE DSECT
*
WORKSTG DSECT
*
COMPCODE DS F
REASON DS F
BUFFLEN DS F
OPTIONS DS F
HCONN DS F
HOBJ DS F
*
BUFFER DS CL80
BUFFER_LEN EQU *-BUFFER
*
WMD CMQMDA DSECT=NO,LIST=NO
WPMO CMQPMOA DSECT=NO,LIST=NO
*
CALLLST CALL ,(0,0,0,0,0,0,0,0,0,0),VL,MF=L
*
:
END

```

### ***MQPUT1 kullanılarak ileti konması***

Bu örnek, bir kuyruğu açmak için MQPUT1 çağrısının nasıl kullanılacağını gösterir, kuyruğa tek bir ileti koyar ve kuyruğu kapatır.

Bu alma, IBM MQ ile verilen örnek uygulamalardan alınmıyor.

```

:
*
*   CONNECT TO QUEUE MANAGER
*
CONN DS 0H
:
*
*   R4,R5,R6,R7 = WORK REGISTER.
*
PUT DS 0H
*
MVC WOD_AREA,MQOD_AREA INITIALIZE WORKING VERSION OF
*                               MQOD WITH DEFAULTS
MVC WOD_OBJECTNAME,Q_NAME SPECIFY Q NAME FOR PUT1
*
LA R4,MQMD SET UP ADDRESSES AND
LA R5,MQMD_LENGTH LENGTH FOR USE BY MVCL
LA R6,WMD INSTRUCTION, AS MQMD IS
LA R7,WMD_LENGTH OVER 256 BYES LONG.
MVCL R6,R4 INITIALIZE WORKING VERSION
*                               OF MESSAGE DESCRIPTOR

```

```

*
*      MVC  WPMO_AREA,MQPMO_AREA      INITIALIZE WORKING MQPMO
*
*      LA   R5,BUFFER_LEN             RETRIEVE THE BUFFER LENGTH
*      ST   R5,BUFFLEN                AND SAVE IT FOR MQM USE
*
*      MVC  BUFFER,TEST_MSG           SET THE MESSAGE TO BE PUT
*
*      * ISSUE MQI PUT REQUEST USING REENTRANT FORM OF CALL MACRO
*
*      HCONN WAS SET BY PREVIOUS MQCONN REQUEST
*      HOBJ WAS SET BY PREVIOUS MQOPEN REQUEST
*
*      CALL MQPUT1,                    X
*      (HCONN,                         X
*       LMQOD,                         X
*       LMQMD,                         X
*       LMQPMO,                       X
*       BUFFERLENGTH,                 X
*       BUFFER,                       X
*       COMPCODE,                     X
*       REASON),VL,MF=(E,CALLST)
*
*      LA   R5,MQCC_OK
*      C    R5,COMPCODE
*      BNE  BADCALL
*
*      :
BADCALL DS 0H
*
*

```

```

*      CONSTANTS
*
*      CMQMDA DSECT=NO,LIST=YES,PERSISTENCE=MQPER_PERSISTENT
*      CMQPMOA DSECT=NO,LIST=YES
*      CMQODA DSECT=NO,LIST=YES
*      CMQA
*
*      TEST_MSG DC CL80'THIS IS ANOTHER TEST MESSAGE'
*      Q_NAME   DC CL48'TEST.QUEUE.NAME'
*
*      WORKING STORAGE DSECT
*
*      WORKSTG DSECT
*
*      COMPCODE DS F
*      REASON   DS F
*      BUFFLEN  DS F
*      OPTIONS  DS F
*      HCONN    DS F
*      HOBJ     DS F
*
*      BUFFER   DS CL80
*      BUFFER_LEN EQU *-BUFFER
*
*      WOD      CMQODA DSECT=NO,LIST=YES      WORKING VERSION OF MQOD
*      WMD      CMQMDA DSECT=NO,LIST=NO
*      WPMO     CMQPMOA DSECT=NO,LIST=NO
*
*      CALLLST CALL ,(0,0,0,0,0,0,0,0,0,0,0),VL,MF=L
*
*      :
*      END

```

### ***İleti alınıyor***

Bu örnek, bir iletiyi kuyruktan kaldırmak için MQGET çağrısının nasıl kullanılacağını gösterir.

Bu alma, IBM MQile verilen örnek uygulamalardan alınmıyor.

```

*
*      CONNECT TO QUEUE MANAGER

```

```

*
CONN    DS   0H
:
*
*      OPEN A QUEUE FOR GET
*
OPEN    DS   0H
:
*
*      R4,R5,R6,R7 = WORK REGISTER.
*
GET     DS   0H
      LA   R4,MQMD                SET UP ADDRESSES AND
      LA   R5,MQMD_LENGTH         LENGTH FOR USE BY MVCL
      LA   R6,WMD                 INSTRUCTION, AS MQMD IS
      LA   R7,WMD_LENGTH         OVER 256 BYES LONG.
      MVCL R6,R4                 INITIALIZE WORKING VERSION
*                                 OF MESSAGE DESCRIPTOR
*
*      MVC  WGMO_AREA,MQGMO_AREA  INITIALIZE WORKING MQGMO
*
      LA   R5,BUFFER_LEN         RETRIEVE THE BUFFER LENGTH
      ST   R5,BUFFLEN           AND SAVE IT FOR MQM USE
*
*
*      ISSUE MQI GET REQUEST USING REENTRANT FORM OF CALL MACRO
*
*      HCONN WAS SET BY PREVIOUS MQCONN REQUEST
*      HOBJ WAS SET BY PREVIOUS MQOPEN REQUEST
*
      CALL MQGET,                X
          (HCONN,                X
           HOBJ,                  X
           WMD,                    X
           WGMO,                   X
           BUFFLEN,                 X
           BUFFER,                  X
           DATALEN,                X
           COMPCODE,                 X
           REASON),                 X
          VL,MF=(E,CALLLST)
*
      LA   R5,MQCC_OK
      C    R5,COMPCODE
      BNE BADCALL
*
      :
BADCALL DS   0H
:

```

```

*
*      CONSTANTS
*
*      CMQMDA DSECT=NO,LIST=YES
*      CMQGMOA DSECT=NO,LIST=YES
*      CMQA
*
*      WORKING STORAGE DSECT
*
WORKSTG DSECT
*
COMPCODE DS F
REASON   DS F
BUFFLEN  DS F
DATALEN  DS F
OPTIONS  DS F
HCONN    DS F
HOBJ     DS F
*
BUFFER   DS CL80
BUFFER_LEN EQU *-BUFFER
*
WMD      CMQMDA DSECT=NO,LIST=NO
WGMO     CMQGMOA DSECT=NO,LIST=NO
*
CALLLST  CALL , (0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0),VL,MF=L
*

```



```
⋮  
END
```

### ***Bekleme seçeneğini kullanarak ileti alınması***

Bu örnek, MQGET çağrısının bekleme seçeneğinin nasıl kullanılacağını gösterir.

Bu kod kesilen iletileri kabul eder. Bu alma, IBM MQ ile verilen örnek uygulamalardan alınmıyor.

```
⋮  
* CONNECT TO QUEUE MANAGER  
CONN DS 0H  
⋮  
* OPEN A QUEUE FOR GET  
OPEN DS 0H  
⋮  
* R4,R5,R6,R7 = WORK REGISTER.  
GET DS 0H  
LA R4,MQMD SET UP ADDRESSES AND  
LA R5,MQMD_LENGTH LENGTH FOR USE BY MVCL  
LA R6,WMD INSTRUCTION, AS MQMD IS  
LA R7,WMD_LENGTH OVER 256 BYES LONG.  
MVCL R6,R4 INITIALIZE WORKING VERSION  
* OF MESSAGE DESCRIPTOR  
  
*  
MVC WGMO_AREA,MQGMO_AREA INITIALIZE WORKING MQGMO  
L R5,=AL4(MQGMO_WAIT)  
A R5,=AL4(MQGMO_ACCEPT_TRUNCATED_MSG)  
ST R5,WGMO_OPTIONS  
MVC WGMO_WAITINTERVAL,TWO_MINUTES WAIT UP TO TWO  
MINUTES BEFORE  
FAILING THE  
CALL  
  
*  
LA R5,BUFFER_LEN RETRIEVE THE BUFFER LENGTH  
ST R5,BUFFLEN AND SAVE IT FOR MQM USE  
  
*  
* ISSUE MQI GET REQUEST USING REENTRANT FORM OF CALL MACRO  
*  
* HCONN WAS SET BY PREVIOUS MQCONN REQUEST  
* HOBJ WAS SET BY PREVIOUS MQOPEN REQUEST  
*  
CALL MQGET, X  
(HCONN, X  
HOBJ, X  
WMD, X  
WGMO, X  
BUFFLEN, X  
BUFFER, X  
DATALEN, X  
COMP CODE, X  
REASON), X  
VL,MF=(E,CALLST)  
  
*  
LA R5,MQCC_OK DID THE MQGET REQUEST  
C R5,COMP CODE WORK OK?  
BE GETOK YES, SO GO AND PROCESS.  
LA R5,MQCC_WARNING NO, SO CHECK FOR A WARNING.  
C R5,COMP CODE IS THIS A WARNING?  
BE CHECK_W YES, SO CHECK THE REASON.  
  
*  
LA R5,MQRC_NO_MSG_AVAILABLE IT MUST BE AN ERROR.  
IS IT DUE TO AN EMPTY  
C R5,REASON QUEUE?  
BE NOMSG YES, SO HANDLE THE ERROR  
B BADCALL NO, SO GO TO ERROR ROUTINE  
  
*  
CHECK_W DS 0H  
LA R5,MQRC_TRUNCATED_MSG_ACCEPTED IS THIS A  
TRUNCATED  
C R5,REASON MESSAGE?  
BE GETOK YES, SO GO AND PROCESS.  
B BADCALL NO, SOME OTHER WARNING  
  
*  
NOMSG DS 0H  
⋮
```

```
GETOK DS 0H
      :
```

```
BADCALL DS 0H
      :
```

```
*
*      CONSTANTS
*
*      CMQMDA DSECT=NO,LIST=YES
*      CMQGMOA DSECT=NO,LIST=YES
*      CMQA
*
*      TWO_MINUTES DC F'120000'      GET WAIT INTERVAL
*
*      WORKING STORAGE DSECT
```

```
*
*      WORKSTG DSECT
*
*      COMPCODE DS F
*      REASON DS F
*      BUFFLEN DS F
*      DATALEN DS F
*      OPTIONS DS F
*      HCONN DS F
*      HOBJ DS F
*
*      BUFFER DS CL80
*      BUFFER_LEN EQU *-BUFFER
*
*      WMD CMQMDA DSECT=NO,LIST=NO
*      WGMO CMQGMOA DSECT=NO,LIST=NO
*
*      CALLLST CALL ,(0,0,0,0,0,0,0,0,0,0,0,0),VL,MF=L
*
*      :
*      END
```

### ***Sinyal kullanarak ileti alınıyor***

Bu örnek, bir sinyal ayarlamak için MQGET çağrısının nasıl kullanılacağını gösterir. Böylece, kuyruğa uygun bir ileti geldiğinde size bildirim gönderilir.

Bu alma, IBM MQ ile verilen örnek uygulamalardan alınmıştır.

```
:
*
*      CONNECT TO QUEUE MANAGER
*
*      CONN DS 0H
*      :
*
*      OPEN A QUEUE FOR GET
*
*      OPEN DS 0H
*      :
*
*      R4,R5,R6,R7 = WORK REGISTER.
*
*      GET DS 0H
*      LA R4,MQMD          SET UP ADDRESSES AND
*      LA R5,MQMD_LENGTH  LENGTH FOR USE BY MVCL
*      LA R6,WMD           INSTRUCTION, AS MQMD IS
*      LA R7,WMD_LENGTH   OVER 256 BYES LONG.
*      MVCL R6,R4         INITIALIZE WORKING VERSION
*                          OF MESSAGE DESCRIPTOR
*
```

```
*
*      MVC WGMO_AREA,MQGMO_AREA  INITIALIZE WORKING MQGMO
*      LA R5,MQGMO_SET_SIGNAL
*      ST R5,WGMO_OPTIONS
*      MVC WGMO_WAITINTERVAL,FIVE_MINUTES  WAIT UP TO FIVE
*                                          MINUTES BEFORE
```

```

*                                     FAILING THE CALL
*
XC  SIG_ECB,SIG_ECB  CLEAR THE ECB
LA  R5,SIG_ECB      GET THE ADDRESS OF THE ECB
ST  R5,WGMO_SIGNAL1 AND PUT IT IN THE WORKING
*                                     MQGMO
*
LA  R5,BUFFER_LEN   RETRIEVE THE BUFFER LENGTH
ST  R5,BUFFLEN      AND SAVE IT FOR MQM USE
*
*
*   ISSUE MQI GET REQUEST USING REENTRANT FORM OF CALL MACRO
*
*   HCONN WAS SET BY PREVIOUS MQCONN REQUEST
*   HOBJ WAS SET BY PREVIOUS MQOPEN REQUEST
*
CALL  MQGET,          X
      (HCONN,         X
      HOBJ,           X
      WMD,            X
      WGMO,           X
      BUFFLEN,        X
      BUFFER,         X
      DATALEN,       X
      COMPCODE,       X
      REASON),        X
      VL,MF=(E,CALLST)
*
LA  R5,MQCC_OK       DID THE MQGET REQUEST
C   R5,COMPCODE      WORK OK?
BE  GETOK            YES, SO GO AND PROCESS.
LA  R5,MQCC_WARNING NO, SO CHECK FOR A WARNING.
C   R5,COMPCODE      IS THIS A WARNING?
BE  CHECK_W          YES, SO CHECK THE REASON.
B   BADCALL          NO, SO GO TO ERROR ROUTINE
*

```

```

CHECK_W DS 0H
LA R5,MQRC_SIGNAL_REQUEST_ACCEPTED
C R5,REASON SIGNAL REQUEST SIGNAL SET?
BNE BADCALL NO, SOME ERROR OCCURRED
B DOWORK YES, SO DO SOMETHING
*                                     ELSE
*
CHECKSIG DS 0H
CLC SIG_ECB+1(3),=AL3(MQEC_MSG_ARRIVED)
*                                     IS A MESSAGE AVAILABLE?
BE GET YES, SO GO AND GET IT
*
CLC SIG_ECB+1(3),=AL3(MQEC_WAIT_INTERVAL_EXPIRED)
*                                     HAVE WE WAITED LONG ENOUGH?
BE NOMSG YES, SO SAY NO MSG AVAILABLE
B BADCALL IF IT'S ANYTHING ELSE
*                                     GO TO ERROR ROUTINE.
*
DOWORK DS 0H
:
TM SIG_ECB,X'40' HAS THE SIGNAL ECB BEEN POSTED?
BO CHECKSIG YES, SO GO AND CHECK WHY
B DOWORK NO, SO GO AND DO MORE WORK
*
NOMSG DS 0H
:
GETOK DS 0H
:
BADCALL DS 0H
:
*
*   CONSTANTS
*
CMQMDA DSECT=NO,LIST=YES
CMQGMOA DSECT=NO,LIST=YES
CMQA
*
FIVE_MINUTES DC F'300000' GET SIGNAL INTERVAL
*
*   WORKING STORAGE DSECT
*
WORKSTG DSECT

```



```

MVC M00_MSG,M01_MSG1 Invalid request
BR R6 Return to caller

```

\*

```

UPDINHBT DS 0H
MVC UPDTYPE,CINHIBIT Indicate action type
LA R0,MQQA_GET_INHIBITED Load attribute value
ST R0,INTATTRS+0 Place in field
LA R0,MQQA_PUT_INHIBITED Load attribute value
ST R0,INTATTRS+4 Place in field
B UPDCALL Go and do call

```

\*

```

UPDALLOW DS 0H
MVC UPDTYPE,CALLOWED Indicate action type
LA R0,MQQA_GET_ALLOWED Load attribute value
ST R0,INTATTRS+0 Place in field
LA R0,MQQA_PUT_ALLOWED Load attribute value
ST R0,INTATTRS+4 Place in field
B UPDCALL Go and do call

```

\*

```

UPDCALL DS 0H
CALL MQSET, C
(HCONN, C
HOBJ, C
SELECTORCOUNT, C
SELECTOR, C
INTATTRCOUNT, C
INTATTRS, C
CHARATTRLENGTH, C
CHARATTRS, C
COMPCODE, C
REASON), C
VL,MF=(E,CALLLIST)

```

\*

```

LA R0,MQCC_OK Load expected compcode
C R0,COMPCODE Was set successful?
:
```

```

* SECTION NAME : INQUIRE *
* FUNCTION : Inquires on the objects attributes *
* CALLED BY : PROCESS *
* CALLS : OPEN, CLOSE, CODES *
* RETURN : To Register 6 *
INQUIRE DS 0H
:
```

\* Initialize the variables for the inquire call

\*

```

SR R0,R0 Clear register zero
ST R0,CHARATTRLENGTH Set char length to zero
LA R0,2 Load to set
ST R0,SELECTORCOUNT selectors add
ST R0,INTATTRCOUNT integer attributes

```

\*

```

LA R0,MQIA_INHIBIT_GET Load attribute value
ST R0,SELECTOR+0 Place in field
LA R0,MQIA_INHIBIT_PUT Load attribute value
ST R0,SELECTOR+4 Place in field
CALL MQINQ, C
(HCONN, C
HOBJ, C
SELECTORCOUNT, C
SELECTOR, C
INTATTRCOUNT, C
INTATTRS, C
CHARATTRLENGTH, C
CHARATTRS, C
COMPCODE, C
REASON), C
VL,MF=(E,CALLLIST)
LA R0,MQCC_OK Load expected compcode
C R0,COMPCODE Was inquire successful?
:
```

## PL/I örnekleri

PL/I kullanımı yalnızca z/OS tarafından desteklenir. Bu konular grubu, PL/I örneklerini kullanarak teknikleri gösterir.

### ***Kuyruk yöneticiyle bağlantı kuruluyor***

Bu örnek, bir programı z/OS toplu kipte bir kuyruk yöneticisine bağlamak için MQCONN çağrısının nasıl kullanılacağını gösterir.

Bu alma, IBM MQ ile verilen örnek uygulamalardan alınmıştır.

```
%INCLUDE SYSLIB(CMQP);
%INCLUDE SYSLIB(CMQEPP);
:
/*****
/* STRUCTURE BASED ON PARAMETER INPUT AREA (PARAM) */
*****/
DCL 1 INPUT_PARAM      BASED(ADDR(PARAM)),
    2 PARAM_LENGTH     FIXED BIN(15),
    2 PARAM_MQMNNAME   CHAR(48);
:
/*****
/* WORKING STORAGE DECLARATIONS */
*****/
DCL MQMNNAME           CHAR(48);
DCL COMPCODE           BINARY FIXED (31);
DCL REASON              BINARY FIXED (31);
DCL HCONN              BINARY FIXED (31);
:
/*****
/* COPY QUEUE MANAGER NAME PARAMETER */
/* TO LOCAL STORAGE */
*****/
MQMNNAME = ' ';
MQMNNAME = SUBSTR(PARAM_MQMNNAME,1,PARAM_LENGTH);
:
/*****
/* CONNECT FROM THE QUEUE MANAGER */
*****/
CALL MQCONN (MQMNNAME, /* MQM SYSTEM NAME */
            HCONN, /* CONNECTION HANDLE */
            COMPCODE, /* COMPLETION CODE */
            REASON); /* REASON CODE */
:
/*****
/* TEST THE COMPLETION CODE OF THE CONNECT CALL. */
/* IF THE CALL HAS FAILED ISSUE AN ERROR MESSAGE */
/* SHOWING THE COMPLETION CODE AND THE REASON CODE. */
*****/
IF COMPCODE = MQCC_OK
  THEN DO;
  :
  CALL ERROR_ROUTINE;
END;
```

### ***Kuyruk yöneticisinden bağlantı kesiliyor***

This example demonstrates how to use the MQDISC call to disconnect a program from a queue manager in z/OS batch.

Bu alma, IBM MQ ile verilen örnek uygulamalardan alınmıştır.

```
%INCLUDE SYSLIB(CMQP);
%INCLUDE SYSLIB(CMQEPP);
:
/*****
/* WORKING STORAGE DECLARATIONS */
*****/
DCL COMPCODE           BINARY FIXED (31);
DCL REASON              BINARY FIXED (31);
DCL HCONN              BINARY FIXED (31);
:
/*****
/* DISCONNECT FROM THE QUEUE MANAGER */
*****/
```

```

/*****/
CALL MQDISC (HCONN, /* CONNECTION HANDLE */
             COMPCODE, /* COMPLETION CODE */
             REASON); /* REASON CODE */
/*****/
/* TEST THE COMPLETION CODE OF THE DISCONNECT CALL. */
/* IF THE CALL HAS FAILED ISSUE AN ERROR MESSAGE */
/* SHOWING THE COMPLETION CODE AND THE REASON CODE. */
/*****/
IF COMPCODE = MQCC_OK
  THEN DO;
  :
  CALL ERROR_ROUTINE;
END;

```

### ***Dinamik kuyruk yaratılması***

Bu örnek, dinamik bir kuyruk yaratmak için MQOPEN çağrısının nasıl kullanılacağını gösterir.

Bu alma, IBM MQ ile verilen örnek uygulamalardan alınmıştır.

```

%INCLUDE SYSLIB(CMQP);
%INCLUDE SYSLIB(CMQEPP);
:
/*****/
/* WORKING STORAGE DECLARATIONS */
/*****/
DCL COMPCODE          BINARY FIXED (31);
DCL REASON            BINARY FIXED (31);
DCL HCONN            BINARY FIXED (31);
DCL HOBJ             BINARY FIXED (31);
DCL OPTIONS          BINARY FIXED (31);
:
DCL MODEL_QUEUE_NAME CHAR(48) INIT('PL1.REPLY.MODEL');
DCL DYNAMIC_NAME_PREFIX CHAR(48) INIT('PL1.TEMPQ.*');
DCL DYNAMIC_QUEUE_NAME CHAR(48) INIT(' ');
:
/*****/
/* LOCAL COPY OF OBJECT DESCRIPTOR */
/*****/
DCL 1 LMQOD LIKE MQOD;
:
/*****/
/* SET UP OBJECT DESCRIPTOR FOR OPEN OF REPLY QUEUE */
/*****/
LMQOD.OBJECTTYPE =MQOT_Q;
LMQOD.OBJECTNAME = MODEL_QUEUE_NAME;
LMQOD.DYNAMICQNAME = DYNAMIC_NAME_PREFIX;
OPTIONS = MQOO_INPUT_EXCLUSIVE;

CALL MQOPEN (HCONN,
            LMQOD,
            OPTIONS,
            HOBJ,
            COMPCODE,
            REASON);

/*****/
/* TEST THE COMPLETION CODE OF THE OPEN CALL. */
/* IF THE CALL HAS FAILED ISSUE AN ERROR MESSAGE */
/* SHOWING THE COMPLETION CODE AND THE REASON CODE. */
/* IF THE CALL HAS SUCCEEDED THEN EXTRACT THE NAME OF */
/* THE NEWLY CREATED DYNAMIC QUEUE FROM THE OBJECT */
/* DESCRIPTOR. */
/*****/
IF COMPCODE = MQCC_OK
  THEN DO;
  :
  CALL ERROR_ROUTINE;
END;
ELSE
  DYNAMIC_QUEUE_NAME = LMQOD_OBJECTNAME;

```

### ***Varolan bir kuyruğun açılması***

Bu örnek, varolan bir kuyruğu açmak için MQOPEN çağrısının nasıl kullanılacağını gösterir.

Bu alma, IBM MQ ile verilen örnek uygulamalardan alınmıştır.

```
%INCLUDE SYSLIB(CMQP);
%INCLUDE SYSLIB(CMQEPP);
:
/*****/
/* WORKING STORAGE DECLARATIONS */
/*****/
DCL COMPCODE          BINARY FIXED (31);
DCL REASON            BINARY FIXED (31);
DCL HCONN            BINARY FIXED (31);
DCL HOBJ            BINARY FIXED (31);
DCL OPTIONS          BINARY FIXED (31);
:
DCL QUEUE_NAME        CHAR(48) INIT('PL1.LOCAL.QUEUE');
:
/*****/
/* LOCAL COPY OF OBJECT DESCRIPTOR */
/*****/
DCL 1 LMQOD LIKE MQOD;
:
/*****/
/* SET UP OBJECT DESCRIPTOR FOR OPEN OF REPLY QUEUE */
/*****/
LMQOD.OBJECTTYPE = MQOT_Q;
LMQOD.OBJECTNAME = QUEUE_NAME;
OPTIONS = MQOO_INPUT_EXCLUSIVE;

CALL MQOPEN (HCONN,
             LMQOD,
             OPTIONS,
             HOBJ,
             COMPCODE,
             REASON);

/*****/
/* TEST THE COMPLETION CODE OF THE OPEN CALL. */
/* IF THE CALL HAS FAILED ISSUE AN ERROR MESSAGE */
/* SHOWING THE COMPLETION CODE AND THE REASON CODE. */
/*****/
IF COMPCODE /= MQCC_OK
  THEN DO;
  :
  CALL ERROR_ROUTINE;
END;
```

### ***Kuyruğun Kapatılması***

Bu örnek, MQCLOSE çağrısının nasıl kullanılacağını gösterir.

Bu alma, IBM MQ ile verilen örnek uygulamalardan alınmıştır.

```
%INCLUDE SYSLIB(CMQP);
%INCLUDE SYSLIB(CMQEPP);
:
/*****/
/* WORKING STORAGE DECLARATIONS */
/*****/
DCL COMPCODE          BINARY FIXED (31);
DCL REASON            BINARY FIXED (31);
DCL HCONN            BINARY FIXED (31);
DCL HOBJ            BINARY FIXED (31);
DCL OPTIONS          BINARY FIXED (31);
:
/*****/
/* SET CLOSE OPTIONS */
/*****/
OPTIONS=MQCO_NONE;

/*****/
/* CLOSE QUEUE */
/*****/
CALL MQCLOSE (HCONN, /* CONNECTION HANDLE */
             HOBJ, /* OBJECT HANDLE */
             OPTIONS, /* CLOSE OPTIONS */
             COMPCODE, /* COMPLETION CODE */
             REASON); /* REASON CODE */
```



```

/*****
/* TEST THE COMPLETION CODE OF THE CLOSE CALL.          */
/* IF THE CALL HAS FAILED ISSUE AN ERROR MESSAGE        */
/* SHOWING THE COMPLETION CODE AND THE REASON CODE.     */
*****/
IF COMPCODE = MQCC_OK
  THEN DO;
  :
  CALL ERROR_ROUTINE;
END;

```

## **MQPUT kullanarak ileti koyma**

Bu örnek, bağlamı kullanarak MQPUT çağrısının nasıl kullanılacağını gösterir.

Bu alma, IBM MQ ile verilen örnek uygulamalardan alınmıştır.

```

%INCLUDE SYSLIB(CMQP);
%INCLUDE SYSLIB(CMQEPP);
:
/*****
/* WORKING STORAGE DECLARATIONS                          */
*****/
DCL COMPCODE          BINARY FIXED (31);
DCL REASON            BINARY FIXED (31);
DCL HCONN             BINARY FIXED (31);
DCL HOBJ              BINARY FIXED (31);
DCL OPTIONS           BINARY FIXED (31);
DCL BUFFLEN          BINARY FIXED (31);
DCL BUFFER            CHAR(80);
:
DCL PL1_TEST_MESSAGE CHAR(80)
INIT('***** THIS IS A TEST MESSAGE *****');
:
/*****
/* LOCAL COPY OF MESSAGE DESCRIPTOR                      */
/* AND PUT MESSAGE OPTIONS                              */
*****/
DCL 1 LMQMD LIKE MQMD;
DCL 1 LMQPMO LIKE MQPMO;
:
/*****
/* SET UP MESSAGE DESCRIPTOR                            */
*****/
LMQMD.MSGTYPE = MQMT_DATAGRAM;
LMQMD.PRIORITY = 1;
LMQMD.PERSISTENCE = MQPER_PERSISTENT;
LMQMD.REPLYTOQ = ' ';
LMQMD.REPLYTOQMGR = ' ';
LMQMD.MSGID = MQMI_NONE;
LMQMD.CORRELID = MQCI_NONE;

/*****
/* SET UP PUT MESSAGE OPTIONS                            */
*****/
LMQPMO.OPTIONS = MQPMO_NO_SYNCPOINT;

/*****
/* SET UP LENGTH OF MESSAGE BUFFER AND THE MESSAGE     */
*****/
BUFFLEN = LENGTH(BUFFER);
BUFFER = PL1_TEST_MESSAGE;
/*****
/*
/* HCONN WAS SET BY PREVIOUS MQCONN REQUEST.           */
/* HOBJ WAS SET BY PREVIOUS MQOPEN REQUEST.            */
/*
*****/
CALL MQPUT (HCONN,
           HOBJ,
           LMQMD,
           LMQPMO,
           BUFFLEN,
           BUFFER,
           COMPCODE,
           REASON);

```

```

/*****/
/* TEST THE COMPLETION CODE OF THE PUT CALL. */
/* IF THE CALL HAS FAILED ISSUE AN ERROR MESSAGE */
/* SHOWING THE COMPLETION CODE AND THE REASON CODE. */
/*****/
      IF COMPCODE = MQCC_OK
        THEN DO;
          :
          :
          CALL ERROR_ROUTINE;
        END;

```

## ***MQPUT1 kullanılarak ileti konması***

Bu örnek, MQPUT1 çağrısının nasıl kullanılacağını gösterir.

Bu alma, IBM MQ ile verilen örnek uygulamalardan alınmıyor.

```

%INCLUDE SYSLIB(CMQEPP);
%INCLUDE SYSLIB(CMQP);
:
/*****/
/* WORKING STORAGE DECLARATIONS */
/*****/
DCL COMPCODE      BINARY FIXED (31);
DCL REASON        BINARY FIXED (31);
DCL HCONN         BINARY FIXED (31);
DCL OPTIONS       BINARY FIXED (31);
DCL BUFFLEN       BINARY FIXED (31);
DCL BUFFER        CHAR(80);
:
DCL REPLY_TO_QUEUE CHAR(48) INIT('PL1.REPLY.QUEUE');
DCL QUEUE_NAME     CHAR(48) INIT('PL1.LOCAL.QUEUE');
DCL PL1_TEST_MESSAGE CHAR(80)
INIT('***** THIS IS ANOTHER TEST MESSAGE *****');
:
/*****/
/* LOCAL COPY OF OBJECT DESCRIPTOR, MESSAGE DESCRIPTOR */
/* AND PUT MESSAGE OPTIONS */
/*****/
DCL 1 LMQOD LIKE MQOD;
DCL 1 LMQMD LIKE MQMD;
DCL 1 LMQPMO LIKE MQPMO;
:
/*****/
/* SET UP OBJECT DESCRIPTOR AS REQUIRED. */
/*****/
LMQOD.OBJECTTYPE = MQOT_Q;
LMQOD.OBJECTNAME = QUEUE_NAME;

/*****/
/* SET UP MESSAGE DESCRIPTOR AS REQUIRED. */
/*****/
LMQMD.MSGTYPE = MQMT_REQUEST;
LMQMD.PRIORITY = 5;
LMQMD.PERSISTENCE = MQPER_PERSISTENT;
LMQMD.REPLYTOQ = REPLY_TO_QUEUE;
LMQMD.REPLYTOQMGR = 'T';
LMQMD.MSGID = MQMI_NONE;
LMQMD.CORRELID = MQCI_NONE;

/*****/
/* SET UP PUT MESSAGE OPTIONS AS REQUIRED */
/*****/
LMQPMO.OPTIONS = MQPMO_NO_SYNCPOINT;

/*****/
/* SET UP LENGTH OF MESSAGE BUFFER AND THE MESSAGE */
/*****/
BUFFLEN = LENGTH(BUFFER);
BUFFER = PL1_TEST_MESSAGE;

CALL MQPUT1 (HCONN,
             LMQOD,
             LMQMD,
             LMQPMO,

```

```

BUFFLEN,
BUFFER,
COMPCODE,
REASON);

/*****
/* TEST THE COMPLETION CODE OF THE PUT1 CALL.          */
/* IF THE CALL HAS FAILED ISSUE AN ERROR MESSAGE SHOWING */
/* THE COMPLETION CODE AND THE REASON CODE.          */
*****/
IF COMPCODE = MQCC_OK
THEN DO;
:
CALL ERROR_ROUTINE;
END;

```

## İleti alınıyor

Bu örnek, bir iletiyi kuyruktan kaldırmak için MQGET çağrısının nasıl kullanılacağını gösterir.

Bu alma, IBM MQ ile verilen örnek uygulamalardan alınmıştır.

```

%INCLUDE SYSLIB(CMQP);
%INCLUDE SYSLIB(CMQEPP);
:
/*****
/* WORKING STORAGE DECLARATIONS                      */
*****/
DCL COMPCODE      BINARY FIXED (31);
DCL REASON        BINARY FIXED (31);
DCL HCONN        BINARY FIXED (31);
DCL HOBJ         BINARY FIXED (31);
DCL BUFFLEN      BINARY FIXED (31);
DCL DATALEN     BINARY FIXED (31);
DCL BUFFER       CHAR(80);
:

/*****
/* LOCAL COPY OF MESSAGE DESCRIPTOR AND              */
/* GET MESSAGE OPTIONS                               */
*****/
DCL 1 LMQMD LIKE MQMD;
DCL 1 LMQGMO LIKE MQGMO;
:

/*****
/* SET UP MESSAGE DESCRIPTOR AS REQUIRED.             */
/* MSGID AND CORRELID IN MQMD SET TO NULLS SO FIRST */
/* AVAILABLE MESSAGE WILL BE RETRIEVED.             */
*****/
LMQMD.MSGID = MQMI_NONE;
LMQMD.CORRELID = MQCI_NONE;

/*****
/* SET UP GET MESSAGE OPTIONS AS REQUIRED.           */
*****/
LMQGMO.OPTIONS = MQGMO_NO_SYNCPOINT;

/*****
/* SET UP LENGTH OF MESSAGE BUFFER.                 */
*****/
BUFFLEN = LENGTH(BUFFER);

/*****
/*
/* HCONN WAS SET BY PREVIOUS MQCONN REQUEST.        */
/* HOBJ WAS SET BY PREVIOUS MQOPEN REQUEST.         */
/*
*****/

CALL MQGET (HCONN,
           HOBJ,
           LMQMD,
           LMQGMO,
           BUFFERLEN,
           BUFFER,
           DATALEN,
           COMPCODE,
           REASON);

```

```

/*****/
/* TEST THE COMPLETION CODE OF THE GET CALL. */
/* IF THE CALL HAS FAILED ISSUE AN ERROR MESSAGE */
/* SHOWING THE COMPLETION CODE AND THE REASON CODE. */
/*****/
IF COMPCODE = MQCC_OK
  THEN DO;
  :
  CALL ERROR_ROUTINE;
END;

```

## Bekleme seçeneğini kullanarak ileti alınması

Bu örnek, bekleme seçeneği ile MQGET çağrısının nasıl kullanılacağını ve kesilmiş iletilerin nasıl kabul edileceğini gösterir.

Bu alma, IBM MQ ile verilen örnek uygulamalardan alınmıştır.

```

%INCLUDE SYSLIB(CMQP);
%INCLUDE SYSLIB(CMQEPP);
:
/*****/
/* WORKING STORAGE DECLARATIONS */
/*****/
DCL COMPCODE          BINARY FIXED (31);
DCL REASON            BINARY FIXED (31);
DCL HCONN             BINARY FIXED (31);
DCL HOBJ              BINARY FIXED (31);
DCL BUFFLEN          BINARY FIXED (31);
DCL DATALEN         BINARY FIXED (31);
DCL BUFFER            CHAR(80);
:
/*****/
/* LOCAL COPY OF MESSAGE DESCRIPTOR AND GET MESSAGE */
/* OPTIONS */
/*****/
DCL 1 LMQMD LIKE MQMD;
DCL 1 LMQGMO LIKE MQGMO;
:
/*****/
/* SET UP MESSAGE DESCRIPTOR AS REQUIRED. */
/* MSGID AND CORRELID IN MQMD SET TO NULLS SO FIRST */
/* AVAILABLE MESSAGE WILL BE RETRIEVED. */
/*****/
LMQMD.MSGID = MQMI_NONE;
LMQMD.CORRELID = MQCI_NONE;

/*****/
/* SET UP GET MESSAGE OPTIONS AS REQUIRED. */
/* WAIT INTERVAL SET TO ONE MINUTE. */
/*****/
LMQGMO.OPTIONS = MQGMO_WAIT +
                 MQGMO_ACCEPT_TRUNCATED_MSG +
                 MQGMO_NO_SYNCPOINT;
LMQGMO.WAITINTERVAL=60000;

/*****/
/* SET UP LENGTH OF MESSAGE BUFFER. */
/*****/
BUFFLEN = LENGTH(BUFFER);

/*****/
/*
/* HCONN WAS SET BY PREVIOUS MQCONN REQUEST. */
/* HOBJ WAS SET BY PREVIOUS MQOPEN REQUEST. */
/*
/*****/

CALL MQGET (HCONN,
            HOBJ,
            LMQMD,
            LMQGMO,
            BUFFLEN,
            BUFFER,
            DATALEN,
            COMPCODE,

```

```

REASON);

/*****
/* TEST THE COMPLETION CODE OF THE GET CALL.          */
/* TAKE APPROPRIATE ACTION BASED ON COMPLETION CODE AND */
/* REASON CODE.                                       */
*****/

SELECT (COMPCODE);
  WHEN (MQCC_OK) DO; /* GET WAS SUCCESSFUL */
  :
  END;
  WHEN (MQCC_WARNING) DO;
  IF REASON = MQRC_TRUNCATED_MSG_ACCEPTED
  THEN DO; /* GET WAS SUCCESSFUL */
  :
  END;
  ELSE DO;
  :
  CALL ERROR_ROUTINE;
  END;
  END;
  WHEN (MQCC_FAILED) DO;
  :
  CALL ERROR_ROUTINE;
  END;
  END;
  OTHERWISE;
END;

```

### ***Sinyal kullanarak ileti alınıyor***

MQGET çağrısının sinyalizasyon ile nasıl kullanılacağını gösteren bir kod çıkarması.

**Sinyalizasyon yalnızca IBM MQ for z/OS** ile kullanılabilir.

Bu alma, IBM MQ ile verilen örnek uygulamalardan alınmıştır.

```

%INCLUDE SYSLIB(CMQP);
%INCLUDE SYSLIB(CMQEPP);
:
/*****
/* WORKING STORAGE DECLARATIONS          */
*****/
DCL COMPCODE          BINARY FIXED (31);
DCL REASON            BINARY FIXED (31);
DCL HCONN             BINARY FIXED (31);
DCL HOBJ              BINARY FIXED (31);
DCL DATALEN         BINARY FIXED (31);
DCL BUFFLEN          BINARY FIXED (31);
DCL BUFFER            CHAR(80);
:
DCL ECB_FIXED          FIXED BIN(31);
DCL 1 ECB_OVERLAY     BASED(ADDR(ECB_FIXED)),
      3 ECB_WAIT      BIT,
      3 ECB_POSTED    BIT,
      3 ECB_FLAG3_8   BIT(6),
      3 ECB_CODE      PIC'999';
:
/*****
/* LOCAL COPY OF MESSAGE DESCRIPTOR AND GET MESSAGE */
/* OPTIONS                                           */
*****/
DCL 1 LMQMD LIKE MQMD;
DCL 1 LMQGMO LIKE MQGMO;
:
/*****
/* CLEAR ECB FIELD.                                */
*****/
ECB_FIXED = 0;
:
/*****
/* SET UP MESSAGE DESCRIPTOR AS REQUIRED.          */
/* MSGID AND CORRELID IN MQMD SET TO NULLS SO FIRST */
/* AVAILABLE MESSAGE WILL BE RETRIEVED.          */
*****/
LMQMD.MSGID = MQMI_NONE;
LMQMD.CORRELID = MQCI_NONE;
/*****

```

```

/* SET UP GET MESSAGE OPTIONS AS REQUIRED.          */
/* WAIT INTERVAL SET TO ONE MINUTE.              */
/*****
  LMQGMO.OPTIONS = MQGMO_SET_SIGNAL +
                  MQGMO_NO_SYNCPOINT;
  LMQGMO.WAITINTERVAL=60000;
  LMQGMO.SIGNAL1 = ADDR(ECB_FIXED);
*****/

```

```

/*****
/* SET UP LENGTH OF MESSAGE BUFFER.              */
/* CALL MESSAGE RETRIEVAL ROUTINE.              */
/*****
  BUFFLEN = LENGTH(BUFFER);
  CALL GET_MSG;
*****/

```

```

/*****
/* TEST THE COMPLETION CODE OF THE GET CALL.    */
/* TAKE APPROPRIATE ACTION BASED ON COMPLETION CODE AND REASON CODE.
/*****

```

```

SELECT;
  WHEN ((COMPCODE = MQCC_OK) &
        (REASON = MQCC_NONE)) DO
    :
    CALL MSG_ROUTINE;
    :
  END;
  WHEN ((COMPCODE = MQCC_WARNING) &
        (REASON = MQRC_SIGNAL_REQUEST_ACCEPTED)) DO;
    :
    CALL DO_WORK;
    :
  END;
  WHEN ((COMPCODE = MQCC_FAILED) &
        (REASON = MQRC_SIGNAL_OUTSTANDING)) DO;
    :
    CALL DO_WORK;
    :
  END;
  OTHERWISE DO;          /* FAILURE CASE */
/*****
/* ISSUE AN ERROR MESSAGE SHOWING THE COMPLETION CODE
/* AND THE REASON CODE.
/*****
  CALL ERROR_ROUTINE;
  :
  END;
END;
:

```

```

DO_WORK: PROC;
:
  IF ECB_POSTED
  THEN DO;
    SELECT(ECB_CODE);
    WHEN(MQEC_MSG_ARRIVED) DO;
      :
      CALL GET_MSG;
      :
    END;
    WHEN(MQEC_WAIT_INTERVAL_EXPIRED) DO;
      :
      CALL NO_MSG;
      :
    END;
    OTHERWISE DO;          /* FAILURE CASE */
/*****
/* ISSUE AN ERROR MESSAGE SHOWING THE COMPLETION CODE
/* AND THE REASON CODE.
/*****
  CALL ERROR_ROUTINE;
  :
  END;

```

```

        END;
        END;
        :
END DO_WORK;
GET_MSG: PROC;

```

```

/*****/
/*
/* HCONN WAS SET BY PREVIOUS MQCONN REQUEST.          */
/* HOBJ WAS SET BY PREVIOUS MQOPEN REQUEST.           */
/* MD AND GMO SET UP AS REQUIRED.                      */
/*                                                     */
/*****/

        CALL MQGET (HCONN,
                    HOBJ,
                    LMQMD,
                    LMQGMO,
                    BUFLLEN,
                    BUFFER,
                    DATALEN,
                    COMPCODE,
                    REASON);

END GET_MSG;

NO_MSG: PROC;
:
END NO_MSG;

```

### ***Bir nesnenin özniteliklerinin sorulmasına neden oluyor***

Bu örnek, bir kuyruğun öznitelikleriyle ilgili olarak sorgulamak için MQINQ çağrısının nasıl kullanılacağını gösterir.

Bu alma, IBM MQ ile verilen örnek uygulamalardan alınmıyor.

```

%INCLUDE SYSLIB(CMQP);
%INCLUDE SYSLIB(CMQEPP);
:
/*****/
/* WORKING STORAGE DECLARATIONS                      */
/*****/
DCL COMPCODE          BINARY FIXED (31);
DCL REASON            BINARY FIXED (31);
DCL HCONN             BINARY FIXED (31);
DCL HOBJ              BINARY FIXED (31);
DCL OPTIONS           BINARY FIXED (31);
DCL SELECTORCOUNT   BINARY FIXED (31);
DCL INTATTRCOUNT   BINARY FIXED (31);
DCL 1 SELECTOR_TABLE,
    3 SELECTORS(5)    BINARY FIXED (31);
DCL 1 INTATTR_TABLE,
    3 INTATTRS(5)     BINARY FIXED (31);
DCL CHARATTRLENGTH   BINARY FIXED (31);
DCL CHARATTRS        CHAR(100);
:

/*****/
/* SET VARIABLES FOR INQUIRE CALL                    */
/* INQUIRE ON THE CURRENT QUEUE DEPTH                */
/*****/

        SELECTORS(01) = MQIA_CURRENT_Q_DEPTH;

        SELECTORCOUNT = 1;
        INTATTRCOUNT = 1;

        CHARATTRLENGTH = 0;
/*****/
/*
/* HCONN WAS SET BY PREVIOUS MQCONN REQUEST.          */
/* HOBJ WAS SET BY PREVIOUS MQOPEN REQUEST.           */
/*                                                     */
/*****/

```

```

CALL MQINQ (HCONN,
           HOBJ,
           SELECTORCOUNT,
           SELECTORS,
           INTATTRCOUNT,
           INTATTRS,
           CHARATTRLENGTH,
           CHARATTRS,
           COMPCODE,
           REASON);

```

```

/*****/
/* TEST THE COMPLETION CODE OF THE INQUIRE CALL.          */
/* IF THE CALL HAS FAILED ISSUE AN ERROR MESSAGE SHOWING */
/* THE COMPLETION CODE AND THE REASON CODE.              */
/*****/
IF COMPCODE = MQCC_OK
  THEN DO;
  :
  CALL ERROR_ROUTINE;
END;

```

### ***Kuyruğun Özniteliklerinin Ayarlanmasını***

Bu örnek, bir kuyruğun özniteliklerini değiştirmek için MQSET çağrısının nasıl kullanılacağını gösterir.

Bu alma, IBM MQ ile verilen örnek uygulamalardan alınmıyor.

```

%INCLUDE SYSLIB(CMQP);
%INCLUDE SYSLIB(CMQEPP);
:
/*****/
/* WORKING STORAGE DECLARATIONS                          */
/*****/
DCL COMPCODE          BINARY FIXED (31);
DCL REASON            BINARY FIXED (31);
DCL HCONN             BINARY FIXED (31);
DCL HOBJ              BINARY FIXED (31);
DCL OPTIONS           BINARY FIXED (31);
DCL SELECTORCOUNT   BINARY FIXED (31);
DCL INTATTRCOUNT    BINARY FIXED (31);
DCL 1 SELECTOR_TABLE,
   3 SELECTORS(5)      BINARY FIXED (31);
DCL 1 INTATTR_TABLE,
   3 INTATTRS(5)      BINARY FIXED (31);
DCL CHARATTRLENGTH    BINARY FIXED (31);
DCL CHARATTRS         CHAR(100);
:

/*****/
/* SET VARIABLES FOR SET CALL                            */
/* SET GET AND PUT INHIBITED                             */
/*****/

SELECTORS(01) = MQIA_INHIBIT_GET;
SELECTORS(02) = MQIA_INHIBIT_PUT;

INTATTRS(01) = MQQA_GET_INHIBITED;
INTATTRS(02) = MQQA_PUT_INHIBITED;

SELECTORCOUNT = 2;
INTATTRCOUNT  = 2;

CHARATTRLENGTH = 0;

```

```

/*****/
/*
/* HCONN WAS SET BY PREVIOUS MQCONN REQUEST.             */
/* HOBJ WAS SET BY PREVIOUS MQOPEN REQUEST.              */
/*
/*****/
CALL MQSET (HCONN,
           HOBJ,
           SELECTORCOUNT,
           SELECTORS,

```



```

                INTATTRCOUNT,
                INTATTRS,
                CHARATTRLENGTH,
                CHARATTRS,
                COMPCODE,
                REASON);

/*****
/* TEST THE COMPLETION CODE OF THE SET CALL.          */
/* IF THE CALL HAS FAILED ISSUE AN ERROR MESSAGE SHOWING */
/* THE COMPLETION CODE AND THE REASON CODE.          */
*****/
        IF COMPCODE ^= MQCC_OK
            THEN DO;
                :
                CALL ERROR_ROUTINE;
            END;

```

## Değişmezler

İş gereksinimlerinizi karşılayacak görevleri gerçekleştirmek için bu kısımdaki başvuru bilgilerini kullanın.

### IBM MQ COPY, üstbilgi, include ve module dosyaları

Bu bilgiler, genel-programlama arabirimi bilgilerini kullanır.

Bu bölümde, çeşitli programlama dilleri için MQI 'yı kullanmanıza yardımcı olacak bilgiler yer almaktadır.

#### C üstbilgi dosyaları

Üstbilgi dosyaları, MQI kullanan C uygulama programlarını yazmanıza yardımcı olmak için sağlanır.

C üstbilgi dosyaları aşağıdaki tabloda özetlenir:

Çizelge 1. C üstbilgi dosyaları-çağrı prototipleri, veri tipleri, dönüş kodları, sabitler ve yapılar					
Dosya adı	Tanım	IBM i	AIX and Linux® sistemleri	Windows	z/OS
<b>Çağrı prototipleri, veri tipleri, dönüş kodları, sabitler ve yapılar</b>					
CMQC	MQI tanımlamaları	C	C	C	C
CMQBC	MQAI tanımlamaları	C	C	C	
CMQEC	Arabirim Giriş Noktaları tanımlaması (CMQC, CMQXC ve CMQZC içerir)		C	C	
CMQCFC	PCF tanımlamaları	C	C	C	C
CMQPSC	Yayınlama/Abone Olma tanımları	C	C	C	C
CMQXC	Kanal ve çıkış tanımları	C	C	C	C
CMQZC	Kurulabilir hizmet tanımlamaları	C	C	C	
<b>Anahtar:</b> C= Dosyalar sağlandı					

#### COBOL COPY dosyaları

Çeşitli COPY dosyaları, MQI kullanan COBOL uygulama programlarını yazmanıza yardımcı olmak için sağlanmıştır.

Çizelge 2. COBOL kopya dosyaları-dönüş kodları, sabitler ve yapılar

Dosya adı	Tanım	IBM i	AIX and Linux	Windows	z/OS
<b>Dönüş kodları ve değişmezleri</b>					
CMQX	MQI tanımlamaları	V	V	V	V
CMQCFX	PCF tanımlamaları	V	V	V	V
CMQPSI	Yayınlama/Abone Olma tanımları	V	V	V	V
CMQXX	Kanal ve çıkış tanımları	V	V	V	V
<b>Yapılar</b>					
CMQAIRx	MQAIR-Kimlik doğrulama bilgileri kaydı		V L	V L	
CMQBOX	MQBO-Başlat seçenekleri	V L	V L	V L	
CMQCDX	MQCD-Kanal tanımlaması	V L	V L	V L	V L
CMQCFBFx	MQCFBF-PCF byte dizgisi süzgeç değiştirgesi	V L	V L	V L	V L
CMQCFBSX	MQCFBS-PCF byte dizilimi parametresi	V L	V L	V L	V L
CMQCFGRX	MQCFGR-PCF grup parametresi	V L	V L	V L	V L
CMQCFHx	MQCFH-PCF üstbilgisi	V L	V L	V L	V L
CMQCFIFX	MQCFIF-PCF tamsayı süzgeç değiştirgesi	V L	V L	V L	V L
CMQCFIX	MQCFIL-PCF tamsayı listesi değiştirgesi	V L	V L	V L	V L
CMQCFINX	MQCFIN-PCF tamsayı değiştirgesi	V L	V L	V L	V L
CMQCFSFx	MQCFSF-PCF dizgi süzgeç değiştirgesi	V L	V L	V L	V L
CMQCFSLX	MQCFSL-PCF dizgi listesi değiştirgesi	V L	V L	V L	V L
CMQCFSTX	MQCFST-PCF dizgisi değiştirgesi	V L	V L	V L	V L
CMQCFXLx	MQCFIL64 -PCF 64-bitlik tamsayı listesi değiştirgesi	V L	V L	V L	V L
CMQCFXNx	MQCFIN64 -PCF 64 bit tamsayı değiştirgesi	V L	V L	V L	V L
CMQCHARVX	MQCHARV-Değişken uzunluklu dizgi	V L	V L	V L	V L
CMQCIHx	MQCIH- CICS bridge üstbilgisi	V L	V L	V L	V L
CMQCNOx	MQCNO-Bağlantı seçenekleri	V L	V L	V L	V L
CMQCSPx	MQCSP-Güvenlik değiştirgeleri	V L	V L	V L	V L
CMQCXPx	MQCXP-Kanal çıkış değiştirgeleri	V L			V L
CMQDHx	MQDH-Dağıtım üstbilgisi	V L	V L	V L	V L
CMQDLHx	MQDLH-Dead-letter üstbilgisi	V L	V L	V L	V L

Çizelge 2. COBOL kopya dosyaları-dönüş kodları, sabitler ve yapılar (devamı var)

Dosya adı	Tanım	IBM i	AIX and Linux	Windows	z/OS
CMQDXPx	MQDXP-Veri dönüştürme çıkış deęiřtirgeleri	V L		V L	
CMQEPhX	MQEPh-Embedded PCF üstbilgisi	V L	V L	V L	V L
CMQGMox	MQGMo-İleti alma seçenekleri	V L	V L	V L	V L
CMQIHx	MQIIH- IMS bilgi üstbilgisi	V L	V L	V L	V L
CMQMDx	MQMD-İleti tanımlayıcısı	V L	V L	V L	V L
CMQMD1x	MQMD1 -İleti tanımlayıcısı sürüm 1	V L	V L	V L	V L
CMQMD2x	MQMD2 -İleti tanımlayıcı sürüm 2	V L	V L	V L	V L
CMQMDEX	MQMDE-İleti tanımlayıcısı genişletti	V L	V L	V L	V L
CMQODx	MQOD-Nesne tanımlayıcısı	V L	V L	V L	V L
CMQORX	MQOR-Nesne kaydı	V L	V L	V L	V L
CMQPMox	MQPMo-put iletisi seçenekleri	V L	V L	V L	V L
CMQRFHx	MQRFH-Kurallar ve biçimleme üstbilgisi	V L	V L	V L	V L
CMQRFH2x	MQRFH2 -Kural ve biçimlendirme üstbilgisi 2	V L	V L	V L	V L
CMQRMHx	MQRMH-Başvuru iletisi üstbilgisi	V L	V L	V L	V L
CMQRRX	MQRR-Yanıt kaydı	V L	V L	V L	
CMQSCOx	MQSCO-TLS configuratooon seçenekleri		V L	V L	
CMQTMx	MQTM-Tetikleme iletisi	V L		V L	V L
CMQTMcx	MQTMc-Tetikleyici ileti karakteri	V L	V L		
CMQTMc2x	MQTMc2 -Tetikleyici iletisi 2 karakter	V L	V L	V L	V L
CMQWIHx	MQWIH-İř bilgisi üstbilgisi	V L	V L	V L	V L
CMQXQHx	MQXQH-İletim kuyruęu üstbilgisi	V L	V L	V L	V L

**Anahtar:**

- Sağlanan başlangıç deęerlerine sahip dosyalar, x = V
- Başlangıç deęerleri sağlanmadan dosyalar, x = L

### z/OS PL/I içerme dosyaları

PL/I programlama dili için bir dizi INCLUDE (INCLUDE) dosyası sağlanır. Bu dosyalar yalnızca z/OS üzerinde kullanılabilir.

Çizelge 3. PL/I, kütük-veri tiplerini, dönüş kodlarını, deęişmezleri ve yapıları içerir

Dosya adı	Tanım	IBM i	AIX and Linux	Windows	z/OS
<b>Veri tipleri, dönüş kodları, sabitler ve yapılar</b>					

Çizelge 3. PL/I, kütük-veri tiplerini, dönüş kodlarını, değişmezleri ve yapıları içerir (devamı var)

Dosya adı	Tanım	IBM i	AIX and Linux	Windows	z/OS
CMQP	MQI tanımlamaları				P
CMQCFP	PCF tanımlamaları				P
CMQEPP	Giriş noktası tanımları				P
CMQPSP	Yayınlama/Abone Olma tanımları				P
CMQXP	Kanal ve çıkış tanımları				P

**Anahtar:** P= Dosya sağlandı

### IBM i RPG kopyalama dosyaları

RPG COPY dosyaları RPG programlama dili için sağlanır. Bu dosyalar yalnızca IBM üzerinde kullanılabilir.

Çizelge 4. RPG kopya dosyaları-dönüş kodları, sabitler ve yapılar

Dosya adı	Tanım	IBM i	AIX and Linux	Windows	z/OS
<b>Dönüş kodları ve değişmezleri</b>					
CMQX	MQI tanımlamaları	G R			
CMQCFX	PCF tanımlamaları	G			
CMQPSI	Yayınlama/Abone Olma tanımları	G			
CMQXX	Kanal ve çıkış tanımları	G R			
<b>Yapılar</b>					
CMQBOX	MQBO-Başlat seçenekleri	G H			
CMQCDX	MQCD-Kanal tanımlaması	G H R			
CMQCFBFx	MQCFBF-PCF byte dizgisi süzgeç değiştirgesi	G H			
CMQCFBSX	MQCFBS-PCF byte dizilimi parametresi	G H			
CMQCFGRX	MQCFGR-PCF grup parametresi	G H			
CMQCFHx	MQCFH-PCF üstbilgisi	G H			
CMQCFIFX	MQCFIF-PCF tamsayı süzgeç değiştirgesi	G H			
CMQCFIX	MQCFIL-PCF tamsayı listesi değiştirgesi	G H			
CMQCFINX	MQCFIN-PCF tamsayı değiştirgesi	G H			
CMQCFSFx	MQCFSF-PCF dizgi süzgeç değiştirgesi	G H			
CMQCFSLX	MQCFSL-PCF dizgi listesi değiştirgesi	G H			
CMQCFSTX	MQCFST-PCF dizgisi değiştirgesi	G H			

Çizelge 4. RPG kopya dosyaları-dönüş kodları, sabitler ve yapılar (devamı var)

Dosya adı	Tanım	IBM i	AIX and Linux	Windows	z/OS
CMQCFXLx	MQCFIL64 -PCF 64-bitlik tamsayı listesi değiştirgesi	G H			
CMQCFXNx	MQCFIN64 -PCF 64 bit tamsayı değiştirgesi	G H			
CMQCHARVX	MQCHARV-Değişken uzunluklu dizgi	G H			
CMQCIHx	MQCIH- CICS bridge üstbilgisi	G H			
CMQCN0x	MQCN0-Bağlantı seçenekleri	G H			
CMQCSPx	MQCS P-Güvenlik değiştirgeleri	G H			
CMQCPx	MQCP-Kanal çıkış değiştirgeleri	G H R			
CMQDHx	MQDH-Dağıtım üstbilgisi	G H R			
CMQDLHx	MQDLH-Dead-letter üstbilgisi	G H R			
CMQDXPx	MQDXP-Veri dönüştürme çıkış değiştirgeleri	G H R			
CMQEPHX	MQEPH-Embedded PCF üstbilgisi	G H			
CMQGM0X	MQGM0-İleti alma seçenekleri	G H R			
CMQIHx	MQIIH- IMS bilgi üstbilgisi	G H R			
CMQMDx	MQMD-İleti tanımlayıcısı	G H R			
CMQMD1x	MQMD1 -İleti tanımlayıcısı sürüm 1	G H R			
CMQMD2x	MQMD2 -İleti tanımlayıcı sürüm 2	G H			
CMQMDEX	MQMDE-İleti tanımlayıcısı genişletti	G H R			
CMQODx	MQOD-Nesne tanımlayıcısı	G H R			
CMQORX	MQOR-Nesne kaydı	G H R			
CMQPM0x	MQPM0-put iletisi seçenekleri	G H R			
CMQPPX	MQPPX-Yayınlama/Abone Olma yönlendirme çıkış değiştirgeleri	G H			
CMQRFHx	MQRFH-Kurallar ve biçimleme üstbilgisi	G H			
CMQRFH2x	MQRFH2 -Kural ve biçimlendirme üstbilgisi 2	G H			
CMQRMHx	MQRMH-Başvuru iletisi üstbilgisi	G H R			
CMQRRX	MQRR-Yanıt kaydı	G H R			
CMQTMx	MQTM-Tetikleme iletisi	G H R			
CMQTMcx	MQTMc-Tetikleyici ileti karakteri	G H R			
CMQTM2x	MQTM2 -Tetikleyici ileti 2 karakter	G H R			
CMQWIHx	MQWIH-İş bilgisi üstbilgisi	G H			

Çizelge 4. RPG kopya dosyaları-dönüş kodları, sabitler ve yapılar (devamı var)

Dosya adı	Tanım	IBM i	AIX and Linux	Windows	z/OS
CMQXQHx	MQXQH-İletim kuyruğu üstbilgisi	G H R			

**Anahtar:**

- Durağan bağ için dosya, kullanıma hazırlandı, x = G sağlandı
- Durağan bağ için dosya, kullanıma hazırlanmadı, sağlanan x = H
- Dinamik bağlantı için dosya, kullanıma hazırlandı, sağlandı, x = R

### Windows Visual Basic modül dosyaları

Üstbilgi (ya da form) dosyaları, MQI kullanan Visual Basic uygulama programlarını yazmanıza yardımcı olmak için sağlanmıştır. Bu üstbilgi kütükleri yalnızca 32 bit sürümlerde sağlanır.

Çizelge 5. Visual Basic module files-call bildirimleri, data types, return codes, deconstructies

Dosya adı	Tanım	IBM i	AIX and Linux sistemleri	Windows	z/OS
<b>Çağrı bildirimleri, veri tipleri, dönüş kodları, sabitler ve yapılar</b>					
CMQB	MQI tanımlamaları			B	
CMQBB	MQAI tanımlamaları			B	
CMQCFB	PCF tanımlamaları			B	
CMQXB	Kanal ve çıkış tanımları			B	

**Anahtar:** B= Dosya sağlandı

### z/OS z/OS Assembler COPY dosyaları

Çeşitli COPY dosyaları, MQI kullanan z/OS Assembler uygulama programlarını yazmanıza yardımcı olmak için sağlanmıştır.

Çizelge 6. z/OS Assembler kopya dosyaları-veri tipleri, dönüş kodları, sabitler ve yapılar

Dosya adı	Tanım	IBM i	AIX and Linux	Windows	z/OS
<b>Veri tipleri, dönüş kodları ve sabitler</b>					
CMQA	MQI tanımlamaları				A
CMQCFA	PCF tanımlamaları				A
CMQPSA	Yayınlama/Abone Olma tanımları				A
CMQVERA	Yapı sürümü denetimi				A
MQXA	Kanal ve çıkış tanımları				A
<b>Yapılar</b>					
CMQCDA	MQCD-Kanal tanımlaması				
CMQCFBFA	MQCFBF-PCF byte dizgisi süzgeç değiştirgesi				

Çizelge 6. z/OS Assembler kopya dosyaları-veri tipleri, dönüş kodları, sabitler ve yapılar (devamı var)

Dosya adı	Tanım	IBM i	AIX and Linux	Windows	z/OS
CMQCFBSA	MQCFBS-PCF byte dizilimi parametresi				A
CMQCFGRA	MQCFGR-PCF grup parametresi				A
CMQCFHA	MQCFH-PCF üstbilgisi				A
CMQCFIFA	MQCFIF-PCF tamsayı süzgeç değiştirgesi				A
CMQCFILA	MQCFIL-PCF tamsayı listesi değiştirgesi				A
CMQCFINA	MQCFIN-PCF tamsayı değiştirgesi				A
CMQCFSA	MQCFSF-PCF dizgi süzgeç değiştirgesi				A
CMQCFSLA	MQCFSL-PCF dizgi listesi değiştirgesi				A
CMQCFSTA	MQCFST-PCF dizgisi değiştirgesi				A
CMQCFXLA	MQCFIL64 -PCF 64-bitlik tamsayı listesi değiştirgesi				A
CMQCFXNA	MQCFIN64 -PCF 64 bit tamsayı değiştirgesi				A
CMQCHARVA	MQCHARV-Değişken uzunluklu dizgi				A
CMQCIHA	MQCIH- CICS bridge üstbilgisi				A
CMQCNOA	MQCNO-Bağlantı seçenekleri				A
CMQCSPA	MQCSP-Güvenlik değiştirgeleri				A
CMQCXPA	MQCXP-Kanal çıkış değiştirgeleri				A
CMQDHA	MQDH-Dağıtım üstbilgisi				A
CMQDLHA	MQDLH-Dead-letter üstbilgisi				A
CMQDXPA	MQDXP-Veri dönüştürme çıkış değiştirgeleri				A
CMQEPHA	MQEPH-Embedded PCF üstbilgisi				A
CMQGMOA	MQGMO-İleti alma seçenekleri				A
CMQIIHA	MQIIH- IMS bilgi üstbilgisi				A
CMQMDA	MQMD-İleti tanımlayıcısı				A
CMQMD1A	MQMD1 -İleti tanımlayıcısı sürüm 1				A
CMQMD2A	MQMD2 -İleti tanımlayıcı sürüm 2				A
CMQMDEA	MQMDE-İleti tanımlayıcısı genişletti				A
CMQODA	MQOD-Nesne tanımlayıcısı				A
CMQORA	MQOR-Nesne kaydı				A

Çizelge 6. z/OS Assembler kopya dosyaları-veri tipleri, dönüş kodları, sabitler ve yapılar (devamı var)

Dosya adı	Tanım	IBM i	AIX and Linux	Windows	z/OS
CMQPMOA	MQPMO-put iletisi seçenekleri				A
CMQRFHA	MQRFH-Kurallar ve biçimleme üstbilgisi				A
CMQRFH2A	MQRFH2 -Kural ve biçimlendirme üstbilgisi 2				A
CMQRMHA	MQRMH-Başvuru iletisi üstbilgisi				A
CMQTMMA	MQTM-Tetikleme iletisi				A
CMQTMCA	MQTMCA -Tetikleyici iletisi 2 karakter				A
CMQWCRA	MQWCR-Küme iş yükü kümesi kaydı				A
CMQWDRA	MQWDR-Küme iş yükü hedef kaydı				A
CMQWDR1A	MQWDR1 -Küme iş yükü hedef kaydı sürüm 1				A
CMQWDR2A	MQWDR2 -Küme iş yükü hedef kaydı sürüm 2				A
CMQWIHA	MQWIH-İş bilgisi üstbilgisi				A
CMQWQRK	MQWQR-Küme iş yükü kuyruğu kaydı				A
CMQWQR1A	MQWQR1 -Küme iş yükü kuyruğu kaydı sürüm 1				A
CMQWQR2A	MQWQR2 -Küme iş yükü kuyruğu kaydı sürüm 2				A
CMQWXP	MQWXP-Küme iş yükü çıkış değiştirgeleri				A
CMQWXP1A	MQWXP1 -Küme iş yükü çıkış değiştirgeleri sürüm 1				A
CMQWXP2A	MQWXP2 -Küme iş yükü çıkış değiştirgeleri sürüm 2				A
CMQWXP3A	MQWXP3 -Küme iş yükü çıkış değiştirgeleri sürüm 3				A
CMQXPA	MQXP- CICS API-çıkış çıkış değiştirgeleri				A
CMQXQHA	MQXQH-İletim kuyruğu üstbilgisi				A
CMQXWDA	MQXWD-Çıkış bekleme tanımlayıcısı				A

**Anahtar:** A= Sağlanan dosya



## MQ\_\* (Dizgi Uzunlukları)

Çizelge 7. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQ_ABEND_CODE_LENGTH	4	X'00000004'
MQ_ACCOUNTING_TOKEN_LENGTH	32	X'00000020'
MQ_APPL_FUNCTION_NAME_LENGTH	10	X'0000000A'
MQ_APPL_IDENTITY_DATA_LENGTH	32	X'00000020'
MQ_APPL_LENNGTH	28	X'0000001C'
MQ_APPL_ORIGIN_DATA_LENGTH	4	X'00000004'
MQ_APPL_TAG_LENGTH	28	X'0000001C'
MQ_ARM_SUFFIX_LENGTH	2	X'00000002'
MQ_ATTEN_ID_LENGTH	4	X'00000004'
MQ_AUTH_INFO_CONN_NAME_LENGTH	264	X'00000108'
MQ_AUTH_INFO_DESC_LENGTH	64	X'00000040'
MQ_AUTH_INFO_NAME_LENGTH	48	X'00000030'
MQ_AUTH_INFO_OCSP_URL_LENGTH	256	X'00000100'
MQ_AUTHENTICATOR_LENGTH	8	X'00000008'
MQ_AUTO_REORG_CATALOG_LENGTH	44	X'0000002C'
MQ_AUTO_REORG_TIME_LENGTH	4	X'00000004'
MQ_BATCH_INTERFACE_ID_LENGTH	8	X'00000008'
MQ_BRIDGE_NAME_LENGTH	24	X'00000018'
MQ_CANCEL_KOD_UZUNLUĞU	4	X'00000004'
MQ_CF_STRUC_DESC_LENGTH	64	X'00000040'
MQ_CF_STRUC_NAME_LENGTH	12	X'0000000C'
MQ_CHANNEL_DATE_LENGTH	12	X'0000000C'
MQ_CHANNEL_DESC_LENGTH	64	X'00000040'
MQ_CHANNEL_NAME_LENGTH	20	X'00000014'
MQ_CHANNEL_TIME_LENGTH	8	X'00000008'
MQ_CHINIT_SERVICE_PARM_LENGTH	32	X'00000020'
MQ_CICSS_FILE_NAME_LENGTH	8	X'00000008'
MQ_CLIENT_ID_LENGTH	23	X'00000017'
MQ_CLUSTER_NAME_LENGTH	48	X'00000030'
MQ_CONN_NAME_LENGTH	264	X'00000108'
MQ_CONN_TAG_LENGTH	128	X'00000080'
MQ_CONNECTION_ID_LENGTH	24	X'00000018'
MQ_COREL_ID_LENGTH	24	X'00000018'
MQ_CREATION_DATE_LENGTH	12	X'0000000C'
MQ_CREATION_TIME_LENGTH	8	X'00000008'
MQ_DATE_LENGTH	12	X'0000000C'
MQ_DISTINGUISH_NAME_LENGTH	1024	X'00000400'
MQ_DNS_GROUP_NAME_LENGTH	18	X'00000012'

Çizelge 7. Değişmezlerin değerleri (devamı var)		
Ad	Ondalık değer	Onaltılı değer
MQ_EXIT_DATA_LENGTH	32	X'00000020'
MQ_EXIT_INFO_NAME_LENGTH	48	X'00000030'
MQ_EXIT_NAME_LENGTH	(value differs by platform or version)	
MQ_EXIT_PD_AREA_LENGTH	48	X'00000030'
MQ_EXIT_USER_AREA_LENGTH	16	X'00000010'
MQ_KOLAYLIK_UZUNLUK	8	X'00000008'
MQ_KOLAYLIK_UZUNLUK_UZUNLUK	4	X'00000004'
MQ_FORMAT_LENGTH	8	X'00000008'
MQ_FUNCTION_LENGTH	4	X'00000004'
MQ_GROUP_ID_LENGTH	24	X'00000018'
MQ_LDAP_PASSWORD_LENGTH	32	X'00000020'
MQ_LISTENER_NAME_LENGTH	48	X'00000030'
MQ_LISTENER_DESC_LENGTH	64	X'00000040'
MQ_LOCAL_ADDRESS_LENGTH	48	X'00000030'
MQ_LTERM_OVERRIDE_LENGTH	8	X'00000008'
MQ_LU_NAME_LENGTH	8	X'00000008'
MQ_LUWID_LENGTH	16	X'00000010'
MQ_MAX_EXIT_NAME_LENGTH	128	X'00000080'
MQ_MAX_MCA_USER_ID_LENGTH	64	X'00000040'
MQ_MAX_PROPERTY_NAME_LENGTH	4095	X'00000FFF'
MQ_MAX_USER_ID_LENGTH	64	X'00000040'
MQ_MCA_JOB_NAME_LENGTH	28	X'0000001C'
MQ_MCA_NAME_LENGTH	20	X'00000014'
MQ_MCA_USER_DATA_LENGTH	32	X'00000020'
MQ_MCA_USER_LENGTH	(value differs by platform or version)	
MQ_MFS_MAP_NAME_LENGTH	8	X'00000008'
MQ_MODE_NAME_TH	8	X'00000008'
MQ_MSG_HEADER_LENGTH	4000	X'00000FA0'
MQ_MSG_ID_LENGTH	24	X'00000018'
MQ_MSG_TOKEN_LENGTH	16	X'00000010'
MQ_NAMELIST_DESC_LENGTH	64	X'00000040'
MQ_NAMELIST_NAME_LENGTH	48	X'00000030'
MQ_NHA_INSTANCE_NAME_LENGTH	48	X'00000030'
MQ_OBJECT_INSTANCE_ID_LENGTH	24	X'00000018'
MQ_OBJECT_NAME_LENGTH	48	X'00000030'
MQ_PASS_TICKET_APPL_LENGTH	8	X'00000008'
MQ_PASSWORD_LENGTH	12	X'0000000C'
MQ_PROCESS_APPL_ID_LENGTH	256	X'00000100'
MQ_PROCESS_DESC_LENGTH	64	X'00000040'

Çizelge 7. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQ_PROCESS_ENV_DATA_LENGTH	128	X'00000080'
MQ_PROCESS_NAME_LENGTH	48	X'00000030'
MQ_PROCESS_USER_DATA_LENGTH	128	X'00000080'
MQ_PROGRAM_ADI_UZUNLUK	20	X'00000014'
MQ_PUT_APPL_LENNGTH	28	X'0000001C'
MQ_PUT_DATE_LENGTH	8	X'00000008'
MQ_PUT_TIME_LENGTH	8	X'00000008'
MQ_Q_DESC_LENGTH	64	X'00000040'
MQ_Q_MGR_DESC_LENGTH	64	X'00000040'
MQ_Q_MGR_IDENTIFIER_LENGTH	48	X'00000030'
MQ_Q_MGR_NAME_LENGTH	48	X'00000030'
MQ_Q_NAME_LENGTH	48	X'00000030'
MQ_QSG_NAME_LENGTH	4	X'00000004'
MQ_REMOTE_SYS_ID_LENGTH	4	X'00000004'
MQ_SECURITY_ID_LENGTH	40	X'00000028'
MQ_SELECTOR_LENGTH	10240	X'00002800'
MQ_SERVICE_ARGS_LENGTH	255	X'000000FF'
MQ_SERVICE_COMMAND_LENGTH	255	X'000000FF'
MQ_SERVICE_DESC_LENGTH	64	X'00000040'
MQ_SERVICE_NAME_LENGTH	32	X'00000020'
MQ_SERVICE_PATH_LENGTH	255	X'000000FF'
MQ_SERVICE_STEP_LENGTH	8	X'00000008'
MQ_SHORT_CONN_NAME_LENGTH	20	X'00000014'
MQ_SHORT_DNAME_LENGTH	256	X'00000100'
MQ_SSL_CIPHER_SPEC_LENGTH	32	X'00000020'
MQ_SSL_CRYPTOHARDWARE_LENGTH	256	X'00000100'
MQ_SSL_HANDSHAKE_STAGE_LENGTH	32	X'00000020'
MQ_SSL_KEY_LIBRARY_LENGTH	44	X'0000002C'
MQ_SSL_KEY_MEMBER_LENGTH	8	X'00000008'
MQ_SSL_KEY_REPOSITORY_LENGTH	256	X'00000100'
MQ_SSL_PEER_NAME_LENGTH	1024	X'00000400'
MQ_SSL_SHORT_PEER_NAME_LENGTH	256	X'00000100'
MQ_START_CODE_LENGTH	4	X'00000004'
MQ_STORAGE_CLASS_DESC_LENGTH	64	X'00000040'
MQ_STORAGE_CLASS_LENGTH	8	X'00000008'
MQ_SUB_IDENTITY_LENGTH	128	X'00000080'
MQ_SUB_POINT_LENGTH	128	X'00000080'
MQ_SUITE_B_128_BIT	2	X'00000002'
MQ_SUITE_B_192_BIT	4	X'00000004'

Çizelge 7. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQ_SUITE_B_NONE	1	X'00000001'
MQ_SUITE_B_NOT_VAR	0	X'00000000'
MQ_TCP_NAME_LENGTH	8	X'00000008'
MQ_TIME_LENGTH	8	X'00000008'
MQ_TOPIC_DESC_LENGTH	64	X'00000040'
MQ_TOPIC_NAME_LENGTH	48	X'00000030'
MQ_TOPIC_STR_LENGTH	10240	X'00002800'
MQ_TOTAL_EXIT_DATA_LENGTH	999	X'000003E7'
MQ_TOTAL_EXIT_NAME_LENGTH	999	X'000003E7'
MQ_TP_NAME_LENGTH	64	X'00000040'
MQ_TPIPE_NAME_LENGTH	8	X'00000008'
MQ_TRAN_INSTANCE_ID_LENGTH	16	X'00000010'
MQ_TRANSACTION_ID_LENGTH	4	X'00000004'
MQ_TRIGGER_DATA_LENGTH	64	X'00000040'
MQ_TRIGGER_PROGRAM_NAME_LENGTH	8	X'00000008'
MQ_TRIGGER_TERM_ID_LENGTH	4	X'00000004'
MQ_TRIGGER_TRANS_ID_LENGTH	4	X'00000004'
MQ_USER_LENGTH	12	X'0000000C'
MQ_VERSION_LENGTH	8	X'00000008'
MQ_XCF_GROUP_NAME_LENGTH	8	X'00000008'
MQ_XCF_MEMBER_NAME_LENGTH	16	X'00000010'

### MQ\_\* (Komut biçimi Dizgi Uzunlukları)

Çizelge 8. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQ_ARCHIVE_PFX_LENGTH	36	X'00000024'
MQ_ARCHIVE_UNIT_LENGTH	8	X'00000008'
MQ_ASID_LENGTH	4	X'00000004'
MQ_AUTH_PROFILE_NAME_LENGTH	48	X'00000030'
MQ_CF_LEID_LENGTH	12	X'0000000C'
MQ_KOMUT_MQSC_LENGTH	32768	X'00008000'
MQ_DATA_SET_NAME_LENGTH	44	X'0000002C'
MQ_DB2_NAME_LENGTH	4	X'00000004'
MQ_DSG_NAME_LENGTH	8	X'00000008'
MQ_ENTITY_NAME_LENGTH	1024	X'00000400'
MQ_ENV_INFO_LENGTH	96	X'00000060'
MQ_IP_ADDRESS_LENGTH	48	X'00000030'
MQ_LOG_COREL_ID_LENGTH	8	X'00000008'
MQ_LOG_EXTENT_NAME_LENGTH	24	X'00000018'
MQ_LOG_PATH_LENGTH	1024	X'00000400'

Çizelge 8. Değişmezlerin değerleri (devamı var)		
Ad	Ondalık değer	Onaltılı değer
MQ_LRSN_LENGTH	12	X'0000000C'
MQ_ORIGIN_NAME_LENGTH	8	X'00000008'
MQ_PSB_NAME_LENGTH	8	X'00000008'
MQ_PST_ID_LENGTH	8	X'00000008'
MQ_Q_MGR_CPF_LENGTH	4	X'00000004'
MQ_RESPONSE_ID_LENGTH	24	X'00000018'
MQ_RBA_LENGTH	16	X'00000010'
MQ_SECURITY_PROFILE_LENGTH	40	X'00000028'
MQ_SERVICE_COMPONENT_LENGTH	48	X'00000030'
MQ_SUB_NAME_LENGTH	10240	X'00002800'
MQ_SYSP_SERVICE_LENGTH	32	X'00000020'
MQ_SYSTEM_NAME_LENGTH	8	X'00000008'
MQ_TASK_NUMBER_LENGTH	8	X'00000008'
MQ_TPIPE_PFX_LENGTH	4	X'00000004'
MQ_UOW_ID_LENGTH	256	X'00000100'
MQ_USER_DATA_LENGTH	10240	X'00002800'
MQ_VOLSER_LENGTH	6	X'00000006'

### MQACH\_\* (API çıkış zinciri alanı üstbilgi yapısı)

Çizelge 9. Değişmezlerin yapıları	
Ad	Yapı
MQACH_STRUCT_ID	"ACH-"
MQACH_STRUC_ID_ARRAY	'A','C','H','-'

**Not:** - simgesi tek bir boş karakteri temsil eder.

Çizelge 10. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQACH_VERSION_1	1	X'00000001'
MQACH_CURRENT_VERSION	1	X'00000001'
MQACH_LENGTH_1	(value differs by platform or version)	(value differs by platform or version)
MQACH_CURRENT_LENGTH	(value differs by platform or version)	(value differs by platform or version)

### MQACT\_\* (Hesap Belirteci)

Çizelge 11. Değişmez adlar ve değerler	
Ad	Değer
MQACT_NONE	X'00...00' (32 boş değer)
MQACT_NONE_DIZISI	'\0','\0',... (32 boş değer)

## MQACT\_\* (Komut biçimi İşlem Seçenekleri)

Çizelge 12. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQACT_FORCE_REMOVE	1	X'00000001'
MQACT_ADVANCE_LOG	2	X'00000002'
MQACT_COLLECT_STATISTICS	3	X'00000003'
MQACT_PUBSUB	4	X'00000004'

## MQACTP\_\* (İşlem)

Çizelge 13. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQACTP_YENİ	0	X'00000000'
MQAKP_ILERI	1	X'00000001'
MQACTP_CEVAPLA	2	X'00000002'
MQACP_REPORT	3	X'00000003'

## MQACTT\_\* (Hesap Simgesi Tipleri)

Çizelge 14. Değişmezlerin değerleri	
Ad	Onaltılı değer
MQACTT_UNKNOWN	X'00'
MQACTT_CICS_LUOW_ID	X'01'
MQACTT_OS2_DEFAULT	X'04'
MQACCT_DOS_VARSAYILAN	X'05'
MQACTT_UNIX_NUMERIC_ID	X'06'
MQACTT_OS400_ACCOUNT_TOKEN	X'08'
MQACTT_WINDOWS_VARSAYI	X'09'
MQACTT_NT_SECURITY_ID	X'0B'
MQACCT_USER	X'19'

## MQADOPT\_\* (Yeni MCA Denetimlerini ve Yeni MCA Tiplerini Aopt)

### Yeni MCA Denetimlerini Kabul Et

Çizelge 15. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQADOPT_CHECK_NONE	0	X'00000000'
MQADOPT_CHECK_ALL	1	X'00000001'
MQADOPT_CHECK_Q_MGR_NAME	2	X'00000002'
MQADOPT_CHECK_NET_ADDR	4	X'00000004'

## Yeni MCA Tiplerini Benimse

Çizelge 16. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQADOPT_TYPE_NO	0	X'00000000'
MQADOPT_TYPE_ALL	1	X'00000001'
MQADOPT_TYPE_SVR	2	X'00000002'
MQADOPT_TYPE_SDR	4	X'00000004'
MQADOPT_TYPE_RCVR	8	X'00000008'
MQADOPT_TYPE_CLUSRCVR	16	X'00000010'

## MQAIR\_\* (Kimlik doğrulama bilgileri kayıt yapısı)

Çizelge 17. Değişmezlerin yapıları	
Ad	Yapı
MQAIR_STRUCT_ID	"AIR↵"
MQAIR_STRUC_ID_ARRAY	'A', 'I', 'R', '↵'

**Not:** ↵ simgesi tek bir boş karakteri temsil eder.

Çizelge 18. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQAIR_VERSION_1	1	X'00000001'
MQAIR_VERSION_2	2	X'00000002'
MQAIR_CURRENT_SÜRÜM	2	X'00000002'

## MQAIT\_\* (Kimlik Doğrulama Bilgileri Tipi)

Çizelge 19. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQAIT_TÜMÜ	0	X'00000000'
MQAIT_CRL_LDAP	1	X'00000001'
MQAIT_OCSP	2	X'00000002'
MQAIT_IDPW_OS	3	X'00000003'
MQAIT_IDPW_LDAP	4	X'00000004'

## MQAS\_\* (Komut biçimi Zamanuyumsuz Durum Değerleri)

Çizelge 20. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQAS_NONE	0	X'00000000'
MQAS_BAŞLATMA	1	X'00000001'
MQAS_START_SABIT	2	X'00000002'
MQAS_DURDU	3	X'00000003'
MQAS_ASKI	4	X'00000004'
MQAS_JARTIDED_TEMPORARY	5	X'00000005'
MQAS_ETKİN	6	X'00000006'

Çizelge 20. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQAS_DEVREDİŞİ	7	X'00000007'

## MQAT\_\* (Koyma Uygulaması Tipleri)

Çizelge 21. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQAT_BILINMIYOR	-1	X'FFFFFFFF'
MQAT_NO_CONTEXT	0	X'00000000'
MQAT_CICS	1	X'00000001'
MQAT_MVS	2	X'00000002'
MQAT_OS390	2	X'00000002'
MQAT_ZOS	2	X'00000002'
MQAT_IMS	3	X'00000003'
MQAT_OS2	4	X'00000004'
MQAT_DOS	5	X'00000005'
MQAT_AIX	6	X'00000006'
MQAT_UNIX	6	X'00000006'
MQAT_QMGR	7	X'00000007'
MQAT_OS400	8	X'00000008'
MQAT_WINDOWS	9	X'00000009'
MQAT_CICS_VSE	10	X'0000000A'
MQAT_WINDOWS_NT	11	X'0000000B'
MQAT_VMS	12	X'0000000C'
MQAT_GUARDIAN	13	X'0000000D'
MQAT_NSK	13	X'0000000D'
MQAT_VOS	14	X'0000000E'
MQAT_OPEN_TP1	15	X'0000000F'
MQAT_VM	18	X'00000012'
MQAT_IMS_BRIDGE	19	X'00000013'
MQAT_XCF	20	X'00000014'
MQAT_CICS_BRIDGE	21	X'00000015'
MQAT_NOTES_AGENT	22	X'00000016'
MQAT_TPF	23	X'00000017'
MQAT_USER	25	X'00000019'
MQAT_BROKER	26	X'0000001A'
MQAT_QMGR_YAYINLAMA	26	X'0000001A'
MQAT_JAVA	28	X'0000001C'
MQAT_DQM	29	X'0000001D'
MQAT_CHANNEL_INITIATOR	30	X'0000001E'
MQAT_WLM	31	X'0000001F'



Çizelge 21. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQAT_BATCH	32	X'00000020'
MQAT_RRS_BATCH	33	X'00000021'
MQAT_SIB	34	X'00000022'
MQAT_DEFAULT	(value differs by platform or version)	(value differs by platform or version)
MQAT_USER_FIRST	65536	X'00010000'
MQAT_USER_SON	99999999	X'3B9AC9FF'

### MQAUTH\_\* (Komut biçimi Yetki Değerleri)

Çizelge 22. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQAUTH_NONE	0	X'00000000'
MQAUTH_ALT_USER_AUTHORITY	1	X'00000001'
MQAUTH_BROWSE	2	X'00000002'
MQAUTH_CHANGE	3	X'00000003'
MQAUTH_CLEAR	4	X'00000004'
MQAUTH_CONNECT	5	X'00000005'
MQAUTH_CREATE	6	X'00000006'
MQAUTH_DELETE	7	X'00000007'
MQAUTH_DISPLAY	8	X'00000008'
MQAUTH_INPUT	9	X'00000009'
MQAUTH_SORGULAMA	10	X'0000000A'
MQAUTH_OUTPUT	11	X'0000000B'
MQAUTH_PASS_ALL_CONTEXT	12	X'0000000C'
MQAUTH_PASPAS_IDENTITY_CONTEXT	13	X'0000000D'
MQAUTH_SET	14	X'0000000E'
MQAUTH_SET_ALL_CONTEXT	15	X'0000000F'
MQAUTH_SET_IDENTITY_CONTEXT	16	X'00000010'
MQAUTH_CONTROL	17	X'00000011'
MQAUTH_CONTROL_EXTENDED	18	X'00000012'
MQAUTH_YAYINLA	19	X'00000013'
MQAUTH_ABLE	20	X'00000014'
MQAUTH_ÖZGEÇMİŞ	21	X'00000015'
MQAUTH_SYSTEM	22	X'00000016'

### MQAUTHOPT\_\* (Komut biçimi Yetki Seçenekleri)

Çizelge 23. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQAUTHOPT_KÜMÜLATIF	256	X'00000100'
MQAUTHOPT_ENTITY_EXPLICIT	1	X'00000001'

Çizelge 23. Değişmezlerin değerleri (devamı var)		
Ad	Ondalık değer	Onaltılı değer
MQAUTHOPT_ENTITY_SET	2	X'00000002'
MQAUTHOPT_NAME_ALL_MATCHING	32	X'00000020'
MQAUTHOPT_NAME_AS_JOKER	64	X'00000040'
MQAUTHOPT_NAME_EXPLICIT	16	X'00000010'

### MQAXC\_\* (API çıkış bağlamı yapısı)

Çizelge 24. Değişmezlerin yapıları	
Ad	Yapı
MQAXC_STRUCT_ID	"AXC~"
MQAXC_STRUC_ID_ARRAY	'A','X','C','~'

**Not:** ~ simgesi tek bir boş karakteri temsil eder.

Çizelge 25. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQAXC_VERSION_1	1	X'00000001'
MQAXC_CURRENT_VERSION	1	X'00000001'

### MQAXP\_\* (API çıkış değiştirgesi yapısı)

Çizelge 26. Değişmezlerin yapıları	
Ad	Yapı
MQAXP_STRUCT_ID	"AXP~"
MQAXP_STRUC_ID_ARRAY	'A','X','P','~'

**Not:** ~ simgesi tek bir boş karakteri temsil eder.

Çizelge 27. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQAXP_VERSION_1	1	X'00000001'
MQAXP_VERSION_2	2	X'00000002'
MQAXP_CURRENT_VERSION	2	X'00000002'

### MQBA\_\* (Bayt Öznitelik Seçicileri)

Çizelge 28. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQBA_FIRST	6001	X'00001771'
MQBA_SON	8000	X'00001F40'

### MQBACF\_\* (Komut biçimi Byte Değiştirgesi Tipleri)

Çizelge 29. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQBACF_FIRST	7001	X'00001B59'

Çizelge 29. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQBACF_EVENT_ACCOUNTING_TOKEN	7001	X'00001B59'
MQBACF_EVENT_SECURITY_ID	7002	X'00001B5A'
MQBACF_RESPONSE_SET	7003	X'00001B5B'
MQBACF_RESPEONSE_ID	7004	X'00001B5C'
MQBACF_EXTERNAL_UOW_ID	7005	X'00001B5D'
MQBACF_CONNECTION_ID	7006	X'00001B5E'
MQBACF_GENERIC_CONNECTION_ID	7007	X'00001B5F'
MQBACF_ORIGIN_UOW_ID	7008	X'00001B60'
MQBACF_Q_MGR_UOW_ID	7009	X'00001B61'
MQBACF_ACCOUNTING_TOKEN	7010	X'00001B62'
MQBACF_COREL_ID	7011	X'00001B63'
MQBACF_GROUP_ID	7012	X'00001B64'
MQBACF_MSG_ID	7013	X'00001B65'
MQBACF_CF_LEID	7014	X'00001B66'
MQBACF_DESTINATION_COREL_ID	7015	X'00001B67'
MQBACF_ALT_ID	7016	X'00001B68'
KULLANILAN MQBACF_LAST_SEN	7016	X'00001B68'

### MQBL\_\* ( mqAddDizgisi ve mqSetDizgisi için Arabellek Uzunluğu)

Çizelge 30. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQBL_NULL_TERMINATED	-1	X'FFFFFFFF'

### MQBMHO\_\* (Arabellekten ileti işleme seçenekleri ve yapısı)

#### Arabellek-ileti tanıtıcısı seçenekleri yapısı

Çizelge 31. Değişmezlerin yapıları

Ad	Yapı
MQBMHO_STRUC_ID	"BMHO"
MQBMHO_STRUC_ID_ARRAY	'B', 'M', 'H', 'O'

**Not:** ~ simgesi tek bir boş karakteri temsil eder.

Çizelge 32. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQBMHO_VERSION_1	1	X'00000001'
MQBMHO_CURRENT_VERSION	1	X'00000001'

## Arabellek-İleti Tanıtıcısı Seçenekleri

Çizelge 33. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQBMHO_YOK	0	X'00000000'
MQBMHO_DELETE_PROPERTIES	1	X'00000001'

## MQBND\_\* (Varsayılan Bağ Tanımları)

Çizelge 34. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQBND_BIND_ON_AçIK	0	X'00000000'
MQBND_BIND_NOT_FIXED	1	X'00000001'
MQBND_BIND_ON_GROUP	2	X'00000002'

## MQBO\_\* (Başlangıç seçenekleri ve yapısı)

### Seçenekler yapısını başlat

Çizelge 35. Değişmezlerin yapıları	
Ad	Yapı
MQBO_STRUC_ID	"B0↯"
MQBO_STRUC_ID_ARRAY	'B', '0', '↯', '↯'

**Not:** ↯ simgesi tek bir boş karakteri temsil eder.

Çizelge 36. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQBO_VERSION_1	1	X'00000001'
MQBO_CURRENT_VERSION	1	X'00000001'

## Başlangıç Seçenekleri

Çizelge 37. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQBO_NONE	0	X'00000000'

## MQBT\_\* (Komut biçimi Köprü Tipleri)

Çizelge 38. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQBT_OTMA	1	X'00000001'

## MQCA\_\* (Karakter Özniteliği Seçicileri)

Çizelge 39. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQCA_ADMIN_TOPIC_NAME	2105	X'00000839'
MQCA_ALTERATION_DATE	2027	X'000007EB'

Çizelge 39. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQCA_ALTERATION_TIME	2028	X'000007EC'
MQCA_APPL_ID	2001	X'000007D1'
MQCA_AUTH_INFO_CONN_NAME	2053	X'00000805'
MQCA_AUTH_INFO_DESC	2046	X'000007FE'
MQCA_AUTH_INFO_NAME	2045	X'000007FD'
MQCA_AUTH_INFO_OCSP_URL	2109	X'0000083D'
MQCA_AUTO_REORG_CATALOG	2091	X'0000082B'
MQCA_AUTO_REORG_START_TIME	2090	X'0000082A'
MQCA_BACKOUT_REQ_Q_NAME	2019	X'000007E3'
MQCA_BASE_OBJECT_NAME	2002	X'000007D2'
MQCA_BASE_Q_NAME	2002	X'000007D2'
MQCA_BATCH_INTERFACE_ID	2068	X'00000814'
MQCA_CF_STRUC_DESC	2052	X'00000804'
MQCA_CF_STRUC_NAME	2039	X'000007F7'
MQCA_CHANNEL_AUTO_DEF_EXIT	2026	X'000007EA'
MQCA_CHILD	2101	X'00000835'
MQCA_CHINIT_SERVICE_PARM	2076	X'0000081C'
MQCA_CICS_FILE_NAME	2060	X'0000080C'
MQCA_CLUS_CHL_NAME	2124	X'0000084C'
MQCA_CLUSTER_DATE	2037	X'000007F5'
MQCA_CLUSTER_NAME	2029	X'000007ED'
MQCA_CLUSTER_NAMELIST	2030	X'000007EE'
MQCA_CLUSTER_Q_MGR_NAME	2031	X'000007EF'
MQCA_CLUSTER_TIME	2038	X'000007F6'
MQCA_CLUSTER_WORKLOAD_DATA	2034	X'000007F2'
MQCA_CLUSTER_WORKLOAD_EXIT	2033	X'000007F1'
MQCA_COMMAND_INPUT_Q_NAME	2003	X'000007D3'
MQCA_COMMAND_REPLY_Q_NAME	2067	X'00000813'
MQCA_CREATION_DATE	2004	X'000007D4'
MQCA_CREATION_TIME	2005	X'000007D5'
MQCA_DEAD_LETTER_Q_NAME	2006	X'000007D6'
MQCA_DEF_XMIT_Q_NAME	2025	X'000007E9'
MQCA_DNS_GROUP	2071	X'00000817'
MQCA_ENV_DATA	2007	X'000007D7'
MQCA_FIRST	2001	X'000007D1'
MQCA_IGQ_USER_ID	2041	X'000007F9'
MQCA_INITIATION_Q_NAME	2008	X'000007D8'
MQCA_LAST	4000	X'00000FA0'
MQCA_LAST_USED	2109	X'0000083D'

Çizelge 39. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQCA_LDAP_PASSWORD	2048	X'00000800'
MQCA_LDAP_USER_NAME	2047	X'000007FF'
MQCA_LU_GROUP_NAME	2072	X'00000818'
MQCA_LU_NAME	2073	X'00000819'
MQCA_LU62_ARM_SUFFIX	2074	X'0000081A'
MQCA_MODEL_DURABLE_Q	2096	X'00000830'
MQCA_MODEL_NON_DURABLE_Q	2097	X'00000831'
MQCA_MONITOR_Q_NAME	2066	X'00000812'
MQCA_NAMELIST_DESC	2009	X'000007D9'
MQCA_NAMELIST_NAME	2010	X'000007DA'
MQCA_NAMES	2020	X'000007E4'
MQCA_PARENT	2102	X'00000836'
MQCA_PASS_TICKET_APPL	2086	X'00000826'
MQCA_PROCESS_DESC	2011	X'000007DB'
MQCA_PROCESS_NAME	2012	X'000007DC'
MQCA_Q_DESC	2013	X'000007DD'
MQCA_Q_MGR_DESC	2014	X'000007DE'
MQCA_Q_MGR_IDENTIFIER	2032	X'000007F0'
MQCA_Q_MGR_NAME	2015	X'000007DF'
MQCA_Q_NAME	2016	X'000007E0'
MQCA_QSG_NAME	2040	X'000007F8'
MQCA_REMOTE_Q_MGR_NAME	2017	X'000007E1'
MQCA_REMOTE_Q_NAME	2018	X'000007E2'
MQCA_REPOSITORY_NAME	2035	X'000007F3'
MQCA_REPOSITORY_NAMELIST	2036	X'000007F4'
MQCA_RESUME_DATE	2098	X'00000832'
MQCA_RESUME_TIME	2099	X'00000833'
MQCA_SERVICE_DESC	2078	X'0000081E'
MQCA_SERVICE_NAME	2077	X'0000081D'
MQCA_SERVICE_START_ARGS	2080	X'00000820'
MQCA_SERVICE_START_COMMAND	2079	X'0000081F'
MQCA_SERVICE_STOP_ARGS	2082	X'00000822'
MQCA_SERVICE_STOP_COMMAND	2081	X'00000821'
MQCA_STDERR_DESTINATION	2084	X'00000824'
MQCA_STDOUT_DESTINATION	2083	X'00000823'
MQCA_SSL_CRL_NAMELIST	2050	X'00000802'
MQCA_SSL_CRYPTO_HARDWARE	2051	X'00000803'
MQCA_SSL_KEY_LIBRARY	2069	X'00000815'
MQCA_SSL_KEY_MEMBER	2070	X'00000816'

Çizelge 39. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQCA_SSL_KEY_REPOSITORY	2049	X'00000801'
MQCA_STORAGE_CLASS	2022	X'000007E6'
MQCA_STORAGE_CLASS_DESC	2042	X'000007FA'
MQCA_SYSTEM_LOG_Q_NAME	2065	X'00000811'
MQCA_TCP_NAME	2075	X'0000081B'
MQCA_TOPIC_DESC	2093	X'0000082D'
MQCA_TOPIC_NAME	2092	X'0000082C'
MQCA_TOPIC_STRING_FILTER	2108	X'0000083C'
MQCA_TOPIC_STRING	2094	X'0000082E'
MQCA_TPIPE_NAME	2085	X'00000825'
MQCA_TRIGGER_CHANNEL_NAME	2064	X'00000810'
MQCA_TRIGGER_DATA	2023	X'000007E7'
MQCA_TRIGGER_PROGRAM_NAME	2062	X'0000080E'
MQCA_TRIGGER_TERM_ID	2063	X'0000080F'
MQCA_TRIGGER_TRANS_ID	2061	X'0000080D'
MQCA_USER_DATA	2021	X'000007E5'
MQCA_USER_LIST	4000	X'00000FA0'
MQCA_VERSION	2120	X'00000848'
MQCA_XCF_GROUP_NAME	2043	X'000007FB'
MQCA_XCF_MEMBER_NAME	2044	X'000007FC'
MQCA_XMIT_Q_NAME	2024	X'000007E8'

### MQCACF\_\* (Komut biçimi Karakter Değiştirgesi Tipleri)

Çizelge 40. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQCACF_FIRST	3001	X'00000BB9'
MQCACF_FROM_Q_NAME	3001	X'00000BB9'
MQCACF_TO_Q_NAME	3002	X'00000BBA'
MQCACF_FROM_PROCESS_NAME	3003	X'00000BBB'
MQCACF_TO_PROCESS_NAME	3004	X'00000BBC'
MQCACF_FROM_NAMELIST_NAME	3005	X'00000BBD'
MQCACF_TO_NAMELI_ADı	3006	X'00000BBE'
MQCACF_FROM_CHANNEL_NAME	3007	X'00000BBF'
MQCACF_TO_CHANNEL_NAME	3008	X'00000BC0'
MQCACF_FROM_AUTH_INFO_NAME	3009	X'00000BC1'
MQCACF_TO_AUTH_INFO_NAME	3010	X'00000BC2'
MQCACF_Q_ADLARı	3011	X'00000BC3'
MQCACF_PROCESS_NAMES	3012	X'00000BC4'
MQCACF_NAMELIST_NAMES	3013	X'00000BC5'

Çizelge 40. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQCACF_ESCAPE_TEXT	3014	X'00000BC6'
MQCACF_LOCAL_Q_NAMES	3015	X'00000BC7'
MQCACF_MODEL_Q_NAMES	3016	X'00000BC8'
MQCACF_ALIAS_Q_NAMES	3017	X'00000BC9'
MQCACF_REMOTE_Q_NAMES	3018	X'00000BCA'
MQCACF_SENDER_CHANNEL_ADS	3019	X'00000BCB'
MQCACF_SERVER_CHANNEL_ADS	3020	X'00000BCC'
MQCACF_REQUESTER_CHANNEL_ADS	3021	X'00000BCD'
MQCACF_RECEIVER_CHANNEL_NAMES	3022	X'00000BCE'
MQCACF_OBJECT_Q_MGR_NAME	3023	X'00000BCF'
MQCACF_APPL_ADı	3024	X'00000BD0'
MQCACF_USER_IDENTIFIER	3025	X'00000BD1'
MQCACF_AUX_ERROR_DATA_STR_1	3026	X'00000BD2'
MQCACF_AUX_ERROR_DATA_STR_2	3027	X'00000BD3'
MQCACF_AUX_ERROR_DATA_STR_3	3028	X'00000BD4'
MQCACF_BRIDGE_NAME	3029	X'00000BD5'
MQCACF_STREAM_NAME	3030	X'00000BD6'
MQCACF_KONUSU	3031	X'00000BD7'
MQCACF_PARENT_Q_MGR_NAME	3032	X'00000BD8'
MQCACF_COREL_ID	3033	X'00000BD9'
MQCACF_PUBLISH_ZAMAN damgası	3034	X'00000BDA'
MQCACF_STRING_DATA	3035	X'00000BDB'
MQCACF_SUPPORTED_STREAM_NAME	3036	X'00000BDC'
MQCACF_REG_KONUSU	3037	X'00000BDD'
MQCACF_REG_TIME	3038	X'00000BDE'
MQCACF_REG_USER_ID	3039	X'00000BDF'
MQCACF_CHILD_Q_MGR_NAME	3040	X'00000BE0'
MQCACF_REG_STREAM_NAME	3041	X'00000BE1'
MQCACF_REG_Q_MGR_NAME	3042	X'00000BE2'
MQCACF_REG_Q_NAME	3043	X'00000BE3'
MQCACF_REG_COREL_ID	3044	X'00000BE4'
MQCACF_EVENT_USER_ID	3045	X'00000BE5'
MQCACF_OBJECT_NAME	3046	X'00000BE6'
MQCACF_EVENT_Q_MGR	3047	X'00000BE7'
MQCACF_AUTH_INFO_NAMES	3048	X'00000BE8'
MQCACF_EVENT_APPL_IDENTITY	3049	X'00000BE9'
MQCACF_EVENT_APPL_ADı	3050	X'00000BEA'
MQCACF_EVENT_APPL_ORIGIN	3051	X'00000BEB'
MQCACF_XX_ENCODE_CASE_ONE subscription_name	3052	X'00000BEC'



Çizelge 40. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQCACF_REG_ALT_ADı	3053	X'00000BED'
MQCACF_SUBSCRIPTION_IDENTITY	3054	X'00000BEE'
MQCACF_REG_SUB_IDENTITY	3055	X'00000BEF'
MQCACF_XX_ENCODE_CASE_ONE subscription_user_data	3056	X'00000BF0'
MQCACF_REG_SUB_USER_DATA	3057	X'00000BF1'
MQCACF_APPL_TAG	3058	X'00000BF2'
MQCACF_DATA_SET_NAME	3059	X'00000BF3'
MQCACF_UOW_START_DATE	3060	X'00000BF4'
MQCACF_UOW_START_TIME	3061	X'00000BF5'
MQCACF_UOW_LOG_START_DATE	3062	X'00000BF6'
MQCACF_UOW_LOG_START_TIME	3063	X'00000BF7'
MQCACF_UOW_LOG_EXTENT_NAME	3064	X'00000BF8'
MQCACF_PRINCIPAL_ENTITY_NAMES	3065	X'00000BF9'
MQCACF_GROUP_ENTITY_NAMES	3066	X'00000BFA'
MQCACF_AUTH_PROFILE_NAME	3067	X'00000BFB'
MQCACF_ENTITY_NAME	3068	X'00000BFC'
MQCACF_SERVICE_COMPONENT	3069	X'00000BFD'
MQCACF_RESPONSE_Q_MGR_NAME	3070	X'00000BFE'
MQCACF_CURRENT_LOG_EXTENT_NAME	3071	X'00000BFF'
MQCACF_RESTART_LOG_EXTENT_NAME	3072	X'00000C00'
MQCACF_MEDIA_LOG_EKREN_ADı	3073	X'00000C01'
MQCACF_LOG_PATH	3074	X'00000C02'
MQCACF_COMMAND_MQSC	3075	X'00000C03'
MQCACF_Q_MGR_CPF	3076	X'00000C04'
MQCACF_USAGE_LOG_RBA	3078	X'00000C06'
MQCACF_USAGE_LOG_LRSN	3079	X'00000C07'
MQCACF_COMMAND_SCOPE	3080	X'00000C08'
MQCACF_ASID	3081	X'00000C09'
MQCACF_PSB_ADı	3082	X'00000C0A'
MQCACF_PST_ID	3083	X'00000C0B'
MQCACF_TASK_SAYı	3084	X'00000C0C'
MQCACF_TRANSACTION_ID	3085	X'00000C0D'
MQCACF_Q_MGR_UOW_ID	3086	X'00000C0E'
MQCACF_ORIGIN_NAME	3088	X'00000C10'
MQCACFENV_INFO	3089	X'00000C11'
MQCACF_SECURITY_PROFILE	3090	X'00000C12'
MQCACF_CONFIGURATION_DATE	3091	X'00000C13'
MQCACF_CONFIGURATION_TIME	3092	X'00000C14'
MQCACF_FROM_CF_STRUC_NAME	3093	X'00000C15'

Çizelge 40. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQCACF_TO_CF_STRUC_NAME	3094	X'00000C16'
MQCACF_CF_STRUC_NAMES	3095	X'00000C17'
MQCACF_FAIL_DATE	3096	X'00000C18'
MQCACF_FAIL_TIME	3097	X'00000C19'
MQCACF_BACKUP_DATE	3098	X'00000C1A'
MQCACF_BACKUP_TIME	3099	X'00000C1B'
MQCACF_SYSTEM_NAME	3100	X'00000C1C'
MQCACF_CF_STRUC_BACKUP_START	3101	X'00000C1D'
MQCACF_CF_STRUC_BACKUP_END	3102	X'00000C1E'
MQCACF_CF_STRUC_LOG_Q_MGRS	3103	X'00000C1F'
MQCACF_FROM_STORAGE_CLASS	3104	X'00000C20'
MQCACF_TO_STORAGE_CLASS	3105	X'00000C21'
MQCACF_STORAGE_CLASS_NAMES	3106	X'00000C22'
MQCACF_DSG_NAME	3108	X'00000C24'
MQCACF_DB2_NAME	3109	X'00000C25'
MQCACF_SYSP_CMD_USER_ID	3110	X'00000C26'
MQCACF_SYSP_OTMA_GROUP	3111	X'00000C27'
MQCACF_SYSP_OTMA_ÜYESİ	3112	X'00000C28'
MQCACF_SYSP_OTMA_DRU_EXIT	3113	X'00000C29'
MQCACF_SYSP_OTMA_TPIPE_PFX	3114	X'00000C2A'
MQCACF_SYSP_ARCHIVE_PFX1	3115	X'00000C2B'
MQCACF_SYSP_ARCHIVE_UNIT1	3116	X'00000C2C'
MQCACF_SYSP_LOG_COREL_ID	3117	X'00000C2D'
MQCACF_SYSP_UNIT_VOLSER	3118	X'00000C2E'
MQCACF_SYSP_Q_MGR_TIME	3119	X'00000C2F'
MQCACF_SYSP_Q_MGR_DATE	3120	X'00000C30'
MQCACF_SYSP_Q_MGR_RBA	3121	X'00000C31'
MQCACF_SYSP_LOG_RBA	3122	X'00000C32'
MQCACF_SYSP_HİZMET	3123	X'00000C33'
MQCACF_FROM_LISTENER_NAME	3124	X'00000C34'
MQCACF_TO_LISTELER_ADI	3125	X'00000C35'
MQCACF_FROM_SERVICE_NAME	3126	X'00000C36'
MQCACF_TO_SERVICE_NAME	3127	X'00000C37'
MQCACF_LAST_PUT_DATE	3128	X'00000C38'
MQCACF_LAST_PUT_TIME	3129	X'00000C39'
MQCACF_LAST_GET_DATE	3130	X'00000C3A'
MQCACF_LAST_GET_TIME	3131	X'00000C3B'
MQCACF_OPERATION_DATE	3132	X'00000C3C'
MQCACF_OPERATION_TIME	3133	X'00000C3D'

Çizelge 40. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQCACF_ETKİNY_DESC	3134	X'00000C3E'
MQCACF_APPL_IDENTITY_DATA	3135	X'00000C3F'
MQCACF_APPL_ORIGIN_DATA	3136	X'00000C40'
MQCACF_PUT_DATE	3137	X'00000C41'
MQCACF_PUT_TIME	3138	X'00000C42'
MQCACF_REPLY_TO_Q	3139	X'00000C43'
MQCACF_REPLY_TO_Q_MGR	3140	X'00000C44'
MQCACF_RESOLVED_Q_NAME	3141	X'00000C45'
MQCACF_STRUC_ID	3142	X'00000C46'
MQCACF_VALUE_NAME	3143	X'00000C47'
MQCACF_SERVICE_START_DATE	3144	X'00000C48'
MQCACF_SERVICE_START_TIME	3145	X'00000C49'
MQCACF_SYSP_OFFLINE_RBA	3146	X'00000C4A'
MQCACF_SYSP_ARCHIVE_PFX2	3147	X'00000C4B'
MQCACF_SYSP_ARCHIVE_UNIT2	3148	X'00000C4C'
MQCACF_TO_TOPIC_NAME	3149	X'00000C4D'
MQCACF_FROM_TOPIC_NAME	3150	X'00000C4E'
MQCACF_TOPIC_NAMES	3151	X'00000C4F'
MQCACF_SU_ADı	3152	X'00000C50'
MQCACF_DESTINATION_Q_MGR	3153	X'00000C51'
MQCACF_DESTIND	3154	X'00000C52'
MQCACF_SUB_USER_ID	3156	X'00000C54'
MQCACF_SUB_USER_DATA	3159	X'00000C57'
MQCACF_SUB_SELECTOR	3160	X'00000C58'
MQCACF_LAST_PUB_DATE	3161	X'00000C59'
MQCACF_LAST_PUB_TIME	3162	X'00000C5A'
MQCACF_FROM_ALT ADı	3163	X'00000C5B'
MQCACF_TO_ALT ADı	3164	X'00000C5C'
MQCACF_LAST_MSG_TIME	3167	X'00000C5F'
MQCACF_LAST_MSG_DATE	3168	X'00000C60'
MQCACF_SUBSCRIPTION_POINT	3169	X'00000C61'
MQCACF_FILTER	3170	X'00000C62'
MQCACF_NONE	3171	X'00000C63'
MQCACF_ADMIN_TOPIC_NAMES	3172	X'00000C64'
MQCACF_ROUTING_FINGER_PRINT	3173	X'00000C65'
MQCACF_APPL_DESC	3174	X'00000C66'
MQCACF_Q_MGR_START_DATE	3175	X'00000C67'
MQCACF_Q_MGR_START_TIME	3176	X'00000C68'
MQCACF_FROM_COMM_INFO_NAME	3177	X'00000C69'

Çizelge 40. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQCACF_TO_COMM_INFO_NAME	3178	X'00000C6A'
MQCACF_CF_OFFLOAD_SIZE1	3179	X'00000C6B'
MQCACF_CF_OFFLOAD_SIZE2	3180	X'00000C6C'
MQCACF_CF_OFFLOAD_SIZE3	3181	X'00000C6D'
MQCACF_CF_SMDS_GENERIC_AD1	3182	X'00000C6E'
MQCACF_CF_SMDS	3183	X'00000C6F'
MQCACF_RECOVERY_DATE	3184	X'00000C70'
MQCACF_RECOVERY_TIME	3185	X'00000C71'
MQCACF_CF_SMDSCONN	3186	X'00000C72'
MQCACF_CF_STRUC_NAME	3187	X'00000C73'
MQCACF_ALTERNATE_USERID	3188	X'00000C74'
MQCACF_CHAR_ATTRS	3189	X'00000C75'
MQCACF_DYNAMIC_Q_NAME	3190	X'00000C76'
MQCACF_HOST_NAME	3191	X'00000C77'
MQCACF_MQCB_NAME	3192	X'00000C78'
MQCACF_OBJECT_STRING	3193	X'00000C79'
MQCACF_RESOLVED_LOCAL_Q_MGR	3194	X'00000C7A'
MQCACF_RESOLVED_LOCAL_Q_NAME	3195	X'00000C7B'
MQCACF_RESOLVED_OBJECT_STRING	3196	X'00000C7C'
MQCACF_RESOLVED_Q_MGR	3197	X'00000C7D'
MQCACF_SELECTION_STRING	3198	X'00000C7E'
MQCACF_XA_INFO	3199	X'00000C7F'
MQCACF_APPL_FUNC	3200	X'00000C80'
MQCACF_XQH_REMOTE_Q_NAME	3201	X'00000C81'
MQCACF_XQH_REMOTE_Q_MGR	3202	X'00000C82'
MQCACF_XQH_PUT_TIME	3203	X'00000C83'
MQCACF_XQH_PUT_DATE	3204	X'00000C84'
MQCACF_EXCHL_OPERATOR_MESSAGES	3205	X'00000C85'
MQCACF_CSP_USER_IDENTIFIER	3206	X'00000C86'
MQCACF_AMQP_CLIENT_ID	3207	X'00000C87'
MQCACF_ARCHIV_LOG_EXTENT_NAME	3208	X'00000C88'
MQCACF_APPL_IMMOVABLE_DATE	3209	X'00000C89'
MQCACF_APPL_IMMOVABLE_TIME	3210	X'00000C8A'
MQCACF_NHA_INSTANCE_NAME	3211	X'00000C8B'
MQCACF_LAST_KULLANILDI	3211	X'00000C8B'

## MQCACH\_\* (Komut biçimi Karakter Kanalı Parametresi Tipleri)

Çizelge 41. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQCACH_FIRST	3501	X'00000DAD'
MQCACH_CHANNEL_NAME	3501	X'00000DAD'
MQCACH_DESC	3502	X'00000DAE'
MQCACH_MODE_NAME	3503	X'00000DAF'
MQCACH_TP_ADı	3504	X'00000DB0'
MQCACH_XMIT_Q_ADı	3505	X'00000DB1'
MQCACH_CONNECTION_NAME	3506	X'00000DB2'
MQCACH_MCA_ADı	3507	X'00000DB3'
MQCACH_SEC_EXIT_NAME	3508	X'00000DB4'
MQCACH_MSG_EXIT_NAME	3509	X'00000DB5'
MQCACH_SEND_EXIT_NAME	3510	X'00000DB6'
MQCACH_RCV_EXIT_NAME	3511	X'00000DB7'
MQCACH_CHANNEL_ADS	3512	X'00000DB8'
MQCACH_SEC_EXIT_USER_DATA	3513	X'00000DB9'
MQCACH_MSG_EXIT_USER_DATA	3514	X'00000DBA'
MQCACH_SEND_EXIT_USER_DATA	3515	X'00000DBB'
MQCACH_RCV_EXIT_USER_DATA	3516	X'00000DBC'
MQCACH_USER_ID	3517	X'00000DBD'
MQCACH_PAROLASı	3518	X'00000DBE'
MQCACH_LOCAL_ADDRESS	3520	X'00000DC0'
MQCACH_LOCAL_NAME	3521	X'00000DC1'
MQCACH_LAST_MSG_TIME	3524	X'00000DC4'
MQCACH_LAST_MSG_DATE	3525	X'00000DC5'
MQCACH_MCA_USER_ID	3527	X'00000DC7'
MQCACH_CHANNEL_START_TIME	3528	X'00000DC8'
MQCACH_CHANNEL_START_DATE	3529	X'00000DC9'
MQCACH_MCA_JOB_NAME	3530	X'00000DCA'
MQCACH_LAST_LUWID	3531	X'00000DCB'
MQCACH_CURRENT_LUWID	3532	X'00000DCC'
MQCACH_FORMAT_NAME	3533	X'00000DCD'
MQCACH_MR_EXIT_NAME	3534	X'00000DCE'
MQCACH_MR_EXIT_USER_DATA	3535	X'00000DCF'
MQCACH_SSL_CIPHER_SPEC	3544	X'00000DD8'
MQCACH_SSL_PEER_NAME	3545	X'00000DD9'
MQCACH_SSL_HANDSHAKE_STAGE	3546	X'00000DDA'
MQCACH_SSL_SHORT_PEER_NAME	3547	X'00000ddb'
MQCACH_REMOTE_APPL_TAG	3548	X'00000DDC'
MQCACH_SSL_CERT_USER_ID	3549	X'00000DDD'

Çizelge 41. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQCACH_SSL_CERT_ISSUER_NAME	3550	X'00000DDE'
MQCACH_LU_NAME	3551	X'00000DDF'
MQCACH_IP_ADDRESS	3552	X'00000DE0'
MQCACH_TCP_ADİ	3553	X'00000DE1'
MQCACH_LISTENER_NAME	3554	X'00000DE2'
MQCACH_LISTENER_DESC	3555	X'00000DE3'
MQCACH_LISTENER_START_DATE	3556	X'00000DE4'
MQCACH_LISTENER_START_TIME	3557	X'00000DE5'
MQCACH_SSL_KEY_RESET_DATE	3558	X'00000DE6'
MQCACH_SSL_KEY_RESET_TIME	3559	X'00000DE7'
MQCACH_LAST_KULLANILDI	3559	X'00000DE7'

### MQCADSD\_\* (CICS bilgi üstbilgisi ADS açıklayıcıları)

Çizelge 42. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQCADSD_NONE	0	X'00000000'
MQCADSD_SEND	1	X'00000001'
MQCADSD_RECV	16	X'00000010'
MQCADSD_MSGSAYI	256	X'00000100'

### MQCAFTY\_\* (Bağlantı İlişkisi Değerleri)

Çizelge 43. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQCAFTY_NONE	0	X'00000000'
MQCAFTY_PREFERDI	1	X'00000001'

### MQCAMO\_\* (Komut biçimi Karakter İzleme Parametresi Tipleri)

Çizelge 44. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQCAMO_FIRST	2701	X'00000A8D'
MQCAMO_CLOSE_DATE	2701	X'00000A8D'
MQCAMO_CLOSE_TIME	2702	X'00000A8E'
MQCAMO_CONN_DATE	2703	X'00000A8F'
MQCAMO_CONN_TIME	2704	X'00000A90'
MQCAMO_DISC_DATE	2705	X'00000A91'
MQCAMO_DISC_TIME	2706	X'00000A92'
MQCAMO_END_DATE	2707	X'00000A93'
MQCAMO_END_TIME	2708	X'00000A94'
MQCAMO_OPEN_DATE	2709	X'00000A95'
MQCAMO_OPEN_TIME	2710	X'00000A96'

Çizelge 44. Değişmezlerin değerleri (devamı var)		
Ad	Ondalık değer	Onaltılı değer
MQCAMO_START_DATE	2711	X'00000A97'
MQCAMO_START_TIME	2712	X'00000A98'
MQCAMO_LAST_SED	2712	X'00000A98'

### MQCBC\_\* (MQCBC değişmezleri yapısı)

Çizelge 45. Değişmezlerin yapıları	
Ad	Yapı
MQCBC_STRUC_ID	"CBC~"
MQCBC_STRUC_ID_ARRAY	'C', 'B', 'C', '~'

**Not:** ~ simgesi tek bir boş karakteri temsil eder.

Çizelge 46. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQCBC_VERSION_1	1	X'00000001'
MQCBC_CURRENT_VERSION	1	X'00000001'

### MQCBCF\_\* (MQCBC değişmezleri işaretleri)

Çizelge 47. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQCBCF_NONE	0	X'00000000'
MQCBCF_READA_BUFFER_BOŞ	1	X'00000001'

### MQCBCT\_\* (MQCBC değişmezleri Callback tipi)

Çizelge 48. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQCBCT_START_CALL	1	X'00000001'
MQCBCT_STOP_CALL	2	X'00000002'
MQCBCT_REGISTER_CALL	3	X'00000003'
MQCBCT_DEREGISTIER_CALL	4	X'00000004'
MQCBCT_EVENT_CALL	5	X'00000005'
MQCBCT_MSG_REMOVED	6	X'00000006'
MQCBCT_MSG_NOT_REMOVED	7	X'00000007'

### MQCBD\_\* (MQCBD değişmezleri yapısı)

Çizelge 49. Değişmezlerin yapıları	
Ad	Yapı
MQCBD_STRUC_ID	"CBD~"
MQCBD_STRUC_ID_ARRAY	'C', 'B', 'D', '~'

**Not:** ~ simgesi tek bir boş karakteri temsil eder.

Çizelge 50. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQCBD_VERSION_1	1	X'00000001'
MQCBD_CURRENT_VERSION	1	X'00000001'

### MQCBDO\_\* (MQCBD değişmezleri Geri Çağırma Seçenekleri)

Çizelge 51. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQCBDO_NONE	0	X'00000000'
MQCBDO_START_CALL	1	X'00000001'
MQCBDO_STOP_CALL	4	X'00000004'
MQCBDO_REGISTER_CALL	256	X'00000100'
MQCBDO_DEREGISTIER_CALL	512	X'00000200'
MQCBDO_FAIL_IF_QUIESCING	8192	X'00002000'

### MQCBO\_\* (mqCreateBag için Oluştur-Bag Seçenekleri)

Çizelge 52. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQCBO_NONE	0	X'00000000'
MQCBO_USER_BAG	0	X'00000000'
MQCBO_ADMIN_BAG	1	X'00000001'
MQCBO_KOMUT_BAG	16	X'00000010'
MQCBO_SYSTEM_BAG	32	X'00000020'
MQCBO_GROUP_BAG	64	X'00000040'
MQCBO_LIST_FORM_ALLOWD	2	X'00000002'
MQCBO_LIST_FORM_INHIBITED	0	X'00000000'
MQCBO_REORDER_AS_REQUIREMS	4	X'00000004'
MQCBO_DO_NOT_REORDER	0	X'00000000'
MQCBO_CHECK_SELECTS	8	X'00000008'
MQCBO_DO_NOT_CHECK_SELECTS	0	X'00000000'

### MQCBT\_\* (MQCBD değişmezleri bu, Callback İşlevinin tipidir)

Çizelge 53. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQCBT_MESSAGE_CONSUMER	1	X'00000001'
MQCBT_EVENT_HANDLER	2	X'00000002'

### MQCC\_\* (tamamlanma kodları)

Çizelge 54. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQCC_OK	0	X'00000000'
MQCC_UYARI	1	X'00000001'



Çizelge 54. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQCC_FAILED	2	X'00000002'
MQCC_BILINMIYOR	-1	X'FFFFFFFF'

### MQCCSI\_\* (Kodlanmış Karakter Kümesi Tanıtıcıları)

Çizelge 55. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQCCSI_UNDEDINED	0	X'00000000'
MQCCSI_VARSA	0	X'00000000'
MQCCSI_Q_MGR	0	X'00000000'
MQCCSI_INHERIT	-2	X'FFFFFFFE'
MQCCSI_EMBEDDED	-1	X'FFFFFFFF'
MQCCSI_APPL	-3	X'FFFFFFFD'





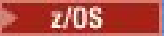

### MQCCT\_\* (CICS bilgi üstbilgisi Conversasyonel Görev Seçenekleri)


Çizelge 56. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQCCT_YES	1	X'00000001'
MQCCT_NO	0	X'00000000'

### MQCD\_\* (Kanal tanımlama yapısı)

Çizelge 57. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQCD_VERSION_1	1	X'00000001'
MQCD_VERSION_2	2	X'00000002'
MQCD_VERSION_3	3	X'00000003'
MQCD_VERSION_4	4	X'00000004'
MQCD_VERSION_5	5	X'00000005'
MQCD_VERSION_6	6	X'00000006'
MQCD_VERSION_7	7	X'00000007'
MQCD_VERSION_8	8	X'00000008'
MQCD_VERSION_9	9	X'00000009'
MQCD_VERSION_10	10	X'0000000A'
 MQCD_VERSION_11	11	X'0000000B'
 MQCD_CURRENT_VERSION	11	X'0000000B'
  MQCD_VERSION_12	12	X'0000000C'
  MQCD_CURRENT_VERSION	12	X'0000000C'
MQCD_LENGTH_4	(value differs by platform or version)	(value differs by platform or version)

Çizelge 57. Değişmezlerin değerleri (devamı var)		
Ad	Ondalık değer	Onaltılı değer
MQCD_LENGTH_5	(value differs by platform or version)	(value differs by platform or version)
MQCD_LENGTH_6	(value differs by platform or version)	(value differs by platform or version)
MQCD_LENGTH_7	(value differs by platform or version)	(value differs by platform or version)
MQCD_LENGTH_8	(value differs by platform or version)	(value differs by platform or version)
MQCD_LENGTH_9	(value differs by platform or version)	(value differs by platform or version)
MQCD_LENGTH_10	(value differs by platform or version)	(value differs by platform or version)
MQCD_LENGTH_11	(value differs by platform or version)	(value differs by platform or version)
 MQCD_LENGTH_12	(value differs by platform or version)	(value differs by platform or version)
MQCD_CURRENT_LENGTH	(value differs by platform or version)	(value differs by platform or version)

### MQCDC\_ \* (Kanal Veri Dönüştürme)

Çizelge 58. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQCDC_SENDER_CONVERSION	1	X'00000001'
MQCDC_NO_SENDER_CONVERSION	0	X'00000000'

### MQCERT\_ \* (Sertifika Geçerlilik Denetimi İlkesi Tipi)

MQ_CERT_VAL_POLICY_DEFAULT	0	X'00000000'
MQ_CERT_VAL_POLICY_ANY	0	X'00000000'
MQ_CERT_VAL_POLICY_RFC5280	1	X'00000001'

### MQCF\_ \* (Yetenek İşaretleri)

Çizelge 59. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQCF_NONE	0	X'00000000'
MQCF_DIST_LISTS	1	X'00000001'

### MQCFAC\_ \* (CICS bilgi üstbilgisi Olanakları)

Çizelge 60. Değişmez adları ve değerleri	
Ad	Onaltılı değer
MQCFAC_NONE	X'00...00' (8 boş değer)
MQCFAC_NONE_DIZISI	'\0', '\0', ... (8 boş değer)

## MQCFBF\_\* (Komut biçimi byte dizgisi süzgeç deęiřtirgesi yapısı)

Çizelge 61. Deęiřmezlerin deęerleri		
Ad	Ondalık deęer	Onaltılı deęer
MQCFBF_STRUC_LENGTH_FIXED	20	X'00000014'

## MQCFBS\_\* (Komut biçimi byte dizgisi deęiřtirgesi yapısı)

Çizelge 62. Deęiřmezlerin deęerleri		
Ad	Ondalık deęer	Onaltılı deęer
MQCFBS_STRUC_LENGTH_FIXED	16	X'00000010'

## MQCFC\_\* (Komut biçimi üstbilgi denetim seęenekleri)

Çizelge 63. Deęiřmezlerin deęerleri		
Ad	Ondalık deęer	Onaltılı deęer
MQCFC_SON	1	X'00000001'
MQCFC_NOT_SON	0	X'00000000'

## MQCFGR\_\* (Komut biçimi grup deęiřtirgesi yapısı)

Çizelge 64. Deęiřmezlerin deęerleri		
Ad	Ondalık deęer	Onaltılı deęer
MQCFGR_STRUC_LENGTH	16	X'00000010'

## MQCFH\_\* (Komut biçimi üstbilgi yapısı)

Çizelge 65. Deęiřmezlerin deęerleri		
Ad	Ondalık deęer	Onaltılı deęer
MQCFH_STRUC_LENGTH	36	X'00000024'
MQCFH_VERSION_1	1	X'00000001'
MQCFH_VERSION_2	2	X'00000002'
MQCFH_VERSION_3	3	X'00000003'
MQCFH_CURRENT_VERSION	3	X'00000003'

## MQCFIF\_\* (Komut biçimi tamsayı süzgeç deęiřtirgesi yapısı)

Çizelge 66. Deęiřmezlerin deęerleri		
Ad	Ondalık deęer	Onaltılı deęer
MQCFIFI_STRUCT_LENGTH	20	X'00000014'

## MQCFIL\_\* (Komut biçimi tamsayı listesi deęiřtirge yapısı)

Çizelge 67. Deęiřmezlerin deęerleri		
Ad	Ondalık deęer	Onaltılı deęer
MQCFIL_STRUCT_LENGTH_FIXED	16	X'00000010'

## MQCFIL64\_\* (Komut biçimi 64-bit tamsayı listesi değiştirge yapısı)

Çizelge 68. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQCFIL64_STRUC_LENGTH_FIXED	16	X'00000010'

## MQCFIN\_\* (Komut biçimi tamsayı değiştirgesi yapısı)

Çizelge 69. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQCFIN_STRUCT_LENGTH	16	X'00000010'

## MQCFIN64\_\* (Komut biçimi 64-bit tamsayı değiştirgesi yapısı)

Çizelge 70. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQCFIN64_STRUC_LENGTH	24	X'00000018'

## MQCFO\_\* (Komut biçimi Yenileme Havuzu Seçenekleri ve Komut biçimi Kuyruk Kaldırma Kuyrukları Seçenekleri)

### Komut Biçimi Yenileme Havuzu Seçenekleri

Çizelge 71. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQCFO_REFRESH_REPOSITORY_YES	1	X'00000001'
MQCFO_REFRESH_REPOSITORY_NO	0	X'00000000'

### Komut Biçimi Kaldırma Kuyrukları Seçenekleri

Çizelge 72. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQCFO_REMOVE_QUEUES_YES	1	X'00000001'
MQCFO_REMOVE_QUEUES_NO	0	X'00000000'

## MQCFOP\_\* (Komut biçimi Süzgeç İşleçleri)

Çizelge 73. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQCFOP_LESS	1	X'00000001'
MQCFOP_EQUAL	2	X'00000002'
MQCFOP_GREATER	4	X'00000004'
MQCFOP_NOT_LESS	6	X'00000006'
MQCFOP_NOT_EQUAL	5	X'00000005'
MQCFOP_NOT_GREATER	3	X'00000003'
MQCFOP_LIKE	18	X'00000012'
MQCFOP_NOT_LIKE	21	X'00000015'
MQCFOP_CONTAINS	10	X'0000000A'

Çizelge 73. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQCFOP_EXCLUDES	13	X'0000000D'
MQCFOP_CONTAINS_GEN	26	X'0000001A'
MQCFOP_EXCLUDES_GEN	29	X'0000001D'

### **MQCFR\_\* (CF Kurtarılabirliği)**

Çizelge 74. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQCFR_YES	1	X'00000001'
MQCFR_NO	0	X'00000000'

### **MQCFSF\_\* (Komut biçimi dizgi süzgeci değiştirge yapısı)**

Çizelge 75. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQCFSF_STRUC_LENGTH_FIXED	24	X'00000018'

### **MQCFSL\_\* (Komut biçimi dizgi listesi değiştirge yapısı)**

Çizelge 76. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQCFSL_STRUC_LENGTH_FIXED	24	X'00000018'

### **MQCFST\_\* (Komut biçimi dizgi değiştirgesi yapısı)**

Çizelge 77. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQCFST_STRUC_LENGTH_FIXED	20	X'00000014'

### **MQCFSTATUS\_\* (Komut biçimi CF Durumu)**

Çizelge 78. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQCFSTATUS_NOT_FOUND	0	X'00000000'
MQCFSTATUS_ETKİN	1	X'00000001'
MQCFSTATUS_IN_RECOVER	2	X'00000002'
MQCFSTATUS_IN_BACKUP	3	X'00000003'
MQCFSTATUS_FAILED	4	X'00000004'
MQCFSTATUS_NONE	5	X'00000005'
MQCFSTATUS_UNKNOWN	6	X'00000006'
MQCFSTATUS_ADMIN_TAMAMLANMAMIŞ	20	X'00000014'
MQCFSTATUS_NEVER_KULLANILDI	21	X'00000015'
MQCFSTATUS_NO_BACKUP	22	X'00000016'
MQCFSTATUS_NOT_FAILED	23	X'00000017'
MQCFSTATUS_NOT_RECOVERABLE	24	X'00000018'

Çizelge 78. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQCFSTATUS_XES_ERROR	25	X'00000019'

### **MQCFT\_\* (Structure Structure Of Structure)**

Çizelge 79. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQCFT_NONE	0	X'00000000'
MQCFT_COMMAND	1	X'00000001'
MQCFT_RESAYET	2	X'00000002'
MQCFT_TINGER	3	X'00000003'
MQCFT_STRING	4	X'00000004'
MQCFT_INTEGER_LIST	5	X'00000005'
MQCFT_STRING_LIST	6	X'00000006'
MQCFT_EVENT	7	X'00000007'
MQCFT_USER	8	X'00000008'
MQCFT_BYTE_STRING	9	X'00000009'
MQCFT_TRACE_ROUTE	10	X'0000000A'
MQCFT_REPORT	12	X'0000000C'
MQCFT_INTEGER_FILTER	13	X'0000000D'
MQCFT_STRING_FILTER	14	X'0000000E'
MQCFT_BYTE_STRING_FILTER	15	X'0000000F'
MQCFT_COMMAND_XR	16	X'00000010'
MQCFT_XR_MSG	17	X'00000011'
MQCFT_XR_ITEM	18	X'00000012'
MQCFT_XR_SUMMARY	19	X'00000013'
MQCFT_GROUP	20	X'00000014'
MQCFT_ISTATISTIKLERI	21	X'00000015'
MQCFT_ACCOUNTING	22	X'00000016'
MQCFT_INTEGER64	23	X'00000017'
MQCFT_INTEGER64_LIST	25	X'00000019'

### **MQCFATYPE\_\* (Komut biçimi CF tipleri)**

Çizelge 80. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQCFATYPE_APPL	0	X'00000000'
MQCFATYPE_ADMIN	1	X'00000001'

## MQCFUNC\_\* ( CICS bilgi üstbilgisi işlevleri)

Çizelge 81. Değişmezlerin yapıları	
Ad	Yapı
MQCFUNC_MQCONN	"CONN"
MQCFUNC_MQGET	"GET~"
MQCFUNC_MQINQ	"INQ~"
MQCFUNC_MQAc	"OPEN"
MQCFUNC_MQPUT	"PUT~"
MQCFUNC_MQPUT1	"PUT1"
MQCFUNC_NONE	"~~~~"
MQCFUNC_MQCONN_ARRAY	'C','O','N','N'
MQCFUNC_MQGET_ARRAY	'G','E','T','~'
MQCFUNC_MQINQ_ARRAY	'I','N','Q','~'
MQCFUNC_MQOPEN_ARRAY	'O','P','E','N'
MQCFUNC_MQPUT_ARRAY	'P','U','T','~'
MQCFUNC_MQPUT1_ARRAY	'P','U','T','1'
MQCFUNC_NONE_DIZISI	'~','~','~','~'

**Not:** ~ simgesi tek bir boş karakteri temsil eder.

## MQCGWI\_\* ( CICS bilgi üstbilgisi Bekleme Aralığı AI)

Çizelge 82. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQCGWI_VARSAYILAN	-2	X'FFFFFFFE'

## MQCHAD\_\* (Kanal Otomatik Tanımlaması)

Çizelge 83. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQCHAD_ENGELLI	0	X'00000000'
MQCHAD_ENABLED	1	X'00000001'

## MQCHNiti\_\* (Komut biçimi Kuşku Durumu Durumu)

Çizelge 84. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQCHIDS_NOT_INDOUBT	0	X'00000000'
MQCHGINDOUBT	1	X'00000001'

## MQCHLD\_\* (Komut biçimi Kanal Atımları)

Çizelge 85. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQCHLD_ALL	-1	X'FFFFFFF'
MQCHLD_VARSAYILAN	1	X'00000001'

Çizelge 85. Değişmezlerin değerleri (devamı var)		
Ad	Ondalık değer	Onaltılı değer
MQCHLD_SHARED	2	X'00000002'
MQCHLD_PRIVATE	4	X'00000004'
MQCHLD_FIXSHARED	5	X'00000005'

### **MQCHS\_\* (Komut biçimi Kanal Durumu)**

Çizelge 86. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQCHS_INACTIVE	0	X'00000000'
MQCHS_BINDING	1	X'00000001'
MQCHS_STARTING	2	X'00000002'
MQCHS_RUNNING	3	X'00000003'
MQCHS_STOPPING	4	X'00000004'
MQCHS_RETRYING	5	X'00000005'
MQCHS_STOPPED	6	X'00000006'
MQCHS_REQUESTING	7	X'00000007'
MQCHS_PAUSED	8	X'00000008'
MQCHS_INITIALIZING	13	X'0000000D'
MQCHS_SWITCHING	14	X'0000000E'

### **MQCHSH\_\* (Komut biçimi Kanal Paylaşılan Yeniden Başlatma Seçenekleri)**

Çizelge 87. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQCHSH_RESTART_NO	0	X'00000000'
MQCHSH_RESTART_YES	1	X'00000001'

### **MQCHSR\_\* (Komut biçimi Kanal Durdurma Seçenekleri)**

Çizelge 88. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQCHSR_STOP_NOT_REQUESTED	0	X'00000000'
MQCHSR_STOP_REQUESTED	1	X'00000001'

### **MQCHSSTATE\_\* (Komut biçimi Kanal Alt Devletleri)**

Çizelge 89. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQCHSSTATE_DİĞER	0	X'00000000'
MQCHSSTATE_END_OF_BATCH	100	X'00000064'
MQCHSSTATE_SENDING	200	X'000000C8'
MQCHSSTATE_ALMA	300	X'0000012C'
MQCHSSTATE_SERIALIZE	400	X'00000190'
MQCHSSTATE_RESYNCHING	500	X'000001F4'



Çizelge 89. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQCHSSTATE_HEARTHEAT	600	X'00000258'
MQCHSSTATE_IN_SCYEXIT	700	X'000002BC'
MQCHSSTATE_IN_RCVEXIT	800	X'00000320'
MQCHSSTATE_IN_SENDEXIT	900	X'00000384'
MQCHSSTATE_IN_MSGEXIT	1000	X'000003E8'
MQCHSSTATE_IN_MREXIT	1100	X'0000044C'
MQCHSSTATE_IN_CHADEXIT	1200	X'000004B0'
MQCHSSTATE_NET_BAĞLANTI	1250	X'000004E2'
MQCHSSTATE_SSL_HANDSDSHAING	1300	X'00000514'
MQCHSSTATE_NAME_SERVER	1400	X'00000578'
MQCHSSTATE_IN_MQPUT	1500	X'000005DC'
MQCHSSTATE_IN_MQGET	1600	X'00000640'
MQCHSSTATE_IN_MQI_CALL	1700	X'000006A4'
MQCHSSTATE_COMPRESSING	1800	X'00000708'

### MQCHT\_\* (Kanal Tipleri)

Çizelge 90. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQCHT_SENDER	1	X'00000001'
MQCHT_SERVER	2	X'00000002'
MQCHT_RECEI	3	X'00000003'
MQCHT_REQUESTER	4	X'00000004'
MQCHT_ALL	5	X'00000005'
MQCHT_CLNTCONN	6	X'00000006'
MQCHT_SVRCONN	7	X'00000007'
MQCHT_CLAUSRCVR	8	X'00000008'
MQCHT_CLUSSDR	9	X'00000009'

### MQCHTAB\_\* (Komut biçimi Kanal Çizelgesi Tipleri)

Çizelge 91. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQCHTAB_Q_MGR	1	X'00000001'
MQCHTAB_CLNTCONN	2	X'00000002'

### MQCI\_\* (İlinti Tanıtıcısı)

Çizelge 92. Değişmez adlar ve değerler

Ad	Değer
MQCI_NONE	X'00...00' (24 boş değer)
MQCI_NONE_DIZISI	'\0', '\0', ... (24 boş değer)
MQCI_NEW_SESSION	X'41405121...'

Çizelge 92. Değişmez adlar ve değerler (devamı var)	
Ad	Değer
MQCI_NEW_SESSION_ARRAY	'\x41', '\x4D', '\51', '\x21', ...

## MQCIH\_\* (CICS bilgi üstbilgisi yapısı ve işaretleri)

### CICS bilgi üstbilgisi yapısı

Çizelge 93. Değişmezlerin yapıları	
Ad	Yapı
MQCIH_STRUC_ID	"CIH~"
MQCIH_STRUC_ID_ARRAY	'C', 'I', 'H', '~'

**Not:** ~ simgesi tek bir boş karakteri temsil eder.

Çizelge 94. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQCIH_VERSION_1	1	X'00000001'
MQCIH_VERSION_2	2	X'00000002'
MQCIH_CURRENT_VERSION	2	X'00000002'
MQCIH_LENGTH_1	164	X'000000A4'
MQCIH_LENGTH_2	180	X'000000B4'
MQCIH_CURRENT_LENGTH	180	X'000000B4'

### CICS bilgi üstbilgisi işaretleri

Çizelge 95. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQCIH_NONE	0	X'00000000'
MQCIH_PASS_EXPECTION	1	X'00000001'
MQCIH_UNlimiED_EXPIRATION	0	X'00000000'
MQCIH_REPLY_WITHOUT_NULLS	2	X'00000002'
MQCIH_REPLY_WITH_NULLS	0	X'00000000'
MQCIH_SYNC_ON_RETURN	4	X'00000004'
MQCIH_NO_SYNC_ON_RETURN	0	X'00000000'

### MQCLCT\_\* (Küme Önbelleği Tipleri)

Çizelge 96. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQCLCT_STATIC	0	X'00000000'
MQCLCT_DYNAMIC	1	X'00000001'

## MQCLRS\_\* (Komut biçimi Konu Dizgisi Kapsamını Temizle)

Çizelge 97. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQCLRS_LOCAL	1	X'00000001'
MQCLRS_GLOBAL	2	X'00000002'

## MQCLRT\_\* (Komut biçimi Konu Dizgisi Tipini Temizle)

Çizelge 98. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQCLRT_ALIKONUR	1	X'00000001'

## MQCLT\_\* (CICS bilgi üstbilgisi Bağlantı Tipleri)

Çizelge 99. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQCLT_PROGRAM	1	X'00000001'
MQCLT_TRANSACTION	2	X'00000002'

## MQCLWL\_\* (Küme İş Yüğü)

Çizelge 100. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQCLWL_USEQ_LOCAL	0	X'00000000'
MQCLWL_USEQ_ANY	1	X'00000001'
MQCLWL_USEQ_AS_Q_MGR	-3	X'FFFFFFFD'

## MQCLXQ\_\* (Küme iletim kuyruğu tipi)

MQCLXQ\_\*, DEFCLXQ kuyruk yöneticisi öznitelide ayarlayabileceğiniz değerlerdir. DEFCLXQ özniteliği, iletileri küme alıcı kanallarına göndermek, iletileri almak için küme gönderen kanalları tarafından varsayılan olarak hangi iletim kuyruğunun seçileceğini denetler.

Çizelge 101. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQCLXQ_SCTQ	0	X'00000000'
MQCLXQ_CHANNEL	1	X'00000001'

### İlgili başvurular

“DefClusterXmitQueueTipi (MQUBE)” sayfa 802

DefClusterXmitQueueType özniteliği, iletileri küme alıcı kanallarına göndermek için küme gönderen kanalları tarafından varsayılan olarak hangi iletim kuyruğunun seçileceğini denetler.

Kuyruk Yöneticisini Değiştir

Sorgu Kuyruğu Yöneticisi

Sorgu Kuyruğu Yöneticisi (Yanıt)

“MQINQ-Nesne özniteliklerinin sorgulması” sayfa 693

MQINQ çağrısı, bir tamsayılar dizisi ve bir nesnenin özniteliklerini içeren bir karakter dizgileri kümesini döndürür.

## MQCMD\_\* (Komut Kodları)

Çizelge 102. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQCMD_NONE	0	X'00000000'
MQCMD_CHANGE_Q_MGR	1	X'00000001'
MQCMD_INQUIRE_Q_MGR	2	X'00000002'
MQCMD_CHANGE_PROCESS	3	X'00000003'
MQCMD_COPY_PROCESS	4	X'00000004'
MQCMD_CREATE_PROCESS	5	X'00000005'
MQCMD_DELETE_PROCESS	6	X'00000006'
MQCMD_INQUIRE_PROCESS	7	X'00000007'
MQCMD_CHANGE_Q	8	X'00000008'
MQCMD_CLEAR_Q	9	X'00000009'
MQCMD_COPY_Q	10	X'0000000A'
MQCMD_CREATE_Q	11	X'0000000B'
MQCMD_DELETE_Q	12	X'0000000C'
MQCMD_INQUIRE_Q	13	X'0000000D'
MQCMD_REFRESH_Q_MGR	16	X'00000010'
MQCMD_RESET_Q_STATS	17	X'00000011'
MQCMD_INQUIRE_Q_ADS	18	X'00000012'
MQCMD_INQUIRE_PROCESS_NAMES	19	X'00000013'
MQCMD_INQUIRE_CHANNEL_ADS	20	X'00000014'
MQCMD_CHANGE_CHANNEL	21	X'00000015'
MQCMD_COPY_CHANNEL	22	X'00000016'
MQCMD_CREATE_CHANNEL	23	X'00000017'
MQCMD_DELETE_CHANNEL	24	X'00000018'
MQCMD_INQUIRE_CHANNEL	25	X'00000019'
MQCMD_PING_KANAL	26	X'0000001A'
MQCMD_RESET_CHANNEL	27	X'0000001B'
MQCMD_START_CHANNEL	28	X'0000001C'
MQCMD_STOP_KANAL	29	X'0000001D'
MQCMD_START_CHANNEL_INIT	30	X'0000001E'
MQCMD_START_CHANNEL_LISTENER	31	X'0000001F'
MQCMD_CHANGE_NAMELIST	32	X'00000020'
MQCMD_COPY_NAMELIST	33	X'00000021'
MQCMD_CREATE_NAMELIST	34	X'00000022'
MQCMD_DELETE_NAMELIST	35	X'00000023'
MQCMD_INQUIRE_NAMELIST	36	X'00000024'
MQCMD_INQUIRE_NAMELIST_NAMES	37	X'00000025'
MQCMD_ESCAPE	38	X'00000026'
MQCMD_RESOLVE_CHANNEL	39	X'00000027'

Çizelge 102. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQCMD_PING_Q_MGR	40	X'00000028'
MQCMD_INQUIRE_Q_STATUS	41	X'00000029'
MQCMD_INQUIRE_CHANNEL_STATUS	42	X'0000002A'
MQCMD_CONFIG_OLAY	43	X'0000002B'
MQCMD_Q_MGR_OLAY	44	X'0000002C'
MQCMD_PERFM_OLAY	45	X'0000002D'
MQCMD_CHANNEL_OLAY	46	X'0000002E'
MQCMD_DELETE_YAYINI	60	X'0000003C'
MQCMD_DEREGISTER_PUBLISHER	61	X'0000003D'
MQCMD_DEREGISTER_SUBSCRIBER	62	X'0000003E'
MQCMD_YAYINLAMA	63	X'0000003F'
MQCMD_REGISTER_PUBLISHER	64	X'00000040'
MQCMD_REGISTER_ABONESI	65	X'00000041'
MQCMD_REQUEST_UPDATE	66	X'00000042'
MQCMD_BROKER_INTERNAL	67	X'00000043'
MQCMD_ETKINLIY_MSG	69	X'00000045'
MQCMD_INQUIRE_CLUSTER_Q_MGR	70	X'00000046'
MQCMD_RESUME_Q_MGR_CLUSTER	71	X'00000047'
MQCMD_SUSPEND_Q_MGR_CLUSTER	72	X'00000048'
MQCMD_REFRESH_CLUSTER	73	X'00000049'
MQCMD_RESET_CLUSTER	74	X'0000004A'
MQCMD_IZLEME_YOLU	75	X'0000004B'
MQCMD_REFRESH_SECURITY	78	X'0000004E'
MQCMD_CHANGE_AUTH_INFO	79	X'0000004F'
MQCMD_COPY_AUTH_INFO	80	X'00000050'
MQCMD_CREATE_AUTH_INFO	81	X'00000051'
MQCMD_DELETE_AUTH_INFO	82	X'00000052'
MQCMD_INQUIRE_AUTH_INFO	83	X'00000053'
MQCMD_INQUIRE_AUTH_INFO_NAMES	84	X'00000054'
MQCMD_INQUIRE_CONNECTION	85	X'00000055'
MQCMD_STOP_CONNECTION	86	X'00000056'
MQCMD_INQUIRE_AUTH_RECS	87	X'00000057'
MQCMD_INQUIRE_ENTITY_AUTH	88	X'00000058'
MQCMD_DELETE_AUTH_REC	89	X'00000059'
MQCMD_SET_AUTH_REC	90	X'0000005A'
MQCMD_LOGGER_EVENT	91	X'0000005B'
MQCMD_RESET_Q_MGR	92	X'0000005C'
MQCMD_CHANGE_LISTENER	93	X'0000005D'
MQCMD_COPY_DINLEYICISI	94	X'0000005E'

Çizelge 102. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQCMD_CREATE_LISTENER	95	X'0000005F'
MQCMD_DELETE_LISTENER	96	X'00000060'
MQCMD_INQUIRE_LISTENER	97	X'00000061'
MQCMD_INQUIRE_LISTENER_STATUS	98	X'00000062'
MQCMD_COMMAND_OLAY	99	X'00000063'
MQCMD_CHANGE_SECURITY	100	X'00000064'
MQCMD_CHANGE_CF_STRUC	101	X'00000065'
MQCMD_CHANGE_STG_CLASS	102	X'00000066'
MQCMD_CHANGE_TRACE	103	X'00000067'
MQCMD_ARCHIV_LOG	104	X'00000068'
MQCMD_BACKUP_CF_STRUC	105	X'00000069'
MQCMD_CREATE_BUFFER_POOL	106	X'0000006A'
MQCMD_CREATE_PAGE_SET	107	X'0000006B'
MQCMD_CREATE_CF_STRUC	108	X'0000006C'
MQCMD_CREATE_STG_CLASS	109	X'0000006D'
MQCMD_COPY_CF_STRUC	110	X'0000006E'
MQCMD_COPY_STG_CLASS	111	X'0000006F'
MQCMD_DELETE_CF_STRUC	112	X'00000070'
MQCMD_DELETE_STG_CLASS	113	X'00000071'
MQCMD_INQUIRE_ARCHIVE	114	X'00000072'
MQCMD_INQUIRE_CF_STRUC	115	X'00000073'
MQCMD_INQUIRE_CF_STRUC_STATUS	116	X'00000074'
MQCMD_INQUIRE_CMD_SERVER	117	X'00000075'
MQCMD_INQUIRE_CHANNEL_INIT	118	X'00000076'
MQCMD_INQUIRE_QSG	119	X'00000077'
MQCMD_INQUIRE_LOG	120	X'00000078'
MQCMD_INQUIRE_SECURITY	121	X'00000079'
MQCMD_INQUIRE_STG_CLASS	122	X'0000007A'
MQCMD_INQUIRE_SYSTEM	123	X'0000007B'
MQCMD_INQUIRE_THREAD	124	X'0000007C'
MQCMD_INQUIRE_TRACE	125	X'0000007D'
MQCMD_INQUIRE_USAGE	126	X'0000007E'
MQCMD_MOVE_Q	127	X'0000007F'
MQCMD_RECOVER_BSDS	128	X'00000080'
MQCMD_RECOVER_CF_STRUC	129	X'00000081'
MQCMD_RESET_TPIPE	130	X'00000082'
MQCMD_RESOLVE_INDOUBT	131	X'00000083'
MQCMD_RESUME_Q_MGR	132	X'00000084'
MQCMD_REVERIFY_SECURITY	133	X'00000085'

Çizelge 102. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQCMD_SET_ARCHEVE	134	X'00000086'
MQCMD_SET_LOG	136	X'00000088'
MQCMD_SET_SYSTEM	137	X'00000089'
MQCMD_START_CMD_SERVER	138	X'0000008A'
MQCMD_START_Q_MGR	139	X'0000008B'
MQCMD_START_TRACE	140	X'0000008C'
MQCMD_STOP_CHANNEL_INIT	141	X'0000008D'
MQCMD_STOP_CHANNEL_LISTENER	142	X'0000008E'
MQCMD_STOP_CMD_SERVER	143	X'0000008F'
MQCMD_STOP_Q_MGR	144	X'00000090'
MQCMD_STOP_TRACE	145	X'00000091'
MQCMD_SUSPEND_Q_MGR	146	X'00000092'
MQCMD_INQUIRE_CF_STRUCTION_NAMES	147	X'00000093'
MQCMD_INQUIRE_STG_CLASS_NAMES	148	X'00000094'
MQCMD_CHANGE_HİZMET	149	X'00000095'
MQCMD_COPY_SERVI	150	X'00000096'
MQCMD_CREATE_SERVICE	151	X'00000097'
MQCMD_DELETE_HİZMET	152	X'00000098'
MQCMD_INQUIRE_SERVICE	153	X'00000099'
MQCMD_INQUIRE_SERVICE_STATUS	154	X'0000009A'
MQCMD_START_HİZMET	155	X'0000009B'
MQCMD_STOP_HİZMET	156	X'0000009C'
MQCMD_DELETE_BUFFER_POOL	157	X'0000009D'
MQCMD_DELETE_PAGE_SET	158	X'0000009E'
MQCMD_CHANGE_BUFFER_POOL	159	X'0000009F'
MQCMD_CHANGE_PAGE_SET	160	X'000000A0'
MQCMD_INQUIRE_Q_MGR_STATUS	161	X'000000A1'
MQCMD_CREATE_LOG	162	X'000000A2'
MQCMD_STATISTICS_MQI	164	X'000000A4'
MQCMD_STATISTICS_Q	165	X'000000A5'
MQCMD_STATISTICS_CHANNEL	166	X'000000A6'
MQCMD_ACCOUNTING_MQI	167	X'000000A7'
MQCMD_ACCOUNTING_Q	168	X'000000A8'
MQCMD_INQUIRE_AUTH_SERVICE	169	X'000000A9'
MQCMD_CHANGE_KONUSU	170	X'000000AA'
MQCMD_COPY_TOPIC	171	X'000000AB'
MQCMD_CREATE_TOPIC	172	X'000000AC'
MQCMD_DELETE_KONU	173	X'000000AD'
MQCMD_INQUIRE_TOPIC	174	X'000000AE'

Çizelge 102. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQCMD_INQUIRE_TOPIC_NAMES	175	X'000000AF'
MQCMD_INQUIRE_SUBSCRIPTION	176	X'000000B0'
MQCMD_CREATE_SUBSCRIPTION	177	X'000000B1'
MQCMD_CHANGE_SUBSCRIPTION	178	X'000000B2'
MQCMD_DELETE_SUBSCRIPTION	179	X'000000B3'
MQCMD_COPY_SUBSCRIPTION	181	X'000000B5'
MQCMD_INQUIRE_SUB_STATUS	182	X'000000B6'
MQCMD_INQUIRE_TOPIC_STATUS	183	X'000000B7'
MQCMD_CLEAR_TOPIC_STRING	184	X'000000B8'
MQCMD_INQUIRE_PUBSUB_STATUS	185	X'000000B9'
MQCMD_PURGE_CHANNEL	195	X'000000C3'

### MQCMDI\_\* (Komut biçimi Komut Bilgileri Değerleri)

Çizelge 103. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQCMDI_CMDScope_ACCEPT	1	X'00000001'
MQCMDI_CMDScope_GENERATÖR	2	X'00000002'
MQCMDI_CMDScope_COMPLETED	3	X'00000003'
MQCMDI_QSG_DISP_COMPLETED	4	X'00000004'
MQCMDI_KOMUT_KABUL edildi	5	X'00000005'
MQCMDI_CLUSTER_REQUEST_QUEUED	6	X'00000006'
MQCMDI_CHANNEL_INIT_STARTED	7	X'00000007'
MQCMDI_RECOVER_STARTED	11	X'0000000B'
MQCMDI_BACKUP_STARTED	12	X'0000000C'
MQCMDI_RECOVER_COMPLETED	13	X'0000000D'
MQCMDI_SEC_TIMER_ZERO	14	X'0000000E'
MQCMDI_REFRESH_CONFIGURATION	16	X'00000010'
MQCMDI_SEC_SIGNOFF_ERROR	17	X'00000011'
MQCMDI_IMS_BRIDGE_SUSPENDED	18	X'00000012'
MQCMDI_DB2_SUSPENDED	19	X'00000013'
MQCMDI_DB2_OBSOLETE_MSGS	20	X'00000014'
MQCMDI_SEC_MERAL	21	X'00000015'
MQCMDI_SEC_MIXEDCASE	22	X'00000016'

### MQCMDL\_\* (Komut Düzeyleri)

Çizelge 104. Değişmez adları ve değerleri

Ad	Değer
MQCMDL_LEVEL_800	800
MQCMDL_LEVEL_801	801
MQCMDL_LEVEL_802	802



Çizelge 104. Değişmez adlar ve değerler (devamı var)

Ad	Değer
MQCMDL_LEVEL_900	900
MQCMDL_LEVEL_901	901
MQCMDL_LEVEL_902	902
MQCMDL_LEVEL_903	903
MQCMDL_LEVEL_904	904
MQCMDL_LEVEL_905	905
MQCMDL_LEVEL_910	910
MQCMDL_LEVEL_912	912
MQCMDL_LEVEL_913	913
MQCMDL_LEVEL_914	914
MQCMDL_LEVEL_915	915
MQCMDL_LEVEL_920	920
MQCMDL_LEVEL_921	921
MQCMDL_LEVEL_922	922
MQCMDL_LEVEL_923	923
MQCMDL_LEVEL_924	924
MQCMDL_LEVEL_925	925

## MQCMHO\_\* (İleti işleme seçenekleri ve yapısı yarat)

### İleti tanıtıcısı seçenekleri yapısını yarat

Çizelge 105. Değişmezlerin yapıları

Ad	Yapı
MQCMHO_STRUC_ID	"CMHO"
MQCMHO_STRUC_ID_ARRAY	'C', 'M', 'H', 'O'

**Not:** ↵ simgesi tek bir boş karakteri temsil eder.

Çizelge 106. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQCMHO_VERSION_1	1	X'00000001'
MQCMHO_CURRENT_VERSION	1	X'00000001'

### İleti Tanıtıcısı Yaratma Seçenekleri

Çizelge 107. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQCMHO_DEFAULT_VALIDATION	0	X'00000000'
MQCMHO_NO_VALIDATION	1	X'00000001'
MQCMHO_VALIDATE	2	X'00000002'
MQCMHO_NONE	0	X'00000000'

## MQCNO\_\* (Bağlantı seçenekleri ve yapısı)

### Bağlantı seçenekleri yapısı

Çizelge 108. Değişmezlerin yapıları	
Ad	Yapı
MQCNO_STRUC_ID	"CNO~"
MQCNO_STRUC_ID_ARRAY	'C', 'N', 'O', '~'

**Not:** ~ simgesi tek bir boş karakteri temsil eder.

Çizelge 109. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQCNO_VERSION_1	1	X'00000001'
MQCNO_VERSION_2	2	X'00000002'
MQCNO_VERSION_3	3	X'00000003'
MQCNO_VERSION_4	4	X'00000004'
MQCNO_VERSION_5	5	X'00000005'
MQCE_CURRENT_VERSION	5	X'00000005'

### Bağlanma Seçenekleri

Çizelge 110. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQCNO_STANDARD_BINDING	0	X'00000000'
MQCNO_FASTPATH_BINDING	1	X'00000001'
MQCNO_SERIALIZE_CONN_TAG_Q_MGR	2	X'00000002'
MQCNO_SERIALIZE_CONN_TAG_QSG	4	X'00000004'
MQCNO_RESTRICT_CONN_TAG_Q_MGR	8	X'00000008'
MQCNO_RESTRICT_CONN_TAG_QSG	16	X'00000010'
MQCNO_HANDLE_SHARE_NONE	32	X'00000020'
MQCNO_HANDLE_SHARE_BLOCK	64	X'00000040'
MQCNO_HANDLE_SHARE_NO_BLOCK	128	X'00000080'
MQCNO_SHARED_BINDING	256	X'00000100'
MQCNO_ISOLATED_BINDING	512	X'00000200'
MQCNO_LOCAL_BINBINA	1024	X'00000400'
MQCNO_CLIENT_BINDING	2048	X'00000800'
MQCNO_ACCOUNTING_MQI_ENABLED	4096	X'00001000'
MQCNO_ACCOUNTING_MQI_DISABLLANT	8192	X'00002000'
MQCNO_ACCOUNTING_Q_ENABLED	16384	X'00004000'
MQCNO_ACCOUNTING_Q_DISABET	32768	X'00008000'
MQCNO_NO_CONV_SHARING	65536	X'00010000'
MQCNO_ALL_CONVS_SHARE	262144	X'00040000'
MQCNO_CD_FOR_OUTPUT_ONLY	524288	X'00080000'
MQCNO_USE_CD_SEÇİMİ	1048576	X'00100000'

Çizelge 110. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQCNO_RECONNECT	16777216	X'01000000'
MQCNO_RECONNECT_AS_DEF	0	X'00000000'
MQCNO_RECONNECT_DISABLE	33554432	X'02000000'
MQCNO_RECONNECT_Q_MGR	67108864	X'04000000'
MQCNO_ACTIVITY_TRACE_ENABLED	134217728	X'08000000'
MQCNO_ACTIVITY_TRACE_DISABLDI	268435456	X'10000000'
MQCNO_NONE	0	X'00000000'

### MQCO\_\* (Kapatma Seçenekleri)

Çizelge 111. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQCO_IMMEDIATE	0	X'00000000'
MQCO_NONE	0	X'00000000'
MQCO_DELETE	1	X'00000001'
MQCO_DELETE_PURGE	2	X'00000002'
MQCO_KEEP_SUB	4	X'00000004'
MQCO_REMOVE_SUB	8	X'00000008'
MQCO_QUIESCE	32	X'00000020'

### MQCODL\_\* (CICS bilgi üstbilgisi Çıkış Veri Uzunluğu)

Çizelge 112. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQCODL_AS_INPUT	-1	X'FFFFFFFF'

### MQCOMPRESS\_\* (Kanal Sıkıştırma)

Çizelge 113. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQCOMPRESS_NOT_VAR	-1	X'FFFFFFFF'
MQCOMPRESS_NONE	0	X'00000000'
MQCOMPRESS_RLE	1	X'00000001'
MQCOMPRESS_ZLIBFAST	2	X'00000002'
MQCOMPRESS_ZLIBHIGH	4	X'00000004'
MQCOMPRESS_SYSTEM	8	X'00000008'
MQCOMPRESS_ANY	268435455	X'0FFFFFFFF'

### MQCONNID\_\* (Bağlantı Tanıtıcısı)

Çizelge 114. Değişmez adlar ve değerler

Ad	Değer
MQCONNID_NONE	X'00...00' (24 boş değer)
MQCONNID_NONE_DIZISI	'\0', '\0', ... (24 boş değer)

## MQCOPY\_\* (Özellik Kopyası Seçenekleri)

Çizelge 115. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQCOPY_NONE	0	X'00000000'
MQCOPY_ALL	1	X'00000001'
MQCOPY_FORWY	2	X'00000002'
MQCOPY_YAYIN	4	X'00000004'
MQCOPY_REPLY	8	X'00000008'
MQCOPY_REPORT	16	X'00000010'
MQCOPY_DEFAULT	22	X'00000016'

## MQCQT\_\* (Küme Kuyruğu Tipleri)

Çizelge 116. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQCQT_LOCAL_Q	1	X'00000001'
MQCQL_ALIAS_Q	2	X'00000002'
MQCQT_REMOTE_Q	3	X'00000003'
MQCQT_Q_MGR_ALIAS	4	X'00000004'

## MQCRC\_\* (CICS bilgi üstbilgisi Dönüş Kodları)

Çizelge 117. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQCRC_OK	0	X'00000000'
MQCRC_CICS_EXEC_ERROR	1	X'00000001'
MQCRC_MQ_API_ERROR	2	X'00000002'
MQCRC_BRIDGE_ERROR	3	X'00000003'
MQCRC_BRIDGE_ABEND	4	X'00000004'
MQCRC_APPLICATION_ABEND	5	X'00000005'
MQCRC_SECURITY_ERROR	6	X'00000006'
MQCRC_PROGRAM_NOT_YOK	7	X'00000007'
MQCRC_BRIDGE_TIMEOUT	8	X'00000008'
MQCRC_TRANSID_NOT_VAR	9	X'00000009'

## MQCS\_\* (MQCBC değişmezleri tüketici durumu)

Çizelge 118. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQCS_NONE	0	X'00000000'
MQCS_ASKI_GEÇICI	1	X'00000001'
MQCS_SUSPENDED_USER_ACTION	2	X'00000002'
MQCS_ASKISI	3	X'00000003'
MQCS_DURDU	4	X'00000004'

## MQCSC\_\* ( CICS bilgi üstbilgisi Başlangıç Kodları)

Çizelge 119. Değişmezlerin yapıları	
Ad	Yapı
MQCSC_START	"S~~"
MQCSC_STARTDATA	"SD~~"
MQCSC_TERMINPUT	"TD~~"
MQCSC_NONE	"~~~"
MQCSC_START_ARRAY	'S', '~', '~', '~', '~'
MQCSC_STARTDATA_ARRAY	'S', 'D', '~', '~', '~'
MQCSC_TERMINPUT_ARRAY	'T', 'D', '~', '~', '~'
MQCSC_NONE_DIZISI	'~', '~', '~', '~', '~'

**Not:** ~ simgesi tek bir boş karakteri temsil eder.

## MQCSP\_\* (Bağlantı güvenlik değiştirgeleri yapısı ve kimlik denetimi tipleri)

### Bağlantı güvenliği parametreleri yapısı

Çizelge 120. Değişmezlerin yapıları	
Ad	Yapı
MQCSP_STRUC_ID	"CSP~"
MQCSP_STRUC_ID_ARRAY	'C', 'S', 'P', '~'

**Not:** ~ simgesi tek bir boş karakteri temsil eder.

Çizelge 121. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQCSP_VERSION_1	1	X'00000001'
MQCSP_XX_ENCODE_CASE_ONE current_version	1	X'00000001'

### Bağlantı güvenliği parametreleri Kimlik Doğrulama Tipleri

Çizelge 122. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQCSP_AUTH_NONE	0	X'00000000'
MQCSP_AUTH_USER_ID_AND_PWD	1	X'00000001'

## MQCSP\_\* (Komut Sunucusu Seçenekleri)

Çizelge 123. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQCSP_CONVERT_NO	0	X'00000000'
MQCSP_CONVERT_YES	1	X'00000001'
MQCSP_DLQ_NO	0	X'00000000'
MQCSP_DLQ_YES	1	X'00000001'

## MQCT\_\* (Kuyruk Yöneticisi Bağlantı Etiketi)

Çizelge 124. Değişmez adlar ve değerler	
Ad	Değer
MQCT_NONE	X'00...00' (128 boş değer)
MQCT_NONE_DIZISI	'\0', '\0', ... (128 boş değer)

## MQCTES\_\* (CICS bilgi üstbilgisi Görev Bitiş Durumu)

Çizelge 125. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQCTES_NOSYNC	0	X'00000000'
MQCTES_COMMIT	256	X'00000100'
MQCTES_BACKUT	4352	X'00001100'
MQCTES_ENDTASK	65536	X'00010000'

## MQCTLO\_\* (MQCTL seçenekleri yapısı ve Tüketici Denetimi Seçenekleri)

### MQCTL seçenekleri yapısı

Çizelge 126. Değişmezlerin yapıları	
Ad	Yapı
MQCTLO_STRUC_ID	"CTLO"
MQCTLO_STRUC_ID_ARRAY	'C', 'T', 'L', 'O'

**Not:** ↵ simgesi tek bir boş karakteri temsil eder.

Çizelge 127. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQCTLO_VERSION_1	1	X'00000001'
MQCTLO_CURRENT_VERSION	1	X'00000001'

### MQCTL Seçenekleri Tüketici Denetimi Seçenekleri

Çizelge 128. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQCTLO_NONE	0	X'00000000'
MQCTLO_THREAD_AFFINITY	1	X'00000001'
MQCTLO_FAIL_IF_QUIESCING	8192	X'00002000'

## MQCUOWC\_\* (CICS bilgi üstbilgisi İş Denetimleri-İş Birimi Denetim Birimi)

Çizelge 129. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
YALNIZCIK	273	X'00000111'
MQCUOWC_CONTINY	65536	X'00010000'
İLK MQCUOWC_FIRST	17	X'00000011'
MQCUOWC_ORTA	16	X'00000010'

Çizelge 129. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQCUOWC_SON	272	X'00000110'
MQCUOWC_COMMIT	256	X'00000100'
MQCUOWC_BACKOUT	4352	X'00001100'

## MQDXP\_\* (Kanal çıkış değiştirgesi yapısı)

Çizelge 130. Değişmezlerin yapıları

Ad	Yapı
MQDXP_STRUC_ID	"CXP~"
MQDXP_STRUC_ID_ARRAY	'C', 'X', 'P', '~'

**Not:** ~ simgesi tek bir boş karakteri temsil eder.

Çizelge 131. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQDXP_VERSION_1	1	X'00000001'
MQDXP_VERSION_2	2	X'00000002'
MQDXP_VERSION_3	3	X'00000003'
MQDXP_VERSION_4	4	X'00000004'
MQDXP_VERSION_5	5	X'00000005'
MQDXP_VERSION_6	6	X'00000006'
MQDXP_VERSION_7	7	X'00000007'
MQDXP_VERSION_8	8	X'00000008'
MQDXP_VERSION_9	9	X'00000009'
MQDXP_XX_ENCODE_CASE_ONE current_version	9	X'00000009'

## MQDC\_\* (Hedef Sınıf)

Çizelge 132. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQDC_MANAGED	1	X'00000001'
MQDC_PROVIDER	2	X'00000002'

## MQDCC\_\* (Dönüştürme Seçenekleri, Maskeler ve Üreticiler)

### Dönüştürme Seçenekleri

Çizelge 133. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQDCC_DEFAULT_CONVERSION	1	X'00000001'
MQDCC_FILL_TARGET_BUFFER	2	X'00000002'
MQDCC_INT_DEFAULT_CONVERSION	4	X'00000004'
MQDCC_SOURCE_ENC_NATIVE	(value differs by platform or version)	(value differs by platform or version)

Çizelge 133. Değişmezlerin değerleri (devamı var)		
Ad	Ondalık değer	Onaltılı değer
MQDCC_SOURCE_ENC_NORMAL	16	X'00000010'
MQDCC_SOURCE_ENC_RECYD	32	X'00000020'
MQDCC_SOURCE_ENC_UNDEFINED	0	X'00000000'
MQDCC_TARGET_ENC_NATIVE	(value differs by platform or version)	(value differs by platform or version)
MQDC_TARGET_ENC_NORMAL	256	X'00000100'
MQDCC_TARGET_ENC_REVERSED	512	X'00000200'
MQDCC_TARGET_ENC_UNDEFED	0	X'00000000'
MQDCC_NONE	0	X'00000000'

### Dönüştürme Seçenekleri Maskeleri ve Üreticiler

Çizelge 134. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQDCC_SOURCE_ENC_MASK	240	X'000000F0'
MQDCC_TARGET_MASK	3840	X'00000F00'
MQDCC_SOURCE_ENC_FACTOR	16	X'00000010'
MQDCC_TARGET_ENC_FACTOR	256	X'00000100'

### MQDELO\_\* (Yayınlama/Abone Olma Silme Seçenekleri)

Çizelge 135. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQDELO_NONE	0	X'00000000'
MQDELO_LOCAL	4	X'00000004'

### MQDH\_\* (Dağıtım üstbilgisi yapısı)

Çizelge 136. Değişmezlerin yapıları	
Ad	Yapı
MQDH_STRUC_ID	"DH↯"
MQDH_STRUC_ID_ARRAY	'D', 'H', '↯', '↯'

**Not:** ↯ simgesi tek bir boş karakteri temsil eder.

Çizelge 137. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQDH_VERSION_1	1	X'00000001'
MQDH_XX_ENCODE_CASE_ONE current_version	1	X'00000001'

### MQDHF\_\* (Dağıtım üstbilgisi İşaretleri)

Çizelge 138. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQDHF_NEW_MSG_IDS	1	X'00000001'



Çizelge 138. Değişmezlerin değerleri (devamı var)		
Ad	Ondalık değer	Onaltılı değer
MQDHF_NONE	0	X'00000000'

### **MQDISCONNECT\_ \* (Komut biçimi Bağlantı Kesme Tipleri)**

Çizelge 139. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQDISCONNECT_NORMAL	0	X'00000000'
MQDISCONNECT_IMPLICIT	1	X'00000001'
MQDISCONNECT_Q_MGR	2	X'00000002'

### **MQDL\_ \* (Dağıtım Listeleri)**

Çizelge 140. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQDL_DESTEKLENEN	1	X'00000001'
MQDL_NOT_SUPPORTANT	0	X'00000000'

### **MQDLH\_ \* (Ölü harf üstbilgi yapısı)**

Çizelge 141. Değişmezlerin yapıları	
Ad	Yapı
MQDLH_STRUC_ID	"DLH↵"
MQDLH_STRUC_ID_ARRAY	'D', 'L', 'H', '↵'

**Not:** ↵ simgesi tek bir boş karakteri temsil eder.

Çizelge 142. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQDLH_VERSION_1	1	X'00000001'
MQDLH_CURRENT_VERSION	1	X'00000001'

### **MQDLV\_ \* (Persistent/kalıcı Olmayan İleti Teslimi)**

Çizelge 143. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQDLV_AS_PARENT	0	X'00000000'
MQDLV_ALL	1	X'00000001'
MQDLV_ALL_DUR	2	X'00000002'
MQDLV_ALL_AVAIL	3	X'00000003'

## MQDMHO\_\* (İleti tanıtıcısı seçeneklerini ve yapısını sil)

### İleti tanıtıcısı seçenekleri yapısını sil

Çizelge 144. Değişmezlerin yapıları	
Ad	Yapı
MQDMHO_STRUC_ID	"DMHO"
MQDMHO_STRUC_ID_ARRAY	'D', 'M', 'H', 'O'

**Not:** – simgesi tek bir boş karakteri temsil eder.

Çizelge 145. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQDMHO_VERSION_1	1	X'00000001'
MQDMHO_CURRENT_VERSION	1	X'00000001'

### İleti Tanıtıcısı Seçeneklerini Sil

Çizelge 146. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQDMHO_NONE	0	X'00000000'

## MQDMPO\_\* (İleti özelliği seçeneklerini ve yapısını sil)

### İleti özelliği seçenekleri yapısını sil

Çizelge 147. Değişmezlerin yapıları	
Ad	Yapı
MQDMPO_STRUCT_ID	"DMPO"
MQDMPO_STRUC_ID_ARRAY	'D', 'M', 'P', 'O'

**Not:** – simgesi tek bir boş karakteri temsil eder.

Çizelge 148. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQDMPO_VERSION_1	1	X'00000001'
MQDMPO_CURRENT_VERSION	1	X'00000001'

### İleti Özelliği Seçeneklerini Sil

Çizelge 149. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQDMPO_DEL_FIRST	0	X'00000000'
MQDMPO_DEL_PROP_UNDER_CURSOR	1	X'00000001'
MQDMPO_NONE	0	X'00000000'

## MQDNSWLM\_\* (DNS WLM)

Çizelge 150. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQDNSWLM_NO	0	X'00000000'
MQDNSWLM_YES	1	X'00000001'

## MQDT\_\* (Hedef Tipleri)

Çizelge 151. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQDT_APPL	1	X'00000001'
MQDT_BROKER	2	X'00000002'

## MQDXP\_\* (Dönüştürme çıkışı parametre yapısı)

Çizelge 152. Değişmezlerin yapıları	
Ad	Yapı
MQDXP_STRUC_ID	"DXP↵"
MQDXP_STRUC_ID_ARRAY	'D','X','P','↵'

**Not:** ↵ simgesi tek bir boş karakteri temsil eder.

Çizelge 153. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQDXP_VERSION_1	1	X'00000001'
MQDXP_VERSION_2	2	X'00000002'
MQDXP_CURRENT_VERSION	2	X'00000002'

## MQEC\_\* (İşaret Değerleri)

Çizelge 154. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQEC_MSG_GELDI	2	X'00000002'
MQEC_WAIT_INTERVAL_EXERD	3	X'00000003'
MQEC_WAIT_IPTAL edildi	4	X'00000004'
MQEC_Q_MGR QUIESCING	5	X'00000005'
MQEC_CONNECTION QUIESCING	6	X'00000006'

## MQE\_\* (Süre bitimi)

Çizelge 155. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQE_UNSENSIZ	-1	X'FFFFFFFF'

## MQENC\_\* (Kodlama)

## MQENC\_\* (Kodlama)

Çizelge 156. Değişmezlerin altyapıya göre değerleri

Ad	Altyapı	Ondalık değer	Onaltılı değer
MQENC_NATIVE	IBM i	273	X'00000111 '
	Linux	546	X'00000222 '
	SPARC üzerindeLinux	273	X'00000111 '
	x86üzerindeLinux	546	X'00000222 '
	AIX and Linux	273	X'00000111 '
	Windows	546	X'00000222 '
	Mikro Focus COBOL on Windows	17	X'00000011 '
	z/OS	785	X'00000311 '

## MQENC\_\* (Kodlama Mister)

Çizelge 157. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQENC_INTEGER_MASK	15	X'0000000F '
MQENC_DECIMAL_MASK	240	X'000000F0 '
MQENC_FLOAT_MASK	3840	X'00000F00 '
MQENC_RESERVE_MASK	-4096	X'FFFFFF000 '

## MQENC\_\* (İkili Integer 'lar için Encodings)

Çizelge 158. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQENC_INTEGER_UNDEFIN	0	X'00000000 '
MQENC_INTEGER_NORMAL	1	X'00000001 '
MQENC_INTEGER_TERS	2	X'00000002 '

## MQENC\_\* (Paketlenmiş Ondalık Tamsayıcılara ilişkin Encodings)

Çizelge 159. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQENC_DECIMAL_UNDEFINED	0	X'00000000 '
MQENC_DECIMAL_NORMAL	16	X'00000010 '
MQENC_DECIMAL_TERDI	32	X'00000020 '

## MQENC\_\* (Değişken Nokta Numaraları için Encodings)

Çizelge 160. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQENC_FLOAT_UNDEFIN	0	X'00000000 '
MQENC_FLOAT_IEEE_NORMAL	256	X'00000100 '

Çizelge 160. Değişmezlerin değerleri (devamı var)		
Ad	Ondalık değer	Onaltılı değer
MQENC_FLOAT_IEEEEE_TERS	512	X'00000200'
MQENC_FLOAT_S390	768	X'00000300'
MQENC_FLOAT_TNS	1024	X'00000400'

## MQEPH\_\* (Embedded komut biçimi üstbilgi yapısı ve işaretleri)

### Gömülü komut biçimi üstbilgi yapısı

Çizelge 161. Değişmezlerin yapıları	
Ad	Yapı
MQEPH_STRUCT_ID	"EPH↵"
MQEF_STRUC_ID_ARRAY	'E', 'P', 'H', '↵'

**Not:** ↵ simgesi tek bir boş karakteri temsil eder.

Çizelge 162. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQEF_STRUCT_LENGTH_FIXED	68	X'00000044'
MQEPH_VERSION_1	1	X'00000001'
MQEPH_CURRENT_VERSION	1	X'00000001'

### Gömülü komut biçimi üstbilgi işaretleri

Çizelge 163. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQEPH_NONE	0	X'00000000'
MQEF_CCSDID_EMBEDDED	1	X'00000001'

## MQET\_\* (Komut biçimi Çıkış Tipleri)

Çizelge 164. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQET_MQSC	1	X'00000001'

## MQEVO\_\* (Komut biçimi Olay Originleri)

Çizelge 165. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQEVO_DİĞER	0	X'00000000'
MQEVO_CONSOLE	1	X'00000001'
MQEVO_INIT	2	X'00000002'
MQEVO_MSG	3	X'00000003'
MQEVO_MQSET	4	X'00000004'
MQEVO_DAHILI	5	X'00000005'
MQEVO_MQSUB	6	X'00000006'

Çizelge 165. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQEVO_CTLMSG	7	X'00000007'
MQEVO_REST	8	X'00000008'

### MQEVN\_\* (Komut biçimi Olay Kaydı)

Çizelge 166. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQEVN_DISABLE	0	X'00000000'
MQEVN_ENABLED	1	X'00000001'
MQEVN_EXCEPTION	2	X'00000002'
MQEVN_NO_DISPLAY	3	X'00000003'

### MQEXPI\_\* (Süre Bitimi Tarama Aralığı)

Çizelge 167. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQEXPI_OFF	0	X'00000000'

### MQFB\_\* (Geribildirim Değerleri)

Çizelge 168. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQFB_YOK	0	X'00000000'
MQFB_SYSTEM_FIRST	1	X'00000001'
MQFB_QUIT	256	X'00000100'
MQFB_BITIM	258	X'00000102'
MQFB_COA	259	X'00000103'
MQFB_COD	260	X'00000104'
MQFB_CHANNEL_COMPLETED	262	X'00000106'
MQFB_CHANNEL_FAIL_RETRY	263	X'00000107'
MQFB_CHANNEL_FAIL	264	X'00000108'
MQFB_APPL_CANNOT_BI_STARTED	265	X'00000109'
MQFB_TM_HATASI	266	X'0000010A'
MQFB_APPL_TYPE_ERROR	267	X'0000010B'
MQFB_STOPPED_BY_MSG_EXIT	268	X'0000010C'
MQFB_ACTIVITY	269	X'0000010D'
MQFB_XMIT_Q_MSG_ERROR	271	X'0000010F'
MQFB_PAN	275	X'00000113'
MQFB_NAN	276	X'00000114'
MQFB_STOPPED_BY_CHAD_EXIT	277	X'00000115'
MQFB_STOPPED_BY_PUBSUB_EXIT	279	X'00000117'
MQFB_NOT_A_REPOSITORY_MSG	280	X'00000118'
MQFB_BIND_OPEN_CLUSRCVR_DEL	281	X'00000119'

Çizelge 168. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQFB_MAX_ACTIVITIES	282	X'0000011A'
MQFB_NOT_ILETILDI	283	X'0000011B'
MQFB_NOT_TESLİM	284	X'0000011C'
MQFB_UNSUPPORTED_FORWARDING	285	X'0000011D'
MQFB_UNSUPPORTED_DELIVERY	286	X'0000011E'
MQFB_DATA_LENGTH_ZERO	291	X'00000123'
MQFB_DATA_LENGTH_NEGATIF	292	X'00000124'
MQFB_DATA_LENGTH_TOO_BIG	293	X'00000125'
MQFB_BUFFER_OVERFLOW	294	X'00000126'
MQFB_LENGTH_OFF_BY_ONE	295	X'00000127'
MQFB_IH_ERROR	296	X'00000128'
MQFB_NOT_AUTHORIZED_FOR_IMS	298	X'0000012A'
MQFB_IMS_ERROR	300	X'0000012C'
MQFB_IMS_FIRST	301	X'0000012D'
MQFB_IMS_SON	399	X'0000018F'
MQFB_CICS_INTERNAL_ERROR	401	X'00000191'
MQFB_CICS_NOT_YETKİLİ	402	X'00000192'
MQFB_CICS_BRIDGE_FAILURE	403	X'00000193'
MQFB_CICS_CORREL_ID_ERROR	404	X'00000194'
MQFB_CICS_CCSD_ERROR	405	X'00000195'
MQFB_CICS_ENCODING_ERROR	406	X'00000196'
MQFB_CICS_CIH_ERROR	407	X'00000197'
MQFB_CICS_UOW_ERROR	408	X'00000198'
MQFB_CICS_COMMAREA_ERROR	409	X'00000199'
MQFB_CICS_APPL_NOT_STARTED	410	X'0000019A'
MQFB_CICS_APPL_ABENLENDİ	411	X'0000019B'
MQFB_CICS_DLQ_ERROR	412	X'0000019C'
MQFB_CICS_UOW_BACKED_OUT	413	X'0000019D'
MQFB_PUBLICATIONS_ON_REQUEST	501	X'000001F5'
MQFB_SUBSCRIBER_IS_PUBLISHER	502	X'000001F6'
MQFB_MSG_SCOPE_UYUŞMAZLIĞI	503	X'000001F7'
MQFB_SELECTOR_MISMATCH	504	X'000001F8'
MQFB_IMS_NACK_1A_REASON_FIRST	600	X'00000258'
MQFB_IMS_NACK_1A_REASON_LAST	855	X'00000357'
MQFB_SYSTEM_SON	65535	X'0000FFFF'
MQFB_APPL_FIRST	65536	X'00010000'
MQFB_APPL_SON	999999999	X'3B9AC9FF'

## MQFC\_\* (Komut biçimi Güç Seçenekleri)

Çizelge 169. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQFC_YES	1	X'00000001'
MQFC_NO	0	X'00000000'

## MQFMT\_\* (Biçimler)

Çizelge 170. Değişmez adlar ve değerler	
Ad	Değer
MQFMT_NONE	"-----"
MQFMT_ADMIN	"MQADMIN"
MQFMT_CHANNEL_COMPLETED	"MQCHCOM"
MQFMT_CICS	"MQCICS"
MQFMT_COMMAND_1	"MQCMD1"
MQFMT_COMMAND_2	"MQCMD2"
MQFMT_DEAD_LETTER_HEADER	"MQDEAD"
MQFMT_DIST_HEADER	"MQHDIST"
MQFMT_EMBEDDED_PCF	"MQHEPCF"
MQFMT_OLAY	"MQEVENT"
MQFMT_IMS	"MQIMS"
MQFMT_IMS_VAR_STRING	"MQIMSVS"
MQFMT_MD_EXTENSION	"MQHMDE"
MQFMT_PCF	"MQPCF"
MQFMT_REF_MSG_HEADER	"MQHREF"
MQFMT_RF_HEADER	"MQHRF"
MQFMT_RF_HEADER_1	"MQHRF1"
MQFMT_RF_HEADER_2	"MQHRF2"
MQFMT_STRING	"MQSTR"
MQFMT_TETIKLEYICISI	"MQTRIG"
MQFMT_WORK_INFO_HEADER	"MQHWIH"
MQFMT_XMIT_Q_HEADER	"MQXMIT"
MQFMT_NONE_DIZISI	' ',' ',' ',' ',' ',' ',' ',' '
MQFMT_ADMIN_ARRAY	'M','Q','A','D','M','I','N',' '
MQFMT_CHANNEL_COMPLETED_ARRAY	'M','Q','C','H','C','O','M',' '
MQFMT_CICS_ARRAY	'M','Q','C','I','C','S',' ',' '
MQFMT_COMMAND_1_ARRAY	'M','Q','C','M','D','1',' ',' '
MQFMT_COMMAND_2_ARRAY	'M','Q','C','M','D','2',' ',' '
MQFMT_DEAD_LETTER_HEADER_ARRAY	'M','Q','D','E','A','D',' ',' '
MQFMT_DIST_HEADER_ARRAY	'M','Q','H','D','I','S','T',' '
MQFMT_EMBEDDED_PCF_ARRAY	'M','Q','H','E','P','C','F',' '
MQFMT_EVENT_ARRAY	'M','Q','E','V','E','N','T',' '



Çizelge 170. Değişmez adlar ve değerler (devamı var)

Ad	Değer
MQFMT_IMS_ARRAY	'M','Q','I','M','S','-', '-', '-'
MQFMT_IMS_VAR_STRING_ARRAY	'M','Q','I','M','S','V','S','-', '-'
MQFMT_MD_EXTENSION_ARRAY	'M','Q','H','M','D','E','-', '-', '-'
MQFMT_PCF_ARRAY	'M','Q','P','C','F','-', '-', '-'
MQFMT_REF_MSG_HEADER_ARRAY	'M','Q','H','R','E','F','-', '-', '-'
MQFMT_RF_HEADER_ARRAY	'M','Q','H','R','F','-', '-', '-'
MQFMT_RF_HEADER_1_ARRAY	'M','Q','H','R','F','-', '-', '-'
MQFMT_RF_HEADER_2_ARRAY	'M','Q','H','R','F','2','-', '-', '-'
MQFMT_STRING_ARRAY	'M','Q','S','T','R','-', '-', '-'
MQFMT_TRIGGER_ARRAY	'M','Q','T','R','I','G','-', '-', '-'
MQFMT_WORK_INFO_HEADER_ARRAY	'M','Q','H','W','I','H','-', '-', '-'
MQFMT_XMIT_Q_HEADER_ARRAY	'M','Q','X','M','I','T','-', '-', '-'

**Not:** - simgesi tek bir boş karakteri temsil eder.

### **MQFUN\_ \* (Uygulama İşlevi Tipleri)**

Çizelge 171. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQFUN_TYPE_UNKNOWN	0	X'00000000'
MQFUN_TYPE_JVM	1	X'00000001'
MQFUN_TYPE_PROGRAM	2	X'00000002'
MQFUN_TYPE_PROCEDURE	3	X'00000003'
MQFUN_TYPE_USERDEF	4	X'00000004'
MQFUN_TYPE_COMMAND	5	X'00000005'

### **MQGA\_ \* (Grup Özniteliği Seçicileri)**

Çizelge 172. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQGA_FIRST	8001	X'00001F41'
MQGA_SON	9000	X'00002328'

### **MQGACF\_ \* (Komut biçimi Grup Parametresi Tipleri)**

Çizelge 173. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQGACF_FIRST	8001	X'00001F41'
MQGACF_COMMAND_CONTEXT	8001	X'00001F41'
MQGACF_COMMAND_DATA	8002	X'00001F42'
MQGACF_TRACE_ROUTE	8003	X'00001F43'
MQGACF_OPERATION	8004	X'00001F44'
MQGACF_ETKİNLİK	8005	X'00001F45'

Çizelge 173. Değişmezlerin değerleri (devamı var)		
Ad	Ondalık değer	Onaltılı değer
MQGACF_EMBEDDED_MQMD	8006	X'00001F46'
MQGACF_MESSAGE	8007	X'00001F47'
MQGACF_MQMD	8008	X'00001F48'
MQGACF_VALUE_NAMING	8009	X'00001F49'
MQGACF_Q_ACCOUNTING_DATA	8010	X'00001F4A'
MQGACF_Q_STATISTICS_DATA	8011	X'00001F4B'
MQGACF_CHL_STATISTICS_DATA	8012	X'00001F4C'
MQGACF_LAST_KULLANILDI	8012	X'00001F4C'

### MQGI\_\* (Grup Tanıtıcısı)

Çizelge 174. Değişmez adlar ve değerler	
Ad	Değer
MQGI_NONE	X'00...00' (24 boş değer)
MQGI_NONE_DIZISI	'\0', '\0', ... (24 boş değer)

### MQGMO\_\* (İleti seçeneklerini ve yapısını al)

#### İleti seçenekleri yapısını al

Çizelge 175. Değişmezlerin yapıları	
Ad	Yapı
MQGMO_STRUC_ID	"GMO~"
MQGMO_STRUC_ID_ARRAY	'G', 'M', 'O', '~'

**Not:** ~ simgesi tek bir boş karakteri temsil eder.

Çizelge 176. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQGMO_VERSION_1	1	X'00000001'
MQGMO_VERSION_2	2	X'00000002'
MQGMO_VERSION_3	3	X'00000003'
MQGMO_VERSION_4	4	X'00000004'
MQGMO_CURRENT_VERSION	4	X'00000004'

#### İleti Seçeneklerini Al

Çizelge 177. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQGMO_BEKLE	1	X'00000001'
MQGMO_NO_BEKLEME	0	X'00000000'
MQGMO_SET_SIGNAL	8	X'00000008'
MQGMO_FAIL_IF QUIESCING	8192	X'00002000'
MQGMO_SYNCPOINT	2	X'00000002'

Çizelge 177. Değişmezlerin değerleri (devamı var)		
Ad	Ondalık değer	Onaltılı değer
MQGMO_SYNCPOINT_IF_PERSISTENT	4096	X'00001000'
MQGMO_NO_SYNCPOINT	4	X'00000004'
MQGMO_MARK_SKIP_BACKUT	128	X'00000080'
MQGMO_BROWSE_FIRST	16	X'00000010'
MQGMO_BROWSE_NEXT	32	X'00000020'
MQGMO_BROWSE_MSG_UNDER_CURSOR	2048	X'00000800'
MQGMO_BROWSE_HANDLE	17825808	X'01100010'
MQGMO_BROWSE_CO_OP	18874384	X'01200010'
MQGMO_MSG_UNDER_CURSOR	256	X'00000100'
MQGMO_LOCK	512	X'00000200'
MQGMO_UNLOCK	1024	X'00000400'
MQGMO_ACCEPT_TRUNCATED_MSG	64	X'00000040'
MQGMO_CONVERT	16384	X'00004000'
MQGMO_LOGICAL_ORDER	32768	X'00008000'
MQGMO_COMPLE_MSG	65536	X'00010000'
MQGMO_ALL_MSGS_AVALABILIR	131072	X'00020000'
MQGMO_ALL_SEGMENTS_AVALABILIR	262144	X'00040000'
MQGMO_MARK_BROWSE_HANDLE	1048576	X'00100000'
MQGMO_MARK_BROWSE_CO_OP	2097152	X'00200000'
MQGMO_UNMARK_BROWSE_CO_OP	4194304	X'00400000'
MQGMO_UNMARK_BROWSE_HANDLE	8388608	X'00800000'
MQGMO_UNMARKET_BROWSE_MSG	16777216	X'01000000'
MQGMO_PROPERTIES_FORCE_MQRFH2	33554432	X'02000000'
MQGMO_NO_ÖZELLİKLERİ	67108864	X'04000000'
MQGMO_PROPERTIES_IN_HANDLE	134217728	X'08000000'
MQGMO_PROPERTIES_COMPATIBILITY	268435456	X'10000000'
MQGMO_PROPERTIES_AS_Q_DEF	0	X'00000000'
MQGMO_NONE	0	X'00000000'

### MQGS\_\* (Grup Durumu)

Çizelge 178. Değişmez adlar ve değerler	
Ad	Değer
MQGS_NOT_IN_GROUP	'-'
MQGS_MSG_IN_GROUP	'G'
MQGS_LAST_MSG_IN_GROUP	'L'

**Not:** - simgesi tek bir boş karakteri temsil eder.

## MQHA\_\* (Tanıtıcı Seçiciler)

Çizelge 179. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQHA_FIRST	4001	X'00000FA1'
MQHA_SIM_HANDLE	4001	X'00000FA1'
KULLANILAN MQHA_LAST_	4001	X'00000FA1'
MQHA_SON	6000	X'00001770'

## MQHB\_\* (çanta çekme noktaları)

Çizelge 180. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQHB_UNUSABLE_HBAG	-1	X'FFFFFFFF'
MQHB_NONE	-2	X'FFFFFFFE'

## MQHC\_\* (Bağlantı Elleri)

Çizelge 181. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQHC_DEF_HCONN	0	X'00000000'
MQHC_UNUSABLE_HCONN	-1	X'FFFFFFFF'
MQHC_UNASSOCIATED_HCONN	-3	X'FFFFFFFD'

## MQHM\_\* (İleti tanıtıcısı)

Çizelge 182. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQHM_UNUSABLE_HMSG	-1	X'FFFFFFFF'
MQHM_NONE	0	X'00000000'

## MQHO\_\* (Nesne Tanıtıcısı)

Çizelge 183. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQHO_UNUSABLE_HOBJ	-1	X'FFFFFFFF'
MQHO_NONE	0	X'00000000'

## MQHSTATE\_\* (Komut biçimi tanıtıcı durumları)

Çizelge 184. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQHSTATE_DEVREDİŞİ	0	X'00000000'
MQHSTATE_ETKİN	1	X'00000001'

## MQIA\_\* (Tamsayı Özniteliği Seçicileri)

Çizelge 185. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQIA_ACCOUNTING_CONN_OVERRIDE	136	X'00000088'
MQIA_ACCOUNTING_INTERVAL	135	X'00000087'
MQIA_ACCOUNTING_MQI	133	X'00000085'
MQIA_ACCOUNTING_Q	134	X'00000086'
MQIA_ACTIVE_CHANNELS	100	X'00000064'
MQIA_ACTIVITY_CONN_OVERRIDE	239	X'000000EF'
MQIA_ACTIVITY_RECORDING	138	X'0000008A'
MQIA_ACTIVITY_TRACE	240	X'000000F0'
MQIA_ADOPTNEWMCA_CHECK	102	X'00000066'
MQIA_ADOPTNEWMCA_INTERVAL	104	X'00000068'
MQIA_ADOPTNEWMCA_TYPE	103	X'00000067'
MQIA_ADOPT_CONTEXT	260	X'00000104'
<span style="background-color: #e0e0e0;">MQ Adv. VUE</span> MQIA_ADVANCED_CAPABILITY	273	X'00000111'
MQIA_AMQP_CAPABILITY	265	X'00000109'
MQIA_APPL_TYPE	1	X'00000001'
MQIA_ARCHIVE	60	X'0000003C'
MQIA_AUTHENTICATION_FAIL_DELAY	259	X'00000103'
MQIA_AUTHENTICATION_METHOD	266	X'0000010A'
MQIA_AUTH_INFO_TYPE	66	X'00000042'
MQIA_AUTHORITY_EVENT	47	X'0000002F'
MQIA_AUTO_REORG_INTERVAL	174	X'000000AE'
MQIA_AUTO_REORGANIZATION	173	X'000000AD'
MQIA_BACKOUT_THRESHOLD	22	X'00000016'
MQIA_BASE_TYPE	193	X'000000C1'
MQIA_BATCH_INTERFACE_AUTO	86	X'00000056'
MQIA_BRIDGE_EVENT	74	X'0000004A'
MQIA_CF_LEVEL	70	X'00000046'
MQIA_CF_RECOVER	71	X'00000047'
MQIA_CHANNEL_AUTO_DEF	55	X'00000037'
MQIA_CHANNEL_AUTO_DEF_EVENT	56	X'00000038'
MQIA_CHANNEL_EVENT	73	X'00000049'
MQIA_CHECK_CLIENT_BINDING	258	X'00000102'
MQIA_CHECK_LOCAL_BINDING	257	X'00000101'
MQIA_CHINIT_ADAPTERS	101	X'00000065'
MQIA_CHINIT_CONTROL	119	X'00000077'
MQIA_CHINIT_DISPATCHERS	105	X'00000069'
MQIA_CHINIT_TRACE_AUTO_START	117	X'00000075'
MQIA_CHINIT_TRACE_TABLE_SIZE	118	X'00000076'

Çizelge 185. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQIA_CLUSTER_OBJECT_STATE	256	X'00000100'
MQIA_CLUSTER_PUB_ROUTE	255	X'000000FF'
MQIA_CLUSTER_Q_TYPE	59	X'0000003B'
MQIA_CLUSTER_WORKLOAD_LENGTH	58	X'0000003A'
MQIA_CLWL_MRU_CHANNELS	97	X'00000061'
MQIA_CLWL_Q_RANK	95	X'0000005F'
MQIA_CLWL_Q_PRIORITY	96	X'00000060'
MQIA_CLWL_USEQ	98	X'00000062'
MQIA_CMD_SERVER_AUTO	87	X'00000057'
MQIA_CMD_SERVER_CONTROL	120	X'00000078'
MQIA_CMD_SERVER_CONVERT_MSG	88	X'00000058'
MQIA_CMD_SERVER_DLQ_MSG	89	X'00000059'
MQIA_CODED_CHAR_SET_ID	2	X'00000002'
MQIA_COMM_EVENT	232	X'000000E8'
MQIA_COMMAND_EVENT	99	X'00000063'
MQIA_COMMAND_LEVEL	31	X'0000001F'
MQIA_CONFIGURATION_EVENT	51	X'00000033'
MQIA_CPI_LEVEL	27	X'0000001B'
MQIA_CURRENT_Q_DEPTH	3	X'00000003'
MQIA_DEF_BIND	61	X'0000003D'
MQIA_DEF_CLUSTER_XMIT_Q_TYPE	250	X'000000FA'
MQIA_DEF_INPUT_OPEN_OPTION	4	X'00000004'
MQIA_DEF_PERSISTENCE	5	X'00000005'
MQIA_DEF_PRIORITY	6	X'00000006'
MQIA_DEF_PUT_RESPONSE_TYPE	184	X'000000B8'
MQIA_DEF_READ_AHEAD	188	X'000000BC'
MQIA_DEFINITION_TYPE	7	X'00000007'
MQIA_DISPLAY_TYPE	262	X'00000106'
MQIA_DIST_LISTS	34	X'00000022'
MQIA_DNS_WLM	106	X'0000006A'
MQIA_DURABLE_SUB	175	X'000000AF'
MQIA_EXPIRY_INTERVAL	39	X'00000027'
MQIA_FIRST	1	X'00000001'
MQIA_GROUP_UR	221	X'000000DD'
MQIA_HARDEN_GET_BACKOUT	8	X'00000008'
MQIA_HIGH_Q_DEPTH	36	X'00000024'
MQIA_IGQ_PUT_AUTHORITY	65	X'00000041'
MQIA_INDEX_TYPE	57	X'00000039'
MQIA_INHIBIT_EVENT	48	X'00000030'

Çizelge 185. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQIA_INHIBIT_GET	9	X'00000009'
MQIA_INHIBIT_PUB	181	X'000000B5'
MQIA_INHIBIT_PUT	10	X'0000000A'
MQIA_INHIBIT_SUB	182	X'000000B6'
MQIA_INTRA_GROUP_queuing	64	X'00000040'
MQIA_IP_ADDRESS_VERSION	93	X'0000005D'
MQIA_KEY_REUSE_COUNT	267	X'0000010B'
MQIA_LAST	2000	X'000007D0'
MQIA_LAST_USED	267	X'0000010B'
MQIA_LDAP_AUTHORMD	263	X'00000107'
MQIA_LDAP_NESTGRP	264	X'00000108'
MQIA_LDAP_SECURE_COMM	261	X'00000105'
MQIA_LISTENER_PORT_NUMBER	85	X'00000055'
MQIA_LISTENER_TIMER	107	X'0000006B'
MQIA_LOGGER_EVENT	94	X'0000005E'
MQIA_LU62_CHANNELS	108	X'0000006C'
MQIA_LOCAL_EVENT	49	X'00000031'
MQIA_MSG_MARK_BROWSE_INTERVAL	68	X'00000044'
MQIA_MAX_CHANNELS	109	X'0000006D'
MQIA_MAX_CLIENTS	172	X'000000AC'
MQIA_MAX_GLOBAL_LOCKS	83	X'00000053'
MQIA_MAX_HANDLES	11	X'0000000B'
MQIA_MAX_LOCAL_LOCKS	84	X'00000054'
MQIA_MAX_MSG_LENGTH	13	X'0000000D'
MQIA_MAX_OPEN_Q	80	X'00000050'
MQIA_MAX_PRIORITY	14	X'0000000E'
MQIA_MAX_PROPERTIES_LENGTH	192	X'000000C0'
MQIA_MAX_Q_DEPTH	15	X'0000000F'
MQIA_MAX_Q_TRIGGERS	90	X'0000005A'
MQIA_MAX_RECOVERY_TASKS	171	X'000000AB'
MQIA_MAX_UNCOMMITTED_MSGS	33	X'00000021'
MQIA_MCAST_BRIDGE	233	X'000000E9'
MQIA_MONITOR_INTERVAL	81	X'00000051'
MQIA_MONITORING_AUTO_CLUSSDR	124	X'0000007C'
MQIA_MONITORING_CHANNEL	122	X'0000007A'
MQIA_MONITORING_Q	123	X'0000007B'
MQIA_MSG_DELIVERY_SEQUENCE	16	X'00000010'
MQIA_MSG_DEQ_COUNT	38	X'00000026'
MQIA_MSG_ENQ_COUNT	37	X'00000025'

Çizelge 185. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQIA_NAME_COUNT	19	X'00000013'
MQIA_NAMELIST_TYPE	72	X'00000048'
MQIA_NPM_CLASS	78	X'0000004E'
MQIA_NPM_DELIVERY	196	X'000000C4'
MQIA_OPEN_INPUT_COUNT	17	X'00000011'
MQIA_OPEN_OUTPUT_COUNT	18	X'00000012'
MQIA_OUTBOUND_PORT_MAX	140	X'0000008C'
MQIA_OUTBOUND_PORT_MIN	110	X'0000006E'
MQIA_PAGESET_ID	62	X'0000003E'
MQIA_PERFORMANCE_EVENT	53	X'00000035'
MQIA_PLATFORM	32	X'00000020'
MQIA_PM_DELIVERY	195	X'000000C3'
MQIA_PROPERTY_CONTROL	190	X'000000BE'
MQIA_PROT_POLICY_CAPABILITY	251	X'000000FB'
MQIA_PROXY_SUB	199	X'000000C7'
MQIA_PUB_COUNT	215	X'000000D7'
MQIA_PUB_SCOPE	219	X'000000DB'
MQIA_PUBSUB_CLUSTER	249	X'000000F9'
MQIA_PUBSUB_MAXMSG_RETRY_COUNT	206	X'000000CE'
MQIA_PUBSUB_MODE	187	X'000000BB'
MQIA_PUBSUB_NP_MSG	203	X'000000CB'
MQIA_PUBSUB_NP_RESP	205	X'000000CD'
MQIA_PUBSUB_SYNC_PT	207	X'000000CF'
MQIA_Q_DEPTH_HIGH_EVENT	43	X'0000002B'
MQIA_Q_DEPTH_HIGH_LIMIT	40	X'00000028'
MQIA_Q_DEPTH_LOW_EVENT	44	X'0000002C'
MQIA_Q_DEPTH_LOW_LIMIT	41	X'00000029'
MQIA_Q_DEPTH_MAX_EVENT	42	X'0000002A'
MQIA_Q_SERVICE_INTERVAL	54	X'00000036'
MQIA_Q_SERVICE_INTERVAL_EVENT	46	X'0000002E'
MQIA_Q_TYPE	20	X'00000014'
MQIA_Q_USERS	82	X'00000052'
MQIA_QMGR_CFCONLOS	245	X'000000F5'
MQIA_QMOPT_CONS_COMMS_MSGS	155	X'0000009B'
MQIA_QMOPT_CONS_CRITICAL_MSGS	154	X'0000009A'
MQIA_QMOPT_CONS_ERROR_MSGS	153	X'00000099'
MQIA_QMOPT_CONS_INFO_MSGS	151	X'00000097'
MQIA_QMOPT_CONS_REORG_MSGS	156	X'0000009C'
MQIA_QMOPT_CONS_SYSTEM_MSGS	157	X'0000009D'



Çizelge 185. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQIA_QMOPT_CONS_WARNING_MSGS	152	X'00000098'
MQIA_QMOPT_CSMT_ON_ERROR	150	X'00000096'
MQIA_QMOPT_INTERNAL_DUMP	170	X'000000AA'
MQIA_QMOPT_LOG_COMMS_MSGS	162	X'000000A2'
MQIA_QMOPT_LOG_CRITICAL_MSGS	161	X'000000A1'
MQIA_QMOPT_LOG_ERROR_MSGS	160	X'000000A0'
MQIA_QMOPT_LOG_INFO_MSGS	158	X'0000009E'
MQIA_QMOPT_LOG_REORG_MSGS	163	X'000000A3'
MQIA_QMOPT_LOG_SYSTEM_MSGS	164	X'000000A4'
MQIA_QMOPT_LOG_WARNING_MSGS	159	X'0000009F'
MQIA_QMOPT_TRACE_COMMS	166	X'000000A6'
MQIA_QMOPT_TRACE_CONVERSION	168	X'000000A8'
MQIA_QMOPT_TRACE_REORG	167	X'000000A7'
MQIA_QMOPT_TRACE_MQI_CALLS	165	X'000000A5'
MQIA_QMOPT_TRACE_SYSTEM	169	X'000000A9'
MQIA_QSG_DISP	63	X'0000003F'
MQIA_READ_AHEAD	189	X'000000BD'
MQIA_RECEIVE_TIMEOUT	111	X'0000006F'
MQIA_RECEIVE_TIMEOUT_MIN	113	X'00000071'
MQIA_RECEIVE_TIMEOUT_TYPE	112	X'00000070'
MQIA_REMOTE_EVENT	50	X'00000032'
MQIA_RETENTION_INTERVAL	21	X'00000015'
MQIA_REVERSE_DNS_LOOKUP	254	X'000000FE'
MQIA_SCOPE	45	X'0000002D'
MQIA_SECURITY_CASE	141	X'0000008D'
MQIA_SERVICE_CONTROL	139	X'0000008B'
MQIA_SERVICE_TYPE	121	X'00000079'
MQIA_SHAREABILITY	23	X'00000017'
MQIA_SHARED_Q_Q_MGR_NAME	77	X'0000004D'
MQIA_SSL_EVENT	75	X'0000004B'
MQIA_SSL_FIPS_REQUIRED	92	X'0000005C'
MQIA_SSL_RESET_COUNT	76	X'0000004C'
MQIA_SSL_TASKS	69	X'00000045'
MQIA_START_STOP_EVENT	52	X'00000034'
MQIA_STATISTICS_CHANNEL	129	X'00000081'
MQIA_STATISTICS_AUTO_CLUSSDR	130	X'00000082'
MQIA_STATISTICS_INTERVAL	131	X'00000083'
MQIA_STATISTICS_MQI	127	X'0000007F'
MQIA_STATISTICS_Q	128	X'00000080'

<i>Çizelge 185. Değişmezlerin değerleri (devamı var)</i>		
<b>Ad</b>	<b>Ondalık değer</b>	<b>Onaltılı değer</b>
MQIA_SUB_COUNT	204	X'000000CC'
MQIA_SUB_SCOPE	218	X'000000DA'
MQIA_SYNCPOINT	30	X'0000001E'
MQIA_TCP_CHANNELS	114	X'00000072'
MQIA_TCP_KEEP_ALIVE	115	X'00000073'
MQIA_TCP_STACK_TYPE	116	X'00000074'
MQIA_TIME_SINCE_RESET	35	X'00000023'
MQIA_TOPIC_DEF_PERSISTENCE	185	X'000000B9'
MQIA_TOPIC_NODE_COUNT	253	X'000000FD'
MQIA_TOPIC_TYPE	208	X'000000D0'
MQIA_TRACE_ROUTE_RECORDING	137	X'00000089'
MQIA_TREE_LIFE_TIME	183	X'000000B7'
MQIA_TRIGGER_CONTROL	24	X'00000018'
MQIA_TRIGGER_DEPTH	29	X'0000001D'
MQIA_TRIGGER_INTERVAL	25	X'00000019'
MQIA_TRIGGER_MSG_PRIORITY	26	X'0000001A'
MQIA_TRIGGER_TYPE	28	X'0000001C'
MQIA_TRIGGER_RESTART	91	X'0000005B'
MQIA_USAGE	12	X'0000000C'
MQIA_USE_DEAD_LETTER_Q	234	X'000000EA'
MQIA_USER_LIST	2000	X'000007D0'
MQIA_WILDCARD_OPERATION	216	X'000000D8'
MQIA_XR_CAPABILITY	243	X'000000F3'

### **MQIACF\_\* (Komut biçimi Tamsayı Değiştirgeci Tipleri)**

<i>Çizelge 186. Değişmezlerin değerleri</i>		
<b>Ad</b>	<b>Ondalık değer</b>	<b>Onaltılı değer</b>
MQIACF_FIRST	1001	X'000003E9'
MQIACF_Q_MGR_ATTRS	1001	X'000003E9'
MQIACF_Q_ATTRS	1002	X'000003EA'
MQIACF_PROCESS_ATTRS	1003	X'000003EB'
MQIACF_NAMELIST_ATTRS	1004	X'000003EC'
MQIACF_FORCE	1005	X'000003ED'
MQIACF_REPLACE	1006	X'000003EE'
MQIACF_PURGE	1007	X'000003EF'
MQIACF QUIESCE	1008	X'000003F0'
MQIACF_MODE	1008	X'000003F0'
MQIACF_ALL	1009	X'000003F1'
MQIACF_EVENT_APPL_TIP	1010	X'000003F2'

Çizelge 186. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQIACF_EVENT_ORIGIN	1011	X'000003F3'
MQIACF_PARAMETER_ID	1012	X'000003F4'
MQIACF_ERROR_ID	1013	X'000003F5'
MQIACF_ERROR_IDENTIFIER	1013	X'000003F5'
MQIACF_SELECTOR	1014	X'000003F6'
MQIACF_CHANNEL_ATTRS	1015	X'000003F7'
MQIACF_OBJECT_TYPE	1016	X'000003F8'
MQIACF_ESCAPE_TYPE	1017	X'000003F9'
MQIACF_ERROR_OFFSET	1018	X'000003FA'
MQIACF_AUTH_INFO_ATTRS	1019	X'000003FB'
MQIACF_NEDEN_NITELEYICISI	1020	X'000003FC'
MQIACF_COMMAND	1021	X'000003FD'
MQIACF_OPEN_SEÇENEKLER	1022	X'000003FE'
MQIACF_OPEN_TYPE	1023	X'000003FF'
MQIACF_PROCESS_ID	1024	X'00000400'
MQIACF_THREAD_ID	1025	X'00000401'
MQIACF_Q_STATUS_ATTRS	1026	X'00000402'
MQIACF_UNCOMMITTED_MSGS	1027	X'00000403'
MQIACF_HANDLE_STATE	1028	X'00000404'
MQIACF_AUX_ERROR_DATA_INT_1	1070	X'0000042E'
MQIACF_AUX_ERROR_DATA_INT_2	1071	X'0000042F'
MQIACF_CONV_REASON_CODE	1072	X'00000430'
MQIACF_BRIDGE_TIP	1073	X'00000431'
MQIACF_SORGU	1074	X'00000432'
MQIACF_WAIT_INTERVAL	1075	X'00000433'
MQIACF_OPTIONS	1076	X'00000434'
MQIACF_ROKER_OPTIONS	1077	X'00000435'
MQIACF_REFRESH_TYPE	1078	X'00000436'
MQIACF_SEQUENCE_NUMARASI	1079	X'00000437'
MQIACF_INTEGER_DATA	1080	X'00000438'
MQIACF_REGISTRATION_OPTIONS	1081	X'00000439'
MQIACF_YAYINLAMA SEÇENEKLERI	1082	X'0000043A'
MQIACF_CLUSTER_INFO	1083	X'0000043B'
MQIACF_Q_MGR_DEFINTION_TYPE	1084	X'0000043C'
MQIACF_Q_MGR_TIP	1085	X'0000043D'
MQIACF_ACTION	1086	X'0000043E'
MQIACF_SUSPEND	1087	X'0000043F'
MQIACF_ROKER_COUNT	1088	X'00000440'
MQIACF_APPL_COUNT	1089	X'00000441'

Çizelge 186. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQIACF_ANONIM_COUNT	1090	X'00000442'
MQIACF_REG_REG_OPTIONS	1091	X'00000443'
MQIACF_DELEE	1092	X'00000444'
MQIACF_CLUSTER_Q_MGR_ATTRS	1093	X'00000445'
MQIACF_REFRESH_INTERVAL	1094	X'00000446'
MQIACF_REFRESH_REPOSITORY	1095	X'00000447'
MQIACF_REMOVE_QUEUES	1096	X'00000448'
MQIACF_OPEN_INPUT_TYPE	1098	X'0000044A'
MQIACF_OPEN_OUTPUT	1099	X'0000044B'
MQIACF_OPEN_KÜMESİ	1100	X'0000044C'
MQIACF_OPEN_SORGULAMA	1101	X'0000044D'
MQIACF_OPEN_BROWSE	1102	X'0000044E'
MQIACF_Q_STATUS_TIP	1103	X'0000044F'
MQIACF_Q_HANDLE	1104	X'00000450'
MQIACF_Q_STATUS	1105	X'00000451'
MQIACF_SECURITY_Türü	1106	X'00000452'
MQIACF_CONNECTION_ATTRS	1107	X'00000453'
MQIACF_CONNECT_SEÇENEK	1108	X'00000454'
MQIACF_CONN_INFO_TYPE	1110	X'00000456'
MQIACF_CONN_INFO_CONN	1111	X'00000457'
MQIACF_CONN_INFO_HANDLE	1112	X'00000458'
MQIACF_CONN_INFO_ALL	1113	X'00000459'
MQIACF_AUTH_PROFILE_ATTRS	1114	X'0000045A'
MQIACF_AUTHORIZATION_LIST	1115	X'0000045B'
MQIACF_AUTH_ADD_AUTHS	1116	X'0000045C'
MQIACF_AUTH_REMOVE_AUTHS	1117	X'0000045D'
MQIACF_ENTITY_TYPE	1118	X'0000045E'
MQIACF_COMMAND_INFO	1120	X'00000460'
MQIACF_CMDScope_Q_MGR_COUNT	1121	X'00000461'
MQIACF_Q_MGR_SYSTEM	1122	X'00000462'
MQIACF_Q_MGR_OLAY	1123	X'00000463'
MQIACF_Q_MGR_DQM	1124	X'00000464'
MQIACF_Q_MGR_CLUSTER	1125	X'00000465'
MQIACF_QSG_DISPS	1126	X'00000466'
MQIACF_UOW_STATE	1128	X'00000468'
MQIACF_SECURITY_Öğesi	1129	X'00000469'
MQIACF_CF_STRUC_STATUS	1130	X'0000046A'
MQIACF_UOW_TYPE	1132	X'0000046C'
MQIACF_CF_STRUC_ATTRS	1133	X'0000046D'

Çizelge 186. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQIACF_EXCLUDE_INTERVAL	1134	X'0000046E'
MQIACF_CF_STATUS_TIP	1135	X'0000046F'
MQIACF_CF_STATUS_SUMMARY	1136	X'00000470'
MQIACF_CF_STATUS_CONNECT	1137	X'00000471'
MQIACF_CF_STATUS_BACKUP	1138	X'00000472'
MQIACF_CF_STRUC_TYPE	1139	X'00000473'
MQIACF_CF_STRUC_SIZE_MAX	1140	X'00000474'
MQIACF_CF_STRUC_SIZE_KULLANILDI	1141	X'00000475'
MQIACF_CF_STRUC_ENTRIES_MAX	1142	X'00000476'
MQIACF_CF_STRUC_ENTRIES_KULLANILDI	1143	X'00000477'
MQIACF_CF_STRUC_BACKUP_SIZE	1144	X'00000478'
MQIACF_MOVE_TIP	1145	X'00000479'
MQIACF_MOVE_TYPE_MOVE	1146	X'0000047A'
MQIACF_MOVE_TYPE_ADI	1147	X'0000047B'
MQIACF_Q_MGR_NUMARASI	1148	X'0000047C'
MQIACF_Q_MGR_STATUS	1149	X'0000047D'
MQIACF_DB2_CONN_STATUS	1150	X'0000047E'
MQIACF_SECURITY_ATTRS	1151	X'0000047F'
MQIACF_SECURITY_TIMEOUT	1152	X'00000480'
MQIACF_SECURITY_INTERVAL	1153	X'00000481'
MQIACF_SECURITY_SWITCH	1154	X'00000482'
MQIACF_SECURITY_SETTING	1155	X'00000483'
MQIACF_STORAGE_CLASS_ATTRS	1156	X'00000484'
MQIACF_USAGE_TIP	1157	X'00000485'
MQIACF_BUFFER_POOL_ID	1158	X'00000486'
MQIACF_USAGE_TOKAL_SAYFALARI	1159	X'00000487'
MQIACF_USAGE_UNUSED_PAGES	1160	X'00000488'
MQIACF_USAGE_PERSIST_PAGES	1161	X'00000489'
MQIACF_USAGE_NONPERSIST_PAGES	1162	X'0000048A'
MQIACF_USAGE_RESTART_EXTANS	1163	X'0000048B'
MQIACF_USAGE_EXPAND_COUNT	1164	X'0000048C'
MQIACF_PAGESET_STATUS	1165	X'0000048D'
MQIACF_USAGE_TOKAL_BUFFERS	1166	X'0000048E'
MQIACF_USAGE_DATA_SET_TYPE	1167	X'0000048F'
MQIACF_USAGE_PAGESET	1168	X'00000490'
MQIACF_USAGE_DATA_KÜMESI	1169	X'00000491'
MQIACF_USAGE_BUFFER_POOL	1170	X'00000492'
MQIACF_MOVE_COUNT	1171	X'00000493'
MQIACF_EXPIRY_Q_COUNT	1172	X'00000494'

Çizelge 186. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQIACF_CONFIGURATION_OBJECTS	1173	X'00000495'
MQIACF_CONFIGURATION_EVENTS	1174	X'00000496'
MQIACF_SYSP_TYPE	1175	X'00000497'
MQIACF_SYSP_DEALLOC_INTERVAL	1176	X'00000498'
MQIACF_SYSP_MAX_ARCHEVE	1177	X'00000499'
MQIACF_SYSP_MAX_READ_TAPES	1178	X'0000049A'
MQIACF_SYSP_IN_BUFFER_SIZE	1179	X'0000049B'
MQIACF_SYSP_OUT_BUFFER_SIZE	1180	X'0000049C'
MQIACF_SYSP_OUT_BUFFER_COUNT	1181	X'0000049D'
MQIACF_SYSP_ARCHEVE	1182	X'0000049E'
MQIACF_SYSP_DUAL_ETKİN	1183	X'0000049F'
MQIACF_SYSP_DUAL_ARCHE	1184	X'000004A0'
MQIACF_SYSP_DUAL_BSDS	1185	X'000004A1'
MQIACF_SYSP_MAX_CONNS	1186	X'000004A2'
MQIACF_SYSP_MAX_CONNS_ASK	1187	X'000004A3'
MQIACF_SYSP_MAX_CONNS_BACK	1188	X'000004A4'
MQIACF_SYSP_EXIT_INTERVAL	1189	X'000004A5'
MQIACF_SYSP_EXIT_GÖREVLERİ	1190	X'000004A6'
MQIACF_SYSP_CHKPOINT_COUNT	1191	X'000004A7'
MQIACF_SYSP_OTMA_INTERVAL	1192	X'000004A8'
MQIACF_SYSP_Q_INDEX_DEFER	1193	X'000004A9'
MQIACF_SYSP_DB2_TASKS	1194	X'000004AA'
MQIACF_SYSP_RESLEVEL_AUDIT	1195	X'000004AB'
MQIACF_SYSP_ROUTING_CODE	1196	X'000004AC'
MQIACF_SYSP_SMF_ACCOUNTING	1197	X'000004AD'
MQIACF_SYSP_SMF_STATS	1198	X'000004AE'
MQIACF_SYSP_SMF_INTERVAL	1199	X'000004AF'
MQIACF_SYSP_TRACE_SİNİFİ	1200	X'000004B0'
MQIACF_SYSP_TRACE_SIZE	1201	X'000004B1'
MQIACF_SYSP_WLM_INTERVAL	1202	X'000004B2'
MQIACF_SYSP_ALLOC_UNIT	1203	X'000004B3'
MQIACF_SYSP_ARCHIV_RETAIN	1204	X'000004B4'
MQIACF_SYSP_ARCHIV_WTOR	1205	X'000004B5'
MQIACF_SYSP_BLOCK_SIZE	1206	X'000004B6'
MQIACF_SYSP_KATALARI	1207	X'000004B7'
MQIACF_SYSP_COMPACT	1208	X'000004B8'
MQIACF_SYSP_ALLOC_PRIMARY	1209	X'000004B9'
MQIACF_SYSP_ALLOC_SECONARY	1210	X'000004BA'
MQIACF_SYSP_PROTECT	1211	X'000004BB'

Çizelge 186. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQIACF_SYSP_QUIESCE_INTERVAL	1212	X'000004BC'
MQIACF_SYSP_TIMESTAMP	1213	X'000004BD'
MQIACF_SYSP_UNIT_ADRESI	1214	X'000004BE'
MQIACF_SYSP_UNIT_STATUS	1215	X'000004BF'
MQIACF_SYSP_LOG_COPY	1216	X'000004C0'
MQIACF_SYSP_LOG_KULLANILDI	1217	X'000004C1'
MQIACF_SYSP_LOG_SUSPEND	1218	X'000004C2'
MQIACF_SYSP_OFFLOAD_STATUS	1219	X'000004C3'
MQIACF_SYSP_TOKAL_LOGS	1220	X'000004C4'
MQIACF_SYSP_FULL_LOGS	1221	X'000004C5'
MQIACF_LISTENER_ATTRS	1222	X'000004C6'
MQIACF_LISTENER_STATUSUS_ATTRS	1223	X'000004C7'
MQIACF_SERVICE_ATTRS	1224	X'000004C8'
MQIACF_SERVICE_STATUS_ATTRS	1225	X'000004C9'
MQIACF_Q_TIME_INDICATOR	1226	X'000004CA'
MQIACF_OLDEST_MSG_AGE	1227	X'000004CB'
MQIACF_AUTH_OPTIONS	1228	X'000004CC'
MQIACF_Q_MGR_STATUS_ATTRS	1229	X'000004CD'
MQIACF_CONNECTION_COUNT	1230	X'000004CE'
MQIACF_Q_MGR_FACICE	1231	X'000004CF'
MQIACF_CHINIT_STATUS	1232	X'000004D0'
MQIACF_CMD_SERVER_STATUS	1233	X'000004D1'
MQIACF_ROUTE_AYRINTILARI	1234	X'000004D2'
MQIACF_RECORDED_ACTIVITIES	1235	X'000004D3'
MQIACF_MAX_ETKINLIĞI	1236	X'000004D4'
MQIACF_DISCONTINUITY_COUNT	1237	X'000004D5'
MQIACF_ROUTE_BIRIKIM	1238	X'000004D6'
MQIACF_ROUTE_DELIÇE	1239	X'000004D7'
MQIACF_OPERATION_TIP	1240	X'000004D8'
MQIACF_BACKUT_COUNT	1241	X'000004D9'
MQIACF_COMP_CODE	1242	X'000004DA'
MQIACFENCODING	1243	X'000004DB'
MQIACF_EXPIRY	1244	X'000004DC'
MQIACF_GERIBILDIRIM	1245	X'000004DD'
MQIACF_MSG_FLAGS	1247	X'000004DF'
MQIACF_MSG_LENNGTH	1248	X'000004E0'
MQIACF_MSG_TYPE	1249	X'000004E1'
MQIACF_OFFSET	1250	X'000004E2'
MQIACF_ORIGINAL_LENGTH	1251	X'000004E3'

Çizelge 186. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQIACF_PERSISTENCE	1252	X'000004E4'
MQIACF_PRIORITY	1253	X'000004E5'
MQIACF_REASON_CODE	1254	X'000004E6'
MQIACF_REPORT	1255	X'000004E7'
MQIACF_VERSION	1256	X'000004E8'
MQIACF_UNRECORDED_ACTIVITIES	1257	X'000004E9'
MQIACF_MONITORING	1258	X'000004EA'
MQIACF_ROUTE_FORWINA	1259	X'000004EB'
MQIACF_SERVICE_STATUS	1260	X'000004EC'
MQIACF_Q_TYPS	1261	X'000004ED'
MQIACF_USER_ID_DESTEĞİ	1262	X'000004EE'
MQIACF_INTERFACE_VERSION	1263	X'000004EF'
MQIACF_AUTH_SERVICE_ATTRS	1264	X'000004F0'
MQIACF_USAGE_EXPAND_TYPE	1265	X'000004F1'
MQIACF_SYSP_CLUSTER_CACHE	1266	X'000004F2'
MQIACF_SYSP_DB2_BLOB_TASKS	1267	X'000004F3'
MQIACF_SYSP_WLM_INT_UNITS	1268	X'000004F4'
MQIACF_TOPIC_ATTRS	1269	X'000004F5'
MQIACF_PUBUBLE ÖZELLİĞİ	1271	X'000004F7'
MQIACF_DESTINATION_SİNİFİ	1273	X'000004F9'
MQIACF_DURABLE_SUBSCRIPTION	1274	X'000004FA'
MQIACF_SUBSCRIPTION_SCOPE	1275	X'000004FB'
MQIACF_VARIABLE_USER_ID	1277	X'000004FD'
MQIACF_REQUEST_ONLY	1280	X'00000500'
MQIACF_PUB_PRIORITY	1283	X'00000503'
MQIACF_SUB_ATTRS	1287	X'00000507'
MQIACF_WILDCARD_SCHEMA	1288	X'00000508'
MQIACF_SUB_TYPE	1289	X'00000509'
MQIACF_MESSAGE_COUNT	1290	X'0000050A'
MQIACF_Q_MGR_PUBSUB	1291	X'0000050B'
MQIACF_Q_MGR_SÜRÜMÜ	1292	X'0000050C'
MQIACF_SUB_STATUS_ATTRS	1294	X'0000050E'
MQIACF_TOPIC_STATUS	1295	X'0000050F'
MQIACF_TOPIC_SUB	1296	X'00000510'
MQIACF_TOPIC_PUB	1297	X'00000511'
MQIACF_RETAINED_YAYINLARI	1300	X'00000514'
MQIACF_TOPIC_STATUS_ATTRS	1301	X'00000515'
MQIACF_TOPIC_STATUS_TIP	1302	X'00000516'
MQIACF_XX_ENCODE_CASE_ONE sub_options	1303	X'00000517'




Çizelge 186. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQIACF_YAYINLAMA SAYISI	1304	X'00000518'
MQIACF_PLEAR_TIP	1305	X'00000519'
MQIACF_CLEU_SCOPE	1306	X'0000051A'
MQIACF_SUB_DÜZEYİ	1307	X'0000051B'
MQIACF_ASYNC_STATE	1308	X'0000051C'
MQIACF_SUB_SUMMARY	1309	X'0000051D'
MQIACF_OBSOLETE_MSGS	1310	X'0000051E'
MQIACF_PUBUBE DURUMU	1311	X'0000051F'
MQIACF_PS_STATUS_TIP	1314	X'00000522'
MQIACF_PUBSUB_STATUS_ATTRS	1318	X'00000526'
MQIACF_SELECTOR_TYPE	1321	X'00000529'
MQIACF_MCAST_REL_INDICATOR	1351	X'00000547'
MQIACF_CHATUTH_TYPE	1352	X'00000548'
MQXR_DIAGNOSTICS_TYPE	1354	X'0000054A'
MQIACF_CHATUTH_ATTRS	1355	X'0000054B'
MQIACF_OPERATION_ID	1356	X'0000054C'
MQIACF_API_CALLER_TYPE	1357	X'0000054D'
MQIACF_API_ENVIRONMENT	1358	X'0000054E'
MQIACF_TRACE_AYRINTILAR	1359	X'0000054F'
MQIACF_HOBJ	1360	X'00000550'
MQIACF_CALL_YAZISI	1361	X'00000551'
MQIACF_MQCB_OPERATION	1362	X'00000552'
MQIACF_MQCB_TYPE	1363	X'00000553'
MQIACF_MQCB_SEÇENEĞİ	1364	X'00000554'
MQIACF_CLOSE_OPTIONS	1365	X'00000555'
MQIACF_CTL_OPERATION	1366	X'00000556'
MQIACF_GET_OPTIONS	1367	X'00000557'
MQIACF_RECS_PUNENT	1368	X'00000558'
MQIACF_KNOWN_DEST_COUNT	1369	X'00000559'
MQIACF_UNKNOWN_DEST_COUNT	1370	X'0000055A'
MQIACF_INVALID_DEST_COUNT	1371	X'0000055B'
MQIACF_RESOLVED_TYPE	1372	X'0000055C'
MQIACF_PUT_OPTIONS	1373	X'0000055D'
MQIACF_BUFFER_LENGTH	1374	X'0000055E'
MQIACF_TRACE_DATA_LENGTH	1375	X'0000055F'
MQIACF_SMDS_EXPANDST	1376	X'00000560'
MQIACF_STRUC_LENGTH	1377	X'00000561'
MQIACF_ITEM_SAYISI	1378	X'00000562'
MQIACF_EXPIRY_TIME	1379	X'00000563'

Çizelge 186. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQIACF_CONNECT_TIME	1380	X'00000564'
MQIACF_DISCONNECT_TIME	1381	X'00000565'
MQIACF_HSUB	1382	X'00000566'
MQIACF_SUBRQ_SEÇENEKLER	1383	X'00000567'
MQIACF_XA_RMID	1384	X'00000568'
MQIACF_XA_FLAGS	1385	X'00000569'
MQIACF_XA_RETCODE	1386	X'0000056A'
MQIACF_XA_HANDLE	1387	X'0000056B'
MQIACF_XA_RETVAL	1388	X'0000056C'
MQIACF_STATUS_TYPE	1389	X'0000056D'
MQIACF_XA_SAYISI	1390	X'0000056E'
MQIACF_SELECTOR_COUNT	1391	X'0000056F'
MQIACF_SELECTS	1392	X'00000570'
MQIACF_INTATTR_COUNT	1393	X'00000571'
MQIACF_INTATTRS	1394	X'00000572'
MQIACF_SUBRQ_ACTION	1395	X'00000573'
MQIACF_NUM_PUBS	1396	X'00000574'
MQIACF_POINTER_SIZE	1397	X'00000575'
MQIACF_REMOVE_AUTHREC	1398	X'00000576'
MQIACF_XR_ATTRS	1399	X'00000577'
MQIACF_APPL_FUNCTION_TYPE	1400	X'00000578'
MQIACF_AMQP_ATTRS	1401	X'00000579'
MQIACF_EXPORT_TYPE	1402	X'0000057A'
MQIACF_EXPORT_ATTRS	1403	X'0000057B'
MQIACF_SYSTEM_OBJECTS	1404	X'0000057C'
MQIACF_CONNECTION_SWAP	1405	X'0000057D'
MQIACF_AMQP_DIAGNOSTICS_TYPE	1406	X'0000057E'
MQIACF_BUFFER_POOL_LOCATION	1408	X'00000580'
MQIACF_LDAP_CONNECTION_STATUS	1409	X'00000581'
MQIACF_SYSP_MAX_ACE_POOL	1410	X'00000582'
MQIACF_PAGECLAS	1411	X'00000583'
MQIACF_AUTH_REC_TYPE	1412	X'00000584'
MQIACF_SYSP_MAX_CONC_OFFLOADS	1413	X'00000585'
MQIACF_SYSP_ZHYPERWRITE	1414	X'00000586'
MQIACF_Q_MGR_STATUS_LOG	1415	X'00000587'
MQIACF_ARCHIVE_LOG_SIZE	1416	X'00000588'
MQIACF_MEDIA_LOG_SIZE	1417	X'00000589'
MQIACF_RESTART_LOG_SIZE	1418	X'0000058A'
MQIACF_REUSABLE_LOG_SIZE	1419	X'0000058B'

Çizelge 186. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQIACF_LOG_IN_USE	1420	X'0000058C'
MQIACF_LOG_UTILIZATION	1421	X'0000058D'
 MQIACF_IGNORE_STATE	1423	X'0000058F'
MQIACF_MOVBABLE_APPL_COUNT	1424	X'00000590'
MQIACF_APPL_INFO_ATTRS	1425	X'00000591'
MQIACF_APPL_MOVEABLE	1426	X'00000592'
MQIACF_REMOTE_QMGR_ETKİN	1427	X'00000593'
MQIACF_APPL_INFO_TYPE	1428	X'00000594'
MQIACF_APPL_INFO_APPL	1429	X'00000595'
MQIACF_APPL_INFO_QMGR	1430	X'00000596'
MQIACF_APPL_INFO_LOCAL	1431	X'00000597'
MQIACF_APPL_IMMOVABLE_COUNT	1432	X'00000598'
MQIACF_DENGELENDİ	1433	X'00000599'
MQIACF_BALSTATE	1434	X'0000059A'
MQIACF_APPL_IMMOVABLE_REASON	1435	X'0000059B'
MQIACF_DS_ENCRYPTED	1436	X'0000059C'
MQIACF_CUR_Q_FILE_SIZE	1437	X'0000059D'
MQIACF_CUR_MAX_FILE_SIZE	1438	X'0000059E'
MQIACF_BALNCILIK_TIP	1439	X'0000059F'
MQIACF_BALNCIN_SEÇENEK	1440	X'000005A0'
MQIACF_BALANCING_TIMEOUT	1441	X'000005A1'
MQIACF_SYSP_SMF_STAT_TIME_SECS	1442	X'000005A2'
MQIACF_SYSP_SMF_ACCT_TIME_MINS	1443	X'000005A3'
MQIACF_SYSP_SMF_ACCT_TIME_SECS	1444	X'000005A4'
 MQIACF_LAST_KULLANILDI	1441	X'000005A1'
 MQIACF_LAST_KULLANILDI	1444	X'000005A4'

## MQIACH\_\* (Komut biçimi Tamsayı Kanal Tipleri)

Çizelge 187. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQIACH_FIRST	1501	X'000005DD'
MQIACH_XMIT_PROTOCOL_TYPE	1501	X'000005DD'
MQIACH_BATCH_SIZE	1502	X'000005DE'
MQIACH_DISC_INTERVAL	1503	X'000005DF'
MQIACH_SHORT_TIMER	1504	X'000005E0'
MQIACH_SHORT_RETRY	1505	X'000005E1'
MQIACH_LONG_TIMER	1506	X'000005E2'
MQIACH_LONG_RETRY	1507	X'000005E3'

Çizelge 187. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQIACH_PUT_AUTHORITY	1508	X'000005E4'
MQIACH_SEQUENCE_NUMBER_WRAP	1509	X'000005E5'
MQIACH_MAX_MSG_LENGTH	1510	X'000005E6'
MQIACH_CHANNEL_TYPE	1511	X'000005E7'
MQIACH_DATA_COUNT	1512	X'000005E8'
MQIACH_NAME_COUNT	1513	X'000005E9'
MQIACH_MSG_SEQUENCE_NUMBER	1514	X'000005EA'
MQIACH_DATA_CONVERSION	1515	X'000005EB'
MQIACH_IN_DOUBT	1516	X'000005EC'
MQIACH_MCA_TYPE	1517	X'000005ED'
MQIACH_SESSION_COUNT	1518	X'000005EE'
MQIACH_ADAPTER	1519	X'000005EF'
MQIACH_COMMAND_COUNT	1520	X'000005F0'
MQIACH_SOCKET	1521	X'000005F1'
MQIACH_PORT	1522	X'000005F2'
MQIACH_CHANNEL_INSTANCE_TYPE	1523	X'000005F3'
MQIACH_CHANNEL_INSTANCE_ATTRS	1524	X'000005F4'
MQIACH_CHANNEL_ERROR_DATA	1525	X'000005F5'
MQIACH_CHANNEL_TABLE	1526	X'000005F6'
MQIACH_CHANNEL_STATUS	1527	X'000005F7'
MQIACH_INDOUBT_STATUS	1528	X'000005F8'
MQIACH_LAST_SEQ_NUMBER	1529	X'000005F9'
MQIACH_LAST_SEQUENCE_NUMBER	1529	X'000005F9'
MQIACH_CURRENT_MSGS	1531	X'000005FB'
MQIACH_CURRENT_SEQ_NUMBER	1532	X'000005FC'
MQIACH_CURRENT_SEQUENCE_NUMBER	1532	X'000005FC'
MQIACH_SSL_RETURN_CODE	1533	X'000005FD'
MQIACH_MSGS	1534	X'000005FE'
MQIACH_BYTES_SENT	1535	X'000005FF'
MQIACH_BYTES_RCVD	1536	X'00000600'
MQIACH_BYTES_RECEIVED	1536	X'00000600'
MQIACH_BATCHES	1537	X'00000601'
MQIACH_BUFFERS_SENT	1538	X'00000602'
MQIACH_BUFFERS_RCVD	1539	X'00000603'
MQIACH_BUFFERS_RECEIVED	1539	X'00000603'
MQIACH_LONG_RETRIES_LEFT	1540	X'00000604'
MQIACH_SHORT_RETRIES_LEFT	1541	X'00000605'
MQIACH_MCA_STATUS	1542	X'00000606'
MQIACH_STOP_REQUESTED	1543	X'00000607'


Çizelge 187. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQIACH_MR_COUNT	1544	X'00000608'
MQIACH_MR_INTERVAL	1545	X'00000609'
MQIACH_NPM_SPEED	1562	X'0000061A'
MQIACH_HB_INTERVAL	1563	X'0000061B'
MQIACH_BATCH_INTERVAL	1564	X'0000061C'
MQIACH_NETWORK_PRIORITY	1565	X'0000061D'
MQIACH_KEEP_ALIVE_INTERVAL	1566	X'0000061E'
MQIACH_BATCH_HB	1567	X'0000061F'
MQIACH_SSL_CLIENT_AUTH	1568	X'00000620'
MQIACH_ALLOC_RETRY	1570	X'00000622'
MQIACH_ALLOC_FAST_TIMER	1571	X'00000623'
MQIACH_ALLOC_SLOW_TIMER	1572	X'00000624'
MQIACH_DISC_RETRY	1573	X'00000625'
MQIACH_PORT_NUMBER	1574	X'00000626'
MQIACH_HDR_COMPRESSION	1575	X'00000627'
MQIACH_MSG_COMPRESSION	1576	X'00000628'
MQIACH_CLWL_CHANNEL_RANK	1577	X'00000629'
MQIACH_CLWL_CHANNEL_PRIORITY	1578	X'0000062A'
MQIACH_CLWL_CHANNEL_WEIGHT	1579	X'0000062B'
MQIACH_CHANNEL_DISP	1580	X'0000062C'
MQIACH_INBOUND_DISP	1581	X'0000062D'
MQIACH_CHANNEL_TYPES	1582	X'0000062E'
MQIACH_ADAPS_STARTED	1583	X'0000062F'
MQIACH_ADAPS_MAX	1584	X'00000630'
MQIACH_DISPS_STARTED	1585	X'00000631'
MQIACH_DISPS_MAX	1586	X'00000632'
MQIACH_SSLTASKS_STARTED	1587	X'00000633'
MQIACH_SSLTASKS_MAX	1588	X'00000634'
MQIACH_CURRENT_CHL	1589	X'00000635'
MQIACH_CURRENT_CHL_MAX	1590	X'00000636'
MQIACH_CURRENT_CHL_TCP	1591	X'00000637'
MQIACH_CURRENT_CHL_LU62	1592	X'00000638'
MQIACH_ACTIVE_CHL	1593	X'00000639'
MQIACH_ACTIVE_CHL_MAX	1594	X'0000063A'
MQIACH_ACTIVE_CHL_PAUSED	1595	X'0000063B'
MQIACH_ACTIVE_CHL_STARTED	1596	X'0000063C'
MQIACH_ACTIVE_CHL_STOPPED	1597	X'0000063D'
MQIACH_ACTIVE_CHL_RETRY	1598	X'0000063E'
MQIACH_LISTENER_STATUS	1599	X'0000063F'

Çizelge 187. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQIACH_SHARED_CHL_RESTART	1600	X'00000640'
MQIACH_LISTENER_CONTROL	1601	X'00000641'
MQIACH_BACKLOG	1602	X'00000642'
MQIACH_XMITQ_TIME_INDICATOR	1604	X'00000644'
MQIACH_NETWORK_TIME_INDICATOR	1605	X'00000645'
MQIACH_EXIT_TIME_INDICATOR	1606	X'00000646'
MQIACH_BATCH_SIZE_INDICATOR	1607	X'00000647'
MQIACH_XMITQ_MSGS_AVAILABLE	1608	X'00000648'
MQIACH_CHANNEL_SUBSTATE	1609	X'00000649'
MQIACH_SSL_KEY_RESETS	1610	X'0000064A'
MQIACH_COMPRESSION_RATE	1611	X'0000064B'
MQIACH_COMPRESSION_TIME	1612	X'0000064C'
MQIACH_MAX_XMIT_SIZE	1613	X'0000064D'
MQIACH_DEF_CHANNEL_DISP	1614	X'0000064E'
MQIACH_SHARING_CONVERSATIONS	1615	X'0000064F'
MQIACH_MAX_SHARING_CONVS	1616	X'00000650'
MQIACH_CURRENT_SHARING_CONVS	1617	X'00000651'
MQIACH_MAX_INSTANCES	1618	X'00000652'
MQIACH_MAX_INSTS_PER_CLIENT	1619	X'00000653'
MQIACH_CLIENT_CHANNEL_WEIGHT	1620	X'00000654'
MQIACH_CONNECTION_AFFINITY	1621	X'00000655'
MQIACH_AUTH_INFO_TYPES	1622	X'00000656'
MQIACH_RESET_REQUESTED	1623	X'00000657'
MQIACH_BATCH_DATA_LIMIT	1624	X'00000658'
MQIACH_MSG_HISTORY	1625	X'00000659'
MQIACH_MULTICAST_PROPERTIES	1626	X'0000065A'
MQIACH_NEW_SUBSCRIBER_HISTORY	1627	X'0000065B'
MQIACH_MC_HB_INTERVAL	1628	X'0000065C'
MQIACH_USE_CLIENT_ID	1629	X'0000065D'
MQIACH_MQTT_KEEP_ALIVE	1630	X'0000065E'
MQIACH_IN_DOUBT_IN	1631	X'0000065F'
MQIACH_IN_DOUBT_OUT	1632	X'00000660'
MQIACH_MSGS_SENT<	1633	X'00000661'
MQIACH_MSGS_RECEIVED	1634	X'00000662'
MQIACH_MSGS_RCVD	1634	X'00000662'
MQIACH_PENDING_OUT	1635	X'00000663'
MQIACH_AVAILABLE_CIPHERSPECS	1636	X'00000664'
MQIACH_MATCH	1637	X'00000665'
MQIACH_USER_SOURCE	1638	X'00000666'

Çizelge 187. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQIACH_WARNING	1639	X'00000667'
MQIACH_DEF_RECONNECT	1640	X'00000668'
MQIACH_CHANNEL_SUMMARY_ATTRS	1642	X'0000066A'
MQIACH_PROTOCOL	1643	X'0000066B'
MQIACH_AMQPKEEPALIVE	1644	X'0000066C'
MQIACH_SECURITY_PROTOCOL	1645	X'0000066D'
 MQIACH_SPL_PROTECTION	1646	X'0000066E'
MQIACH_LAST_USED	1646	X'0000066E'

### MQIAMO\_\* (Komut biçimi Tamsayı İzleme Parametresi Tipleri)

Çizelge 188. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQIAMO_FIRST	701	X'000002BD'
MQIAMO_AVG_BATCH_SIZE	702	X'000002BE'
MQIAMO_AVG_Q_TIME	703	X'000002BF'
MQIAMO_BACKOUTS	704	X'000002C0'
MQIAMO_BROWSES	705	X'000002C1'
MQIAMO_BROWSE_MAX_BYTES	706	X'000002C2'
MQIAMO_BROWSE_MIN_BYTES	707	X'000002C3'
MQIAMO_BROWSES_FAILED	708	X'000002C4'
MQIAMO_KAPANIR	709	X'000002C5'
MQIAMO_COMMITS	710	X'000002C6'
MQIAMO_COMMITS_FAILED	711	X'000002C7'
MQIAMO_CONNS	712	X'000002C8'
MQIAMO_CONNS_MAX	713	X'000002C9'
MQIAMO_DISKLER	714	X'000002CA'
MQIAMO_DISKS_IMPLICIT	715	X'000002CB'
MQIAMO_DISC_TYPE	716	X'000002CC'
MQIAMO_EXIT_TIME_AVG	717	X'000002CD'
MQIAMO_EXIT_TIME_MAX	718	X'000002CE'
MQIAMO_EXIT_TIME_MIN	719	X'000002CF'
MQIAMO_FULL_BATCHES	720	X'000002D0'
MQIAMO_GENERATED_MSGS	721	X'000002D1'
MQIAMO_GETS	722	X'000002D2'
MQIAMO_GET_MAX_BYTES	723	X'000002D3'
MQIAMO_GET_MIN_BYTES	724	X'000002D4'
MQIAMO_GETS_FAILED	725	X'000002D5'
MQIAMO_INCOMPLETE_BATCHES	726	X'000002D6'
MQIAMO_INQS	727	X'000002D7'

Çizelge 188. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQIAMO_MSGS	728	X'000002D8'
MQIAMO_NET_TIME_AVG	729	X'000002D9'
MQIAMO_NET_TIME_MAX	730	X'000002DA'
MQIAMO_NET_TIME_MIN	731	X'000002DB'
MQIAMO_OBJECT_COUNT	732	X'000002DC'
MQIAMO_AÇAR	733	X'000002DD'
MQIAMO_PUT1S	734	X'000002DE'
MQIAMO_TAKS	735	X'000002DF'
MQIAMO_PUT_MAX_BYTES	736	X'000002E0'
MQIAMO_PUT_MIN_BYTES	737	X'000002E1'
MQIAMO_PUT_RETRIES	738	X'000002E2'
MQIAMO_Q_MAX_DEPTH	739	X'000002E3'
MQIAMO_Q_MIN_DEPTH	740	X'000002E4'
MQIAMO_Q_TIME_AVG	741	X'000002E5'
MQIAMO_Q_TIME_MAX	742	X'000002E6'
MQIAMO_Q_TIME_MIN	743	X'000002E7'
MQIAMO_SETS	744	X'000002E8'
MQIAMO_CONNS_FAILED	749	X'000002ED'
MQIAMO_OPENS_FAILED	751	X'000002EF'
MQIAMO_INQS_FAILED	752	X'000002F0'
MQIAMO_SETS_FAILED	753	X'000002F1'
MQIAAMI_PUTS_FAILED	754	X'000002F2'
MQIAMO_PUT1S_FAILED	755	X'000002F3'
MQIAMO_CLOSES_FAILED	757	X'000002F5'
MQIAMO_MSGS_EXHID	758	X'000002F6'
MQIAMO_MSGS_NOT_KUYRUKLANAN	759	X'000002F7'
MQIAMO_MSGS_TEMIZLENDI	760	X'000002F8'
MQIAMO_SUBS_DUR	764	X'000002FC'
MQIAMO_SUBS_NDUR	765	X'000002FD'
MQIAMO_SUBS_FAILED	766	X'000002FE'
MQIAMO_SUBRQS	767	X'000002FF'
MQIAMO_SUBRQS_FAILED	768	X'00000300'
MQIAMO_CBS	769	X'00000301'
MQIAMO_CBS_FAILED	770	X'00000302'
MQIAMO_CTLs	771	X'00000303'
MQIAMO_CTLs_FAILED	772	X'00000304'
MQIAMO_STATS	773	X'00000305'
MQIAMO_STATS_FAILED	774	X'00000306'
MQIAMO_SUB_DUR_HIGHWATER	775	X'00000307'



Çizelge 188. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQIAMO_SUB_DUR_LOWWATER	776	X'00000308'
MQIAMO_SUB_NDUR_HIGHWATER	777	X'00000309'
MQIAMO_SUB_NDUR_LOWWATER	778	X'0000030A'
MQIAMO_TOPIC_KOYAR	779	X'0000030B'
MQIAMO_TOPIC_PUTS_FAILED	780	X'0000030C'
MQIAMO_TOPIC_PUT1S	781	X'0000030D'
MQIAMO_TOPIC_PUT1S_FAILED	782	X'0000030E'
MQIAMO_PUBLISH_MSG_COUNT	784	X'00000310'
MQIAMO_UNSUBS_DUR	786	X'00000312'
MQIAMO_UNSUBS_NDUR	787	X'00000313'
MQIAMO_UNSUBS_FAILED	788	X'00000314'
MQIAMO_INTERVAL	789	X'00000315'
MQIAMO_MSGS_GÖNDERİLDİ	790	X'00000316'
MQIAMO_BYTES_GÖNDERİLDİ	791	X'00000317'
MQIAMO_REPAIR_BYTES	792	X'00000318'
MQIAMO_FEEDBACK_MODE	793	X'00000319'
MQIAMO_RIABILITY_TYPE	794	X'0000031A'
MQIAMO_LATE_JOIN_MARK	795	X'0000031B'
MQIAMO_NACKS_RCVD	796	X'0000031C'
MQIAMO_REPAIR_PKTS	797	X'0000031D'
MQIAMO_HISTORY_PKTS	798	X'0000031E'
MQIAMO_PENDING_PKTS	799	X'0000031F'
MQIAMO_PKT_RATE	800	X'00000320'
MQIAMO_MCAST_XMIT_RATE	801	X'00000321'
MQIAMO_MCAST_BATCH_TIME	802	X'00000322'
MQIAMO_MCAST_HEARBEAT	803	X'00000323'
MQIAMO_DEST_DATA_KAPISI	804	X'00000324'
MQIAMO_DEST_REPAIR_PORT	805	X'00000325'
MQIAAMI_ACKS_RCVD	806	X'00000326'
MQIAMO_ACTIVE_ACKERS	807	X'00000327'
MQIAMO_PKTS_SENT	808	X'00000328'
MQIAMO_TOTAL_REPAIR_PKTS	809	X'00000329'
MQIAMO_TOTAL_PKTS_GÖNDERİLDİ	810	X'0000032A'
MQIAMO_TOTAL_MSGS_GÖNDERİLDİ	811	X'0000032B'
MQIAMO_TOTAL_BYTES_GÖNDERİLDİ	812	X'0000032C'
MQIAMO_NUM_STREAMS	813	X'0000032D'
MQIAMO_ACK_GERIBILDIRIM	814	X'0000032E'
MQIAMO_NACK_GERIBILDIRIM	815	X'0000032F'
MQIAMO_PKTS_LOST	816	X'00000330'

Çizelge 188. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQIAMO_MSGS_RCVD	817	X'00000331'
MQIAMO_MSG_BYTES_RCVD	818	X'00000332'
MQIAMO_MSGS_TESLIM	819	X'00000333'
MQIAAMO_PKTS_PROCESSE	820	X'00000334'
MQIAMO_PKTS_DLVD	821	X'00000335'
MQIAMO_PKTS_ATILDI	822	X'00000336'
MQIAMO_PKTS_YINELENEN	823	X'00000337'
MQIAMO_NACKS_CREATED	824	X'00000338'
MQIAMO_NACK_PKTS_SENT	825	X'00000339'
MQIAMO_REPAIR_PKTS_RQSTD	826	X'0000033A'
MQIAMO_REPAIR_PKTS_RCVD	827	X'0000033B'
MQIAAMO_PKTS_ONREIR	828	X'0000033C'
MQIAMO_TOTAL_MSGS_RCVD	829	X'0000033D'
MQIAMO_TOTAL_MSGS_BYTES_RCVD	830	X'0000033E'
MQIAMO_TOTAL_REPAIR_PKTS_RCVD	831	X'0000033F'
MQIAMO_TOTAL_REPAIR_PKTS_RQSTD	832	X'00000340'
MQIAMO_TOTAL_MSGS_PROCESITT	833	X'00000341'
MQIAMO_TOTAL_MSGS_SELECTED	834	X'00000342'
MQIAMO_TOTAL_MSGS_EXHID	835	X'00000343'
MQIAMO_TOTAL_MSGS_TESLIM edildi	836	X'00000344'
MQIAMO_TOTAL_MSGS_RED	837	X'00000345'
MQIAMO_LAST_KULLANILDI	837	X'00000345'

### **MQIAMO64\_\* (Komut biçimi 64-bit Tamsayı İzleme Parametresi Tipleri)**

Çizelge 189. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQIAMO64_AVG_Q_TIME	703	X'000002BF'
MQIAMO64_Q_TIME_AVG	741	X'000002E5'
MQIAMO64_Q_TIME_MAX	742	X'000002E6'
MQIAMO64_Q_TIME_MIN	743	X'000002E7'
MQIAMO64_BROWSE_BYTES	745	X'000002E9'
MQIAMO64_BYTES	746	X'000002EA'
MQIAMO64_GET_BYTES	747	X'000002EB'
MQIAMO64_PUT_BYTES	748	X'000002EC'
MQIAMO64_TOPIC_PUT_BYTES	783	X'0000030F'
MQIAMO64_PUBLISH_MSG_BYTES	785	X'00000311'

## MQIASY\_\* (Tamsayı Sistem Seçicileri)

Çizelge 190. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
İLK OLARAK	-1	X'FFFFFFFF'
MQIASY_CODED_CHAR_SET_ID	-1	X'FFFFFFFF'
MQIASY_TIP	-2	X'FFFFFFFE'
MQIASY_COMMAND	-3	X'FFFFFFFD'
MQIASY_MSG_SEQ_NUMARASI	-4	X'FFFFFFFC'
MQIASY_CONTROL	-5	X'FFFFFFFB'
MQIASY_COMP_CODE	-6	X'FFFFFFFA'
MQIASY_REASON	-7	X'FFFFFFF9'
MQIASY_BAG_SEÇENEKL	-8	X'FFFFFFF8'
MQIASY_Sürümü	-9	X'FFFFFFF7'
MQIASY_LAST_KULLANILDI	-9	X'FFFFFFF7'
MQIASY_SON	-2000	X'FFFFFF830'

## MQIAUT\_\* (IMS bilgi üstbilgisi kimliği doğrulayıcı)

Çizelge 191. Değişmez adlar ve değerler	
Ad	Değer
MQIAUT_NONE	"ֆֆֆֆֆֆֆֆ"
MQIAUT_NONE_DIZISI	'ֆ','ֆ','ֆ','ֆ','ֆ','ֆ','ֆ','ֆ','ֆ','ֆ'

**Not:** ֆ simgesi tek bir boş karakteri temsil eder.

## MQIAV\_\* (Tamsayı Özniteliği Değerleri)

Çizelge 192. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQIAV_NOT_GEÇERLI	-1	X'FFFFFFFF'
MQIAV_UNDEFINED	-2	X'FFFFFFFE'

## MQICM\_\* (IMS bilgi üstbilgisi Kesinleştir Kipleri)

Çizelge 193. Değişmez adlar ve değerler	
Ad	Değer
MQICM_COMMIT_THEN_SEND	'0'
MQICM_SEND_THEN_COMMIT	'1'

## MQIDO\_\* (Komut biçimi Kuşku Seçenekleri Seçenekleri)

Çizelge 194. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQIDO_COMMIT	1	X'00000001'
MQIDO_BACKUT	2	X'00000002'

## MQIEP\_\* (Arabirim giriş noktaları)

### Bağlantı güvenliği parametreleri yapısı

Çizelge 195. Değişmezlerin yapıları	
Ad	Yapı
MQIEP_STRUC_ID	"IEP~"
MQIEP_STRUC_ID_ARRAY	'I', 'E', 'P', '~'

**Not:** ~ simgesi tek bir boş karakteri temsil eder.

Çizelge 196. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQIEP_VERSION_1	1	X'00000001'
MQDXP_CURRENT_VERSION	1	X'00000001'

## MQIGQ\_\* (Intra-Group queuing)

Çizelge 197. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQIGQ_DISABLE	0	X'00000000'
MQIGQ_ENABLED	1	X'00000001'

## MQIGQPA\_\* (Intra-Group queuing put Authority)

Çizelge 198. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQIGQPA_VARSAYILAN	1	X'00000001'
MQIGQPA_CONTEXT	2	X'00000002'
MQIGQPA_ONLY_IGQ	3	X'00000003'
MQIGQPA_ALTERNATE_OR_IGQ	4	X'00000004'

## MQIIH\_\* (IMS bilgi üstbilgisi yapısı ve işaretleri)

### IMS bilgi üstbilgisi yapısı

Çizelge 199. Değişmezlerin yapıları	
Ad	Yapı
MQIIH_STRUC_ID	"IIH~"
MQIIH_STRUC_ID_ARRAY	'I', 'I', 'H', '~'

**Not:** ~ simgesi tek bir boş karakteri temsil eder.

Çizelge 200. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQIIH_VERSION_1	1	X'00000001'
MQIIH_CURRENT_VERSION	1	X'00000001'
MQIIH_LENGTH_1	84	X'00000054'

## IMS bilgi üstbilgisi işaretleri

Çizelge 201. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQIH_NONE	0	X'00000000'
MQIIH_PASS_EXPIRATION	1	X'00000001'
MQIH_UNlimiED_EXPIRATION	0	X'00000000'
MQIH_REPLY_FORMAT_NONE	8	X'00000008'
MQIH_IGNORE_PURG	16	X'00000010'
MQIIH_CM0_REQUEST_RESPONSE	32	X'00000020'

## MQIMPO\_\* (Sorgula ilişkin ileti özelliği seçenekleri ve yapısı)

### Sorgu iletisi özellik seçenekleri yapısı

Çizelge 202. Değişmezlerin yapıları	
Ad	Yapı
MQIMPO_STRUC_ID	"IMPO"
MQIMPO_STRUC_ID_ARRAY	'I','M','P','O'

**Not:** ↵ simgesi tek bir boş karakteri temsil eder.

Çizelge 203. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQIMPO_VERSION_1	1	X'00000001'
MQIMPO_CURRENT_VERSION	1	X'00000001'

## Sorgu İleti Özelliği Seçenekleri

Çizelge 204. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQIMPO_CONVERT_TYPE	2	X'00000002'
MQIMPO_QUERY_LENGTH	4	X'00000004'
MQIMPO_INQ_FIRST	0	X'00000000'
MQIMPO_INQ_Next (SONRAKI)	8	X'00000008'
MQIMPO_INQ_PROP_UNDER_CURSOR	16	X'00000010'
MQIMPO_CONVERT_VALUE	32	X'00000020'
MQIMPO_NONE	0	X'00000000'

## MQINBD\_\* (Komut biçimi Gelen Düzenleri)

Çizelge 205. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQINBD_Q_MGR	0	X'00000000'
MQINBD_GROUP	3	X'00000003'

## MQIND\_\* (Özel Dizin Değerleri)

Çizelge 206. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQIND_NONE	-1	X'FFFFFFFF'
MQIND_ALL	-2	X'FFFFFFFE'

## MQIPADDR\_\* (IP Adresi Sürümleri)

Çizelge 207. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQIPADDR_IPV4	0	X'00000000'
MQIPADDR_IPV6	1	X'00000001'

## MQISS\_\* (IMS bilgi üstbilgisi Güvenlik Kapsamları)

Çizelge 208. Değişmez adlar ve değerler	
Ad	Değer
MQISS_CHECK	'C'
MQISS_FULL	'F'

## MQIT\_\* (Dizin Tipleri)

Çizelge 209. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQIT_YOK	0	X'00000000'
MQIT_MSG_ID	1	X'00000001'
MQIT_COREL_ID	2	X'00000002'
MQIT_MSG_TOKEN	4	X'00000004'
MQIT_GROUP_ID	5	X'00000005'

## MQITEM\_\* (mqInquireItemInfo için Öğe Tipi)

Çizelge 210. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQITEM_INTEGER	1	X'00000001'
MQITEM_STRING	2	X'00000002'
MQITEM_BAG	3	X'00000003'
MQITEM_BYTE_STRING	4	X'00000004'
MQITEM_INTEGER_FILTER	5	X'00000005'
MQITEM_STRING_FILTER	6	X'00000006'
MQITEM_INTEGER64	7	X'00000007'
MQITEM_BYTE_STRING_FILTER	8	X'00000008'

## MQITII\_\* (IMS bilgi üstbilgisi Hareket Eşgörünümü Tanıtıcısı)

Çizelge 211. Değişmez adlar ve değerler	
Ad	Değer
MQITII_NONE	X'00...00' (16 boş değer)
MQITII_NONE_DIZISI	'\0', '\0', ... (16 boş değer)

## MQITS\_\* (IMS bilgi üstbilgisi hareket durumları)

Çizelge 212. Değişmez adlar ve değerler	
Ad	Değer
MQITS_IN_MUHABBET	'C'
MQITS_NOT_IN_SOHBET	'-'
MQITS_ARCHITECTED	'A'

**Not:** - simgesi tek bir boş karakteri temsil eder.

## MQKAI\_\* (KeepAlive Aralığı)

Çizelge 213. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQKI_AUTO	-1	X'FFFFFFFF'

## MQMASTER\_\* (Ana denetim)

Çizelge 214. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQMASTER_NO	0	X'00000000'
MQMASTER_EVET	1	X'00000001'

## MQMCAS\_\* (Komut biçimi İleti Kanalı Aracısı Durumu)

Çizelge 215. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQMCAS_DURDU	0	X'00000000'
MQMCAS_RUNNING	3	X'00000003'

## MQMCAT\_\* (MCA Tipleri)

Çizelge 216. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQMCAT_PROCESI	1	X'00000001'
MQMCAT_THREAD	2	X'00000002'

## MQMCD\_\* (Yayınlama/Abone Olma Seçenekleri Etiket Bilgileri)

### Yayınlama/Abone Olma Seçenekleri Etiket İletisi İçerik Tanımlayıcısı (mcd) Etiketleri

Çizelge 217. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQMCD_KLASÖRÜ	1	X'00000001'

### Yayınlama/Abone Olma Seçenekleri Etiket Etiketleri adları

Çizelge 218. Değişmez adlar ve değerler

Ad	Değer
MQMCD_MSG_DOMAIN	"Msd"
MQMCD_MSG_KÜMESİ	"Set"
MQMCD_MSG_TYPE	"Type"
MQMCD_MSG_BIÇIMI	"Fmt"

### Yayınlama/Abone Olma Seçenekleri Etiket XML Etiketleri adları

Çizelge 219. Değişmez adlar ve değerler

Ad	Değer
MQMCD_MSG_DOMAIN_B	"<Msd>"
MQMCD_MSG_DOMAIN_E	"</Msd>"
MQMCD_MSG_SET_B	"<Set>"
MQMCD_MSG_SET_E	"</Set>"
MQMCD_MSG_TYPE_B	"<Type>"
MQMCD_MSG_TYPE_E	"</Type>"
MQMCD_MSG_FORMAT_B	"<Fmt>"
MQMCD_MSG_FORMAT_E	"</Fmt>"

### Yayınlama/Abone Olma Seçenekleri Etiket Etiketleri değerleri

Çizelge 220. Değişmez adlar ve değerler

Ad	Değer
MQMCD_DOMAIN_YOK	"none"
MQMCD_DOMAIN_NEON	"neon"
MQMCD_DOMAIN_MRM	"mrm"
MQMCD_DOMAIN_JMS_NONE	"jms_none"
MQMCD_DOMAIN_JMS_TEXT	"jms_text"
MQMCD_DOMAIN_JMS_OBJECT	"jms_object"
MQMCD_DOMAIN_JMS_MAP	"jms_map"
MQMCD_DOMAIN_JMS_STREAM	"jms_stream"
MQMCD_DOMAIN_JMS_BYTES	"jms_bytes"



## MQMD\_\* (İleti tanımlayıcı yapısı)

Çizelge 221. Değişmezlerin yapıları	
Ad	Yapı
MQMD_STRUC_ID	"MD↵"
MQMD_STRUC_ID_ARRAY	'M', 'D', '↵', '↵'

**Not:** ↵ simgesi tek bir boş karakteri temsil eder.

Çizelge 222. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQMD_VERSION_1	1	X'00000001'
MQMD_VERSION_2	2	X'00000002'
MQMD_CURRENT_VERSION	2	X'00000002'

## MQMDE\_\* (İleti tanımlayıcı uzantısı yapısı)

Çizelge 223. Değişmezlerin yapıları	
Ad	Yapı
MQMDE_STRUC_ID	"MDE↵"
MQMDE_STRUC_ID_ARRAY	'M', 'D', 'E', '↵'

**Not:** ↵ simgesi tek bir boş karakteri temsil eder.

Çizelge 224. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQMDE_VERSION_2	2	X'00000002'
MQMDE_CURRENT_SÜRÜM	2	X'00000002'
MQMDE_LENGTH_2	72	X'00000048'

## MQMDEF\_\* (İleti tanımlayıcısı uzantı işaretleri)

Çizelge 225. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQMDEF_NONE	0	X'00000000'

## MQMDS\_\* (İleti Teslim Sırası)

Çizelge 226. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQMDS_PRIORITY	0	X'00000000'
MQMDS_FIFO	1	X'00000001'

## MQMF\_\* (İleti İşaretleri)

Çizelge 227. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQMF_SEGMENTATION_ENGELLENDI	0	X'00000000'
MQMF_SEGMENTATION_ALLOW	1	X'00000001'

Çizelge 227. Değişmezlerin değerleri (devamı var)		
Ad	Ondalık değer	Onaltılı değer
MQMF_MSG_IN_GROUP	8	X'00000008'
MQMF_LAST_MSG_IN_GROUP	16	X'00000010'
MQMF_SEGMENT	2	X'00000002'
MQMF_LAST_SEGMENT	4	X'00000004'
MQMF_NONE	0	X'00000000'

## MQMHBO\_\* (Arabellek seçenekleri ve yapıyla ilgili ileti tanıtıcısı)

### İleti tanıtıcısı-arabellek seçenekleri yapısı

Çizelge 228. Değişmezlerin yapıları	
Ad	Yapı
MQMHBO_STRUCT_ID	"MHBO"
MQMHBO_STRUC_ID_ARRAY	'M', 'H', 'B', 'O'

**Not:** ↵ simgesi tek bir boş karakteri temsil eder.

Çizelge 229. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQMHBO_VERSION_1	1	X'00000001'
MQMHBO_CURRENT_VERSION	1	X'00000001'

### İleti Tanıtıcısı-Arabellek Seçenekleri

Çizelge 230. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQMHBO_PROPERTIES_IN_MQRFH2	1	X'00000001'
MQMHBO_DELETE_PROPERTIES	2	X'00000002'
MQMHBO_NONE	0	X'00000000'

## MQMI\_\* (İleti Tanıtıcısı)

Çizelge 231. Değişmez adları ve değerleri	
Ad	Değer
MQMI_NONE	X'00...00' (24 boş değer)
MQMI_NONE_DIZISI	'\0', '\0', ... (24 boş değer)

## MQMMBI\_\* (İleti İşareti-Göz At Aralığı)

Çizelge 232. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQMMBI_UNSI_NMA	-1	X'FFFFFFFF'

## MQMO\_\* (Eşleme Seçenekleri)

Çizelge 233. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQMO_MATCH_MSG_ID	1	X'00000001'
MQMO_MATCH_COREL_ID	2	X'00000002'
MQMO_MATCH_GROUP_ID	4	X'00000004'
MQMO_MATCH_MSG_SEQ_NUMBER	8	X'00000008'
MQMO_MATCH_OFFSET	16	X'00000010'
MQMO_MATCH_MSG_TOKEN	32	X'00000020'
MQMO_NONE	0	X'00000000'

## MQMODE\_\* (Komut biçimi Kipi Seçenekleri)

Çizelge 234. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQMODE_FORCE	0	X'00000000'
MQMODE_QUIESCE	1	X'00000001'
MQMODE_TERMINATE	2	X'00000002'

## MQMON\_\* (İzleme Değerleri)

Çizelge 235. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQMON_NOT_VAR	-1	X'FFFFFFFF'
MQMON_NONE	-1	X'FFFFFFFF'
MQMON_Q_MGR	-3	X'FFFFFFFD'
MQMON_OFF	0	X'00000000'
MQMON_ON	1	X'00000001'
MQMON_DEVRE DIŞI	0	X'00000000'
MQMON_ENABLED	1	X'00000001'
MQMON_LOW	17	X'00000011'
MQMON_ORTAMI	33	X'00000021'
MQMON_YÜKSEK	65	X'00000041'

## MQMT\_\* (İleti Tipleri)

Çizelge 236. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQMT_SYSTEM_FIRST	1	X'00000001'
MQMT_REQUEST	1	X'00000001'
MQMT_REPLY	2	X'00000002'
MQMT_DATAGRAM	8	X'00000008'
MQMT_REPORT	4	X'00000004'
MQMT_MQE_FIELDS_FROM_MQE	112	X'00000070'

Çizelge 236. Değişmezlerin değerleri (devamı var)		
Ad	Ondalık değer	Onaltılı değer
MQMT_MQE_FIELDS	113	X'00000071'
MQMT_SYSTEM_SON	65535	X'0000FFFF'
MQMT_APPL_FIRST	65536	X'00010000'
MQMT_APPL_SON	99999999	X'3B9AC9FF'

### MQMTOK\_\* (İleti Simgesi)

Çizelge 237. Değişmez adlar ve değerler	
Ad	Değer
MQMTOK_NONE	X'00...00' (16 boş değer)
MQMTOKT_NONE_DIZISI	'\0', '\0', ... (16 boş değer)

Çizelge 238. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQMTOK_NONE	X'00...00'	(16 nulls)
MQMTOKT_NONE_DIZISI	'\0', '\0', ...	(16 nulls)

### MQNC\_\* (Ad Sayısı)

Çizelge 239. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQNC_MAX_NAMELIST_NAME_COUNT	256	X'00000100'

### MQNPM\_\* (Kalıcı Olmayan İleti Sınıfı)

Çizelge 240. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQNPM_CLASS_NORMAL	0	X'00000000'
MQNPM_CLASS_YükSEK	10	X'0000000A'

### MQNPMS\_\* (NonPersistent-İleti Hızları)

Çizelge 241. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQNPMS_NORMAL	1	X'00000001'
MQNPMS_FAST	2	X'00000002'

### MQNT\_\* (Ad Listesi Tipleri)

Çizelge 242. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQNT_NONE	0	X'00000000'
MQNT_Q	1	X'00000001'
MQNT_CLUSTER	2	X'00000002'
MQNT_AUTH_INFO	4	X'00000004'

Çizelge 242. Değişmezlerin değerleri (devamı var)		
Ad	Ondalık değer	Onaltılı değer
MQNT_ALL	1001	X'000003E9'

### **MQNVS\_\* (Ad/Değer Dizgisi için adlar)**

Çizelge 243. Değişmez adlar ve değerler	
Ad	Değer
MQNVS_APPL_TYPE	"OPT_APP_GRP↵"
MQNVS_MSG_TYPE	"OPT_MSG_TYPE↵"

**Not:** ↵ simgesi tek bir boş karakteri temsil eder.

### **MQOA\_\* (Nesne özniteliklerine ilişkin Seçiciler için sınırlar)**

Çizelge 244. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQOA_FIRST	1	X'00000001'
MQOA_SON	9000	X'00002328'

### **MQOD\_\* (Nesne tanımlayıcı yapısı)**

Çizelge 245. Değişmezlerin yapıları	
Ad	Yapı
MQOD_STRUC_ID	"OD↵"
MQOD_STRUC_ID_ARRAY	'0', 'D', '↵', '↵'

**Not:** ↵ simgesi tek bir boş karakteri temsil eder.

Çizelge 246. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQOD_VERSION_1	1	X'00000001'
MQOD_VERSION_2	2	X'00000002'
MQOD_VERSION_3	3	X'00000003'
MQOD_VERSION_4	4	X'00000004'
MQOD_CURRENT_VERSION	4	X'00000004'
MQOD_CURRENT_LENGTH	(value differs by platform or version)	(value differs by platform or version)

### **MQOLI\_\* (Nesne Eşgörünümü Tanıtıcısı)**

Çizelge 247. Değişmez adlar ve değerler	
Ad	Değer
MQOL_NONE	X'00...00' (24 boş değer)
MQOII_NONE_DIZISI	'\0', '\0', ... (24 boş değer)

## MQOL\_\* (Özgün Uzunluk)

Çizelge 248. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQOL_TANIMSIZ	-1	X'FFFFFFFF'

## MQOM\_\* (Sorgu Grubuna ilişkin Kullanım Dışı Db2 İletileri seçenekleri)

Çizelge 249. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQOM_NO	0	X'00000000'
MQOM_YES	1	X'00000001'

## MQOO\_\* (Açma Seçenekleri)

Çizelge 250. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQOO_BIND_AS_Q_DEF	0	X'00000000'
MQOO_READ_AHEAD_AS_Q_DEF	0	X'00000000'
MQOO_INPUT_AS_Q_DEF	1	X'00000001'
MQOO_INPUT_SHARED	2	X'00000002'
MQOO_INPUT_EXCLUSIVE	4	X'00000004'
MQOO_BROWSE	8	X'00000008'
MQOO_OUTPUT	16	X'00000010'
MQOO_SORGULAMA	32	X'00000020'
MQOO_SET	64	X'00000040'
MQOO_SAVE_ALL_CONTEXT	128	X'00000080'
MQOO_PASS_IDENTITY_CONTEXT	256	X'00000100'
MQOO_PASS_ALL_CONTEXT	512	X'00000200'
MQOO_SET_IDENTITY_CONTEXT	1024	X'00000400'
MQOO_SET_ALL_CONTEXT	2048	X'00000800'
MQOO_ALTERNATE_USER_AUTHORITY	4096	X'00001000'
MQOO_FAIL_IF QUIESCING	8192	X'00002000'
MQOO_BIND_ON_Açık	16384	X'00004000'
MQOO_BIND_NOT_FIXED	32768	X'00008000'
MQOO_CO_OP	131072	X'00020000'
MQOO_RESOLVE_LOCAL_TOPIC	262144	X'00040000'
MQOO_NO_READ_AHEAD	524288	X'00080000'
MQOO_READ_AHEAD	1048576	X'00100000'
MQOO_BIND_ON_GROUP	4194304	X'00400000'

## MQOO\_\* (yalnızca C++ dilinde kullanılır)

Çizelge 251. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQOO_RESOLVE_NAMES	65536	X'00010000'
MQOO_RESOLVE_LOCAL_Q	262144	X'00040000'

## MQOP\_\* (MQCTL ve MQCB için işlem kodları)

### MQCTL ' ye ilişkin işlem kodları

Çizelge 252. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQOP_START	1	X'00000001'
MQOP_START_SABE	2	X'00000002'
MQOP_STOP	4	X'00000004'

### MQCB için işlem kodları

Çizelge 253. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQOP_KAYDOL	256	X'00000100'
MQOP_DEREGISTER	512	X'00000200'

### MQCTL ve MQCB ' ye ilişkin işlem kodları

Çizelge 254. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQOP_SUSPEND	65536	X'00010000'
MQOP_RESUME	131072	X'00020000'

## MQOPEN\_\* (MQOPEN\_PRIV yapısıyla ilgili değerler)

Çizelge 255. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQOPEN_PRIV_VERSION_1	1	X'00000001'
MQOPEN_PRIV_CURRENT_VERSION	1	X'00000001'

## MQOPER\_\* (Etkinlik İşlemleri)

Çizelge 256. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQOPER_SYSTEM_FIRST	0	X'00000000'
MQOPER_BILINMIYOR	0	X'00000000'
MQOPER_BROWSE	1	X'00000001'
MQOPER_AT	2	X'00000002'
MQOPER_GET	3	X'00000003'

Çizelge 256. Değişmezlerin değerleri (devamı var)		
Ad	Ondalık değer	Onaltılı değer
MQOPER_PUT	4	X'00000004'
MQOPER_PUT_REPLY	5	X'00000005'
MQOPER_PUT_REPORT	6	X'00000006'
MQOPER_RECEIVE	7	X'00000007'
MQOPER_SEND	8	X'00000008'
MQOPER_TRANSFORM	9	X'00000009'
MQOPER_YAYINLAMA	10	X'0000000A'
MQOPER_EXCLUDED_PUBLISH	11	X'0000000B'
MQOPER_DISCARDED_YAYINLAMA	12	X'0000000C'
MQOPER_SYSTEM_SON	65535	X'0000FFFF'
MQOPER_APPL_FIRST	65536	X'00010000'
MQOPER_APPL_SON	99999999	X'3B9AC9FF'

## MQOT\_\* (Nesne Tipleri ve Genişletilmiş Nesne Tipleri)

### Nesne Tipleri

Çizelge 257. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQOT_NONE	0	X'00000000'
MQOT_Q	1	X'00000001'
MQOT_NAMELIST	2	X'00000002'
MQOT_PROCESS	3	X'00000003'
MQOT_STORAGE_CLASS	4	X'00000004'
MQOT_Q_MGR	5	X'00000005'
MQOT_KANAL	6	X'00000006'
MQOT_AUTH_INFO	7	X'00000007'
MQOT_KONUSU	8	X'00000008'
MQOT_CF_STRUC	10	X'0000000A'
MQOT_LISTENER	11	X'0000000B'
MQOT_SERVICE	12	X'0000000C'
MQOT_RESERVED_1	999	X'000003E7'

### Genişletilmiş Nesne Tipleri

Çizelge 258. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQOT_ALL	1001	X'000003E9'
MQOT_ALIAS_Q	1002	X'000003EA'
MQOT_MODEL_Q	1003	X'000003EB'
MQOT_LOCAL_Q	1004	X'000003EC'
MQOT_REMOTE_Q	1005	X'000003ED'



Çizelge 258. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQOT_SENDER_CHANNEL	1007	X'000003EF'
MQOT_SERVER_CHANNEL	1008	X'000003F0'
MQOT_REQUESTER_CHANNEL	1009	X'000003F1'
MQOT_RECEIVER_CHANNEL	1010	X'000003F2'
MQOT_CURRENT_CHANNEL	1011	X'000003F3'
MQOT_SAVED_CHANNEL	1012	X'000003F4'
MQOT_SVRCONN_CHANNEL	1013	X'000003F5'
MQOT_CLNTCONN_CHANNEL	1014	X'000003F6'
MQOT_SHORT_CHANNEL	1015	X'000003F7'
MQOT_CHLAUTH	1016	X'000003F8'
MQOT_REMOTE_Q_MGR_NAME	1017	X'000003F9'
MQOT_PROT_POLICY	1019	X'000003FB'
MQOT_TT_KANAL	1020	X'000003FC'
MQOT_AMQP_KANAL	1021	X'000003FD'
MQOT_AUTH_REC	1022	X'000003FE'

## MQPA\_\* (Koyma Yetkisi)

Çizelge 259. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQPA_DEFAULT	1	X'00000001'
MQPA_CONTEXT	2	X'00000002'
MQPA_ONLY_MCA	3	X'00000003'
MQPA_ALTERNATE_OR_MCA	4	X'00000004'

## MQPD\_\* (Özellik tanımlayıcısı, destek ve bağlam)

### Özellik tanımlayıcı yapısı

Çizelge 260. Değişmezlerin yapıları

Ad	Yapı
MQPD_STRUC_ID	"PD~"
MQPD_STRUC_ID_ARRAY	'P', 'D', '~', '~'

**Not:** ~ simgesi tek bir boş karakteri temsil eder.

Çizelge 261. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQPD_VERSION_1	1	X'00000001'
MQPD_CURRENT_VERSION	1	X'00000001'

**Not:** ~ simgesi tek bir boş karakteri temsil eder.

## Özellik Tanımlayıcı Seçenekleri

Çizelge 262. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQPD_NONE	0	X'00000000'

## Özellik Desteği Seçenekleri

Çizelge 263. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQPD_SUPPORT_OPTIONAL	1	X'00000001'
MQPD_SUPPORT_REQUIREY	1048576	X'00100000'
MQPD_SUPPORT_REQUIRED_IF_LOCAL	1024	X'00000400'
MQPD_REJECT_UNSUP_MASK	-1048576	X'FFF00000'
MQPD_ACCEPT_UNSUP_IF_XMIT_MASK	1047552	X'000FFC00'
MQPD_ACCEPT_UNSUP_MASK	1023	X'000003FF'

## Özellik Bağlamı

Çizelge 264. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQPD_NO_CONTEXT	0	X'00000000'
MQPD_USER_CONTEXT	1	X'00000001'

## MQPER\_\* (Kalıcılık Değerleri)

Çizelge 265. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQPER_PERSISTENCE_AS_PARENT	-1	X'FFFFFFFF'
MQPER_NOT_PERSISTENT	0	X'00000000'
MQPER_PERISPER	1	X'00000001'
MQPER_PERSISTENCE_AS_Q_DEF	2	X'00000002'
MQPER_PERSISTENCE_AS_TOPIC_DEF	2	X'00000002'

## MQPL\_\* (Platformlar)

Çizelge 266. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQPL_MVS	1	X'00000001'
MQPL_OS390	1	X'00000001'
MQPL_ZOS	1	X'00000001'
MQPL_OS2	2	X'00000002'
MQPL_AIX	3	X'00000003'
MQPL_UNIX	3	X'00000003'
MQPL_OS400	4	X'00000004'
MQPL_WINDOWS	5	X'00000005'

Çizelge 266. Değişmezlerin değerleri (devamı var)		
Ad	Ondalık değer	Onaltılı değer
MQPL_WINDOWS_NT	11	X'0000000B'
MQPL_VMS	12	X'0000000C'
MQPL_NSK	13	X'0000000D'
MQPL_OPEN_TP1	15	X'0000000F'
MQPL_VM	18	X'00000012'
MQPL_TPF	23	X'00000017'
MQPL_VSE	27	X'0000001B'
MQP_APPLIANCE	28	X'0000001C'
MQP_NATIVE	1	X'00000001'

## MQPMO\_\* (Yayınlama maskesi için ileti seçenekleri ve yapısı kullanın)

### İleti seçenekleri yapısını koy

Çizelge 267. Değişmezlerin yapıları	
Ad	Yapı
MQPMO_STRUC_ID	"PMO~"
MQPMO_STRUC_ID_ARRAY	'P', 'M', 'O', '~'

**Not:** ~ simgesi tek bir boş karakteri temsil eder.

Çizelge 268. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQPMO_VERSION_1	1	X'00000001'
MQPMO_VERSION_2	2	X'00000002'
MQPMO_VERSION_3	3	X'00000003'
MQPMO_CURRENT_VERSION	3	X'00000003'
MQPMO_CURRENT_LENGTH	(value differs by platform or version)	(value differs by platform or version)

### İleti Koyma Seçenekleri

Çizelge 269. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQPMO_SYNCPOINT	2	X'00000002'
MQPMO_NO_SYNCPOINT	4	X'00000004'
MQPMO_DEFAULT_CONTEXT	32	X'00000020'
MQPMO_NEW_MSG_ID	64	X'00000040'
MQPMO_NEW_CORREL_ID	128	X'00000080'
MQPMO_PASS_IDENTITY_CONTEXT	256	X'00000100'
MQPMO_PASS_ALL_CONTEXT	512	X'00000200'
MQPMO_SET_IDENTITY_CONTEXT	1024	X'00000400'
MQPMO_SET_ALL_CONTEXT	2048	X'00000800'

Çizelge 269. Değişmezlerin değerleri (devamı var)		
Ad	Ondalık değer	Onaltılı değer
MQPMO_ALTERNATE_USER_AUTHORITY	4096	X'00001000'
MQPMO_FAIL_IF_QUIESCING	8192	X'00002000'
MQPMO_NO_BAĞLAMı	16384	X'00004000'
MQPMO_LOGICAL_ORDER	32768	X'00008000'
MQPMO_ASYNC_RESPONSE	65536	X'00010000'
MQPMO_SYNC_RESPONSE	131072	X'00020000'
MQPMO_RESOLVE_LOCAL_Q	262144	X'00040000'
MQPMO_TUT	2097152	X'00200000'
MQPMO_MD_FOR_OUTPUT_ONLY	8388608	X'00800000'
MQPMO_SCOP_QMGR	67108864	X'04000000'
MQPMO_SUPPRESS_REPLYTO	134217728	X'08000000'
MQPMO_NOT_OWN_SUBS	268435456	X'10000000'
MQPMO_RESPONSE_AS_Q_DEF	0	X'00000000'
MQPMO_RESPONSE_AS_TOPIC_DEF	0	X'00000000'
MQPMO_NONE	0	X'00000000'

### Yayınlama maskesine ilişkin İleti Seçenekleri

Çizelge 270. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQPMO_PUB_OPTIONS_MASK	2097152	X'00200000'

### MQPMRF\_\* (İleti Kaydı Alanları Koy)

Çizelge 271. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQPMRF_MSG_ID	1	X'00000001'
MQPMRF_CORREL_ID	2	X'00000002'
MQPMRF_GROUP_ID	4	X'00000004'
MQPMRF_FEEDBACK	8	X'00000008'
MQPMRF_ACCOUNTING_TOKEN	16	X'00000010'
MQPMRF_NONE	0	X'00000000'

### MQPO\_\* (Komut biçimi Temizleme Seçenekleri)

Çizelge 272. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQPO_YES	1	X'00000001'
MQPO_NO	0	X'00000000'

## MQPRI\_\* (Öncelik)

Çizelge 273. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQPRI_PRIORITY_AS_Q_DEF	-1	X'FFFFFFFF'
MQPRI_PRIORITY_AS_PARENT	-2	X'FFFFFFFE'
MQPRI_PRIORITY_AS_PUDID	-3	X'FFFFFFFD'
MQPRI_PRIORITY_AS_TOPIC_DEF	-1	X'FFFFFFFF'

## MQPROP\_\* (Kuyruk ve Kanal Özelliği Denetim Değerleri ve Maksimum Özellik Uzunluğu)

### Kuyruk ve Kanal Özellik Denetimi Değerleri

Çizelge 274. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQPROP_UYUMLULUK	0	X'00000000'
MQPROP_NONE	1	X'00000001'
MQPROP_ALL	2	X'00000002'
MQPROP_FORCE_MQRFH2	3	X'00000003'

### Özellik Uzunluğu Üst Sınırı

Çizelge 275. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQPROP_UNRESTRICTED_LENGTH	-1	X'FFFFFFFF'

## MQPRT\_\* (Yanıt Değerlerini Koyma)

Çizelge 276. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQPRT_RESPONSE_AS_PARENT	0	X'00000000'
MQPRT_SYNC_RESPONSE	1	X'00000001'
MQPRT_ASYNC_RESPONSE	2	X'00000002'

## MQPS\_\* (Yayınlama/Abone Olma)

### Komut biçimi Yayınlama/Abone Olma Durumu

Çizelge 277. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQPS_STATUS_DEVREDİŞİ	0	X'00000000'
MQPS_STATUS_STARTING	1	X'00000001'
MQPS_STATUS_DURMA	2	X'00000002'
MQPS_STATUS_ETKİN	3	X'00000003'
MQPS_STATUS_COMPAT	4	X'00000004'
MQPS_STATUS_ERROR	5	X'00000005'

Çizelge 277. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQPS_STATUS_REFRED	6	X'00000006'

### Etiketleri dize olarak Yayınla/Abone Ol

MQPS_COMMAND	"MQPSCommand"
MQPS_COMP_CODE	"MQPSCompCode"
MQPS_COREL_ID	"MQPSCorrelId"
MQPS_DELEE	"MQPSDe10pts"
MQPS_ERROR_ID	"MQPSErrorId"
MQPS_ERROR_POS	"MQPSErrorPos"
MQPS_INTEGER_DATA	"MQPSIntData"
MQPS_PARAMETER_ID	"MQSParmId"
MQPS_YAYINLAMA SEÇENEKLERİ	"MQSPubOpts"
MQPS_PUBLISH_TIMESTAMP	"MQSPubTime"
MQPS_Q_MGR_NAME	"MQPSQMgrName"
MQPS_Q_ADI	"MQPSQName"
MQPS_NEDENİ	"MQPSReason"
MQPS_NEDEN_METNİ	"MQPSReasonText"
MQPS_REGISTRATION_OPTIONS	"MQPSRegOpts"
MQPS_SEQUENCE_NUMBER	"MQPSSeqNum"
MQPS_STREAM_NAME	"MQPSStreamName"
MQPS_STRING_DATA	"MQPSStringData"
MQPS_SUBSCRIPTION_IDENTITY	"MQPSSubIdentity"
MQPS_SUBSCRIPTION_NAME	"MQPSSubName"
MQPS_XX_ENCODE_CASE_ONE subscription_user_data	"MQPSSubUserData"
MQPS_KONU	"MQPSTopic"
MQPS_USER_ID	"MQPSUserId"

### Etiketleri boş olarak içeren dizgiler olarak Yayınla/Abone Ol

MQPS_COMMAND_B	"-MQPSCommand-"
MQPS_COMP_CODE_B	"-MQPSCompCode-"
MQPS_CORREL_ID_B	"-MQPSCorrelId-"
MQPS_DELETE_OPTIONS_B	"-MQPSDe10pts-"
MQPS_ERROR_ID_B	"-MQPSErrorId-"
MQPS_ERROR_POS_B	"-MQPSErrorPos-"
MQPS_INTEGER_DATA_B	"-MQPSIntData-"

MQPS_PARAMETER_ID_B	"~MQPSParamId~"
MQPS_PUBLICATION_OPTIONS_B	"~MQPSPubOpts~"
MQPS_PUBLISH_TIMESTAMP_B	"~MQPSPubTime~"
MQPS_Q_MGR_NAME_B	"~MQPSQMgrName~"
MQPS_Q_NAME_B	"~MQPSQName~"
MQPS_NEDEN_B	"~MQPSReason~"
MQPS_REASON_TEXT_B	"~MQPSReasonText~"
MQPS_REGISTRATION_OPTIONS_B	"~MQPSRegOpts~"
MQPS_SEQUENCE_NUMBER_B	"~MQPSSeqNum~"
MQPS_STREAM_NAME_B	"~MQPSStreamName~"
MQPS_STRING_DATA_B	"~MQPSStringData~"
MQPS_SUBSCRIPTION_IDENTITY_B	"~MQPSSubIdentity~"
MQPS_XX_ENCODE_CASE_ONE subscription_name_b	"~MQPSSubName~"
MQPS_XX_ENCODE_CASE_ONE subscription_user_data_b	"~MQPSSubUserData~"
MQPS_TOPIC_B	"~MQPSTopic~"
MQPS_USER_ID_B	"~MQPSUserId~"

**Not:** ~ simgesi tek bir boş karakteri temsil eder.

### Komut Etiket Değerlerini dizgi olarak Yayınla/Abone Ol

MQPS_DELETE_YAYINI	"DeletePub"
MQPS_DEREGISTER_PUBLISHER	"DeregPub"
MQPS_DEREGISTER_SUBSCRIBER	"DeregSub"
MQPS_YAYINLAMA	"Publish"
MQPS_REGISTER_PUBLISHER	"RegPub"
MQPS_REGISTER_ABONESI	"RegSub"
MQPS_REQUEST_UPDATE	"ReqUpdate"

### Yayınla/Abone Olma Komutu Etiket Değerleri boşluğa alınmış dizgiler olarak

MQPS_DELETE_PUBLICATION_B	"~DeletePub~"
MQPS_DEREGISTER_YAYINHER_B	"~DeregPub~"
MQPS_DEREGISTER_SUBSCRIBER_B	"~DeregSub~"
MQPS_YAYINLAYICI_B	"~Publish~"
MQPS_REGISTER_PUBLISHER_B	"~RegPub~"
MQPS_REGISTER_SUBSCRIBER_B	"~RegSub~"
MQPS_REQUEST_UPDATE_B	"~ReqUpdate~"

**Not:** ~ simgesi tek bir boş karakteri temsil eder.

### Yayınlama/Abone Olma Seçenekleri Etiket Değerleri dizgi olarak

MQPS_ADD_NAME	"AddName"
MQPS_ANONIM	"Anon"
MQPS_CORREL_ID_AS_IDENTITY	"CorrelAsId"
MQPS_DEREGISTER_ALL	"DeregAll"
MQPS_DIRECT_REQUIREMS	"DirectReq"
MQPS_DUPLICATES_OK	"DupsOK"
MQPS_FULL_RESPONSE	"FullResp"
MQPS_INCLUDE_STREAM_NAME	"InclStreamName"
MQPS_INFORMA_IF_ALIKOYMA	"InformIfRet"
MQPS_IS_RETAINED_YAYIN	"IsRetainedPub"
MQPS_JOIN_EXCLUSIVE	"JoinExcl"
MQPS_JOIN_SHARED	"JoinShared"
MQPS_LEAVE_ONLY	"LeaveOnly"
MQPS_LOCAL	"Local"
MQPS_KILITLI	"Locked"
MQPS_NEW_YAYINLAYICI	"NewPubsOnly"
MQPS_NO_ALTERATION	"NoAlter"
MQPS_NO_REGISTRATION	"NoReg"
MQPS_NON_PERDEAL	"NonPers"
MQPS_NONE	"None"
MQPS_OTHER_SUBSCRIBERS_ONLY	"OtherSubsOnly"
MQPS_PERSSI	"Pers"
MQPS_PERSISTENT_AS_YAYINLAMA	"PersAsPub"
MQPS_PERSISTENT_AS_Q	"PersAsQueue"
MQPS_YAYINLAMA_YALNIZ_ISTEĞI	"PubOnReqOnly"
MQPS_RETAIN_YAYINLARI	"RetainPub"
MQPS_VARIABLE_USER_ID	"VariableUserId"

### Yayınlama/Abone Olma Seçenekleri Etiket Değerleri boşluğa alınmış dizgiler olarak

MQPS_ADD_NAME_B	"-AddName-"
MQPS_ANONYMOUS_B	"-Anon-"
MQPS_CORREL_ID_AS_IDENTITY_B	"-CorrelAsId-"
MQPS_DEREGISTIER_ALL_B	"-DeregAll-"
MQPS_DIRECT_REQUESTS_B	"-DirectReq-"
MQPS_DUPLICAES_OK_B	"-DupsOK-"



MQPS_FULL_RESPONSE_B	"¬FullResp¬"
MQPS_INCLUDE_STREAM_NAME_B	"¬InclStreamName¬"
MQPS_BİLGELER_IF_RETAINED_B	"¬InformIfRet¬"
MQPS_IS_RETAINED_PUBLICATION_B	"¬IsRetainedPub¬"
MQPS_JOIN_SÖZLÜZ_B	"¬JoinExcl¬"
MQPS_JOIN_SHARED_B	"¬JoinShared¬"
MQPS_LEAVE_ONLY_B	"¬LeaveOnly¬"
MQPS_LOCAL_B	"¬Local¬"
MQPS_LOCKED_B	"¬Locked¬"
MQPS_NEW_YAYINLAMACI_ONLY_B	"¬NewPubsOnly¬"
MQPS_NO_ALTERATION_B	"¬NoAlter¬"
MQPS_NO_REGISTRATION_B	"¬NoReg¬"
MQPS_NON_PERSISTENT_B	"¬NonPers¬"
MQPS_NONE_B	"¬None¬"
MQPS_OTHER_SUBSCRIBERS_ONLY_B	"¬OtherSubsOnly¬"
MQPS_PERSISTENT_B	"¬Pers¬"
MQPS_PERSISTENT_AS_YAYINLAYICI_B	"¬PersAsPub¬"
MQPS_PERSISTENT_AS_Q_B	"¬PersAsQueue¬"
MQPS_YAYINLAYICI_BAŞLIK_BAŞLIYOR	"¬PubOnReqOnly¬"
MQPS_RETAIN_PUBLICATION_B	"¬RetainPub¬"
MQPS_VARIABLE_USER_ID_B	"¬VariableUserId¬"

**Not:** ¬ simgesi tek bir boş karakteri temsil eder.

### **MQPSC\_\* (Yayınlama/Abone Olma Seçenekleri Etiketleri Yayınlama/Abone Olma Komut Klasörü (psc) Etiketleri)**

<i>Çizelge 278. Değişmezlerin değerleri</i>		
Ad	Ondalık değer	Onaltılı değer
MQPSC_FOLDER_VERSION	1	X'00000001'

### **MQPSC\_\* (Yayınlama/Abone Olma Seçenekleri Etiket Etiketleri adları)**

MQPSC_COMMAND	"Command"
MQPSC_REGISTRATION_OPTION	"RegOpt"
MQPS_PUBLICATION_OPTION	"PubOpt"
MQPSC_DELETE_OPTION	"DelOpt"
MQPSC_KONUSU	"Topic"
MQPSC_SUBSCRIPTION_POINT	"SubPoint"
MQPSC_SÜZGEÇ	"Filter"

MQPSC_Q_MGR_NAME	"QMgrName"
MQPSC_Q_ADı	"QName"
MQPSC_PUBLISH_TIMESTAMP	"PubTime"
MQPSC_SEQUENCE_NUMBER	"SeqNum"
MQPSC_SUBSCRIPTION_NAME	"SubName"
MQPSC_SUBSCRIPTION_IDENTITY	"SubIdentity"
MQPSC_XX_ENCODE_CASE_ONE subscription_user_data	"SubUserData"
MQPSC_COREL_ID	"CorrelId"

**MQPSC\_ \* (Yayınlama/Abone Olma Seçenekleri Etiketleri XML etiketleri adları)**

MQPSC_COMMAND_B	"<Command>"
MQPSC_COMMAND_E	"</Command>"
MQPSC_REGISTRATION_OPTION_B	"<RegOpt>"
MQPSC_REGISTRATION_OPTION_E	"</RegOpt>"
MQPSC_PUBLICATION_OPTION_B	"<PubOpt>"
MQPSC_PUBLICATION_OPTION_E	"</PubOpt>"
MQPSC_DELETE_OPTION_B	"<DelOpt>"
MQPSC_DELETE_OPTION_E	"</DelOpt>"
MQPSC_TOPIC_B	"<Topic>"
MQPSC_TOPIC_E	"</Topic>"
MQPSC_SUBSCRIPTION_POINT_B	"<SubPoint>"
MQPSC_SUBSCRIPTION_POINT_E	"</SubPoint>"
MQPSC_SÜZGEÇ_B	"<Filter>"
MQPSC_SÜZGELER_E	"</Filter>"
MQPSC_Q_MGR_NAME_B	"<QMgrName>"
MQPSC_Q_MGR_NAME_E	"</QMgrName>"
MQPSC_Q_NAME_B	"<QName>"
MQPSC_Q_NAME_E	"</QName>"
MQPSC_PUBLISH_TIMESTAMP_B	"<PubTime>"
MQPSC_PUBLISH_TIMESTAMP_E	"</PubTime>"
MQPSC_SEQUENCE_NUMBER_B	"<SeqNum>"
MQPSC_SEQUENCE_NUMBER_E	"</SeqNum>"
MQPSC_SUBSCRIPTION_NAME_B	"<SubName>"
MQPSC_SUBSCRIPTION_NAME_E	"</SubName>"
MQPSC_SUBSCRIPTION_IDENTITY_B	"<SubIdentity>"
MQPSC_SUBSCRIPTION_IDENTITY_E	"</SubIdentity>"
MQPSC_SUBSCRIPTION_USER_DATA_B	"<SubUserData>"

MQPSC_XX_ENCODE_CASE_ONE subscription_user_data_e	"</SubUserData>"
MQPSC_CORREL_ID_B	"<CorrelId>"
MQPSC_CORREL_ID_E	"</CorrelId>"

**MQPSC\_\* (Yayınlama/Abone Olma Seçenekleri Etiket Yayınlayıcısı Değerleri dizgi olarak)**

MQPSC_DELETE_YAYIN	"DeletePub"
MQPSC_DEREGISTER_SUBSCRIBER	"DeregSub"
MQPSC_YAYINLAMA	"Publish"
MQPSC_REGISTER_SUBSCRIBER	"RegSub"
MQPSC_REQUEST_UPDATE	"ReqUpdate"

**MQPSC\_\* (Yayınlama/Abone Olma Seçenekleri Etiket Adı Değerleri dizgiler olarak)**

MQPSC_ADD_NAME	"AddName"
MQPSC_CORREL_ID_AS_IDENTITY	"CorrelAsId"
MQPSC_DEREGISTER_ALL	"DeregAll"
MQPSC_DUPLICATES_OK	"DupsOK"
MQPSC_FULL_RESPONSE	"FullResp"
MQPSC_INFORM_IF_ALIKOYMA	"InformIfRet"
MQPSC_IS_RETAINED_PUB	"IsRetainedPub"
MQPSC_JOIN_SHARED	"JoinShared"
MQPSC_JOIN_EXCLUSIVE	"JoinExcl"
MQPSC_LEAVE_ONLY	"LeaveOnly"
MQPSC_LOCAL	"Local"
MQPSC_LOCKED	"Locked"
MQPSC_NEW_PUBS_ONLY	"NewPubsOnly"
MQPSC_NO_ALTERATION	"NoAlter"
MQPSC_NON_PERSISTENT	"NonPers"
MQPSC_OTHER_SUBS_ONLY	"OtherSubsOnly"
MQPSC_PERSISTENT	"Pers"
MQPSC_PERSISTENT_AS_YAYINLAMA	"PersAsPub"
MQPSC_PERSISTENT_AS_Q	"PersAsQueue"
MQPSC_NONE	"None"
MQPSC_PUB_ON_REQUEST_ONLY	"PubOnReqOnly"
MQPSC_RETAIN_PUB	"RetainPub"
MQPSC_VARIABLE_USER_ID	"VariableUserId"

## MQPSCR\_\* (Yayınlama/Abone Olma Seçenekleri)

### Yayınlama/Abone Olma Seçenekleri Etiket Yayınlama/Abone Olma Yanıt Klasörü (pscr) Etiketleri

Çizelge 279. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQPSCR_FOLDER_VERSION	1	X'00000001'

### Yayınlama/Abone Olma Seçenekleri Etiket Etiketleri adları

MQPSCR_COMPLETION	"Completion"
MQPSCR_RESPONSE	"Response"
MQPSCR_REASON	"Reason"

### Yayınlama/Abone Olma Seçenekleri Etiket XML etiketi adları

MQPSCR_COMPLETION_B	"<Completion>"
MQPSCR_COMPLETION_E	"</Completion>"
MQPSCR_RESPONSE_B	"<Response>"
MQPSCR_RESPONSE_E	"</Response>"
MQPSCR_REASON_B	"<Reason>"
MQPSCR_REASON_E	"</Reason>"

### Yayınlama/Abone Olma Seçenekleri Etiket Etiketleri değerleri

MQPSCR_OK	"ok"
MQPSCR_UYARI	"warning"
MQPSCR_HATASI	"error"

## MQPSM\_\* (Pub/Alt Kipi)

Çizelge 280. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQPSM_DISABLE	0	X'00000000'
MQPSM_COMPAT	1	X'00000001'
MQPSMENABLED	2	X'00000002'

## MQPSPROP\_\* (Pub/Alt İleti Özellikleri)

Çizelge 281. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQPSPROP_NONE	0	X'00000000'
MQPSPROP_COMPAT	1	X'00000001'
MQPSPROP_RFH2	2	X'00000002'
MQPSPROP_MSGPROP	3	X'00000003'

## MQPSST\_\* (Komut biçimi Pub/Alt Durum Tipi)

Çizelge 282. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQPSST_ALL	0	X'00000000'
MQPSST_LOCAL	1	X'00000001'
MQPSST_PARENT	2	X'00000002'
MQPSST_COÇUK	3	X'00000003'

## MQPUBO\_\* (Yayınlama/Abone Olma Yayın Seçenekleri)

Çizelge 283. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQPUBO_NONE	0	X'00000000'
MQPUBO_CORREL_ID_AS_IDENTITY	1	X'00000001'
MQPUBO_RETAIN_YAYINI	2	X'00000002'
MQPUBO_OTHER_SUBSCRIBERS_ONLY	4	X'00000004'
MQPUBO_NO_REGISTRATION	8	X'00000008'
MQPUBO_IS_RETAINED_YAYINLARI	16	X'00000010'

## MQXPX\_\* (Yayınlama/abone olma yöneltme çıkış değiştirgesi yapısı)

Çizelge 284. Değişmezlerin yapıları	
Ad	Yapı
MQXPX_STRUC_ID	"PXP↯"
MQXPX_STRUC_ID_ARRAY	'P', 'X', 'P', '↯'

**Not:** ↯ simgesi tek bir boş karakteri temsil eder.

Çizelge 285. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQXPX_VERSION_1	1	X'00000001'
MQXPX_XX_ENCODE_CASE_ONE current_version	1	X'00000001'

## MQQA\_\* (Kuyruk öznitelikleri)

### Alma Değerlerini Engelle

Çizelge 286. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQQA_GET_INHIBITED	1	X'00000001'
MQQA_GET_ALLOWD	0	X'00000000'

### Koyma Değerlerini Engelle

Çizelge 287. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQQA_PUT_INHIBITED	1	X'00000001'

Çizelge 287. Değişmezlerin değerleri (devamı var)		
Ad	Ondalık değer	Onaltılı değer
MQQA_PUT_ALLOWED	0	X'00000000'

### Kuyruk Paylaşılabilirliği

Çizelge 288. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQQA_SHAREABLE	1	X'00000001'
MQQA_NOT_SHAREABLE	0	X'00000000'

### Geri Dönüş Sertleşmesi

Çizelge 289. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQQA_BACKUT_HARDENED	1	X'00000001'
MQQA_BACKUT_NOT_HARDENED	0	X'00000000'

### MQQDT\_\* (Kuyruk Tanımı Tipleri)

Çizelge 290. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQQDT_ÖNCEDEN tanımlı	1	X'00000001'
MQQDT_PERMANENT_DYNAMIC	2	X'00000002'
MQQDT_TEMPORARY_DYNAMIC	3	X'00000003'
MQQDT_SHARED_DYNAMIC	4	X'00000004'

### MQQF\_\* (Kuyruk İşaretleri)

Çizelge 291. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQQF_LOCAL_Q	1	X'00000001'
MQQF_CLWL_USEQ_ANY	64	X'00000040'
MQQF_CLWL_USEQ_LOCAL	128	X'00000080'

### MQQMDT\_\* (Komut biçimi Kuyruk Yöneticisi Tanımlaması Tipleri)

Çizelge 292. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQQMDT_REGIDENIT_CLUSTER_SENDER	1	X'00000001'
MQQMDT_AUTO_CLUSTER_SENDER	2	X'00000002'
MQQMDT_AUTO_EXP_CLUSTER_SENDER	4	X'00000004'
MQQMDT_CLUSTER_RECEIVER	3	X'00000003'

## MQQMF\_\* (Kuyruk Yöneticisi İşaretleri)

Çizelge 293. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQQMF_REPOSITORY_Q_MGR	2	X'00000002'
MQQMF_CLUSSDR_USER_DEFINED	8	X'00000008'
MQQMF_CLUSSDR_AUTO_DEFINED	16	X'00000010'
MQQMF_VAR	32	X'00000020'

## MQQMFAC\_\* (Komut biçimi Kuyruk Yöneticisi Olanığı)

Çizelge 294. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQQMFAT_IMS_BRIDGE	1	X'00000001'
MQQMFAC_DB2	2	X'00000002'

## MQQMSTA\_\* (Komut biçimi Kuyruk Yöneticisi Durumu)

Çizelge 295. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQQMSTA_BAŞLATMA	1	X'00000001'
MQQMSTA_RUNING	2	X'00000002'
MQQMSTA QUIESCING	3	X'00000003'

## MQQMT\_\* (Komut biçimi Kuyruk Yöneticisi Tipleri)

Çizelge 296. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQQMT_NORMAL	0	X'00000000'
MQQMT_REPOSITORY	1	X'00000001'

## MQQO\_\* (Komut biçimi Quiesce Seçenekleri)

Çizelge 297. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQQO_YES	1	X'00000001'
MQQO_NO	0	X'00000000'

## MQQSGD\_\* (Kuyruk paylaşımı grubu atımları)

Çizelge 298. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQQSGD_ALL	-1	X'FFFFFFFF'
MQQSGD_Q_MGR	0	X'00000000'
MQQSGD_COPY	1	X'00000001'
MQQSGD_SHARED	2	X'00000002'
MQQSGD_GROUP	3	X'00000003'

Çizelge 298. Değişmezlerin değerleri (devamı var)		
Ad	Ondalık değer	Onaltılı değer
MQQSGD_PRIVATE	4	X'00000004'
MQQSGD_LIVE	6	X'00000006'

### MQQSGS\_\* (Komut biçimi kuyruğu paylaşım grubu durumu)

Çizelge 299. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQQSGS_UNKNOWN	0	X'00000000'
MQQSGS_CREATED	1	X'00000001'
MQQSGS_ETKİN	2	X'00000002'
MQQSGS_INETKİN	3	X'00000003'
MQQSGS_FAILED	4	X'00000004'
MQQSGS_PENDING	5	X'00000005'

### MQQSI\_\* (Komut biçimi Kuyruk Hizmeti-Aralık Olayları)

Çizelge 300. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQQSI_NONE	0	X'00000000'
MQQSI_YükSEK	1	X'00000001'
MQQSI_OK	2	X'00000002'

### MQQSO\_\* (SET, BROWSE, INPUT için Komut biçimi Kuyruk Durumu Açma Seçenekleri)

Çizelge 301. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQQSO_NO	0	X'00000000'
MQQSO_YES	1	X'00000001'
MQQSO_SHARED	1	X'00000001'
MQQSO_EXCLUSIVE	2	X'00000002'

### MQQSOT\_\* (Komut biçimi Kuyruk Durumu Açık Tipleri)

Çizelge 302. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQQSOT_ALL	1	X'00000001'
MQQSOT_INPUT	2	X'00000002'
MQQSOT_OUTPUT	3	X'00000003'

### MQQSUM\_\* (Komut biçimi Kuyruk Durumu Kesinleştirilmemiş İletiler)

Çizelge 303. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQQSUM_YES	1	X'00000001'



Çizelge 303. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQCSUM_NO	0	X'00000000'

## MQQT\_\* (Kuyruk Tipleri ve Genişletilmiş Kuyruk Tipleri)

### Kuyruk Tipleri

Çizelge 304. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQQT_LOCAL	1	X'00000001'
MQQT_MODEL	2	X'00000002'
MQQT_ALIAS	3	X'00000003'
MQQT_REMOTE	6	X'00000006'
MQQT_CLUSTER	7	X'00000007'

### Genişletilmiş Kuyruk Tipleri

Çizelge 305. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQQT_ALL	1001	X'000003E9'

## MQRC\_\* (neden kodları)

Çizelge 306. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQRC_NONE	0	X'00000000'
MQRC_APPL_FIRST	900	X'00000384'
MQRC_APPL_SON	999	X'000003E7'
MQRC_ALIAS_BASE_Q_TYPE_ERROR	2001	X'000007D1'
MQRC_ALREADY_CONNECTED	2002	X'000007D2'
MQRC_BACKED_OUT	2003	X'000007D3'
MQRC_BUFFER_ERROR	2004	X'000007D4'
MQRC_BUFFER_LENGTH_ERROR	2005	X'000007D5'
MQRC_CHAR_ATTR_LENGTH_ERROR	2006	X'000007D6'
MQRC_CHAR_ATTRS_ERROR	2007	X'000007D7'
MQRC_CHAR_ATTRS_TOO_SHORT	2008	X'000007D8'
MQRC_CONNECTION_BROKEN	2009	X'000007D9'
MQRC_DATA_LENGTH_ERROR	2010	X'000007DA'
MQRC_DYNAMIC_Q_NAME_ERROR	2011	X'000007DB'
MQRC_ENVIRONMENT_ERROR	2012	X'000007DC'
MQRC_EXPIRY_ERROR	2013	X'000007DD'
MQRC_FEEDBACK_ERROR	2014	X'000007DE'
MQRC_GET_INHIBITED	2016	X'000007E0'
MQRC_HANDLE_NOT_VAR	2017	X'000007E1'

Çizelge 306. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQRC_HCONN_ERROR	2018	X'000007E2'
MQRC_HOBJ_ERROR	2019	X'000007E3'
MQRC_INHIBIT_VALUE_ERROR	2020	X'000007E4'
MQRC_INT_ATTR_COUNT_ERROR	2021	X'000007E5'
MQRC_INT_ATTR_COUNT_TOO_SMALL	2022	X'000007E6'
MQRC_INT_ATTRS_ARRAY_ERROR	2023	X'000007E7'
MQRC_SYNCPOINT_LIMIT_UVARD	2024	X'000007E8'
MQRC_MAX_CONNS_IMLim_REACHED	2025	X'000007E9'
MQRC_MD_ERROR	2026	X'000007EA'
MQRC_MISSING_REPLY_TO_Q	2027	X'000007EB'
MQRC_MSG_TYPE_ERROR	2029	X'000007ED'
MQRC_MSG_TOO_BIG_FOR_Q	2030	X'000007EE'
MQRC_MSG_TOO_BIG_FOR_Q_MGR	2031	X'000007EF'
MQRC_NO_MSG_AVAILABLE	2033	X'000007F1'
MQRC_NO_MSG_UNDER_CURSOR	2034	X'000007F2'
MQRC_NOT_YETKILI	2035	X'000007F3'
MQRC_NOT_OPEN_FOR_BROWSE	2036	X'000007F4'
MQRC_NOT_OPEN_FOR_INPUT	2037	X'000007F5'
MQRC_NOT_OPEN_FOR_SORGULAMA	2038	X'000007F6'
MQRC_NOT_OPEN_FOR_OUTPUT	2039	X'000007F7'
MQRC_NOT_OPEN_FOR_SET	2040	X'000007F8'
MQRC_OBJECT_CHANGED	2041	X'000007F9'
MQRC_OBJECT_IN_USE	2042	X'000007FA'
MQRC_OBJECT_TYPE_ERROR	2043	X'000007FB'
MQRC_OD_ERROR	2044	X'000007FC'
MQRC_OPTION_NOT_VALID_FOR_TYPE	2045	X'000007FD'
MQRC_OPTIONS_ERROR	2046	X'000007FE'
MQRC_PERSISTENCE_ERROR	2047	X'000007FF'
MQRC_PERSISTENT_NOT_ALLOWED	2048	X'00000800'
MQRC_PRIORITY_EXCEEDS_MAXIMUM	2049	X'00000801'
MQRC_PRIORITY_ERROR	2050	X'00000802'
MQRC_PUT_INHIBITED	2051	X'00000803'
MQRC_Q_DELETED	2052	X'00000804'
MQRC_Q_FULL	2053	X'00000805'
MQRC_Q_NOT_EMPTY	2055	X'00000807'
MQRC_Q_SPACE_NOT_VAR	2056	X'00000808'
MQRC_Q_TYPE_ERROR	2057	X'00000809'
MQRC_Q_MGR_NAME_ERROR	2058	X'0000080A'
MQRC_Q_MGR_NOT_VAR	2059	X'0000080B'

Çizelge 306. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQRC_REPORT_OPTIONS_ERROR	2061	X'0000080D'
MQRC_SECOND_MARK_NOT_ALLOWED	2062	X'0000080E'
MQRC_SECURITY_ERROR	2063	X'0000080F'
MQRC_SELECTOR_COUNT_ERROR	2065	X'00000811'
MQRC_SELECTOR_LIMIT_EXACID	2066	X'00000812'
MQRC_SELECTOR_ERROR	2067	X'00000813'
MQRC_SELECTOR_NOT_FOR_TYPE	2068	X'00000814'
MQRC_SIGNAL_INSTANT	2069	X'00000815'
MQRC_SIGNAL_REQUEST_ACCEPTED	2070	X'00000816'
MQRC_STORAGE_NOT_AVAM	2071	X'00000817'
MQRC_SYNCPOINT_NOT_AVAM	2072	X'00000818'
MQRC_TRIGGER_CONTROL_ERROR	2075	X'0000081B'
MQRC_TRIGGER_DEPTH_ERROR	2076	X'0000081C'
MQRC_TRIGGER_MSG_PRIORITY_ERR	2077	X'0000081D'
MQRC_TRIGGER_TYPE_ERROR	2078	X'0000081E'
MQRC_TRUNCATED_MSG_KABUL edildi	2079	X'0000081F'
MQRC_TRUNCATED_MSG_FAILED	2080	X'00000820'
MQRC_UNKNOWN_ALIAS_BASE_Q	2082	X'00000822'
MQRC_UNKNOWN_OBJECT_NAME	2085	X'00000825'
MQRC_UNKNOWN_OBJECT_Q_MGR	2086	X'00000826'
MQRC_UNKNOWN_REMOTE_Q_MGR	2087	X'00000827'
MQRC_WAIT_INTERVAL_ERROR	2090	X'0000082A'
MQRC_XMIT_Q_TYPE_ERROR	2091	X'0000082B'
MQRC_XMIT_Q_USAGE_ERROR	2092	X'0000082C'
MQRC_NOT_OPEN_FOR_PASS_ALL	2093	X'0000082D'
MQRC_NOT_OPEN_FOR_PASS_IDENT	2094	X'0000082E'
MQRC_NOT_OPEN_FOR_SET_ALL	2095	X'0000082F'
MQRC_NOT_OPEN_FOR_SET_IDENT	2096	X'00000830'
MQRC_CONTEXT_HANDLE_ERROR	2097	X'00000831'
MQRC_CONTEXT_NOT_AVAILABLE	2098	X'00000832'
MQRC_SIGNAL1_ERROR	2099	X'00000833'
MQRC_OBJECT_ALREADY_EXISTS	2100	X'00000834'
MQRC_OBJECT_ZARAR	2101	X'00000835'
MQRC_RESOURCE_SORUNU	2102	X'00000836'
MQRC_ANOTHER_Q_MGR_CONNECTED	2103	X'00000837'
MQRC_UNKNOWN_REPORT_OPTION	2104	X'00000838'
MQRC_STORAGE_CLASS_ERROR	2105	X'00000839'
MQRC_COD_NOT_VALID_FOR_XCF_Q	2106	X'0000083A'
MQRC_XWAIT İPTAL EDİLDİ	2107	X'0000083B'

Çizelge 306. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQRC_XWAIT_ERROR	2108	X'0000083C'
MQRC_SUPPRESSED_BY_EXIT	2109	X'0000083D'
MQRC_FORMAT_ERROR	2110	X'0000083E'
MQRC_SOURCE_CCSDID_ERROR	2111	X'0000083F'
MQRC_SOURCE_INTEGER_ENC_ERROR	2112	X'00000840'
MQRC_SOURCE_DECIMAL_ENC_ERROR	2113	X'00000841'
MQRC_SOURCE_FLOATENC_ERROR	2114	X'00000842'
MQRC_TARGET_CCSDID_HATASI	2115	X'00000843'
MQRC_TARGET_INTEGER_ENC_ERROR	2116	X'00000844'
MQRC_TARGET_DECIMALIM_ERROR	2117	X'00000845'
MQRC_TARGET_FLOATENC_ERROR	2118	X'00000846'
MQRC_NOT_CONVERTED	2119	X'00000847'
MQRC_CONVERTED_MSG_TOO_BÜYÜK	2120	X'00000848'
MQRC_TRUNCATED	2120	X'00000848'
MQRC_NO_EXTERNAL_KATILIMCILARI	2121	X'00000849'
MQRC_PARTICIPANT_NOT_VAR	2122	X'0000084A'
MQRC_OUTCOME_MIXED	2123	X'0000084B'
MQRC_OUTCOME_PENDING	2124	X'0000084C'
MQRC_BRIDGE_STARTED	2125	X'0000084D'
MQRC_BRIDGE_DURDU	2126	X'0000084E'
MQRC_ADAPTER_STORAGE_EKSIKLİK	2127	X'0000084F'
MQRC_UOW_IN_PROGRESS	2128	X'00000850'
MQRC_ADAPTER_CONN_LOAD_ERROR	2129	X'00000851'
MQRC_ADAPTER_SERV_LOAD_ERROR	2130	X'00000852'
MQRC_ADAPTER_DEFS_ERROR	2131	X'00000853'
MQRC_ADAPTER_DEFS_LOAD_ERROR	2132	X'00000854'
MQRC_ADAPTER_CONV_LOAD_ERROR	2133	X'00000855'
MQRC_BO_HATASI	2134	X'00000856'
MQRC_DH_HATASI	2135	X'00000857'
MQRC_MULTIPLE_REASONS	2136	X'00000858'
MQRC_OPEN_FAILED	2137	X'00000859'
MQRC_ADAPTER_DISC_LOAD_ERROR	2138	X'0000085A'
MQRC_CNO_HATASI	2139	X'0000085B'
MQRC_CICS_WAIT_FAILED	2140	X'0000085C'
MQRC_DLH_ERROR	2141	X'0000085D'
MQRC_HEADER_ERROR	2142	X'0000085E'
MQRC_SOURCE_LENGTH_ERROR	2143	X'0000085F'
MQRC_TARGET_LENGTH_ERROR	2144	X'00000860'
MQRC_SOURCE_BUFFER_ERROR	2145	X'00000861'

Çizelge 306. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQRC_TARGET_BUFF_ERROR	2146	X'00000862'
MQRC_IIH_ERROR	2148	X'00000864'
MQRC_PCF_ERROR	2149	X'00000865'
MQRC_DBCS_HATASI	2150	X'00000866'
MQRC_OBJECT_NAME_ERROR	2152	X'00000868'
MQRC_OBJECT_Q_MGR_NAME_ERROR	2153	X'00000869'
MQRC_RECS_PRESENT_ERROR	2154	X'0000086A'
MQRC_OBJECT_RECORDS_ERROR	2155	X'0000086B'
MQRC_RESPONSE_RECORDS_ERROR	2156	X'0000086C'
MQRC_ASID_MISSMACH	2157	X'0000086D'
MQRC_PMO_RECORD_FLAGS_ERROR	2158	X'0000086E'
MQRC_PUT_MSG_RECORDS_ERROR	2159	X'0000086F'
MQRC_CONN_ID_IN_USE	2160	X'00000870'
MQRC_Q_MGR QUIESCING	2161	X'00000871'
MQRC_Q_MGR_STOPPING	2162	X'00000872'
MQRC_DUPLICATE_RECOV_COORD	2163	X'00000873'
MQRC_PMO_ERROR	2173	X'0000087D'
MQRC_API_EXIT_NOT_FOUND	2182	X'00000886'
MQRC_API_EXIT_LOAD_ERROR	2183	X'00000887'
MQRC_REMOTE_Q_NAME_ERROR	2184	X'00000888'
MQRC_INTUTARLMENT_PERSISTENCE	2185	X'00000889'
MQRC_GMO_ERROR	2186	X'0000088A'
MQRC_CICS_BRIDGE_RESTRICTION	2187	X'0000088B'
MQRC_STOPPED_BY_CLUSTER_EXIT	2188	X'0000088C'
MQRC_CLUSTER_RESOLUTION_ERROR	2189	X'0000088D'
MQRC_CONVERTED_STRING_TOO_Büyük	2190	X'0000088E'
MQRC_TMC_ERROR	2191	X'0000088F'
MQRC_PAGEET_FULL	2192	X'00000890'
MQRC_STORAGE_MEDIUM_FULL	2192	X'00000890'
MQRC_PAGEES_HATASI	2193	X'00000891'
MQRC_NAME_NOT_VALID_FOR_TYPE	2194	X'00000892'
MQRC_UNEXPECTED_ERROR	2195	X'00000893'
MQRC_UNKNOWN_XMIT_Q	2196	X'00000894'
MQRC_UNKNOWN_DEF_XMIT_Q	2197	X'00000895'
MQRC_DEF_XMIT_Q_TYPE_ERROR	2198	X'00000896'
MQRC_DEF_XMIT_Q_USAGE_ERROR	2199	X'00000897'
MQRC_MSG_MARKED_BROWSE_CO_OP	2200	X'00000898'
MQRC_NAME_IN_USE	2201	X'00000899'
MQRC_CONNECTION QUIESCING	2202	X'0000089A'

Çizelge 306. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQRC_CONNECTION_DURATION	2203	X'0000089B'
MQRC_ADAPTER_NOT_AVAM	2204	X'0000089C'
MQRC_MSG_ID_ERROR	2206	X'0000089E'
MQRC_CORREL_ID_ERROR	2207	X'0000089F'
MQRC_FILE_SYSTEM_ERROR	2208	X'000008A0'
MQRC_NO_MSG_LOCKED	2209	X'000008A1'
MQRC_SOAP_DOTNET_ERROR	2210	X'000008A2'
MQRC_SOAP_AXIS_ERROR	2211	X'000008A3'
MQRC_SOAP_URL_ERROR	2212	X'000008A4'
MQRC_FILE_NOT_DENETLENDI	2216	X'000008A8'
MQRC_CONNECTION_NOT_YETKILI	2217	X'000008A9'
MQRC_MSG_TOO_BIG_FOR_CHANNEL	2218	X'000008AA'
MQRC_CALL_IN_PROGRESS	2219	X'000008AB'
MQRC_RMH_ERROR	2220	X'000008AC'
MQRC_Q_MGR_ETKİN	2222	X'000008AE'
MQRC_Q_MGR_NOT_ETKİN	2223	X'000008AF'
MQRC_Q_DEPTH_YÜKSEKLİK	2224	X'000008B0'
MQRC_Q_DEPTH_LOW	2225	X'000008B1'
MQRC_Q_SERVICE_INTERN_YÜKSEK	2226	X'000008B2'
MQRC_Q_SERVICE_INTERVAL_OK	2227	X'000008B3'
MQRC_RFH_HEADER_FIELD_ERROR	2228	X'000008B4'
MQRC_RAS_ÖZELİ_HATASI	2229	X'000008B5'
MQRC_UNIT_OF_WORK_NOT_STARTED	2232	X'000008B8'
MQRC_CHANNEL_AUTO_DEF_OK	2233	X'000008B9'
MQRC_CHANNEL_AUTO_DEF_ERROR	2234	X'000008BA'
MQRC_CFH_ERROR	2235	X'000008BB'
MQRC_CFIL_ERROR	2236	X'000008BC'
MQRC_CFIN_ERROR	2237	X'000008BD'
MQRC_CFSL_ERROR	2238	X'000008BE'
MQRC_CFST_ERROR	2239	X'000008BF'
MQRC_INCOMPLE_GROUP	2241	X'000008C1'
MQRC_INCOMPLE_MSG	2242	X'000008C2'
MQRC_INCONSISTENT_CCSIDS	2243	X'000008C3'
MQRC_INCONTINUED_ENCODINGS	2244	X'000008C4'
MQRC_INCONSISTENT_UOW	2245	X'000008C5'
MQRC_INVALID_MSG_UNDER_CURSOR	2246	X'000008C6'
MQRC_MATCH_OPTIONS_ERROR	2247	X'000008C7'
MQRC_MDE_HATASI	2248	X'000008C8'
MQRC_MSG_FLAGS_ERROR	2249	X'000008C9'

Çizelge 306. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQRC_MSG_SEQ_NUMBER_ERROR	2250	X'000008CA'
MQRC_OFFSET_ERROR	2251	X'000008CB'
MQRC_ORIGINAL_LENGTH_ERROR	2252	X'000008CC'
MQRC_SEGMENT_LENGTH_ZERO	2253	X'000008CD'
MQRC_UOW_NOT_VAR	2255	X'000008CF'
MQRC_WRONG_GMO_Sürümü	2256	X'000008D0'
MQRC_WRONG_MD_VERSION	2257	X'000008D1'
MQRC_GROUP_ID_ERROR	2258	X'000008D2'
MQRC_INCONTINUED_BROWSE	2259	X'000008D3'
MQRC_XQH_ERROR	2260	X'000008D4'
MQRC_SRC_ENV_ERROR	2261	X'000008D5'
MQRC_SRC_NAME_ERROR	2262	X'000008D6'
MQRC_DEST_ENV_ERROR	2263	X'000008D7'
MQRC_DEST_NAME_ERROR	2264	X'000008D8'
MQRC_TM_HATASI	2265	X'000008D9'
MQRC_CLUSTER_EXIT_ERROR	2266	X'000008DA'
MQRC_CLUSTER_EXIT_LOAD_ERROR	2267	X'000008DB'
MQRC_CLUSTER_PUT_INHIMATED	2268	X'000008DC'
MQRC_CLUSTER_KAYNAĞI_HATASI	2269	X'000008DD'
MQRC_NO_DESTINATIONS_AVALABILIR	2270	X'000008DE'
MQRC_CONN_TAG_IN_USE	2271	X'000008DF'
MQRC_PARTIALY_CONVERTED	2272	X'000008E0'
MQRC_CONNECTION_ERROR	2273	X'000008E1'
MQRC_OPTION_ENVIRONMENT_ERROR	2274	X'000008E2'
MQRC_CD_HATASI	2277	X'000008E5'
MQRC_CLIENT_CONN_ERROR	2278	X'000008E6'
MQRC_CHANNEL_STOPPED_BY_USER	2279	X'000008E7'
MQRC_HCONFIL_ERROR	2280	X'000008E8'
MQRC_FUNCTION_HATASI	2281	X'000008E9'
MQRC_CHANNEL_STARTED	2282	X'000008EA'
MQRC_CHANNEL_DURDU	2283	X'000008EB'
MQRC_CHANNEL_CONV_ERROR	2284	X'000008EC'
MQRC_SERVICE_NOT_VAR	2285	X'000008ED'
MQRC_INITIALIZATION_FAILED	2286	X'000008EE'
MQRC_TERMINATION_FAILED	2287	X'000008EF'
MQRC_UNKNOWN_Q_NAME	2288	X'000008F0'
MQRC_SERVICE_ERROR	2289	X'000008F1'
MQRC_Q_ALREADY_EXISTS	2290	X'000008F2'
MQRC_USER_ID_NOT_AVAILABLE	2291	X'000008F3'

Çizelge 306. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQRC_UNKNOWN_ENTITY	2292	X'000008F4'
MQRC_UNKNOWN_AUTH_ENTITY	2293	X'000008F5'
MQRC_UNKNOWN_REF_OBJECT	2294	X'000008F6'
MQRC_CHANNEL_ACTIVATED	2295	X'000008F7'
MQRC_CHANNEL_NOT_ETKINED	2296	X'000008F8'
MQRC_UOW_İPTAL	2297	X'000008F9'
MQRC_FUNCTION_NOT_SUPPORUL	2298	X'000008FA'
MQRC_SELECTOR_TYPE_ERROR	2299	X'000008FB'
MQRC_COMMAND_TYPE_ERROR	2300	X'000008FC'
MQRC_MULTIPLE_INSTANCE_ERROR	2301	X'000008FD'
MQRC_SYSTEM_ITEM_NOT_ALTERNATEABLE	2302	X'000008FE'
MQRC_BAG_CONVERSION_ERROR	2303	X'000008FF'
MQRC_SELECTOR_OUT_OF_RANGE	2304	X'00000900'
MQRC_SELECTOR_NOT_UNIQUE	2305	X'00000901'
MQRC_INDEX_NOT_PRESNENT	2306	X'00000902'
MQRC_STRING_ERROR	2307	X'00000903'
MQRC_ENCODING_NOT_SUPPORTANT	2308	X'00000904'
MQRC_SELECTOR_NOT_PRESENT	2309	X'00000905'
MQRC_OUT_SELECTOR_ERROR	2310	X'00000906'
MQRC_STRING_TRUNCATED	2311	X'00000907'
MQRC_SELECTOR_HATA_TIPI	2312	X'00000908'
MQRC_INCONSISTENT_ITEM_TYPE	2313	X'00000909'
MQRC_INDEX_ERROR	2314	X'0000090A'
MQRC_SYSTEM_BAG_NOT_ALTERABLE	2315	X'0000090B'
MQRC_ITETİĞ_COUNT_ERROR	2316	X'0000090C'
MQRC_FORMAT_NOT_SUPPORTANT	2317	X'0000090D'
MQRC_SELECTOR_NOT_SUPPORTANT	2318	X'0000090E'
MQRC_ITETI_VALUE_ERROR	2319	X'0000090F'
MQRC_HBAG_ERROR	2320	X'00000910'
MQRC_PARAMETER_MISSING	2321	X'00000911'
MQRC_CMD_SERVER_NOT_VAR	2322	X'00000912'
MQRC_STRING_LENGTH_ERROR	2323	X'00000913'
MQRC_INQUIRY_COMMAND_ERROR	2324	X'00000914'
MQRC_NESTED_BAG_NOT_SUPPORTANT	2325	X'00000915'
MQRC_BAG_HATA_TIPI	2326	X'00000916'
MQRC_ITEM_TYPE_ERROR	2327	X'00000917'
MQRC_SYSTEM_BAG_NOT_DELETABLE	2328	X'00000918'
MQRC_SYSTEM_ITEM_NOT_DELETABLE	2329	X'00000919'
MQRC_CODED_CHAR_SET_ID_ERROR	2330	X'0000091A'



Çizelge 306. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQRC_MSG_TOKEN_HATASI	2331	X'0000091B'
MQRC_MISSING_WIH	2332	X'0000091C'
MQRC_WIH_ERROR	2333	X'0000091D'
MQRC_RFH_ERROR	2334	X'0000091E'
MQRC_RFH_STRING_ERROR	2335	X'0000091F'
MQRC_RFH_COMMAND_ERROR	2336	X'00000920'
MQRC_RFH_PARM_ERROR	2337	X'00000921'
MQRC_RFH_DUPLICATE_PARM	2338	X'00000922'
MQRC_RFH_PARM_MISSING	2339	X'00000923'
MQRC_CHAR_CONVERSION_ERROR	2340	X'00000924'
MQRC_UCS2_CONVERSION_ERROR	2341	X'00000925'
MQRC_DB2_NOT_AVAILABLE	2342	X'00000926'
MQRC_OBJECT_NOT_UNIQUE	2343	X'00000927'
MQRC_CONN_TAG_NOT_SERBEST	2344	X'00000928'
MQRC_CF_NOT_VAR	2345	X'00000929'
MQRC_CF_STRUC_IN_USE	2346	X'0000092A'
MQRC_CF_STRUC_LIST_HDR_IN_USE	2347	X'0000092B'
MQRC_CF_STRUC_AUTH_FAILED	2348	X'0000092C'
MQRC_CF_STRUC_ERROR	2349	X'0000092D'
MQRC_CONN_TAG_NOT_USABLE	2350	X'0000092E'
MQRC_GLOBAL_UOW_CONFLICTIONS	2351	X'0000092F'
MQRC_LOCAL_UOW_CONFLICTID	2352	X'00000930'
MQRC_ANDLE_IN_USE_FOR_UOW	2353	X'00000931'
MQRC_UOWENLISTMENT_ERROR	2354	X'00000932'
MQRC_UOW_MIX_NOT_SUPPORTANT	2355	X'00000933'
MQRC_WXP_ERROR	2356	X'00000934'
MQRC_CURRENT_RECORD_ERROR	2357	X'00000935'
MQRC_NEXT_OFFSET_ERROR	2358	X'00000936'
MQRC_NO_RECORD_VAR	2359	X'00000937'
MQRC_OBJECT_LEVEL_UYUMSUZ	2360	X'00000938'
MQRC_NEXT_RECORD_ERROR	2361	X'00000939'
MQRC_BACKOUT_THRESHOLD_ERULAD	2362	X'0000093A'
MQRC_MSG_NOT_MATCHED	2363	X'0000093B'
MQRC_JMS_FORMAT_ERROR	2364	X'0000093C'
MQRC_SEGMENTS_NOT_SUPPORTED	2365	X'0000093D'
MQRC_HATA_CF_DÜZEYI	2366	X'0000093E'
MQRC_CONFIG_CREATE_OBJECT	2367	X'0000093F'
MQRC_CONFIG_CHANGE_OBJECT	2368	X'00000940'
MQRC_CONFIG_DELET_OBJECT	2369	X'00000941'

Çizelge 306. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQRC_CONFIG_REFRESH_OBJECT	2370	X'00000942'
MQRC_CHANNEL_SSL_ERROR	2371	X'00000943'
MQRC_AĞILAN_KURULUŞ_NOT_	2372	X'00000944'
MQRC_CF_STRUC_FAILED	2373	X'00000945'
MQRC_API_EXIT_ERROR	2374	X'00000946'
MQRC_API_EXIT_INIT_ERROR	2375	X'00000947'
MQRC_API_EXIT_TERM_ERROR	2376	X'00000948'
MQRC_EXIT_REASON_ERROR	2377	X'00000949'
MQRC_RESERVED_VALUE_ERROR	2378	X'0000094A'
MQRC_NO_DATA_AVAILABLE	2379	X'0000094B'
MQRC_SCO_ERROR	2380	X'0000094C'
MQRC_KEY_REPOSITORY_ERROR	2381	X'0000094D'
MQRC_CRYPTOHARDWARE_ERROR	2382	X'0000094E'
MQRC_AUTH_INFO_REC_COUNT_ERROR	2383	X'0000094F'
MQRC_AUTH_INFO_REC_ERROR	2384	X'00000950'
MQRC_AIR_ERROR	2385	X'00000951'
MQRC_AUTH_INFO_TYPE_ERROR	2386	X'00000952'
MQRC_AUTH_INFO_CONN_NAME_ERROR	2387	X'00000953'
MQRC_LDAP_USER_NAME_ERROR	2388	X'00000954'
MQRC_LDAP_USER_NAME_LENGTH_ERR	2389	X'00000955'
MQRC_LDAP_PASSWORD_ERROR	2390	X'00000956'
MQRC_SSL_ALREADY_INITIALIZED	2391	X'00000957'
MQRC_SSL_CONFIG_ERROR	2392	X'00000958'
MQRC_SSL_INITIALIZATION_ERROR	2393	X'00000959'
MQRC_Q_INDEX_TYPE_ERROR	2394	X'0000095A'
MQRC_CFBS_ERROR	2395	X'0000095B'
MQRC_SSL_NOT_ALLOWED	2396	X'0000095C'
MQRC_JSSE_ERROR	2397	X'0000095D'
MQRC_SSL_PEER_NAME_MISMATCH	2398	X'0000095E'
MQRC_SSL_PEER_NAME_ERROR	2399	X'0000095F'
MQRC_UNSUPPORTED_CIPHER_SUITE	2400	X'00000960'
MQRC_SSL_CERTIFICATE_FRESHENED	2401	X'00000961'
MQRC_SSL_CERT_STORE_ERROR	2402	X'00000962'
MQRC_CLIENT_EXIT_LOAD_ERROR	2406	X'00000966'
MQRC_CLIENT_EXIT_ERROR	2407	X'00000967'
MQRC_UOW_COMMITEND	2408	X'00000968'
MQRC_SSL_KEY_RESET_ERROR	2409	X'00000969'
MQRC_UNKNOWN_BİLEŞİY	2410	X'0000096A'
MQRC_LOGGER_STATUS	2411	X'0000096B'

Çizelge 306. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQRC_COMMAND_MQSC	2412	X'0000096C'
MQRC_COMMAND_PCF	2413	X'0000096D'
MQRC_CFIF_ERROR	2414	X'0000096E'
MQRC_CFSF_ERROR	2415	X'0000096F'
MQRC_CFGR_ERROR	2416	X'00000970'
MQRC_MSG_NOT_ALLOWED_IN_GROUP	2417	X'00000971'
MQRC_FILTER_OPERATOR_ERROR	2418	X'00000972'
MQRC_NESTED_SELECTOR_ERROR	2419	X'00000973'
MQRC_EPH_ERROR	2420	X'00000974'
MQRC_RFH_FORMAT_ERROR	2421	X'00000975'
MQRC_CFBF_ERROR	2422	X'00000976'
MQRC_CLIENT_CHANNEL_CONFLIP	2423	X'00000977'
MQRC_SD_ERROR	2424	X'00000978'
MQRC_TOPIC_STRING_ERROR	2425	X'00000979'
MQRC_STS_ERROR	2426	X'0000097A'
MQRC_NO_ALTYPİSİ	2428	X'0000097C'
MQRC_ALTSCRIPTION_IN_USE	2429	X'0000097D'
MQRC_STAT_TYPE_ERROR	2430	X'0000097E'
MQRC_SUB_USER_DATA_ERROR	2431	X'0000097F'
MQC_SUB_ALREADY_EXISTS	2432	X'00000980'
MQRC_IDENTITY_MISMATCH	2434	X'00000982'
MQRC_ALTER_SUB_HATASI	2435	X'00000983'
MQRC_DURABILITY_NOT_ALLOWALIZE	2436	X'00000984'
MQRC_NO_RETAINED_MSG	2437	X'00000985'
MQRC_SRO_ERROR	2438	X'00000986'
MQRC_SUB_NAME_ERROR	2440	X'00000988'
MQRC_OBJECT_STRING_ERROR	2441	X'00000989'
MQRC_ÖZELLİĞİ _	2442	X'0000098A'
MQRC_SEGMENTATION_NOT_ALLOWEND	2443	X'0000098B'
MQRC_CBD_ERROR	2444	X'0000098C'
MQRC_CTLO_ERROR	2445	X'0000098D'
MQC_NO_CALLBACKS_ACTIVE	2446	X'0000098E'
MQRC_CALLBACK_NOT_REGISTERED	2448	X'00000990'
MQRC_OPTIONS_CHANGED	2457	X'00000999'
MQRC_READ_AHEAD_MSGS	2458	X'0000099A'
MQRC_SELECTOR_SYNTAX_ERROR	2459	X'0000099B'
MQRC_HMSG_ERROR	2460	X'0000099C'
MQRC_CMHO_ERROR	2461	X'0000099D'
MQRC_DMHO_ERROR	2462	X'0000099E'

Çizelge 306. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQRC_SMPO_ERROR	2463	X'0000099F'
MQRC_IMPO_ERROR	2464	X'000009A0'
MQRC_özelliğİN_NAME_TOO_	2465	X'000009A1'
MQRC_PROP_VALUE_NOT_CONVERTED	2466	X'000009A2'
MQRC_PROP_TYPE_NOT_SUPPORTED	2467	X'000009A3'
MQRC_PROPERTY_VALUE_TOO_BIG	2469	X'000009A5'
MQRC_PROP_CONV_NOT_SUPPORTED	2470	X'000009A6'
MQRC_PROPERTY_NOT_VAR	2471	X'000009A7'
MQRC_PROP_NUMBER_FORMAT_ERROR	2472	X'000009A8'
MQRC_özelliği TYPE_	2473	X'000009A9'
MQRC_özellik_TOO_Büyük	2478	X'000009AE'
MQRC_PUT_NOT_ALICILI	2479	X'000009AF'
MQRC_ALIAS_TARGETTYPE_CHANGED	2480	X'000009B0'
MQRC_DMPO_ERROR	2481	X'000009B1'
MQRC_PD_ERROR	2482	X'000009B2'
MQRC_CALLBACK_TYPE_ERROR	2483	X'000009B3'
MQRC_CBD_OPTIONS_ERROR	2484	X'000009B4'
MQRC_MAX_MSG_LENGTH_ERROR	2485	X'000009B5'
MQRC_CALLBACK_ROUTINE_ERROR	2486	X'000009B6'
MQRC_CALLBACK_LINK_ERROR	2487	X'000009B7'
MQRC_OPERATION_HATASI	2488	X'000009B8'
MQRC_BMHO_ERROR	2489	X'000009B9'
MQRC_UNSUPPORTED_özelliği	2490	X'000009BA'
MQRC_PROP_NAME_NOT_CONVERTED	2492	X'000009BC'
MQRC_GET_ENABLED işiği	2494	X'000009BE'
MQRC_MODULE_NOT_FOUND	2495	X'000009BF'
MQRC_MODULE_INVALID	2496	X'000009C0'
MQRC_MODULE_ENTRY_NOT_FOUND	2497	X'000009C1'
MQRC_MIXED_CONTENT_NOT_ALLOWED	2498	X'000009C2'
MQRC_MSG_HANDLE_IN_USE	2499	X'000009C3'
MQRC_HCONN_ASYNC_ETKİN	2500	X'000009C4'
MQRC_MHBO_ERROR	2501	X'000009C5'
MQRC_PUBLICATION_FAILURE	2502	X'000009C6'
MQRC_SUB_INHIBITED	2503	X'000009C7'
MQRC_SELECTOR_ALWAYS_FALSE	2504	X'000009C8'
MQRC_XEPO_ERROR	2507	X'000009CB'
MQRC_DURABILITY_NOT_ALTERABLE	2509	X'000009CD'
MQRC_TOPIC_NOT_ALTERABLE	2510	X'000009CE'
MQRC_XX_ENCODE_CASE_ONE sublevel_not_alterable	2512	X'000009D0'

Çizelge 306. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQRC_PROPERTY_NAME_LENGTH_ERR	2513	X'000009D1'
MQRC_DUPLICATE_GROUP_SUB	2514	X'000009D2'
MQRC_GROUPING_NOT_ALTERABLE	2515	X'000009D3'
MQRC_SELECTOR_INVALID_FOR_TYPE	2516	X'000009D4'
MQRC_HOBJ QUIESCED	2517	X'000009D5'
MQRC_HOBJ QUIESCED_NO_MSGS	2518	X'000009D6'
MQRC_SELECTION_STRING_ERROR	2519	X'000009D7'
MQRC_RES_OBJECT_STRING_ERROR	2520	X'000009D8'
MQRC_CONNECTION_SUSPAN	2521	X'000009D9'
MQRC_INVALID_DESTIND	2522	X'000009DA'
MQRC_INVALID_SUBSCRIPTION	2523	X'000009DB'
MQRC_SELECTOR_NOT_ALTERABLE	2524	X'000009DC'
MQRC_RETAINED_MSG_Q_ERROR	2525	X'000009DD'
MQRC_RETAINED_NOT_TESLIM	2526	X'000009DE'
MQRC_RFH_RESTRICTED_FORMAT_ERR	2527	X'000009DF'
MQRC_CONNECTION_DURDU	2528	X'000009E0'
MQRC_ASYNC_UOW_CONFLICTER	2529	X'000009E1'
MQRC_ASYNC_XA_CONFLICTER	2530	X'000009E2'
MQRC_PUBSUB_INHIBITED	2531	X'000009E3'
MQRC_MSG_HANDLE_COPY_FAILURE	2532	X'000009E4'
MQRC_DEST_CLASS_NOT_ALTERABLE	2533	X'000009E5'
MQRC_OPERATION_NOT_ALLOWD	2534	X'000009E6'
MQRC_ACTION_ERROR	2535	X'000009E7'
MQRC_CHANNEL_NOT_VAR	2537	X'000009E9'
MQRC_HOST_NOT_AVAM	2538	X'000009EA'
MQRC_CHANNEL_CONFIG_ERROR	2539	X'000009EB'
MQRC_UNKNOWN_CHANNEL_NAME	2540	X'000009EC'
MQRC_LOOPING_YAYINI	2541	X'000009ED'
MQRC_ALREADY_JOINED	2542	X'000009EE'
MQRC_CHANNEL_SSL_UYARI	2552	X'000009F8'
MQRC_OCSP_URL_ERROR	2553	X'000009F9'
MQRC_CIPHER_SPEC_NOT_SUITE_B	2591	X'00000A1F'
MQRC_SUITE_B_ERROR	2592	X'00000A20'
MQRC_PASSWORD_PROTECTION_ERROR	2594	X'00000A22'
MQRC_REOPEN_EXCL_INPUT_ERROR	6100	X'000017D4'
MQRC_REOPEN_INQUIRE_ERROR	6101	X'000017D5'
MQRC_REOPEN_SAVED_CONTEXT_ERR	6102	X'000017D6'
MQRC_REOPEN_TEMPORARY_Q_ERROR	6103	X'000017D7'
MQRC_ATTRIBUTE_LOCKED	6104	X'000017D8'

Çizelge 306. Değişmezlerin değerleri (devamı var)		
Ad	Ondalık değer	Onaltılı değer
MQR_CURSOR_NOT_VALID	6105	X'000017D9'
MQR_ENCODING_ERROR	6106	X'000017DA'
MQR_STRUC_ID_ERROR	6107	X'000017DB'
MQR_NULL_POINTER	6108	X'000017DC'
MQR_NO_CONNECTION_REFERENCE	6109	X'000017DD'
MQR_NO_BUFFER	6110	X'000017DE'
MQR_BINARY_DATA_LENGTH_ERROR	6111	X'000017DF'
MQR_BUFFER_NOT_AUTOMAKRED	6112	X'000017E0'
MQR_INSUFFICIENT_BUFFER	6113	X'000017E1'
MQR_INSUFFICIENT_DATA	6114	X'000017E2'
MQR_DATA_TRUNCATED	6115	X'000017E3'
MQR_ZERO_LENGTH	6116	X'000017E4'
MQR_NEGATIVE_LENGTH	6117	X'000017E5'
MQR_NEGATIVE_OFFSET	6118	X'000017E6'
MQR_INCONSTANT_BİÇİMİ	6119	X'000017E7'
MQR_INCONSISTENT_OBJECT_STATE	6120	X'000017E8'
MQR_CONTEXT_OBJECT_NOT_VALID	6121	X'000017E9'
MQR_CONTEXT_OPEN_ERROR	6122	X'000017EA'
MQR_STRUC_LENGTH_ERROR	6123	X'000017EB'
MQR_NOT_CONNECTED	6124	X'000017EC'
MQR_NOT_Açık	6125	X'000017ED'
MQR_DISTRIBUTION_LIST_EMPTY	6126	X'000017EE'
MQR_INCONSISTENT_OPEN_OPTIONS	6127	X'000017EF'
MQR_WRONG_Sürüm	6128	X'000017F0'
MQR_REFERENCE_ERROR	6129	X'000017F1'

## MQRCCF\_\* (Komut biçimi üstbilgisi neden kodları)

Programcı yanıtına ilişkin ek bilgi için [PCF neden kodları](#) başlıklı konuya bakın.

Çizelge 307. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQRCCF_CFH_TYPE_ERROR	3001	X'00000BB9'
MQRCCF_CFH_LENGTH_ERROR	3002	X'00000BBA'
MQRCCF_CFH_VERSION_ERROR	3003	X'00000BBB'
MQRCCF_CFH_MSG_SEQ_NUMBER_ERR	3004	X'00000BBC'
MQRCCF_CFH_CONTROL_ERROR	3005	X'00000BBD'
MQRCCF_CFH_PARM_COUNT_ERROR	3006	X'00000BBE'
MQRCCF_CFH_COMMAND_ERROR	3007	X'00000BBF'
MQRCCF_COMMAND_FAILED	3008	X'00000BC0'
MQRCCF_CFIN_LENGTH_ERROR	3009	X'00000BC1'

Çizelge 307. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQRCCF_CFST_LENGTH_ERROR	3010	X'00000BC2'
MQRCCF_CFST_STRING_LENGTH_ERR	3011	X'00000BC3'
MQRCCF_FORCE_VALUE_ERROR	3012	X'00000BC4'
MQRCCF_STRUCTURE_TYPE_ERROR	3013	X'00000BC5'
MQRCCF_CFIN_PARM_ID_ERROR	3014	X'00000BC6'
MQRCCF_CFST_PARM_ID_ERROR	3015	X'00000BC7'
MQRCCF_MSG_LENGTH_ERROR	3016	X'00000BC8'
MQRCCF_CFIN_DUPLICATE_PARM	3017	X'00000BC9'
MQRCCF_CFST_DUPLICATE_PARM	3018	X'00000BCA'
MQRCCF_PARM_COUNT_TOO_KÜÇÜK	3019	X'00000BCB'
MQRCCF_PARM_COUNT_TOO_BÜYÜK	3020	X'00000BCC'
MQRCCF_Q_ALREADY_IN_CELL	3021	X'00000BCD'
MQRCCF_Q_TYPE_ERROR	3022	X'00000BCE'
MQRCCF_MD_FORMAT_ERROR	3023	X'00000BCF'
MQRCCF_CFSL_LENGTH_ERROR	3024	X'00000BD0'
MQRCCF_REPLACE_VALUE_ERROR	3025	X'00000BD1'
MQRCCF_CFIL_DUPLICATE_DEĞER	3026	X'00000BD2'
MQRCCF_CFIL_COUNT_ERROR	3027	X'00000BD3'
MQRCCF_CFIL_LENGTH_ERROR	3028	X'00000BD4'
MQRCCF QUIESCE_VALUE_ERROR	3029	X'00000BD5'
MQRCCF_MODE_VALUE_ERROR	3029	X'00000BD5'
MQRCCF_MSG_SEQ_NUMBER_ERROR	3030	X'00000BD6'
MQRCCF_PING_DATA_COUNT_ERROR	3031	X'00000BD7'
MQRCCF_PING_DATA_COMPARE_ERROR	3032	X'00000BD8'
MQRCCF_CFSL_PARM_ID_ERROR	3033	X'00000BD9'
MQRCCF_CHANNEL_TYPE_ERROR	3034	X'00000BDA'
MQRCCF_PARM_SEQUENCE_ERROR	3035	X'00000BDB'
MQRCCF_XMIT_PROTOCOL_TYPE_ERR	3036	X'00000BDC'
MQRCCF_BATCH_SIZE_ERROR	3037	X'00000BDD'
MQRCCF_DISC_INT_ERROR	3038	X'00000BDE'
MQRCCF_SHORT_RETRY_ERROR	3039	X'00000BDF'
MQRCCF_SHORT_TIMER_ERROR	3040	X'00000BE0'
MQRCCF_LONG_RETRY_ERROR	3041	X'00000BE1'
MQRCCF_LONG_TIMER_ERROR	3042	X'00000BE2'
MQRCCF_SEQ_NUMBER_WRAP_ERROR	3043	X'00000BE3'
MQRCCF_MAX_MSG_LENGTH_ERROR	3044	X'00000BE4'
MQRCCF_PUT_AUTH_ERROR	3045	X'00000BE5'
MQRCCF_PURGE_VALUE_ERROR	3046	X'00000BE6'
MQRCCF_CFIL_PARM_ID_ERROR	3047	X'00000BE7'

Çizelge 307. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQRCCF_MSG_KESILDI	3048	X'00000BE8'
MQRCCF_CCSDI	3049	X'00000BE9'
MQRCCFENCODING_ERROR	3050	X'00000BEA'
MQRCCF_QUEUES_VALUE_ERROR	3051	X'00000BEB'
MQRCCF_DATA_CONV_VALUE_ERROR	3052	X'00000BEC'
MQRCCF_INDOUBT_VALUE_ERROR	3053	X'00000BED'
MQRCCF_ESCAPE_TYPE_ERROR	3054	X'00000BEE'
MQRCCF_REPOS_VALUE_ERROR	3055	X'00000BEF'
MQRCCF_CHANNEL_TABLE_ERROR	3062	X'00000BF6'
MQRCCF_MCA_TYPE_ERROR	3063	X'00000BF7'
MQRCCF_CHL_INST_TYPE_ERROR	3064	X'00000BF8'
MQRCCF_CHL_STATUS_NOT_FOUND	3065	X'00000BF9'
MQRCCF_CFL_DUPLICATE_PARM	3066	X'00000BFA'
MQRCCF_CFSL_TOTAL_LENGTH_ERROR	3067	X'00000BFB'
MQRCCF_CFSL_COUNT_ERROR	3068	X'00000BFC'
MQRCCF_CFSL_STRING_LENGTH_ERR	3069	X'00000BFD'
MQRCCF_BROKER_DELETED	3070	X'00000BFE'
MQRCCF_STREAM_ERROR	3071	X'00000BFF'
MQRCCF_TOPIC_ERROR	3072	X'00000C00'
MQRCCF_NOT_REGISTERED	3073	X'00000C01'
MQRCCF_Q_MGR_NAME_ERROR	3074	X'00000C02'
MQRCCF_INCORRECT_STREAM	3075	X'00000C03'
MQRCCF_Q_NAME_ERROR	3076	X'00000C04'
MQRCCF_NO_RETAINED_MSG	3077	X'00000C05'
MQRCCF_DUPLICATE_IDENTITY	3078	X'00000C06'
MQRCCF_INCORRECT_Q	3079	X'00000C07'
MQRCCF_CORREL_ID_ERROR	3080	X'00000C08'
MQRCCF_NOT_YETKILI	3081	X'00000C09'
MQRCCF_UNKNOWN_STREAM	3082	X'00000C0A'
MQRCCF_REG_OPTIONS_ERROR	3083	X'00000C0B'
MQRCCF_PUB_OPTIONS_ERROR	3084	X'00000C0C'
MQRCCF_UNKNOWN_BROKER	3085	X'00000C0D'
MQRCCF_Q_MGR_CCSDI_ERROR	3086	X'00000C0E'
MQRCCF_DEL_OPTIONS_ERROR	3087	X'00000C0F'
MQRCCF_CLUSTER_NAME_CONFLICTER	3088	X'00000C10'
MQRCCF_REPOS_NAME_CONFLICT	3089	X'00000C11'
MQRCCF_CLUSTER_Q_USAGE_ERROR	3090	X'00000C12'
MQRCCF_ACTION_VALUE_ERROR	3091	X'00000C13'
MQRCCF_COMMS_LIBRARY_ERROR	3092	X'00000C14'



Çizelge 307. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQRCCF_NETBIOS_NAME_ERROR	3093	X'00000C15'
MQRCCF_BROKER_COMMAND_FAILED	3094	X'00000C16'
MQRCCF_CFST_CONFLICT_PARM	3095	X'00000C17'
MQRCCF_PATH_NOT_VALID	3096	X'00000C18'
MQRCCF_PARM_SYNTAX_ERROR	3097	X'00000C19'
MQRCCF_PWD_LENGTH_ERROR	3098	X'00000C1A'
MQRCCF_FILTER_HATASI	3150	X'00000C4E'
MQRCCF_HATA_KULLANICISI	3151	X'00000C4F'
MQRCCF_DUPLICATE_SUBSCRIPTION	3152	X'00000C50'
MQRCCF_XX_ENCODE_CASE_ONE sub_name_error	3153	X'00000C51'
MQRCCF_SUB_IDENTITY_ERROR	3154	X'00000C52'
MQRCCF_SUBSCRIPTION_IN_USE	3155	X'00000C53'
MQRCCF_SUBSCRIPTION_LOCKED	3156	X'00000C54'
MQRCCF_ALREADY_JOINED	3157	X'00000C55'
MQRCCF_OBJECT_IN_USE	3160	X'00000C58'
MQRCCF_UNKNOWN_FILE_ADI	3161	X'00000C59'
MQRCCF_FILE_NOT_VAR	3162	X'00000C5A'
MQRCCF_DISC_RETRY_ERROR	3163	X'00000C5B'
MQRCCF_ALLOC_RETRY_ERROR	3164	X'00000C5C'
MQRCCF_ALLOC_SLOW_TIMER_ERROR	3165	X'00000C5D'
MQRCCF_ALLOC_FAST_TIMER_ERROR	3166	X'00000C5E'
MQRCCF_PORT_NUMBER_ERROR	3167	X'00000C5F'
MQRCCF_CHL_SYSTEM_NOT_ETKİN	3168	X'00000C60'
MQRCCF_ENTITY_NAME_MISSING	3169	X'00000C61'
MQRCCF_PROFILE_NAME_ERROR	3170	X'00000C62'
MQRCCF_AUTH_VALUE_ERROR	3171	X'00000C63'
MQRCCF_AUTH_VALUE_EKSIK	3172	X'00000C64'
MQRCCF_OBJECT_TYPE_EKSIK	3173	X'00000C65'
MQRCCF_CONNECTION_ID_ERROR	3174	X'00000C66'
MQRCCF_LOG_TYPE_ERROR	3175	X'00000C67'
MQRCCF_PROGRAM_NOT_VAR	3176	X'00000C68'
MQRCCF_PROGRAM_AUTH_FAILED	3177	X'00000C69'
MQRCCF_NONE_BULUNDU	3200	X'00000C80'
MQRCCF_SECURITY_SWITCH_OFF	3201	X'00000C81'
MQRCCF_SECURITY_REFRESH_FAILED	3202	X'00000C82'
MQRCCF_PARM_CONFLIP	3203	X'00000C83'
MQRCCF_COMMAND_INHIBITED	3204	X'00000C84'
MQRCCF_OBJECT_BEING_DELETED	3205	X'00000C85'
MQRCCF_STORAGE_CLASS_IN_USE	3207	X'00000C87'

Çizelge 307. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQRCCF_OBJECT_NAME_RESTRICTED	3208	X'00000C88'
MQRCCF_OBJECT_LIMIT_EXADEDI	3209	X'00000C89'
MQRCCF_OBJECT_OPEN_FORCE	3210	X'00000C8A'
MQRCCF_DISPOSI_CONFLICTION	3211	X'00000C8B'
MQRCCF_Q_MGR_NOT_IN_QSG	3212	X'00000C8C'
MQRCCF_ATTR_VALUE_FIXED	3213	X'00000C8D'
MQRCCF_NAMELISTAN_HATASI	3215	X'00000C8F'
MQRCCF_NO_CHANNEL_INITIATOR	3217	X'00000C91'
MQRCCF_CHANNEL_INITIATOR_ERROR	3218	X'00000C92'
MQRCCF_KOMUT_DÜZEY_ÇAKIŞMASI	3222	X'00000C96'
MQRCCF_Q_ATTR_CONFLICT	3223	X'00000C97'
MQRCCF_EVENTS_DISABLI	3224	X'00000C98'
MQRCCF_COMMAND_SCOPE_ERROR	3225	X'00000C99'
MQRCCF_COMMAND_REPLY_ERROR	3226	X'00000C9A'
MQRCCF_FUNCTION_RESTRICTED	3227	X'00000C9B'
MQRCCF_PARM_MISTELİĞİ	3228	X'00000C9C'
MQRCCF_PARM_VALUE_ERROR	3229	X'00000C9D'
MQRCCF_COMMAND_LENGTH_ERROR	3230	X'00000C9E'
MQRCCF_COMMAND_ORIGIN_ERROR	3231	X'00000C9F'
MQRCCF_LISTENER_CONFLICCE	3232	X'00000CA0'
MQRCCF_LISTEMER_STARTED	3233	X'00000CA1'
MQRCCF_LISTENER_DURDU	3234	X'00000CA2'
MQRCCF_CHANNEL_ERROR	3235	X'00000CA3'
MQRCCF_CF_STRUC_ERROR	3236	X'00000CA4'
MQRCCF_UNKNOWN_USER_ID	3237	X'00000CA5'
MQRCCF_UNEXPECTED_ERROR	3238	X'00000CA6'
MQRCCF_NO_XCF_PARÇA	3239	X'00000CA7'
MQRCCF_CFGR_PARM_ID_ERROR	3240	X'00000CA8'
MQRCCF_CFIF_LENGTH_ERROR	3241	X'00000CA9'
MQRCCF_CFIF_OPERATOR_ERROR	3242	X'00000CAA'
MQRCCF_CFIF_PARM_ID_ERROR	3243	X'00000CAB'
MQRCCF_CFSF_FILTER_VAL_LEN_ERR	3244	X'00000CAC'
MQRCCF_CFSF_LENGTH_ERROR	3245	X'00000CAD'
MQRCCF_CFSF_OPERATOR_ERROR	3246	X'00000CAE'
MQRCCF_CFSF_PARM_ID_ERROR	3247	X'00000CAF'
MQRCCF_TOO_MANY_FILTERS	3248	X'00000CB0'
MQRCCF_LISTENER_RUNNING	3249	X'00000CB1'
MQRCCF_LSTR_STATUS_NOT_FOUND	3250	X'00000CB2'
MQRCCF_SERVICE_RUNS	3251	X'00000CB3'

Çizelge 307. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQRCCF_SERV_STATUS_NOT_FOUND	3252	X'00000CB4'
MQRCCF_SERVICE_DURDURULDU	3253	X'00000CB5'
MQRCCF_CFBS_DUPLICATE_PARM	3254	X'00000CB6'
MQRCCF_CFBS_LENGTH_ERROR	3255	X'00000CB7'
MQRCCF_CFBS_PARM_ID_ERROR	3256	X'00000CB8'
MQRCCF_CFBS_STRING_LENGTH_ERR	3257	X'00000CB9'
MQRCCF_CFGR_LENGTH_ERROR	3258	X'00000CBA'
MQRCCF_CFGR_PARM_COUNT_ERROR	3259	X'00000CBB'
MQRCCF_CONN_NOT_DURDU	3260	X'00000CBC'
MQRCCF_SERVICE_REQUEST_PENDING	3261	X'00000CBD'
MQRCCF_NO_START_CMD	3262	X'00000CBE'
MQRCCF_NO_STOP_CMD	3263	X'00000CBF'
MQRCCF_CFBF LENGH_ERROR	3264	X'00000CC0'
MQRCCF_CFBF_PARM_ID_ERROR	3265	X'00000CC1'
MQRCCF_CFBF_OPERATOR_ERROR	3266	X'00000CC2'
MQRCCF_CFBF_FILTER_VAL_LEN_ERR	3267	X'00000CC3'
MQRCCF_LISTENER_STIL_ETKİN	3268	X'00000CC4'
MQRCCF_DEF_XMIT_Q_CLUS_ERROR	3269	X'00000CC5'
MQRCCF_TOPICSTR_ALREADY_EXISTS	3300	X'00000CE4'
MQRCCF_SHARING_CONVS_ERROR	3301	X'00000CE5'
MQRCCF_SHARING_CONVS_TYPE	3302	X'00000CE6'
MQRCCF_SECURITY_CASE_CONFLICCE	3303	X'00000CE7'
MQRCCF_TOPIC_TYPE_ERROR	3305	X'00000CE9'
MQRCCF_MAX_INSTANCES_ERROR	3306	X'00000CEA'
MQRCCF_MAX_INSTS_PER_CLNT_ERR	3307	X'00000CEB'
MQRCCF_TOPIC_STRING_NOT_FOUND	3308	X'00000CEC'
MQRCCF_SUBSCRIPTION_POINT_ERR	3309	X'00000CED'
MQRCCF_SUB_ALREADY_EXISTS	3311	X'00000CEF'
MQRCCF_UNKNOWN_OBJECT_NAME	3312	X'00000CF0'
MQRCCF_REMOTE_Q_NAME_ERROR	3313	X'00000CF1'
MQRCCF_DURABILITY_NOT_ALLOWALIZE	3314	X'00000CF2'
MQRCCF_HOBJ_ERROR	3315	X'00000CF3'
MQRCCF_DEST_NAME_ERROR	3316	X'00000CF4'
MQRCCF_INVALID_DESTIND	3317	X'00000CF5'
MQRCCF_PUBSUB_INHIBITED	3318	X'00000CF6'
MQRCCF_CHATUTH_TYPE_ERROR	3326	X'00000CFE'
MQRCCF_CHATUTH_ACTION_ERROR	3327	X'00000CFF'
MQRCCF_CHATUTH_USERSRC_ERROR	3335	X'00000D07'
MQRCCF_HATA_CHLAUTH_TYPE	3336	X'00000D08'


Çizelge 307. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQRCCF_CHLAUTH_ALREADY_EXISTS	3337	X'00000D09'
MQRCCF_CHATUTH_NOT_FOUND	3338	X'00000D0A'
MQRCCF_HATA_CHLAUTH_ACTION	3339	X'00000D0B'
MQRCCF_HATA_CHLAUTH_USERSRC	3340	X'00000D0C'
MQRCCF_CHATUTH_WARN_ERROR	3341	X'00000D0D'
MQRCCF_HATA_CHLAUTH_MATCH	3342	X'00000D0E'
MQRCCF_IPADDR_RANGE_CONFLICCE	3343	X'00000D0F'
MQRCCF_CHLAUTH_MAX_EXTEND	3344	X'00000D10'
MQRCCF_IPADDR_ERROR	3345	X'00000D11'
MQRCCF_IPADDR_RANGE_ERROR	3346	X'00000D12'
MQRCCF_PROFILE_NAME_MI_MISSING	3347	X'00000D13'
MQRCCF_CHATUTH_CLNTER_ERROR	3348	X'00000D14'
MQRCCF_CHATUTH_NAME_ERROR	3349	X'00000D15'
MQRCCF_SUITE_B_ERROR	3353	X'00000D19'
MQRCCF_PSCLUS_DISABLED_TOPDEF	3359	X'00000D1F'
MQRCCF_PSSCLUS_TOPIC_EXISTS	3360	X'00000D20'
MQRCCF_INVALID_PROTOCOL	3365	X'00000D25'
MQRCCF_ACCESS_BLOCKED	3382	X'00000D36'
MQRCCF_OBJECT_ALREADY_EXISTS	4001	X'00000FA1'
MQRCCF_OBJECT_HATA_TIPI	4002	X'00000FA2'
MQRCCF_LIKE_OBJECT_HATA_TIPI	4003	X'00000FA3'
MQRCCF_OBJECT_Aç	4004	X'00000FA4'
MQRCCF_ATTR_VALUE_ERROR	4005	X'00000FA5'
MQRCCF_UNKNOWN_Q_MGR	4006	X'00000FA6'
MQRCCF_Q_HATA_TIPI	4007	X'00000FA7'
MQRCCF_OBJECT_NAME_ERROR	4008	X'00000FA8'
MQRCCF_ALLOCATE_FAILED	4009	X'00000FA9'
MQRCCF_ANASISTEM_NOT_VAR	4010	X'00000FAA'
MQRCCF_CONFIGURATION_ERROR	4011	X'00000FAB'
MQRCCF_CONNECTION_REFRET	4012	X'00000FAC'
MQRCCFENTRY_ERROR	4013	X'00000FAD'
MQRCCF_SEND_FAILED	4014	X'00000FAE'
MQRCCF_RECEIVED_DATA_ERROR	4015	X'00000FAF'
MQRCCF_RECEIVE_FAILED	4016	X'00000FB0'
MQRCCF_CONNECTION_CLOSED	4017	X'00000FB1'
MQRCCF_NO_STORSAYI	4018	X'00000FB2'
MQRCCF_NO_COMMS_YÖNETME	4019	X'00000FB3'
MQRCCF_LISTENER_NOT_STARTED	4020	X'00000FB4'
MQRCCF_BIND_FAILED	4024	X'00000FB8'

Çizelge 307. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQRCCF_CHANNEL_INDOUBT	4025	X'00000FB9'
MQRCCF_MQCONN_FAILED	4026	X'00000FBA'
MQRCCF_MQOPEN_FAILED	4027	X'00000FBB'
MQRCCF_MQGET_FAILED	4028	X'00000FBC'
MQRCCF_MQPUT_FAILED	4029	X'00000FBD'
MQRCCF_PING_ERROR	4030	X'00000FBE'
MQRCCF_CHANNEL_IN_USE	4031	X'00000FBF'
MQRCCF_CHANNEL_NOT_BULUNDU	4032	X'00000FC0'
MQRCCF_UNKNOWN_REMOTE_CHANNEL	4033	X'00000FC1'
MQRCCF_REMOTE_QM_UNAVAILABLE	4034	X'00000FC2'
MQRCCF_REMOTE_QM_SONLANDIRMA	4035	X'00000FC3'
MQRCCF_MQINQ_FAILED	4036	X'00000FC4'
MQRCCF_NOT_XMIT_Q	4037	X'00000FC5'
MQRCCF_CHANNEL_DISABET	4038	X'00000FC6'
MQRCCF_USER_EXIT_NOT_VAR	4039	X'00000FC7'
MQRCCF_COMMIT_FAILED	4040	X'00000FC8'
MQRCCF_HATA_CHANNEL_TYPE	4041	X'00000FC9'
MQRCCF_CHANNEL_ALREADY_EXISTS	4042	X'00000FCA'
MQRCCF_DATA_TOO_LARGE	4043	X'00000FCB'
MQRCCF_CHANNEL_NAME_ERROR	4044	X'00000FCC'
MQRCCF_XMIT_Q_NAME_ERROR	4045	X'00000FCD'
MQRCCF_MCA_NAME_ERROR	4047	X'00000FCF'
MQRCCF_SEND_EXIT_NAME_ERROR	4048	X'00000FD0'
MQRCCF_SEC_EXIT_NAME_ERROR	4049	X'00000FD1'
MQRCCF_MSG_EXIT_NAME_ERROR	4050	X'00000FD2'
MQRCCF_RCV_EXIT_NAME_ERROR	4051	X'00000FD3'
MQRCCF_XMIT_Q_NAME_WRONG_TYPE	4052	X'00000FD4'
MQRCCF_MCA_NAME_WRONG_TYPE	4053	X'00000FD5'
MQRCCF_DIST_HATA_TIPI	4054	X'00000FD6'
MQRCCF_SHORT_RETRY_HATA_TIPI	4055	X'00000FD7'
MQRCCF_SHORT_TIMER_HATA_TIPI	4056	X'00000FD8'
MQRCCF_LONG_RETRY_HATA_TIPI	4057	X'00000FD9'
MQRCCF_LONG_TIMER_HATA_TIPI	4058	X'00000FDA'
MQRCCF_PUT_AUTH_HATA_TIPI	4059	X'00000FDB'
MQRCCF_KEEP_ALIV_INT_ERROR	4060	X'00000FDC'
MQRCCF_MISSING_CONN_NAME	4061	X'00000FDD'
MQRCCF_CONN_NAME_ERROR	4062	X'00000FDE'
MQRCCF_MQSET_FAILED	4063	X'00000FDF'
MQRCCF_CHANNEL_NOT_ETKİN	4064	X'00000FE0'

Çizelge 307. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQRCCF_TERMINATED_BY_SEC_EXIT	4065	X'00000FE1'
MQRCCF_DYNAMIC_Q_SCOPE_ERROR	4067	X'00000FE3'
MQRCCF_CELL_DIR_NOT_VAR	4068	X'00000FE4'
MQRCCF_MR_COUNT_ERROR	4069	X'00000FE5'
MQRCCF_MR_COUNT_HATA_TIPI	4070	X'00000FE6'
MQRCCF_MR_EXIT_NAME_ERROR	4071	X'00000FE7'
MQRCCF_MR_EXIT_NAME_WRONG_TYPE	4072	X'00000FE8'
MQRCCF_MR_INTERVAL_ERROR	4073	X'00000FE9'
MQRCCF_MR_INTERVAL_HATA_TIPI	4074	X'00000FEA'
MQRCCF_NPM_SPEED_ERROR	4075	X'00000FEB'
MQRCCF_NPM_SPEED_HATA_TIPI	4076	X'00000FEC'
MQRCCF_HB_INTERVAL_ERROR	4077	X'00000FED'
MQRCCF_HB_INTERVAL_WRONG_TYPE	4078	X'00000FEE'
MQRCCF_CHAD_ERROR	4079	X'00000FEF'
MQRCCF_CHAD_HATA_TIPI	4080	X'00000FF0'
MQRCCF_CHAD_EVENT_ERROR	4081	X'00000FF1'
MQRCCF_CHAD_EVENT_HATA_TIPI	4082	X'00000FF2'
MQRCCF_CHAD_EXIT_ERROR	4083	X'00000FF3'
MQRCCF_CHAD_EXIT_HATA_TIPI	4084	X'00000FF4'
MQRCCF_SUPPRESSED_BY_EXIT	4085	X'00000FF5'
MQRCCF_BATCH_INT_ERROR	4086	X'00000FF6'
MQRCCF_BATCH_INT_HATA_TIPI	4087	X'00000FF7'
MQRCCF_NET_PRIORITY_ERROR	4088	X'00000FF8'
MQRCCF_NET_PRIORITY_HATA_TIPI	4089	X'00000FF9'
MQRCCF_CHANNEL_CLOSED	4090	X'00000FFA'
MQRCCF_Q_STATUS_NOT_FOUND	4091	X'00000FFB'
MQRCCF_SSL_CIPHER_SPEC_ERROR	4092	X'00000FFC'
MQRCCF_SSL_PEER_NAME_ERROR	4093	X'00000FFD'
MQRCCF_SSL_CLIENT_AUTH_ERROR	4094	X'00000FFE'
MQRCCF_RETAINED_NOT_SUPPOSED	4095	X'00000FFF'
 MQRCCF_KWD_VALUE_HATA_TIPI	4096	X'00001000'

### MQRCN\_\* (İstemci Yeniden Bağlan Değişmezleri)

Çizelge 308. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQRCN_NO	0	X'00000000'
MQRCN_YES	1	X'00000001'
MQRCN_Q_MGR	2	X'00000002'

Çizelge 308. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQRCN_DISABLED	3	X'00000003'

### MQRCVTIME\_\* (Alma Zamanasını Tipleri)

Çizelge 309. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQRCVTIME_ÇARP	0	X'00000000'
MQRCVTIME_ADD	1	X'00000001'
MQRCVTIME_EQUAL	2	X'00000002'

### MQREADA\_\* (İleriye Okuma Değerleri)

Çizelge 310. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQREADA_NO	0	X'00000000'
MQREADA_YES	1	X'00000001'
MQREADA_DEVRE DIŞI	2	X'00000002'
MQREADA_ENGELLEYICI	3	X'00000003'
MQREADA_BACKLOG	4	X'00000004'

### MQRECORDING\_\* (Kayıt Seçenekleri)

Çizelge 311. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQRECORDING_DEVRE DIŞI	0	X'00000000'
MQRECORDING_Q	1	X'00000001'
MQRECORDING_MSG	2	X'00000002'

### MQREGO\_\* (Yayınlama/Abone Olma Kayıt Seçenekleri)

Çizelge 312. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQREGO_NONE	0	X'00000000'
MQREGO_CORREL_ID_AS_IDENTITY	1	X'00000001'
MQREGO_ANONYMOUS	2	X'00000002'
MQREGO_LOCAL	4	X'00000004'
MQREGO_DIRECT_REQUIREMS	8	X'00000008'
MQREGO_NEW_YAYINLAYICI	16	X'00000010'
MQREGO_YAYINISI_YALNIZ_ISTEĞİ	32	X'00000020'
MQREGO_DEREGISTIER_ALL	64	X'00000040'
MQREGO_INCLUDE_STREAM_NAME	128	X'00000080'
MQREGO_INFOR_IF_SACTID	256	X'00000100'
MQREGO_DUPLICATES_OK	512	X'00000200'
MQREGO_NON_PERSISTENT	1024	X'00000400'

Çizelge 312. Değişmezlerin değerleri (devamı var)		
Ad	Ondalık değer	Onaltılı değer
MQREGO_PERSISTENT	2048	X'00000800'
MQREGO_PERSISTENT_AS_YAYINLAMA	4096	X'00001000'
MQREGO_PERSISTENT_AS_Q	8192	X'00002000'
MQREGO_ADD_NAME	16384	X'00004000'
MQREGO_NO_ALTERATION	32768	X'00008000'
MQREGO_FULL_RESPONSE	65536	X'00010000'
MQREGO_JOIN_SHARED	131072	X'00020000'
MQREGO_JOIN_EXCLUSIVE	262144	X'00040000'
MQREGO_LEAV_ONLY	524288	X'00080000'
MQREGO_VARIABLE_USER_ID	1048576	X'00100000'
MQREGO_KILITLI	2097152	X'00200000'

## MQRFH\_\* (Kural ve biçimleme üstbilgisi yapısı ve işaretleri)

### Kurallar ve biçimlendirme üstbilgisi yapısı

Çizelge 313. Değişmezlerin yapıları	
Ad	Yapı
MQRFH_STRUC_ID	"RFH↵"
MQRFH_STRUC_ID_ARRAY	'R', 'F', 'H', '↵'

**Not:** ↵ simgesi tek bir boş karakteri temsil eder.

Çizelge 314. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQRFH_VERSION_1	1	X'00000001'
MQRFH_VERSION_2	2	X'00000002'
MQRFH_STRUC_LENGTH_FIXED	32	X'00000020'
MQRFH_STRUC_LENGTH_FIXED_2	36	X'00000024'

### Kurallar ve biçimlendirme üstbilgisi İşaretleri

Çizelge 315. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQRFH_NONE	0	X'00000000'
MQRFH_NO_FLAGS	0	X'00000000'

## MQRFH2\_\* (Yayınlama/Abone Olma Seçenekleri Etiketleri RFH2 En Üst Düzey Klasör Etiketleri)

Çizelge 316. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQRFH2_NAME_VALUE_VERSION	1	X'00000001'



### MQRFH2\_\* (Yayınlama/Abone Olma Seçenekleri Etiket Etiketleri adları)

MQRFH2_PUBSUB_CMD_FOLDER	"psc"
MQRFH2_PUBSUB_RESP_FOLDER	"pscr"
MQRFH2_MSG_CONTENT_FOLDER	"mcd"
MQRFH2_USER_FOLDER	"usr"

### MQRFH2\_\* (Yayınlama/Abone Olma Seçenekleri Etiket XML etiketi adları)

MQRFH2_PUBSUB_CMD_FOLDER_B	"<psc>"
MQRFH2_PUBSUB_CMD_FOLDER_E	"</psc>"
MQRFH2_PUBSUB_RESP_FOLDER_B	"<pscr>"
MQRFH2_PUBSUB_RESP_FOLDER_E	"</pscr>"
MQRFH2_MSG_CONTENT_FOLDER_B	"<mcd>"
MQRFH2_MSG_CONTENT_FOLDER_E	"</mcd>"
MQRFH2_USER_FOLDER_B	"<usr>"
MQRFH2_USER_FOLDER_E	"</usr>"

### MQRL\_\* (Döndürülen Uzunluk)

Çizelge 317. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQRL_UNDEFED	-1	X'FFFFFFFF'

### MQRMH\_\* (Başvuru iletisi üstbilgi yapısı)

Çizelge 318. Değişmezlerin yapıları	
Ad	Yapı
MQRMH_STRUC_ID	"RMH↵"
MQRMH_STRUC_ID_ARRAY	'R', 'M', 'H', '↵'

**Not:** ↵ simgesi tek bir boş karakteri temsil eder.

Çizelge 319. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQRMH_VERSION_1	1	X'00000001'
MQRMH_CURRENT_VERSION	1	X'00000001'

### MQRMHF\_\* (Başvuru iletisi üstbilgi işaretleri)

Çizelge 320. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQRMHF_SON	1	X'00000001'
MQRMHF_NOT_SON	0	X'00000000'

## MQRO\_\* (Rapor Seçenekleri)

Çizelge 321. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQRO_EXCEPTION	16777216	X'01000000'
MQRO_EXCEPTION_WITH_DATA	50331648	X'03000000'
MQRO_EXCEPTION_WITH_FULL_DATA	117440512	X'07000000'
MQRO_EXPIRATION	2097152	X'00200000'
MQRO_EXPIRATION_WITH_DATA	6291456	X'00600000'
MQRO_EXPIRATION_WITH_FULL_DATA	14680064	X'00E00000'
MQRO_COA	256	X'00000100'
MQRO_CO_WITH_DATA	768	X'00000300'
MQRO_COA_WITHL_FULL_DATA	1792	X'00000700'
MQRO_COD	2048	X'00000800'
MQRO_COD_WITH_DATA	6144	X'00001800'
MQRO_COD_WITH_FULL_DATA	14336	X'00003800'
MQRO_PAN	1	X'00000001'
MQRO_NAN	2	X'00000002'
MQRO_ACTIVITY	4	X'00000004'
MQRO_NEW_MSG_ID	0	X'00000000'
MQRO_PASS_MSG_ID	128	X'00000080'
MQRO_COPY_MSG_ID_TO_CORREL_ID	0	X'00000000'
MQRO_PASS_COREL_ID	64	X'00000040'
MQRO_DEAD_LETTER_Q	0	X'00000000'
MQRO_DISCARD_MSG	134217728	X'08000000'
MQRO_PASS_DISCARD_AND_IFADESI	16384	X'00004000'
MQRO_NONE	0	X'00000000'

## MQRO\_\* (Rapor Seçenekleri Mister)

Çizelge 322. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQRO_REJECT_UNSUP_MASK	270270464	X'101C0000'
MQRO_ACCEPT_UNSUP_MASK	-270532353	X'EFE000FF'
MQRO_ACCEPT_UNSUP_IF_XMIT_MASK	261888	X'0003FF00'

## MQROUTE\_\* (Trace-route)

### İzleme-rota Sayısı Üst Sınırı (MQIACF\_MAX\_ACTITRIES)

Çizelge 323. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQROTE_UNLIMITED_ACTIVITIES	0	X'00000000'

## Trace-route Detail (MQIACF\_ROUTE\_DETAIL)

Çizelge 324. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MROUTE_AYRINTISI	2	X'00000002'
MROUTE_DETAIL_MEDIA	8	X'00000008'
MROUTE_AYRINTI_YUKSEK	32	X'00000020'

## Trace-route Forwarding (MQIACF\_ROUTE\_FORWARDING)

Çizelge 325. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MROUTE_FORWARD_ALL	256	X'00000100'
MROUTE_FORWARD_IF_SUPPORTED	512	X'00000200'
MROUTE_FORWARD_REJ_UNSUP_MASK	-65536	X'FFFF0000'

## İzleme Rotası Teslimi (MQIACF\_ROUTE\_DELIHOI)

Çizelge 326. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MROUTE_DELIVER_YES	4096	X'00001000'
MROUTE_DELIVER_NO	8192	X'00002000'
MROUTE_DELIVER_REJ_UNSUP_MASK	-65536	X'FFFF0000'

## Trace-Route Birikimi (MQIACF\_ROUTE\_BIRIKIM)

Çizelge 327. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MROUTE_ACCUMULATE_NONE	65539	X'00010003'
MROUT_ACCUMULATE_IN_MSG	65540	X'00010004'
MROUTE_ACCUMULATE_AND_REPLY	65541	X'00010005'

## MQRP\_ \* (Komut biçimi Değiştirme Seçenekleri)

Çizelge 328. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQRP_YES	1	X'00000001'
MQRP_NO	0	X'00000000'

## MQRQ\_ \* (Komut biçimi Neden Niteleyicileri)

Çizelge 329. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQRQ_CONN_NOT_YETKILI	1	X'00000001'
MQRQ_OPEN_NOT_YETKILI	2	X'00000002'
MQRQ_CLOSE_NOT_YETKILI	3	X'00000003'
MQRQ_CMD_NOT_YETKILI	4	X'00000004'

Çizelge 329. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQRQ_Q_MGR_DURDURULUYOR	5	X'00000005'
MQRQ_Q_MGR_QUIESCING	6	X'00000006'
MQRQ_CHANNEL_STOPPED_OK	7	X'00000007'
MQRQ_CHANNEL_STOPPED_ERROR	8	X'00000008'
MQRQ_CHANNEL_STOPPED_RETRY	9	X'00000009'
MQRQ_CHANNEL_STOPPED_DISABET	10	X'0000000A'
MQRQ_BRIDGE_STOPPED_OK	11	X'0000000B'
MQRQ_BRIDGE_STOPPED_ERROR	12	X'0000000C'
MQRQ_SSL_HANDSHAKE_ERROR	13	X'0000000D'
MQRQ_SSL_CIPHER_SPEC_ERROR	14	X'0000000E'
MQRQ_SSL_CLIENT_AUTH_ERROR	15	X'0000000F'
MQRQ_SSL_PEER_NAME_ERROR	16	X'00000010'
MQRQ_SUB_NOT_YETKILI	17	X'00000011'
MQRQ_SUB_DEST_NOT_YETKILI	18	X'00000012'
MQRQ_SSL_UNKNOWN_REVOCATION	19	X'00000013'
MQRQ_SYS_CONN_NOT_YETKILI	20	X'00000014'
MQRQ_CHANNEL_BLOCKED_ADDRESS	21	X'00000015'
MQRQ_CHANNEL_BLOCKED_USERID	22	X'00000016'
MQRQ_CHANNEL_BLOCKED_NOACCESS	23	X'00000017'
MQRQ_MAX_ACTIVE_KANALS	24	X'00000018'
MQRQ_MAX_KANALS	25	X'00000019'
MQRQ_SVRCONN_INST_LIMIT	26	X'0000001A'
MQRQ_CLIENT_INST_LIMIT!	27	X'0000001B'
MQRQ_CAF_NOT_INSTALLER	28	X'0000001C'
MQRQ_CSP_NOT_YETKILI	29	X'0000001D'
MQRQ_FAILOVER_ALLOWD	30	X'0000001E'
MQRQ_FAILOVER_NOT_IZINLI	31	X'0000001F'
MQRQ_STANBY_ACTIVATED	32	X'00000020'
MQRQ_REPLICA_ASKIDA	33	X'00000021'

### MQRT\_\* (Komut biçimi Yenileme Tipleri)

Çizelge 330. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQRT_YAPILANIŞI	1	X'00000001'
MQRT_EXPIRY	2	X'00000002'
MQRT_NSPROC	3	X'00000003'
MQRT_PROXYSUB	4	X'00000004'

## MQRU\_\* (Yalnızca İstek)

Çizelge 331. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQRU_PUBLISH_ON_REQUEST	1	X'00000001'
MQRU_YAYINLAYICI_	2	X'00000002'

## MQSCA\_\* (TLS İstemcisi Kimlik Doğrulaması)

Çizelge 332. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQSCA_REQUIREM	0	X'00000000'
MQSCA_OPTIONAL	1	X'00000001'

## MQSCO\_\* (TLS yapılandırma seçenekleri)

### TLS yapılandırma seçenekleri yapısı

Çizelge 333. Değişmezlerin yapıları	
Ad	Yapı
MQSCO_STRUC_ID	"SCO~"
MQSCO_STRUC_ID_ARRAY	'S', 'C', 'O', '~'

**Not:** ~ simgesi tek bir boş karakteri temsil eder.

Çizelge 334. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQSCO_VERSION_1	1	X'00000001'
MQSCO_VERSION_2	2	X'00000002'
MQSCO_VERSION_3	3	X'00000003'
MQSCO_VERSION_4	4	X'00000004'
MQSCO_CURRENT_VERSION	4	X'00000004'

**Not:** ~ simgesi tek bir boş karakteri temsil eder.

### TLS yapılandırma seçenekleri Anahtarı İlk Duruma Getirme Sayısı

Çizelge 335. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQSCO_RESET_COUNT_VARSAYILAN	0	X'00000000'

### Komut Biçimi Kuyruk Tanımı Kapsamı

Çizelge 336. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQSCO_Q_MGR	1	X'00000001'
MQSCO_CEL	2	X'00000002'

## MQSCOPE\_\* (Yayınlama kapsamı)

Çizelge 337. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQSCOPE_ALL	0	X'00000000'
MQSCOPE_AS_PARENT	1	X'00000001'
MQSCOPE_QMGR	4	X'00000004'

## MQSCYC\_\* (Güvenlik Senaryosu)

Çizelge 338. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQSCYC_UPPER	0	X'00000000'
MQSCYC_MIXED	1	X'00000001'

## MQSD\_\* (Nesne tanımlayıcı yapısı)

Çizelge 339. Sabit adlar ve yapılar	
Ad	Yapı
MQSD_STRUC_ID	"SD--"
MQSD_STRUC_ID_ARRAY	'S', 'D', '-', '-'

**Not:** - simgesi tek bir boş karakteri temsil eder.

Çizelge 340. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQSD_VERSION_1	1	X'00000001'
MQSD_CURRENT_VERSION	1	X'00000001'

## MQSECITEM\_\* (Komut biçimi Güvenlik Öğeleri)

Çizelge 341. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQSECEĞİN_Tümü	0	X'00000000'
MQSECITIM_MQADMIN	1	X'00000001'
MQSECIMEM_MQNLIST	2	X'00000002'
MQSECITEM_MQPROC	3	X'00000003'
MQSECITEM_MQQUEUE	4	X'00000004'
MQSECITEM_MQCONN	5	X'00000005'
MQSECITEM_MQCMDS	6	X'00000006'
MQSECITEM_MXADMIN	7	X'00000007'
MQSECIMEM_MXNLIST	8	X'00000008'
MQSECITEM_MXPROC	9	X'00000009'
MQSECITEM_MXQUEUE	10	X'0000000A'
MQSECITEM_MXKONUSU	11	X'0000000B'

## MQSECPROK\_\* (Güvenlik Protokolü Tipleri)

Çizelge 342. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQSECPROT_NONE	0	X'00000000'
MQSECPROT_SSLV30	1	X'00000001'
MQSECPROT_TL SV10	2	X'00000002'
MQSECPROT_TL SV12	4	X'00000004'

## MQSECSW\_\* (Komut biçimi Güvenlik Anahtarları ve Anahtar Durumları)

### Komut biçimi Güvenlik Anahtarları

Çizelge 343. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQSECSW_PROCESS	1	X'00000001'
MQSECSW_NAMELIST	2	X'00000002'
MQSECSW_Q	3	X'00000003'
MQSECSW_KONUSU	4	X'00000004'
MQSECSW_CONTEXT	6	X'00000006'
MQSECSW_ALTERNATE_USER	7	X'00000007'
MQSECSW_KOMUTU	8	X'00000008'
MQSECSW_CONNECTION	9	X'00000009'
MQSECSW_SUBSYSTEM	10	X'0000000A'
MQSECSW_COMMAND_RESOURS	11	X'0000000B'
MQSECSW_Q_MGR	15	X'0000000F'
MQSECSW_QSG	16	X'00000010'

### Komut biçimi Güvenlik Anahtarı Durumları

Çizelge 344. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQSECSW_OFF_BULUNDU	21	X'00000015'
MQSECSW_ON_BULUNDU	22	X'00000016'
MQSECSW_OFF_NOT_BULUNDU	23	X'00000017'
MQSECSW_ON_NOT_BULUNDU	24	X'00000018'
MQSECSW_OFF_ERROR	25	X'00000019'
MQSECSW_ON_GEÇERSİZ kılındı	26	X'0000001A'

## MQSECTYPE\_\* (Komut biçimi Güvenlik Tipleri)

Çizelge 345. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQSECTYPE_AUTHSERV	1	X'00000001'
MQSECTYPE_SSL	2	X'00000002'

Çizelge 345. Değişmezlerin değerleri (devamı var)		
Ad	Ondalık değer	Onaltılı değer
MQSECTYPE_CLASSICS	3	X'00000003'

### MQSEG\_\* (Kesimlere ayırma)

Çizelge 346. Değişmez adlar ve değerler	
Ad	Değer
MQSEG_INHIBITED	'↵'
MQSEG_ALLOWED	'A'

**Not:** ↵ simgesi tek bir boş karakteri temsil eder.

### MQSEL\_\* (Özel Seçici Değerleri)

Çizelge 347. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQSEL_ANY_SELECTOR	-30001	X'FFFF8ACF'
MQSEL_ANY_USER_SELECTOR	-30002	X'FFFF8ACE'
MQSEL_ANY_SYSTEM_SELECTOR	-30003	X'FFFF8ACD'
MQSEL_ALL_SELECTORLARI	-30001	X'FFFF8ACF'
MQSEL_ALL_USER_SELECTS	-30002	X'FFFF8ACE'
MQSEL_ALL_SYSTEM_SELECTORS	-30003	X'FFFF8ACD'

### MQSELTYPE\_\* (Seçici Tipleri)

Çizelge 348. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQSELTYPE_NONE	0	X'00000000'
MQSELTYPE_STANDARD	1	X'00000001'
MQSELTYPE_EXTENDED	2	X'00000002'

### MQSID\_\* (Güvenlik Tanıtıcısı)

Çizelge 349. Değişmez adlar ve değerler	
Ad	Değer
MQSID_NONE	X'00...00' (40 boş değer)
MQSID_NONE_DIZISI	'\0', '\0', ... (40 boş değer)

### MQSIDT\_\* (Güvenlik Tanıtıcısı Tipleri)

Çizelge 350. Değişmez adlar ve değerler	
Ad	Onaltılı değer
MQSIDT_NONE	X'00'
MQSIDT_NT_SECURITY_ID	X'01'
MQSIDT_WAS_SECURITY_ID	X'02'



## MQSMPO\_\* (İleti özelliği seçeneklerini ve yapısını ayarla)

### İleti özelliği seçenekleri yapısını ayarla

Çizelge 351. Değişmezlerin yapıları	
Ad	Yapı
MQSMPO_STRUC_ID	"SMPO"
MQSMPO_STRUC_ID_ARRAY	'S', 'M', 'P', 'O'

**Not:** – simgesi tek bir boş karakteri temsil eder.

Çizelge 352. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQSMPO_VERSION_1	1	X'00000001'
MQSMPO_CURRENT_VERSION	1	X'00000001'

### İleti Özelliği Seçeneklerini Ayarla

Çizelge 353. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQSMPO_SET_FIRST	0	X'00000000'
MQSMPPPO_SET_PROP_UNDER_CURSOR	1	X'00000001'
MQSMPPPO_SET_PROP_AFTER_CURSOR	2	X'00000002'
MQSMPO_APPEND_ÖZELLİĞİ	4	X'00000004'
MQSMPPPO_SET_PROP_BEFORE_CURSOR	8	X'00000008'
MQSMPO_NONE	0	X'00000000'

### MQSO\_\* (Abone Olma Seçenekleri)

Çizelge 354. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQSO_NONE	0	X'00000000'
MQSO_NON_DAYANIKLI	0	X'00000000'
MQSO_READ_AHEAD_AS_Q_DEF	0	X'00000000'
MQSO_ALTER	1	X'00000001'
MQSO_CREATE	2	X'00000002'
MQSO_RESUME	4	X'00000004'
MQSO_DAYANIKLI	8	X'00000008'
MQSO_GROUP_SUB	16	X'00000010'
MQSO_MANAGED	32	X'00000020'
MQSO_SET_IDENTITY_CONTEXT	64	X'00000040'
MQSO_FIXED_USERID	256	X'00000100'
MQSO_ANY_USERID	512	X'00000200'
MQSO_PUBLICATIONS_ON_REQUEST	2048	X'00000800'
MQSO_NEW_PUBLICATIONS_ONLY	4096	X'00001000'
MQSO_FAIL_IF QUIESCING	8192	X'00002000'

Çizelge 354. Değişmezlerin değerleri (devamı var)		
Ad	Ondalık değer	Onaltılı değer
MQSO_ALTERNATE_USER_AUTHORITY	262144	X'00040000'
MQSO_WILDCARD_CHAR	1048576	X'00100000'
MQSO_WILDCARD_KONUSU	2097152	X'00200000'
MQSO_SET_COREL_ID	4194304	X'00400000'
MQSO_SCOPE_QMGR	67108864	X'04000000'
MQSO_NO_READ_AHEAD	134217728	X'08000000'
MQSO_READ_AHEAD	268435456	X'10000000'

### MQSP\_\* (Sync Point Availability)

Çizelge 355. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQSP_AVALABILIR	1	X'00000001'
MQSP_NOT_VAR	0	X'00000000'

### V9.2.0 MQSPL\_\* (Güvenlik İlkesi Koruma Seçenekleri)

Çizelge 356. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQSP_PASSSON	0	X'00000000'
MQSPL_REMOVE	1	X'00000001'
MQSPL_AS_POLICY	2	X'00000002'

### MQSQQM\_\* (Paylaşılan Kuyruk Kuyruğu Yöneticisi Adı)

Çizelge 357. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQSQQM_USE	0	X'00000000'
MQSQQM_IGNORE	1	X'00000001'

### MQSR\_\* (İşlem)

Çizelge 358. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQSR_ACTION_YAYINI	1	X'00000001'

### MQSRO\_\* (Abonelik isteği seçenekleri yapısı)

Çizelge 359. Değişmezlerin yapıları	
Ad	Yapı
MQSRO_STRUC_ID	"SRO~"
MQSRO_STRUC_ID_ARRAY	'S', 'R', 'O', '~'

**Not:** ~ simgesi tek bir boş karakteri temsil eder.

Çizelge 360. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQSRO_VERSION_1	1	X'00000001'
MQSRO_CURRENT_VERSION	1	X'00000001'
MQSRO_YOK	0	X'00000000'
MQSRO_FAIL_IF_QUIESCING	8192	X'00002000'

## MQSS\_\* (Bölüm Durumu)

Çizelge 361. Sabit adlar ve yapılar	
Ad	Yapı
MQSS_NOT_A_KESIMI	'-'
MQSS_SEGMENT	'S'
MQSS_LAST_SEGMENT	'L'

**Not:** - simgesi tek bir boş karakteri temsil eder.

## MQSSL\_\* (TLS FIPS Gereksinimleri)

**Not:** AIX, Linux, and Windows işletim sistemlerinde IBM MQ, "IBM Crypto for C" şifreleme modülü aracılığıyla FIPS 140-2 uyumluluğu sağlar. Bu modüle ilişkin sertifika Geçmiş durumuna taşındı. Müşteriler, IBM Crypto for C certificate belgesini görüntüleyip NIST tarafından sağlanan tüm tavsiyeleri dikkate almalıdır. Yeni bir FIPS 140-3 modülü şu anda devam ediyor ve durumu İşlem listesindeki NIST CMVP modüllerinde aranarak görüntülenebilir.

Çizelge 362. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQSSL_FIPS_NO	0	X'00000000'
MQSSL_FIPS_YES	1	X'00000001'

## MQSTAT\_\* (Stat Seçenekleri)

Çizelge 363. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQSTAT_TYPE_ASYNC_ERROR	0	X'00000000'
MQSTAT_TYPE_RECONNECTION	0	X'00000000'
MQSTAT_TYPE_RECONNECTION_ERROR	0	X'00000000'

## MQSTS\_\* (Durum raporlama yapısı yapısı)

Çizelge 364. Değişmezlerin yapıları	
Ad	Yapı
MQSTS_STRUC_ID	"STAT"
MQSTS_STRUC_ID_ARRAY	'S', 'T', 'A', 'T'

**Not:** - simgesi tek bir boş karakteri temsil eder.

Çizelge 365. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQSTS_VERSION_1	1	X'00000001'
MQSTS_CURRENT_VERSION	1	X'00000001'

## MQSUB\_\* (Durabilen abonelikler)

### Sürekli izin verilen abonelikler

Çizelge 366. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQSUB_DURABLE_AS_PARENT	0	X'00000000'
MQSUB_DURABLE_ALLOWD	1	X'00000001'
MQSUB_DURABLE_INHIBITED	2	X'00000002'

### Sürekli abonelikler aralığı

Çizelge 367. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQSUB_DRUBLE_ALL	-1	X'FFFFFFFF'
MQSUB_DRUBLE_YES	1	X'00000001'
MQSUB_DRUBLE_NO	2	X'00000002'

## MQALTTYPE\_\* (Komut biçimi Abonelik Tipleri)

Çizelge 368. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQSUBTYPE_API	1	X'00000001'
MQSUBTYPE_ADMIN	2	X'00000002'
MQSUBTYPE_PROXY	3	X'00000003'
MQSUBTYPE_ALL	-1	X'FFFFFFFF'
MQSUBTYPE_USER	-2	X'FFFFFFFE'

## MQSUS\_\* (Komut biçimi Askıya Alma Durumu)

Çizelge 369. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQSUS_YES	1	X'00000001'
MQSUS_NO	0	X'00000000'

## MQSVC\_\* (Hizmet)

### Hizmet tipleri

Çizelge 370. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQSVC_TYPE_COMMAND	0	X'00000000'

Çizelge 370. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQSVC_TYPE_SERVER	1	X'00000001'

### Hizmet Denetimleri

Çizelge 371. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQSVC_CONTROL_Q_MGR	0	X'00000000'
MQSVC_CONTROL_Q_MGR_START	1	X'00000001'
MQSVC_CONTROL_MANUAL	2	X'00000002'

### Hizmet Durumu

Çizelge 372. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQSVC_STATUS_DURDU	0	X'00000000'
MQSVC_STATUS_STARTING	1	X'00000001'
MQSVC_STATUS_RUNNING	2	X'00000002'
MQSVC_STATUS_DURMA	3	X'00000003'
MQSVC_STATUS_RETRING	4	X'00000004'

### MQSYNCPPOINT\_\* (Pub/Sub geçişine ilişkin komut biçimi Syncpoint değerleri)

Çizelge 373. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQSYNCPPOINT_YES	0	X'00000000'
MQSYNCPPOINT_IFPER	1	X'00000001'

### MQSYSP\_\* (Komut biçimi Sistem Parametre Değerleri)

Çizelge 374. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQSYSP_NO	0	X'00000000'
MQSYSP_YES	1	X'00000001'
MQSYSP_EXTENDED	2	X'00000002'
MQSYSP_TYPE_INITIAL	10	X'0000000A'
MQSYSP_TYPE_SET	11	X'0000000B'
MQSYSP_TYPE_LOG_COPY	12	X'0000000C'
MQSYSP_TYPE_LOG_STATUS	13	X'0000000D'
MQSYSP_TYPE_ARCHIVE_TAPE	14	X'0000000E'
MQSYSP_ALLOC_BLK	20	X'00000014'
MQSYSP_ALLOC_TRK	21	X'00000015'
MQSYSP_ALLOC_CYL	22	X'00000016'

Çizelge 374. Değişmezlerin değerleri (devamı var)		
Ad	Ondalık değer	Onaltılı değer
MQSYSP_STATUS_BUSY	30	X'0000001E'
MQSYSP_STATUS_PREMOUNT	31	X'0000001F'
MQSYSP_STATUS_AVALABILIR	32	X'00000020'
MQSYSP_STATUS_UNKNOWN	33	X'00000021'
MQSYSP_STATUS_ALLOC_ARCHEVE	34	X'00000022'
MQSYSP_STATUS_COPYING_BSDS	35	X'00000023'
MQSYSP_STATUS_COPYING_LOG	36	X'00000024'

## MQTA\_\* (Konu öznitelikleri)

### jokerler

Çizelge 375. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQTA_BLOCK	1	X'00000001'
MQTA_PASPASS	2	X'00000002'

## İzin Verilen Abonelik

Çizelge 376. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQTA_SUB_AS_PARENT	0	X'00000000'
MQTA_SUB_INHIBITED	1	X'00000001'
MQTA_SUB_ALLOWD	2	X'00000002'

## Yetkili Sunucu Alt Yayma

Çizelge 377. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQTA_PROXY_SUB_FORCE	1	X'00000001'
MQTA_PROXY_SUB_FIRSTUSE	2	X'00000002'

## Yayına İzin Verilen

Çizelge 378. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQTA_PUB_AS_PARENT	0	X'00000000'
MQTA_PUB_INHIBITED	1	X'00000001'
MQTA_PUB_ALLOWD	2	X'00000002'

## MQTC\_\* (Tetikleyici Denetimleri)

Çizelge 379. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQTC_OFF	0	X'00000000'

Çizelge 379. Değişmezlerin değerleri (devamı var)		
Ad	Ondalık değer	Onaltılı değer
MQTC_ON	1	X'00000001'

### **MQTCPKEEP\_\* (TCP Keepalive)**

Çizelge 380. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQTCPKEEP_NO	0	X'00000000'
MQTCPKEEP_YES	1	X'00000001'

### **MQTCPSTACK\_\* (TCP Yığın Tipleri)**

Çizelge 381. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQTCPSTACK_SATIR	0	X'00000000'
MQTCPSTACK_MULTIPLE	1	X'00000001'

### **MQTIME\_\* (Komut biçimi zaman birimleri)**

Çizelge 382. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQTIME_UNIT_MINS	0	X'00000000'
MQTIME_UNIT_SECS	1	X'00000001'

### **MQTM\_\* (İleti yapısını tetikle)**

Çizelge 383. Değişmezlerin yapıları	
Ad	Yapı
MQTM_STRUC_ID	"TM~"
MQTM_STRUC_ID_ARRAY	'T', 'M', '~', '~'

**Not:** ~ simgesi tek bir boş karakteri temsil eder.

Çizelge 384. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQTM_VERSION_1	1	X'00000001'
MQTM_CURRENT_VERSION	1	X'00000001'

### **MQTMC\_\* (Tetikleyici ileti karakter biçimi yapısı)**

Çizelge 385. Değişmezlerin yapıları	
Ad	Yapı
MQTMC_STRUC_ID	"TMC~"
MQTMC_STRUC_ID_ARRAY	'T', 'M', 'C', '~'
MQTMC_VERSION_1	"bbb1"
MQTMC_VERSION_2	"bbb2"
MQTMC_CURRENT_VERSION	"bbb2"

Çizelge 385. Değişmezlerin yapıları (devamı var)	
Ad	Yapı
MQTM_VERSION_1_ARRAY	'-', '-', '-', '1'
MQTM_VERSION_2_ARRAY	'-', '-', '-', '2'
MQTM_CURRENT_VERSION_ARRAY	'-', '-', '-', '2'

### MQTOPT\_\* (Konu Tipi)

Çizelge 386. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQTOPT_LOCAL	0	X'00000000'
MQTOPT_CLUSTER	1	X'00000001'
MQTOPT_ALL	2	X'00000002'

### MQTRAXSTR\_\* (Channel Initiator Trace Autostart)

Çizelge 387. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQTRAXSTR_NO	0	X'00000000'
MQTRAXSTR_YES	1	X'00000001'

### MQTSOPE\_\* (Abonelik Kapsamı)

Çizelge 388. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQTSOPE_QMGR	1	X'00000001'
MQTSOPE_ALL	2	X'00000002'

### MQTT\_\* (Tetikleme Tipleri)

Çizelge 389. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQTT_NONE	0	X'00000000'
MQTT_BIRINCI	1	X'00000001'
MQTT_EVERY	2	X'00000002'
MQTT_DERINLIK	3	X'00000003'

### MQTYPE\_\* (Özellik veri tipleri)

Çizelge 390. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQTYPE_AS_SET	0	X'00000000'
MQTYPE_NULL	2	X'00000002'
MQTYPE_BOOLEAN	4	X'00000004'
MQTYPE_BYTE_STRING	8	X'00000008'
MQTYPE_INT8	16	X'00000010'
MQTYPE_INT16	32	X'00000020'



Çizelge 390. Değişmezlerin değerleri (devamı var)		
Ad	Ondalık değer	Onaltılı değer
MQTYPE_INT32	64	X'00000040'
MQTYPE_LONG	64	X'00000040'
MQTYPE_INT64	128	X'00000080'
MQTYPE_FLOAT32	256	X'00000100'
MQTYPE_FLOAT64	512	X'00000200'
MQTYPE_STRING	1024	X'00000400'

### **MQUA\_\* (Yayınlama/Abone Olma Kullanıcı Özniteliği Seçicileri)**

Çizelge 391. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQUA_FIRST	65536	X'00010000'
MQUA_LAST	999999999	X'3B9AC9FF'

### **MQUIDSUPP\_\* (Komut biçimi Kullanıcı Kimliği Desteği)**

Çizelge 392. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQUIDSUPP_NO	0	X'00000000'
MQUIDSUPP_YES	1	X'00000001'

### **MQUNDELIVERED\_\* (Pub/Sub geçişine ilişkin Teslim Edilmemiş Değerlerden Komut Biçimi)**

Çizelge 393. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQUNDELIVERED_NORMAL	0	X'00000000'
MQUNDELIVERED_SAFE	1	X'00000001'
MQUNDELIVERED_ATIN	2	X'00000002'
MQUNDELIVERED_KEEP	3	X'00000003'

### **MQUOWST\_\* (Komut biçimi UOW Durumları)**

Çizelge 394. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQUOWST_NONE	0	X'00000000'
MQUOWST_ACTIVE	1	X'00000001'
MQUOWST_HAZIR	2	X'00000002'
MQUOWST_ÇÖZÜMLENMEMİŞ	3	X'00000003'

### **MQUOWT\_\* (Komut biçimi UOW Tipleri)**

Çizelge 395. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQUOWT_Q_MGR	0	X'00000000'

Çizelge 395. Değişmezlerin değerleri (devamı var)		
Ad	Ondalık değer	Onaltılı değer
MQUOWT_CICS	1	X'00000001'
MPIOT_RRS	2	X'00000002'
MQUOWT_IMS	3	X'00000003'
MQUOWT_XA	4	X'00000004'

### MQUS\_\* (Kuyruk Kullanımları)

Çizelge 396. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQUS_NORMAL	0	X'00000000'
MQUS_ILETIMI	1	X'00000001'

### MQUSAGE\_\* (Komut biçimi Sayfa Kümesi Kullanım Değerleri ve Veri Kümesi Kullanım Değerleri)

#### Komut Biçimi Sayfa Kümesi Kullanım Değerleri

Çizelge 397. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQUSAGE_PS_AVALABILIR	0	X'00000000'
MQUSAGE_PS_TANIMLI	1	X'00000001'
MQUSAGE_PS_OFFLINE	2	X'00000002'
MQUSAGE_PS_NOT_DEFINED	3	X'00000003'
MQUSAGE_PS_SUSPENDED	4	X'00000004'
MQUSAGE_EXPAND_KULLANICI	1	X'00000001'
MQUSAGE_EXPAND_SYSTEM	2	X'00000002'
MQUSAGE_EXPAND_NONE	3	X'00000003'

#### Komut biçimi Veri Kümesi Kullanım Değerleri

Çizelge 398. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQUSAGE_DS_OLDEST_ACTIVE_UOW	10	X'0000000A'
MQUSAGE_DS_OLDEST_PS_RECOVERY	11	X'0000000B'
MQUSAGE_DS_OLDEST_CF_RECOVERY	12	X'0000000C'

### MQVL\_\* (Değer Uzunluğu)

Çizelge 399. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MVQL_NULL_TERMINATED	-1	X'FFFFFFFF'
MQVL_EMPTY_STRING	0	X'00000000'

## MQVU\_\* (Değişken Kullanıcı Kimliği)

Çizelge 400. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQVU_FIXED_USER	1	X'00000001'
MQVU_ANY_USER	2	X'00000002'

## MQWDR\_\* (Küme iş yükü çıkış hedefi kayıt yapısı)

Çizelge 401. Değişmezlerin yapıları	
Ad	Yapı
MQWDR_STRUC_ID	"WDR↵"
MQWDR_STRUC_ID_ARRAY	'W','D','R','↵'

**Not:** ↵ simgesi tek bir boş karakteri temsil eder.

Çizelge 402. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQWDR_VERSION_1	1	X'00000001'
MQWDR_VERSION_2	2	X'00000002'
MQWDR_CURRENT_VERSION	2	X'00000002'
MQWDR_LENGTH_1	124	X'0000007C'
MQWDR_LENGTH_2	136	X'00000088'
MQWDR_CURRENT_LENGTH	136	X'00000088'

## MQWI\_\* (Bekleme Aralığı)

Çizelge 403. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQWI_UNSIKMA	-1	X'FFFFFFFF'

## MQWIH\_\* (İş yükü bilgisi üstbilgi yapısı ve işaretleri)

### İş yükü bilgileri üstbilgi yapısı

Çizelge 404. Değişmezlerin yapıları	
Ad	Yapı
MQWIH_STRUC_ID	"WIH↵"
MQWIH_STRUC_ID_ARRAY	'W','I','H','↵'

**Not:** ↵ simgesi tek bir boş karakteri temsil eder.

Çizelge 405. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQWIH_VERSION_1	1	X'00000001'
MQWIH_CURRENT_VERSION	1	X'00000001'
MQWIH_LENGTH_1	120	X'00000078'
MQWIH_CURRENT_LENGTH	120	X'00000078'

## İş yükü bilgileri üstbilgi işaretleri

Çizelge 406. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQWIH_NONE	0	X'00000000'

## MQWQR\_\* (Küme iş yükü çıkış kuyruğu kayıt yapısı)

Çizelge 407. Değişmezlerin yapıları	
Ad	Yapı
MQWQR_STRUC_ID	"WQR↯"
MQWQR_STRUC_ID_ARRAY	'W', 'Q', 'R', '↯'

**Not:** ↯ simgesi tek bir boş karakteri temsil eder.

Çizelge 408. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQWQR_VERSION_1	1	X'00000001'
MQWQR_VERSION_2	2	X'00000002'
MQWQR_VERSION_3	3	X'00000003'
MQWQR_CURRENT_VERSION	3	X'00000003'
MQWQR_LENGTH_1	200	X'000000C8'
MQWQR_LENGTH_2	208	X'000000D0'
MQWQR_LENGTH_3	212	X'000000D4'
MQWQR_CURRENT_LENGTH	212	X'000000D4'

## MQWS\_\* (Joker Şeması)

Çizelge 409. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQWS_DEFAULT	0	X'00000000'
MQWS_CHAR	1	X'00000001'
MQWS_KONUSU	2	X'00000002'

## MQWXP\_\* (Küme iş yükü çıkış değiştirgesi yapısı)

## MQWXP\_\* (Küme iş yükü çıkış değiştirgesi yapısı)

Çizelge 410. Değişmezlerin yapıları	
Ad	Yapı
MQWXP_STRUC_ID	"WXP↯"
MQWXP_STRUC_ID_ARRAY	'W', 'X', 'P', '↯'

**Not:** ↯ simgesi tek bir boş karakteri temsil eder.

Çizelge 411. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQWXP_VERSION_1	1	X'00000001'

Çizelge 411. Değişmezlerin değerleri (devamı var)

Ad	Ondalık değer	Onaltılı değer
MQWXP_VERSION_2	2	X'00000002'
MQWXP_VERSION_3	3	X'00000003'
MQWXP_VERSION_4	4	X'00000004'
MQWXP_CURRENT_VERSION	4	X'00000004'

### MQWXP\_\* (Küme İş Yüğü İşaretleri)

Çizelge 412. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQWXP_PUT_BY_CLUSTER_CHL	2	X'00000002'

#### İlgili başvurular

“MQWXP içindeki alanlar-Küme iş yüğü çıkış değıştirgesi yapısı” sayfa 1518  
MQWXP -Cluster iş yüğü çıkış değıştirgesi yapısındaki alanların tanımı

### MQXACT\_\* (API Caller Tipleri)

Çizelge 413. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQXACT_EXTERNAL	1	X'00000001'
MQXACT_INTERNAL	2	X'00000002'

### MQXC\_\* (Çıkış Komutları)

Çizelge 414. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQXC_MQAC	1	X'00000001'
MQXC_MQCLOSE	2	X'00000002'
MQXC_MQGET	3	X'00000003'
MQXC_MQPUT	4	X'00000004'
MQXC_MQPUT1	5	X'00000005'
MQXC_MQINQ	6	X'00000006'
MQXC_MQSET	8	X'00000008'
MQXC_MQI	9	X'00000009'
MQXC_MQCMIT	10	X'0000000A'

### MQXCC\_\* (Çıkış Yanıtları)

Çizelge 415. Değişmezlerin değerleri

Ad	Ondalık değer	Onaltılı değer
MQXCC_OK	0	X'00000000'
MQXCC_SUPPRESS_FUNCTION	-1	X'FFFFFFFF'
MQXCC_SKIP_FUNCTION	-2	X'FFFFFFFE'
MQXCC_SEND_AND_REQUEST_SEC_MSG	-3	X'FFFFFFFD'
MQXCC_SEND_SEC_MSG	-4	X'FFFFFFFC'

Çizelge 415. Değişmezlerin değerleri (devamı var)		
Ad	Ondalık değer	Onaltılı değer
MQXCC_SUPRESS_EXIT	-5	X'FFFFFFFFB'
MQXCC_CLOSE_CHANNEL	-6	X'FFFFFFFA'
MQXCC_REQUEST_ACK	-7	X'FFFFFFF9'
MQXCC_FAILED	-8	X'FFFFFFF8'

### MQXDR\_\* (Çıkış Yanıtı)

Çizelge 416. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQXDR_OK	0	X'00000000'
MQXDR_CONVERSION_FAILED	1	X'00000001'

### MQXE\_\* (Ortamlar)

Çizelge 417. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQXE_DİĞER	0	X'00000000'
MQXE_MCA	1	X'00000001'
MQXE_MCA_SVRCONN	2	X'00000002'
MQXE_COMMAND_SERVER	3	X'00000003'
MQXE_MQSC	4	X'00000004'

### MQXEPO\_\* (Giriş Noktası Seçenekleri yapısını ve Çıkış Seçeneklerini Kaydet)

#### Giriş Noktası Seçenekleri Yapısını Kaydet

Çizelge 418. Değişmezlerin yapıları	
Ad	Yapı
MQXEPO_STRUCT_ID	"XEPO"
MQXEPO_STRUC_ID_ARRAY	'X', 'E', 'P', 'O'

**Not:** – simgesi tek bir boş karakteri temsil eder.

Çizelge 419. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQXEPO_VERSION_1	1	X'00000001'
MQXEPO_CURRENT_VERSION	1	X'00000001'

#### Çıkış Seçenekleri

Çizelge 420. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQXEPO_NONE	0	X'00000000'

## MQXF\_\* (API İşlev Tanıtıcıları)

Çizelge 421. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQXF_INIT	1	X'00000001'
MQXF_TERM	2	X'00000002'
MQXF_CONN	3	X'00000003'
MQXF_CONNX	4	X'00000004'
MQXF_DISC	5	X'00000005'
MQXF_Aç	6	X'00000006'
MQXF_CLOSE	7	X'00000007'
MQXF_PUT1	8	X'00000008'
MQXF_PUT	9	X'00000009'
MQXF_GET	10	X'0000000A'
MQXF_DATA_CONV_ON_GET	11	X'0000000B'
MQXF_INQ	12	X'0000000C'
MQXF_SET	13	X'0000000D'
MQXF_BEGIN	14	X'0000000E'
MQXF_CMIT	15	X'0000000F'
MQXF_BACK	16	X'00000010'
MQXF_STAT	18	X'00000012'
MQXF_CB	19	X'00000013'
MQXF_CTL	20	X'00000014'
MQXF_CALLIK	21	X'00000015'
MQXF_SUB	22	X'00000016'
MQXF_SUBRQ	23	X'00000017'
MQXF_XACLOSE	24	X'00000018'
MQXF_XACOMMIT	25	X'00000019'
MQXF_XCOMPLETE	26	X'0000001A'
MQXF_XAEND	27	X'0000001B'
MQXF_XAFORMA	28	X'0000001C'
MQXF_XAAç	29	X'0000001D'
MQXF_XAPREPARE	30	X'0000001E'
MQXF_XARECOVER	31	X'0000001F'
MQXF_XARLORARI	32	X'00000020'
MQXF_XASTART	33	X'00000021'
MQXF_AXREG	34	X'00000022'
MQXF_AXUNREG	35	X'00000023'

## MQXP\_\* (API geiş ıkıř parametresi yapısı)

izelge 422. Deęiřmezlerin yapıları	
Ad	Yapı
MQXP_STRUC_ID	"XP↵"
MQXP_STRUC_ID_ARRAY	'X', 'P', '↵', '↵'

**Not:** ↵ simgesi tek bir boş karakteri temsil eder.

izelge 423. Deęiřmezlerin deęerleri		
Ad	Ondalık deęer	Onaltılı deęer
MQXP_VERSION_1	1	X'00000001'

## MQXPDA\_\* (Sorun Saptama Alanı)

izelge 424. Deęiřmez adları ve deęerler	
Ad	Deęer
MQXPDA_NONE	X'00...00' (48 boş deęer)
MQXPDA_NONE_DIZISI	'\0', '\0', ... (48 boş deęer)

## MQXPT\_\* (İletim Tipleri)

izelge 425. Deęiřmezlerin deęerleri		
Ad	Ondalık deęer	Onaltılı deęer
MQXPT_ALL	-1	X'FFFFFFFF'
MQXPT_LOCAL	0	X'00000000'
MQXPT_LU62	1	X'00000001'
MQXPT_TCP	2	X'00000002'
MQXPT_NETBIOS	3	X'00000003'
MQXPT_SPX	4	X'00000004'
MQXPT_DEFRET	5	X'00000005'
MQXPT_UDP	6	X'00000006'

## MQXQH\_\* (İletim kuyruęu üstbilgi yapısı)

izelge 426. Deęiřmezlerin yapıları	
Ad	Yapı
MQXQH_STRUC_ID	"XQH↵"
MQXQH_STRUC_ID_ARRAY	'X', 'Q', 'H', '↵'

**Not:** ↵ simgesi tek bir boş karakteri temsil eder.

izelge 427. Deęiřmezlerin deęerleri		
Ad	Ondalık deęer	Onaltılı deęer
MQXQH_VERSION_1	1	X'00000001'
MQXQH_CURRENT_VERSION	1	X'00000001'



## MQXR\_\* (Exit Reasons)

Çizelge 428. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQXR_BEDEN	1	X'00000001'
MQXR_AFTER	2	X'00000002'
MQXR_BAĞLANTI	3	X'00000003'
MQXR_INIT	11	X'0000000B'
MQXR_TERM	12	X'0000000C'
MQXR_MSG	13	X'0000000D'
MQXR_XMIT	14	X'0000000E'
MQXR_SEC_MSG	15	X'0000000F'
MQXR_INIT_SEC	16	X'00000010'
MQXR_RETRY	17	X'00000011'
MQXR_AUTO_CLUSSDR	18	X'00000012'
MQXR_AUTO_RECEIVER	19	X'00000013'
MQXR_CLWL_ASI	20	X'00000014'
MQXR_CLWL_PUT	21	X'00000015'
MQXR_CLWL_MOVE	22	X'00000016'
MQXR_CLWL_REPOS	23	X'00000017'
MQXR_CLWL_REPOS_MOVE	24	X'00000018'
MQXREND_BATCH	25	X'00000019'
MQXR_ACK_RECEIVED	26	X'0000001A'
MQXR_AUTO_SVRCONN	27	X'0000001B'
MQXR_AUTO_CLUSRCVR	28	X'0000001C'
MQXR_SEC_PARS	29	X'0000001D'

## MQXR2\_\* (Çıkış Yanıt 2)

Çizelge 429. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQXR2_PUT_WITH_DEF_ACTION	0	X'00000000'
MQXR2_PUT_WITH_DEF_USERID	1	X'00000001'
MQXR2_PUT_WITH_MSG_USERID	2	X'00000002'
MQXR2_USE_AGENT_BUFFER	0	X'00000000'
MQXR2_USE_EXIT_BUFFER	4	X'00000004'
MQXR2_DEFAULT_CONTINUATION	0	X'00000000'
MQXR2_CONTINUE_CHAIN	8	X'00000008'
MQXR2_SUPPRESS_CHAIN	16	X'00000010'
MQXR2_STATIC_CACHE	0	X'00000000'
MQXR2_DYNAMIC_CACHE	32	X'00000020'

## MQXT\_\* (Çıkış Tanıtıcıları)

Çizelge 430. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQXT_API_CROSSING_EXIT	1	X'00000001'
MQXT_API_EXIT	2	X'00000002'
MQXT_CHANNEL_SEC_EXIT	11	X'0000000B'
MQXT_CHANNEL_MSG_EXIT	12	X'0000000C'
MQXT_CHANNEL_SEND_EXIT	13	X'0000000D'
MQXT_CHANNEL_RCV_EXIT	14	X'0000000E'
MQXT_CHANNEL_MSG_RETRY_EXIT	15	X'0000000F'
MQXT_CHANNEL_AUTO_DEF_EXIT	16	X'00000010'
MQXT_CLUSTER_WORKLOAD_EXIT	20	X'00000014'
MQXT_PUBSUB_ROUTING_EXIT	21	X'00000015'

## MQXUA\_\* (Kullanıcı Alanı Değerinden Çık)

Çizelge 431. Değişmez adlar ve değerler	
Ad	Değer
MQXUA_NONE	X'00...00' (16 boş değer)
MQXUA_NONE_DIZISI	'\0', '\0', ... (16 boş değer)

## MQXWD\_\* (Çıkış bekleme tanımlayıcı yapısı)

Çizelge 432. Değişmezlerin yapıları	
Ad	Yapı
MQXWD_STRUC_ID	"XWD~"
MQXWD_STRUC_ID_ARRAY	'X', 'W', 'D', '~'

**Not:** ~ simgesi tek bir boş karakteri temsil eder.

Çizelge 433. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQXWD_VERSION_1	1	X'00000001'

## MQZAC\_\* (Uygulama bağlamı yapısı)

Çizelge 434. Değişmezlerin yapıları	
Ad	Yapı
MQZAC_STRUC_ID	"ZAC~"
MQZAC_STRUC_ID_ARRAY	'Z', 'A', 'C', '~'

**Not:** ~ simgesi tek bir boş karakteri temsil eder.

Çizelge 435. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQZAC_VERSION_1	1	X'00000001'
MQZAC_CURRENT_VERSION	1	X'00000001'

## MQZAD\_\* (Yetki verileri yapısı)

Çizelge 436. Değişmezlerin yapıları	
Ad	Yapı
MQZAD_STRUC_ID	"ZAD~"
MQZAD_STRUC_ID_ARRAY	'Z', 'A', 'D', '~'

**Not:** ~ simgesi tek bir boş karakteri temsil eder.

Çizelge 437. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQZAD_VERSION_1	1	X'00000001'
MQZAD_VERSION_2	2	X'00000002'
MQZAD_CURRENT_VERSION	2	X'00000002'

## MQZAET\_\* (Kurulabilir Hizmetler Varlık Tipleri)

Çizelge 438. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQZAET_NONE	0	X'00000000'
MQZAET_PRINCIPAL	1	X'00000001'
MQZAET_GRUP	2	X'00000002'
MQZAET_UNKNOWN	3	X'00000003'

## MQZAO\_\* (Kurulabilir Hizmetler Yetkilendirmeleri)

Çizelge 439. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQZAO_CONNECT	1	X'00000001'
MQZAO_GÖZAT	2	X'00000002'
MQZAO_INPUT	4	X'00000004'
MQZAO_OUTPUT	8	X'00000008'
MQZAO_SORGULAMA	16	X'00000010'
MQZAO_SET	32	X'00000020'
MQZAO_PASS_IDENTITY_CONTEXT	64	X'00000040'
MQZAO_PASS_ALL_CONTEXT	128	X'00000080'
MQZAO_SET_IDENTITY_CONTEXT	256	X'00000100'
MQZAO_SET_ALL_CONTEXT	512	X'00000200'
MQZAO_ALTERNATE_USER_AUTHORITY	1024	X'00000400'
MQZAO_PUBLISH	2048	X'00000800'
MQZAO_SUBSCREEN	4096	X'00001000'
MQZAO_RESUME	8192	X'00002000'
MQZAO_ALL_MQI	16383	X'00003FFF'
MQZAO_CREATE	65536	X'00010000'
MQZAO_DELETE	131072	X'00020000'
MQZAO_GÖRÜNTÜLE	262144	X'00040000'

Çizelge 439. Değişmezlerin değerleri (devamı var)		
Ad	Ondalık değer	Onaltılı değer
MQZAO_CHANGE	524288	X'00080000'
MQZAO_CLEAR	1048576	X'00100000'
MQZAO_CONTROL	2097152	X'00200000'
MQZAO_CONTROL_EXTENDED	4194304	X'00400000'
MQZAO_YETKILIVER	8388608	X'00800000'
MQZAO_ALL_ADMIN	16646144	X'00FE0000'
MQZAO_ALL	16662527	X'00FE3FFF'
MQZAO_REMOVE	16777216	X'01000000'
MQZAO_NONE	0	X'00000000'

### MQZAS\_\* (Kurulabilir Hizmetler Hizmet Arabirimi Sürümü)

Çizelge 440. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQZAS_VERSION_1	1	X'00000001'
MQZAS_VERSION_2	2	X'00000002'
MQZAS_VERSION_3	3	X'00000003'
MQZAS_VERSION_4	4	X'00000004'
MQZAS_VERSION_5	5	X'00000005'
MQZAS_VERSION_6	6	X'00000006'

### MQZAT\_\* (Kimlik Doğrulama Tipleri)

Çizelge 441. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQZAT_INITIAL_CONTEXT	0	X'00000000'
MQZAT_CHANGE_CONTEXT	1	X'00000001'

### MQZCI\_\* (Kurulabilir Hizmetler Devamı Göstergesi)

Çizelge 442. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQZCI_VARSAYILAN	0	X'00000000'
MQZCI_CONTINY	0	X'00000000'
MQZCI_STOP	1	X'00000001'

### MQZED\_\* (Varlık veri yapısı)

Çizelge 443. Değişmezlerin yapıları	
Ad	Yapı
MQZED_STRUC_ID	"ZED~"
MQZED_STRUC_ID_ARRAY	'Z', 'E', 'D', '~'

**Not:** ~ simgesi tek bir boş karakteri temsil eder.

Çizelge 444. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQZED_VERSION_1	1	X'00000001'
MQZED_VERSION_2	2	X'00000002'
MQZED_CURRENT_VERSION	2	X'00000002'

### MQZFP\_\* (Serbest parametreler yapısı)

Çizelge 445. Değişmezlerin yapıları	
Ad	Yapı
MQZFP_STRUC_ID	"ZFP↵"
MQZFP_STRUC_ID_ARRAY	'Z','F','P','↵'

**Not:** ↵ simgesi tek bir boş karakteri temsil eder.

Çizelge 446. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQZFP_VERSION_1	1	X'00000001'
MQZFP_CURRENT_VERSION	1	X'00000001'

### MQZIC\_\* (Kimlik bağlamı yapısı)

Çizelge 447. Değişmezlerin yapıları	
Ad	Yapı
MQZIC_STRUC_ID	"ZIC↵"
MQZIC_STRUC_ID_ARRAY	'Z','I','C','↵'

**Not:** ↵ simgesi tek bir boş karakteri temsil eder.

Çizelge 448. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQZIC_VERSION_1	1	X'00000001'
MQZIC_CURRENT_VERSION	1	X'00000001'

### MQZID\_\* (hizmetlere ilişkin işlev tanıtıcıları)

#### Tüm hizmetler için ortak olan işlev tanıtıcıları

Çizelge 449. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQZID_INIT	0	X'00000000'
MQZID_TERM	1	X'00000001'

#### Yetki hizmeti için işlev tanıtıcıları

Çizelge 450. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQZID_INIT_AUTHORITY	0	X'00000000'

Çizelge 450. Değişmezlerin değerleri (devamı var)		
Ad	Ondalık değer	Onaltılı değer
MQZID_TERM_AUTHORITY	1	X'00000001'
MQZID_CHECK_AUTHORITY	2	X'00000002'
MQZID_COPY_ALL_AUTHORITY	3	X'00000003'
MQZID_DELETE_AUTHORITY	4	X'00000004'
MQZID_SET_AUTHORITY	5	X'00000005'
MQZID_GET_AUTHORITY	6	X'00000006'
MQZID_GET_AÇIK_YETKI_YETKISI	7	X'00000007'
MQZID_REFRESH_CACHE	8	X'00000008'
MQZID_ENUMERATE_AUTHORITY_DATA	9	X'00000009'
MQZID_AUTHENTICATE_USER	10	X'0000000A'
MQZID_FREE_USER	11	X'0000000B'
MQZID_SORGULAMA	12	X'0000000C'
MQZID_CHECK_PRIVATIONAL	13	X'0000000D'

### Ad hizmeti için işlev tanıtıcıları

Çizelge 451. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQZID_INIT_NAME	0	X'00000000'
MQZID_TERM_ADı	1	X'00000001'
MQZID_LOOKUP_NAME	2	X'00000002'
MQZID_INSERT_NAME	3	X'00000003'
MQZID_DELETE_ADı	4	X'00000004'

### Userid hizmetine ilişkin işlev tanıtıcıları

Çizelge 452. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQZID_INIT_USERID	0	X'00000000'
MQZID_TERM_USERID	1	X'00000001'
MQZID_FIND_USERID	2	X'00000002'

### MQZIO\_\* (Kurulabilir Hizmetler Kullanıma Hazırlama Seçenekleri)

Çizelge 453. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQZI_BIRINCIL	0	X'00000000'
MQZIO_IKINCIL	1	X'00000001'

## MQZNS\_\* (Ad Hizmet Arabirimi Sürümü)

Çizelge 454. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQZNS_VERSION_1	1	X'00000001'

## MQZSE\_\* (Kurulabilir Hizmetler Başlangıcı-Sıralama Göstergesi)

Çizelge 455. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQZSE_START	1	X'00000001'
MQZSE_DEVAM	0	X'00000000'

## MQZSL\_\* (Kurulabilir Hizmetler Seçici Göstergesi)

Çizelge 456. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQZSL_NOT_RETURNND	0	X'00000000'
MQZSL_RETURNED	1	X'00000001'

## MQZTO\_\* (Kurulabilir Hizmetler Sonlandırma Seçenekleri)

Çizelge 457. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQZTO_BIRINCIL	0	X'00000000'
MQZTO_IKINCIL	1	X'00000001'

## MQZUS\_\* (Kullanıcı Kimliği Hizmet Arabirimi Sürümü)

Çizelge 458. Değişmezlerin değerleri		
Ad	Ondalık değer	Onaltılı değer
MQZUS_VERSION_1	1	X'00000001'

## MQI ' da kullanılan veri tipleri

Message Queue Interface (MQI) olanağında kullanılabilecek veri tipleriyle ilgili bilgi. Her veri tipi ile ilgili dillere ilişkin açıklamalar, alanlar ve dil bildirimleri.

### MQI için veri tipleri ve programlama

Temel ve Yapı veri tiplerinin tanıtılması ve C programlama, COBOL programlama ya da High Level Assembler programlaması aracılığıyla MQI 'ın nasıl kullanılacağı.

#### Temel veri tipleri

Bu kısım, MQI ' de (ya da çıkış işlevlerinde) kullanılan veri tipleriyle ilgili bilgileri içerir. Bu bilgiler, aşağıdaki konularda desteklenen programlama dillerinde temel veri tiplerinin nasıl bildirileceğini gösteren örneklerle birlikte ayrıntılı olarak açıklanmıştır.

MQI ' da (ya da çıkış işlevlerinde) kullanılan veri tipleri şunlardır:

- Temel veri tipleri ya da
- Temel veri türlerinin toplamaları (diziler veya yapılar)

İzleyen temel veri tipleri, MQI ' de (ya da çıkış işlevlerinde) kullanılır:

Temel veri tipi adı	Veri tipi	Tanım
MQBOOL	Boole	MQBOOL veri tipi bir boole değeri temsil eder. 0 değeri false değerini gösterir. Başka bir değer, true değerini gösterir. MQHOOOL, MQHOT veri tipi için aynı şekilde hizalanmalıdır.
MQBYTE	Byte	<p>MQBYTE veri tipi, tek bir byte veri tipini gösterir. Byte 'a belirli bir yorum konmaz; ikili bir sayı ya da karakter olarak değil, bit dizgisi olarak kabul edilir. Özel hizalama gerekmez.</p> <p>Farklı karakter kümeleri ya da kodlamalar kullanan kuyruk yöneticileri arasında MQBYTE verileri gönderildiğinde, MQBYTE verileri herhangi bir şekilde dönüştürülmez. MQMD yapısındaki <i>MsgId</i> ve <i>CorrelId</i> alanları şöyle olur.</p> <p>Bazen, kuyruk yöneticisi tarafından tanınmayan bir ana saklama alanı bölgesini göstermek için kullanılan bir MQByte dizisi kullanılır. Örneğin, alan, uygulama iletisi verileri ya da bir yapı içerebilir. Bu alanın sınır hizalaması, içinde bulunan verilerin nitesiyle uyumlu olmalıdır.</p> <p>C programlama dilinde, herhangi bir veri tipi, MQBYTE dizisi olarak gösterilen işlev parametreleri için kullanılabilir. Bunun nedeni, parametrelerin her zaman adresle geçirildiği ve C ' de işlev parametresinin bir işaretçi olarak bildirilmiş olması.</p>



Çizelge 459. Temel veri tipi adları, tipleri ve açıklamaları (devamı var)

Temel veri tipi adı	Veri tipi	Tanım
MQBYTEN	$n$ bayta ilişkin dizgi	<p>Her MQBYTEN veri tipi <math>n</math> byte dizgisini gösterir; burada <math>n</math>, şu değerlerden herhangi birini alabilir: 8, 16, 24, 32, 40 ya da 128. Her bayt, MQBYTE veri tipi tarafından tanımlanır. Özel hizalama gerekmez.</p> <p>Bayt dizilimindeki veriler, dizilimin tanımlı uzunluğundan daha kısaysa, dizgiyi doldurmak için veriler boş değerlerle doldurulmalıdır.</p> <p>Kuyruk yöneticisi uygulamaya bayt dizgileri döndürdüğünde (örneğin, MQGET çağrısında), kuyruk yöneticisi altlıkları, dizginin tanımlı uzunluğuna kadar boş değere sahip olur.</p> <p>Bayt dizgi alanlarının uzunluklarını tanımlamak için, adlandırılan sabitler kullanılabilir. Bunlar <a href="#">“Değişmezler” sayfa 61</a> içinde listelenir.</p>
MQCHAR	Karakter	<p>MQCHAR veri tipi, tek byte 'lık bir karakteri ya da çift baytlık ya da çok baytlı karakterlerden oluşan bir byte 'ı temsil eder. Özel hizalama gerekmez.</p> <p>Farklı karakter kümeleri ya da kodlamalar kullanan kuyruk yöneticileri arasında MQCHAR verileri gönderildiğinde, verilerin doğru olarak yorumlanabilmek için MQCHAR verileri genellikle dönüştürmeyi gerektirir. Kuyruk yöneticisi, MQMD yapısındaki MQCHAR verileri için bunu otomatik olarak yapar. Uygulama iletisi verilerinde MQCHAR verilerinin dönüştürülmesi, MQGET çağrısında MQGMO_CONVERT seçeneği tarafından denetlenir; ek ayrıntılar için <a href="#">“MQGMO-Get-message seçenekleri” sayfa 364</a> 'ta bu seçeneğin tanımına bakın.</p>

Çizelge 459. Temel veri tipi adları, tipleri ve açıklamaları (devamı var)

Temel veri tipi adı	Veri tipi	Tanım
MQCHARn	$n$ karakter dizgisi	<p>Her MQCHARn veri tipi bir <math>n</math> karakteri dizesini gösterir; burada <math>n</math>, şu değerlerden herhangi birini alabilir: 4, 8, 12, 20, 28, 32, 48, 64, 128 ya da 256. Her karakter MQCHAR veri tipi tarafından tanımlanır. Özel hizalama gerekmez.</p> <p>Dizgideki veriler, dizilimin tanımlı uzunluğundan daha kısaysa, dizgiyi doldurmak için verilerin boşluklarla doldurulması gerekir. Bazı durumlarda, boş bir karakter, boşlukları doldurmak için boşluklara; boşluk karakteri ve izleyen karakterler boşluk olarak, dizginin tanımlı uzunluğuna kadar boşluk olarak kabul edilir. Arama ve veri tipi tanımlamalarında boş değer kullanılabileceği yerler.</p> <p>Kuyruk yöneticisi uygulamaya karakter dizgileri döndürdüğünde (örneğin, MQGET çağrısında), kuyruk yöneticisi her zaman, dizginin tanımlı uzunluğuna boşluk olarak boşluk yapar; kuyruk yöneticisi, dizgiyi sınırlamak için boş değerli karakteri kullanmaz.</p> <p>Karakter dizgisi alanlarının uzunluklarını tanımlayan ve "Değişmezler" sayfa 61'inde yer alan, adlandırılmış sabitler kullanılabilir.</p>
MQFLOAT32	32 bitlik kayan noktalı sayı	<p>MQFLOAT32 veri tipi, standart IEEE kayar noktalı biçim kullanılarak gösterilen, 32 bitlik bir kayan noktalı sayıdır. Bir MQFLOAT32, 4 baytlık bir sınır üzerinde hizalanmalıdır.</p> <p>The use of MQFLOAT32 in C on z/OS requires the use of the FLOAT(IEEE) compiler flag.</p> <p>COBOL 'de MQFLOAT32 kullanımı, IEEE biçimindeki kayan noktalı sayıları destekleyen derleyiciler ile sınırlandırılır. Bu, FLOAT (YEREL) derleyici işaretinin kullanılmasını gerektirebilir.</p>

Çizelge 459. Temel veri tipi adları, tipleri ve açıklamaları (devamı var)

Temel veri tipi adı	Veri tipi	Tanım
MQFLOAT64	64 bitlik kayan noktalı sayı	<p>MQFLOAT64 veri tipi, standart IEEE kayar noktalı biçim kullanılarak gösterilen, 64 bitlik kayan noktalı bir sayıdır. Bir MQFLOAT64 , 8 baytlık bir sınır üzerinde hizalanmalıdır.</p> <p>The use of MQFLOAT64 in C on z/OS requires the use of the FLOAT(IEEE) compiler flag.</p> <p>COBOL ' de MQFLOAT64 kullanımı, IEEE formatındaki kayan noktalı sayıları destekleyen derleyiciyle sınırlıdır. Bu, FLOAT (YEREL) derleyici işaretinin kullanılmasını gerektirebilir.</p>
MQHCONFIG	Yapılandırma tutamacı	<p>MQHCONFIG veri tipi, belirli bir kurulabilir hizmet için yapılandırılmakta olan bir yapılanış tanıtıcısını (yani, bileşen) gösterir. Bir yapılandırma tanıtıcısı, doğal sınırı üzerinde hizalanmalıdır.</p> <p>Uygulamaların bu tanıtıcı içinde saklanan verilerin biçimine dayanmamaları gerekir. Geçerli olması durumunda, değerinin daha sonraki MQI çağrılarında kullanılabilir olması amaçlanır, ancak bu amaç dışında herhangi bir anlama sahip olmak amaçlanmaz.</p>
MQHCONN	Bağlantı tanıtıcı	<p>MQHCONN veri tipi bir bağlantı tanıtıcısını, yani belirli bir kuyruk yöneticisiyle olan bağlantıyı gösterir. Bir bağlantı tanıtıcısı, 4 baytlık bir sınır üzerinde hizalanmalıdır.</p> <p>Uygulamaların bu tanıtıcı içinde saklanan verilerin biçimine dayanmamaları gerekir. Geçerli olması durumunda, değerinin daha sonraki MQI çağrılarında kullanılabilir olması amaçlanır, ancak bu amaç dışında herhangi bir anlama sahip olmak amaçlanmaz.</p>

Çizelge 459. Temel veri tipi adları, tipleri ve açıklamaları (devamı var)

Temel veri tipi adı	Veri tipi	Tanım
MQHMSG	İleti tanıtıcısı	<p>MQHMSG veri tipi, bir iletiye erişim veren bir ileti tanıtıcısını gösterir. Bir ileti tanıtıcısı, 8 byte 'lık bir sınırdaki hizalanmalıdır.</p> <p>Uygulamaların bu tanıtıcı içinde saklanan verilerin biçimine dayanmamaları gerekir. Geçerli olması durumunda, değerinin daha sonraki MQI çağrılarında kullanılabilir olması amaçlanır, ancak bu amaç dışında herhangi bir anlama sahip olmak amaçlanmaz.</p>
MQHOBJ	Nesne tanıtıcısı	<p>MQHOBJ veri tipi, bir nesneye erişim veren bir nesne tanıtıcısını gösterir. Bir nesne tanıtıcısı, 4 baytlık bir sınır üzerinde hizalanmalıdır.</p> <p>Uygulamaların bu tanıtıcı içinde saklanan verilerin biçimine dayanmamaları gerekir. Geçerli olması durumunda, değerinin daha sonraki MQI çağrılarında kullanılabilir olması amaçlanır, ancak bu amaç dışında herhangi bir anlama sahip olmak amaçlanmaz.</p>
MQINT8	8 bitlik işaretli tamsayı	<p>The MQINT8 data type is an 8-bit signed integer that can take any value in the range -128 to +127, unless otherwise restricted by the context.</p>
MQINT16	16 bitlik işaretli tamsayı	<p>MQINT16 veri tipi, bağlam tarafından başka bir kısıtlama getirmediği sürece, -32 768-+32 767 aralığında herhangi bir değeri alabilen 16 bitlik bir işaretli tamsayıdır. Bir MQINT16 , 2 baytlık bir sınır üzerinde hizalanmalıdır.</p>
MQINT32	32 bitlik işaretli tamsayı	<p>The MQINT32 data type is a 32-bit signed binary integer that can take any value in the range -2 147 483 648 through +2 147 483 647, unless otherwise restricted by the context.</p> <p><u>MQLONG</u>' un tanımına bakın.</p>

Çizelge 459. Temel veri tipi adları, tipleri ve açıklamaları (devamı var)

Temel veri tipi adı	Veri tipi	Tanım
MQINT64	64 bit imzalı tamsayı	<p>The MQINT64 data type is a 64-bit signed integer that can take any value in the range -9 223 372 036 854 775 808 through +9 223 372 036 854 775 807, unless otherwise restricted by the context.</p> <p>COBOL için geçerli değerler, +999 999 999 999 999 999 999 999 999 999 numaralı 999 999 999 999 numaralı 999 999 999 numaralı 999 999 999 numaralı 999 999 999 numaralı 999 64 bitlik bir tamsayı, 8 baytlık bir sınır üzerinde hizalanmalıdır.</p>
MQLONG	32 bitlik işaretli tamsayı	<p>MQLong veri tipi, bağlam tarafından aksi belirtilmediği sürece, 2 147 483 648 ile + 2 147 483 647 aralığında herhangi bir değeri alabilen 32 bitlik imzalı bir ikili tamsayıdır.</p> <p>COBOL için geçerli aralık, +999 999 999 aralığıyla -999 999 999 ile sınırlanmıştır. Bir MQHOMET, 4 baytlık bir sınırla hizalanmalıdır.</p>
MQPID	İşlem tanıtıcısı	<p>IBM MQ işlem tanıtıcısı.</p> <p>Bu, MQ izleme ve FFST™ dökümlerinde kullanılan aynı tanıtıcıdır, ancak işletim sistemi işlem tanıtıcısından farklı olabilir.</p>
MQPTR	İşaretçi	<p>MQPTR veri tipi, herhangi bir tipteki verilerin adresidir. Bir işaretçinin doğal sınırları üzerinde hizalanması gerekir; bu, IBM üzerinde 16 baytlık bir sınırdır ve diğer platformlarda da 8 baytlık bir sınırdır.</p> <p>Bazı programlama dilleri tip atanmış göstergeleri destekler: MQI, bunları birkaç durumda da kullanır (örneğin, C programlama dilinde, PMQCHAR ve PMQUWN).</p>

Çizelge 459. Temel veri tipi adları, tipleri ve açıklamaları (devamı var)

Temel veri tipi adı	Veri tipi	Tanım
MQTID	İş parçacığı tanıtıcısı	IBM MQ iş parçacığı tanıtıcısı. Bu, MQ izleme ve FFST™ dökümlerinde kullanılan aynı tanıtıcıdır, ancak işletim sistemi iş parçacığı tanıtıcısından farklı olabilir.
MQUINT8	8 bitlik imzalanmamış tamsayı	MQUINT8 veri tipi, bağlam tarafından başka bir kısıtlama getirmediği sürece, 0-+255 aralığında herhangi bir değeri alabilen, 8 bitlik imzalanmamış bir tamsayıdır.
MQUINT16	16 bit işaretli tamsayı	MQUINT16 veri tipi, bağlam tarafından aksi belirtilmediği sürece, 0 ile +65 535 aralığında herhangi bir değeri alabilen, 16 bitlik imzalanmamış bir tamsayıdır. Bir MQUINT16 , 2 baytlık bir sınır üzerinde hizalanmalıdır.
MQUINT32	32 bitlik imzalanmamış tamsayı	MQUINT32 veri tipi, 32 bitlik imzalanmamış bir ikili tamsayıdır. <u>MQULONG</u> tanımına bakın.
MQUINT64	64 bit işaretli tamsayı	The MQUINT64 data type is a 64-bit unsigned integer that can take any value in the range 0 through +18 446 744 073 709 551 615, unless otherwise restricted by the context. COBOL için geçerli aralık, 0 ile +999 999 999 999 999 999 999 numaralı 999 arasında bir değer ile sınırlanır. 64 bitlik bir tamsayı, 8 baytlık bir sınır üzerinde hizalanmalıdır.
MQULONG	32 bitlik imzalanmamış tamsayı	MQULONG veri tipi, bağlam tarafından aksi belirtilmediği sürece, 0-+ 4 294 967 294 aralığında herhangi bir değeri alabilen, 32 bitlik imzalanmamış bir ikili tamsayıdır. COBOL için geçerli aralık, 0 ile +999 999 999 aralığında bir değer ile sınırlanır. Bir MQULONG, 4 baytlık bir sınır üzerinde hizalanmalıdır.
PMQACH	İşaretçi	MQACH tipinde bir veri yapısına ilişkin gösterge

Çizelge 459. Temel veri tipi adları, tipleri ve açıklamaları (devamı var)

Temel veri tipi adı	Veri tipi	Tanım
PMQAIR	İşaretçi	MQAIR tipinde bir veri yapısına ilişkin gösterge
PMQAXC	İşaretçi	MQAXC tipinde bir veri yapısına ilişkin gösterge
PMQAXP	İşaretçi	MQAXP tipinde bir veri yapısına ilişkin gösterge
PMQBMHO	İşaretçi	MQBMHO tipinde bir veri yapısına ilişkin gösterge
PMQBO	İşaretçi	MQBO tipinde bir veri yapısına ilişkin gösterge
PMQBOOL	İşaretçi	MQBOOL tipi veriye işaretçi
PMQBYTE	İşaretçi	MMQBATY tipinde verilere ilişkin gösterge
PMQBYTEN	İşaretçi	MQBYTEN tipinde verilere ilişkin gösterge; burada n 8, 16, 24, 32, 40, 128 olabilir.
PMQCBC	İşaretçi	İmlecın MQCBC tipinde bir veri yapısı
PMQCBD	İşaretçi	MQCBD tipinde bir veri yapısına ilişkin gösterge
PMQCHAR	İşaretçi	MQCHAR tipinde veriye ilişkin gösterge
PMQCHARN	İşaretçi	MQCHARN veri tipinin göstergesi; burada n 4, 8, 12, 20, 28, 32, 48, 64, 128, 256, 264 olabilir.
PMQCHARV	İşaretçi	MQCHARV tipinde bir veri yapısına işaret eden gösterge
PMQCIH	İşaretçi	MQCIH tipinde bir veri yapısına ilişkin gösterge
PMQCMHO	İşaretçi	MQCMHO tipinde bir veri yapısına ilişkin gösterge
PMQCNO	İşaretçi	MQCNO tipinde bir veri yapısına ilişkin gösterge
PMQCSP	İşaretçi	MQCSP tipinde bir veri yapısına işaret eden gösterge
PMQCTLO	İşaretçi	MQCTLO tipinde bir veri yapısına ilişkin gösterge
PMQDH	İşaretçi	MQDH tipinde bir veri yapısına ilişkin gösterge
PMQDHO	İşaretçi	MQDHO tipinde bir veri yapısına ilişkin gösterge

Çizelge 459. Temel veri tipi adları, tipleri ve açıklamaları (devamı var)

Temel veri tipi adı	Veri tipi	Tanım
PMQDLH	İşaretçi	MQDLH tipinde bir veri yapısına ilişkin gösterge
PMQDMHO	İşaretçi	MQDMHO tipinde bir veri yapısına ilişkin gösterge
PMQDMPO	İşaretçi	MQDMMPO tipinde bir veri yapısına işaret eden gösterge
PMQEPH	İşaretçi	MQEPH tipinde bir veri yapısına işaretçi
PMQFLOAT32	İşaretçi	MQFLOAT32tipinde bir veri yapısına ilişkin gösterge
PMQFLOAT64	İşaretçi	MQFLOAT64tipinde bir veri yapısına ilişkin gösterge
PMQFUNC	İşaretçi	İşleve ilişkin gösterge
PMQGM0	İşaretçi	MQGM0 tipinde bir veri yapısına ilişkin gösterge
PMQHCONFIG	İşaretçi	MQHCONFIG tipli verilere g " sterge
PMQHCONN	İşaretçi	MQHCONN tipinde veri işaretçisi
PMQHMSG	İşaretçi	MQHMSG tipli verilere g " sterge
PMQH0BJ	İşaretçi	MQH0BJ tipindeki verileri gösteren gösterge
PMQIL	İşaretçi	MQIIH tipinde bir veri yapısına ilişkin gösterge
PMQIMPO	İşaretçi	MQIMPO tipinde bir veri yapısına ilişkin gösterge
PMQINT8	İşaretçi	MQINT8tipli verilere gösterge
PMQINT16	İşaretçi	MQINT16tipli verilere gösterge
PMQINT32	İşaretçi	MQINT32tipli verilere gösterge
PMQINT64	İşaretçi	MQINT64tipli verilere gösterge
PMQUZA	İşaretçi	İmlecın veri tipi MQUBE
PMQMD	İşaretçi	İmlecın tipi MQMD tipi
PMQMDE	İşaretçi	MQMDE tipinde bir veri yapısına ilişkin gösterge
PMQMD1	İşaretçi	MQMD1tipinde bir veri yapısına ilişkin gösterge
PMQMD2	İşaretçi	MQMD2tipinde bir veri yapısına işaret eden gösterge
PMQMHBO	İşaretçi	MQMHBO tipinde bir veri yapısına işaret eden gösterge
PMQOD	İşaretçi	MQOD tipinde bir veri yapısına işaretçi



Çizelge 459. Temel veri tipi adları, tipleri ve açıklamaları (devamı var)

Temel veri tipi adı	Veri tipi	Tanım
PMQOR	İşaretçi	MQOR tipinde bir veri yapısına ilişkin gösterge
PMQPD	İşaretçi	MQPD tipinde bir veri yapısına işaret eden gösterge
PMQPID	İşaretçi	İşlem tanıtıcısını gösteren gösterge
PMQMD	İşaretçi	MQMD tipinde bir veri yapısına işaret eden gösterge
PMQPMO	İşaretçi	MQPMO tipinde bir veri yapısına ilişkin gösterge
PMQPTR	İşaretçi	MQPTR tipinde veriye ilişkin gösterge
PMQRFH	İşaretçi	MQRFH tipinde bir veri yapısına işaret eden gösterge
PMQRFH2	İşaretçi	MQRFH2 tipinde bir veri yapısı göstergesi
PMQRMH	İşaretçi	MQRMH tipinde bir veri yapısına ilişkin gösterge
PMQRR	İşaretçi	MQRR tipinde bir veri yapısına ilişkin gösterge
PMQSCO	İşaretçi	MQSCO tipinde bir veri yapısına ilişkin gösterge
PMQSD	İşaretçi	MQSD tipinde bir veri yapısına işaret eden gösterge
PMQSMPO	İşaretçi	MQSMPO tipinde bir veri yapısına işaret eden gösterge
PMQSRO	İşaretçi	MQSRO tipinde bir veri yapısına ilişkin gösterge
PMSSTS	İşaretçi	MQSTS tipinde bir veri yapısına işaret eden gösterge
PMQTID	İşaretçi	Bir iş parçacığı tanıtıcısına işaretçi
PMQTM	İşaretçi	MQTM tipinde bir veri yapısına işaret eden gösterge
PMQTM2	İşaretçi	MQTM2 tipinde bir veri yapısına işaret eden gösterge
PMQUINT8	İşaretçi	MQUINT8 veri tipine işaretçi
PMQUINT16	İşaretçi	MQUINT16 veri tipine işaretçi
PMQUINT32	İşaretçi	MQUINT32 veri tipine işaret eden gösterge
PMQUINT64	İşaretçi	MQUINT64 veri tipine işaret eden gösterge

<i>Çizelge 459. Temel veri tipi adları, tipleri ve açıklamaları (devamı var)</i>		
<b>Temel veri tipi adı</b>	<b>Veri tipi</b>	<b>Tanım</b>
PMQULONG	İşaretçi	MQULONG veri tipine ilişkin gösterge
PMQVOID	İşaretçi	
PMQWIH	İşaretçi	MQWIH tipinde bir veri yapısına ilişkin gösterge
PMQXQH	İşaretçi	MQXQH tipinde bir veri yapısına ilişkin gösterge

*C bildirimleri*

<i>Çizelge 460. C veri tipi adları ve gösterimleri</i>	
<b>Veri tipi</b>	<b>Gösterim</b>
MQBOOL	<pre>typedef MQLONG MQBOOL;</pre>
MQBYTE	<pre>typedef unsigned char MQBYTE;</pre>
MQBYTE8	<pre>typedef MQBYTE MQBYTE8[8];</pre>
MQBYTE16	<pre>typedef MQBYTE MQBYTE16[16];</pre>
MQBYTE24	<pre>typedef MQBYTE MQBYTE24[24];</pre>
MQBYTE32	<pre>typedef MQBYTE MQBYTE32[32];</pre>
MQBYTE40	<pre>typedef MQBYTE MQBYTE40[40];</pre>
MQCHAR	<pre>typedef char MQCHAR;</pre>
MQCHAR4	<pre>typedef MQCHAR MQCHAR4[4];</pre>
MQCHAR8	<pre>typedef MQCHAR MQCHAR8[8];</pre>
MQCHAR12	<pre>typedef MQCHAR MQCHAR12[12];</pre>
MQCHAR20	<pre>typedef MQCHAR MQCHAR20[20];</pre>

Çizelge 460. C veri tipi adları ve gösterimleri (devamı var)

Veri tipi	Gösterim
MQCHAR28	<code>typedef MQCHAR MQCHAR28[28];</code>
MQCHAR32	<code>typedef MQCHAR MQCHAR32[32];</code>
MQCHAR48	<code>typedef MQCHAR MQCHAR48[48];</code>
MQCHAR64	<code>typedef MQCHAR MQCHAR64[64];</code>
MQCHAR128	<code>typedef MQCHAR MQCHAR128[128];</code>
MQCHAR256	<code>typedef MQCHAR MQCHAR256[256];</code>
MQFLOAT32	<code>typedef float MQFLOAT32;</code>
MQFLOAT64	<code>typedef double MQFLOAT64;</code>
MQHCONFIG	<code>typedef void MQPOINTER MQHCONFIG;</code>
MQHCONN	<code>typedef MQLONG MQHCONN;</code>
MQHOBJ	<code>typedef MQLONG MQHOBJ;</code>
MQINT8	<code>typedef signed char MQINT8;</code>
MQINT16	<code>typedef short MQINT16;</code>

Çizelge 460. C veri tipi adları ve gösterimleri (devamı var)

Veri tipi	Gösterim
MQINT64	<p><b>UNIX</b> 64 bit UNIX üzerinde:</p> <pre>typedef long;</pre> <p><b>AIX</b> 32 bit AIX üzerinde:</p> <pre>typedef int64_t;</pre> <p><b>IBM i</b> <b>Linux</b> <b>z/OS</b> Linux, IBM i ve z/OS sistemlerinde:</p> <pre>typedef long long;</pre> <p><b>Windows</b> Windows'ta:</p> <pre>typedef _int64;</pre>
MQLONG	<p><b>IBM i</b> IBM i'ta:</p> <pre>typedef long MQLONG;</pre> <p><b>z/OS</b> <b>ALW</b> Diğer platformlarda:</p> <pre>if defined(MQ_64_BIT)     typedef int MQLONG; else     typedef long MQLONG;</pre>
MQPID	<pre>typedef MQLONG MQPID;</pre>
MQPTR	<pre>typedef void MQPOINTER MQPTR;</pre>
MQTID	<pre>typedef MQLONG MQTID;</pre>
MQUINT8	<pre>typedef unsigned char MQUINT8;</pre>
MQUINT16	<pre>typedef unsigned short MQUINT16;</pre>

Çizelge 460. C veri tipi adları ve gösterimleri (devamı var)

Veri tipi	Gösterim
MQUINT64	<p><b>UNIX</b> 64 bit UNIX üzerinde:</p> <pre>typedef unsigned long;</pre> <p><b>AIX</b> 32 bit AIX üzerinde:</p> <pre>typedef uint64_t;</pre> <p><b>IBM i</b> <b>Linux</b> <b>z/OS</b> Linux, IBM i ve z/OS sistemlerinde:</p> <pre>typedef unsigned long long;</pre> <p><b>Windows</b> Windows'ta:</p> <pre>typedef unsigned _int64;</pre>
MQULONG	<p><b>IBM i</b> IBM i'ta:</p> <pre>typedef unsigned long MQULONG;</pre> <p><b>z/OS</b> <b>ALW</b> Diğer platformlarda:</p> <pre>if defined(MQ_64_BIT)     typedef unsigned int MQULONG; else     typedef unsigned long MQULONG;</pre>
PMQBO	<pre>typedef MQBO MQPOINTER PMQBO;</pre>
PMQBOOL	<pre>typedef MQBOOL MQPOINTER PMQBOOL;</pre>
PMQBYTE	<pre>typedef MQBYTE MQPOINTER PMQBYTE;</pre>
PMQBYTE8	<pre>typedef MQBYTE8[8] MQPOINTER PMQBYTE8[8];</pre>
PMQBYTE16	<pre>typedef MQBYTE16[16] MQPOINTER PMQBYTE16[16];</pre>
PMQBYTE24	<pre>typedef MQBYTE24[24] MQPOINTER PMQBYTE24[24];</pre>

Çizelge 460. C veri tipi adları ve gösterimleri (devamı var)

Veri tipi	Gösterim
PMQBYTE32	<code>typedef MQBYTE32[32] MQPOINTER PMQBYTE32[32];</code>
PMQBYTE40	<code>typedef MQBYTE40[40] MQPOINTER PMQBYTE40[40];</code>
PMQBYTE128	<code>typedef MQBYTE128[128] MQPOINTER PMQBYTE128[128];</code>
PMQCHAR	<code>typedef MQCHAR MQPOINTER PMQCHAR;</code>
PMQCHAR4	<code>typedef MQCHAR4[4] MQPOINTER PMQCHAR4[4];</code>
PMQCHAR8	<code>typedef MQCHAR8[8] MQPOINTER PMQCHAR8[8];</code>
PMQCHAR12	<code>typedef MQCHAR12[12] MQPOINTER PMQCHAR12[12];</code>
PMQCHAR20	<code>typedef MQCHAR20[20] MQPOINTER PMQCHAR20[20];</code>
PMQCHAR28	<code>typedef MQCHAR28[28] MQPOINTER PMQCHAR28[28];</code>
PMQCHAR32	<code>typedef MQCHAR32[32] MQPOINTER PMQCHAR32[32];</code>
PMQCHAR48	<code>typedef MQCHAR48[48] MQPOINTER PMQCHAR48[48];</code>
PMQCHAR64	<code>typedef MQCHAR64[64] MQPOINTER PMQCHAR64[64];</code>
PMQCHAR128	<code>typedef MQCHAR128[128] MQPOINTER PMQCHAR128[128];</code>
PMQCHAR256	<code>typedef MQCHAR256[256] MQPOINTER PMQCHAR256[256];</code>
PMQCHAR264	<code>typedef MQCHAR264[264] MQPOINTER PMQCHAR264[264];</code>
PMQCIH	<code>typedef MQCIH MQPOINTER PMQCIH;</code>

Çizelge 460. C veri tipi adları ve gösterimleri (devamı var)

Veri tipi	Gösterim
PMQCNO	<code>typedef MQCNO MQPOINTER PMQCNO;</code>
PMQDLH	<code>typedef MQDLH MQPOINTER PMQDLH;</code>
PMQFUNC	<code>typedef void MQPOINTER PMQFUNC;</code>
PMQFLOAT32	<code>typedef MQFLOAT32 MQPOINTER PMQFLOAT32;</code>
PMQFLOAT64	<code>typedef MQFLOAT64 MQPOINTER PMQFLOAT64;</code>
PMQGM0	<code>typedef MQGM0 MQPOINTER PMQGM0;</code>
PMQHCONFIG	<code>typedef MQHCONFIG MQPOINTER PMQHCONFIG;</code>
PMQHCONN	<code>typedef MQHCONN MQPOINTER PMQHCONN;</code>
PMQH0BJ	<code>typedef MQH0BJ MQPOINTER PMQH0BJ;</code>
PMQIIH	<code>typedef MQIIH MQPOINTER PMQIIH;</code>
PMQINT8	<code>typedef MQINT8 MQPOINTER PMQINT8;</code>
PMQINT16	<code>typedef MQINT16 MQPOINTER PMQINT16;</code>
PMQLONG	<code>typedef MQLONG MQPOINTER PMQLONG;</code>
PMQMD	<code>typedef MQMD MQPOINTER PMQMD;</code>
PMQMD1	<code>typedef MQMD1[1] MQPOINTER PMQMD1[1];</code>
PMQMDE	<code>typedef MQMDE MQPOINTER PMQMDE;</code>

Çizelge 460. C veri tipi adları ve gösterimleri (devamı var)

Veri tipi	Gösterim
PMQOD	<code>typedef MQOD MQPOINTER PMQOD;</code>
PMQPMO	<code>typedef MQPMO MQPOINTER PMQPMO;</code>
PMQPTR	<code>typedef MQPTR MQPOINTER PMQPTR;</code>
PMQRFH	<code>typedef MQRFH MQPOINTER PMQRFH;</code>
PMQRFH2	<code>typedef MQRFH2[2] MQPOINTER PMQRFH2[2];</code>
PMQRMH	<code>typedef MQRMH MQPOINTER PMQRMH;</code>
PMQTM	<code>typedef MQTM MQPOINTER PMQTM;</code>
PMQTM2	<code>typedef MQTM2[2] MQPOINTER PMQTM2[2];</code>
PMQUINT8	<code>typedef MQUINT8 MQPOINTER PMQUINT8;</code>
PMQUINT16	<code>typedef MQUINT16 MQPOINTER PMQUINT16;</code>
PMQULONG	<code>typedef MQULONG MQPOINTER PMQULONG;</code>
PMQVOID	<code>typedef void MQPOINTER PMQVOID;</code>
PMQWIH	<code>typedef MQWIH MQPOINTER PMQWIH;</code>
PMQXQH	<code>typedef MQXQH MQPOINTER PMQXQH;</code>
PPMQBO	<code>typedef PMQBO MQPOINTER PPMQBO;</code>
PPMQBYTE	<code>typedef PMQBYTE MQPOINTER PPMQBYTE;</code>



<i>Çizelge 460. C veri tipi adları ve gösterimleri (devamı var)</i>	
<b>Veri tipi</b>	<b>Gösterim</b>
PPMQCHAR	typedef PMQCHAR MQPOINTER PPMQCHAR;
PPMQCNO	typedef PMQCNO MQPOINTER PPMQCNO;
PPMQGMO	typedef PMQGMO MQPOINTER PPMQGMO;
PPMQHCONN	typedef PMQHCONN MQPOINTER PPMQHCONN;
PPMQHOBJ	typedef PMQHOBJ MQPOINTER PPMQHOBJ;
PPMQLONG	typedef PMQLONG MQPOINTER PPMQLONG;
PPMQMD	typedef PMQMD MQPOINTER PPMQMD;
PPMQOD	typedef PMQOD MQPOINTER PPMQOD;
PPMQPMO	typedef PMQPMO MQPOINTER PPMQPMO;
PPMQULONG	typedef PMQULONG MQPOINTER PPMQULONG;
PPMQVOID	typedef PMQVOID MQPOINTER PPMQVOID;
Burada defined (MQ_64_BIT) , 64 bitlik bir platform anlamına gelir.	

MQPOINTER makro değişkeninin tanımı için bkz. [“Veri türleri” sayfa 264](#) .

*COBOL bildirimleri*

<i>Çizelge 461. COBOL veri tipi adları ve gösterimleri</i>	
<b>Veri tipi</b>	<b>Gösterim</b>
MQBOOL	PIC S9(9) BINARY
MQBYTE	PIC X

Çizelge 461. COBOL veri tipi adları ve gösterimleri (devamı var)

<b>Veri tipi</b>	<b>Gösterim</b>
MQBYTE8	PIC X(8)
MQBYTE16	PIC X(16)
MQBYTE24	PIC X(24)
MQBYTE32	PIC X(32)
MQBYTE40	PIC X(40)
MQCHAR	PIC X
MQCHAR4	PIC X(4)
MQCHAR8	PIC X(8)
MQCHAR12	PIC X(12)
MQCHAR20	PIC X(20)
MQCHAR28	PIC X(28)
MQCHAR32	PIC X(32)
MQCHAR48	PIC X(48)
MQCHAR64	PIC X(64)
MQCHAR128	PIC X(128)
MQCHAR256	PIC X(256)

<i>Çizelge 461. COBOL veri tipi adları ve gösterimleri (devamı var)</i>	
<b>Veri tipi</b>	<b>Gösterim</b>
MQFLOAT32	USAGE COMP-1
MQFLOAT64	USAGE COMP-2
MQHCONN	Açıkz/OS PIC S9(9) COMP-5 Diğer platformlarda PIC S9(9) BINARY
MQHOBJ	PIC S9(9) BINARY
MQINT8	PIC S9(2) BINARY
MQINT16	PIC S9(4) BINARY
MQINT64	PIC S9(18) BINARY
MQLONG	PIC S9(9) BINARY
MQPTR	POINTER
MQUINT8	PIC 9(2) BINARY
MQUINT16	PIC 9(4) BINARY
MQUINT64	PIC 9(18) BINARY
MQULONG	PIC 9(9) BINARY

*PL/I bildirimleri*  
PL/I, z/OSüzerinde desteklenir.

Çizelge 462. PL/I veri tipi adları ve gösterimler

<b>Veri tipi</b>	<b>Gösterim</b>
MQBOOL	fixed bin(31)
MQBYTE	char(1)
MQBYTE8	char(8)
MQBYTE16	char(16)
MQBYTE24	char(24)
MQBYTE32	char(32)
MQBYTE40	char(40)
MQCHAR	char(1)
MQCHAR4	char(4)
MQCHAR8	char(8)
MQCHAR12	char(12)
MQCHAR20	char(20)
MQCHAR28	char(28)
MQCHAR32	char(32)
MQCHAR48	char(48)
MQCHAR64	char(64)

<i>Çizelge 462. PL/I veri tipi adları ve gösterimler (devamı var)</i>	
<b>Veri tipi</b>	<b>Gösterim</b>
MQCHAR128	char(128)
MQCHAR256	char(256)
MQFLOAT32	binary float(21) ieee
MQFLOAT64	binary float(52) ieee
MQHCONN	fixed bin(31)
MQHOBJ	fixed bin(31)
MQINT8	fixed bin(7)
MQINT16	fixed bin(15)
MQINT64	fixed bin(63)
MQLONG	fixed bin(31)
MQPTR	pointer
MQUINT8	fixed bin(8)
MQUINT16	fixed bin(16)
MQUINT64	fixed bin(64)
MQULONG	fixed bin(32)

*System/390 çevirici bildirimleri*

System/390 çevirici programı yalnızca z/OS üzerinde desteklenir.

Çizelge 463. System/390 çevirici veri tipi adları ve gösterimleri

<b>Veri tipi</b>	<b>Gösterim</b>
MQBOOL	DS F
MQBYTE	DS XL1
MQBYTE8	DS XL8
MQBYTE16	DS XL16
MQBYTE24	DS XL24
MQBYTE32	DS XL32
MQBYTE40	DS XL40
MQCHAR	DS CL1
MQCHAR4	DS CL4
MQCHAR8	DS CL8
MQCHAR12	DS CL12
MQCHAR20	DS CL20
MQCHAR28	DS CL28
MQCHAR32	DS CL32
MQCHAR48	DS CL48
MQCHAR64	DS CL64

Çizelge 463. System/390 çevirici veri tipi adları ve gösterimleri (devamı var)

Veri tipi	Gösterim
MQCHAR128	DS CL128
MQCHAR256	DS CL256
MQFLOAT32	DS EB
MQFLOAT64	DS DB
MQHCONN	DS F
MQHOBJ	DS F
MQINT8	DS XL1
MQINT16	DS H
MQINT64	DS D
MQLONG	DS F
MQPTR	DS F
MQUINT8	DS XL1
MQUINT16	DS H
MQUINT64	DS D
MQULONG	DS F

### **Yapı veri tipleri**

Yapı veri tiplerinin özeti, MQI yapılarının tutarlı bir şekilde eşlenmesine ilişkin kurallar ve her yapı veri tipi tanımında kullanılan kurallar.

- “MQI çağrılarında ya da çıkış işlevlerinde kullanılan yapı veri tiplerinin özeti” sayfa 260
- “İleti verilerinde kullanılan yapı veri tiplerinin özeti” sayfa 261
- “MQI yapılarının tutarlı bir şekilde eşlenmesine ilişkin kurallar” sayfa 261
- “Her yapı veri tipi tanımında kullanılan kurallar” sayfa 262

## MQI çağrılarında ya da çıkış işlevlerinde kullanılan yapı veri tiplerinin özeti

<i>Çizelge 464. MQI çağrılarında ya da çıkış işlevlerinde kullanılan yapı veri tipleri</i>		
<b>Yapı</b>	<b>Açıklama</b>	<b>Kullanıldığı yerdeki çağrılar</b>
<a href="#">MQACH</a>	API çıkış zinciri üstbilgisi	
<a href="#">MQAIR</a>	Kimlik doğrulama bilgileri kaydı	<a href="#">MQCONN</a>
<a href="#">MQAXC</a>	API çıkış bağlamı	
<a href="#">MQAXP</a>	API çıkış parametresi	
<a href="#">MQBMHO</a>	İleti işleme seçenekleri için arabellek	<a href="#">MQBUFMH</a>
<a href="#">MQBO</a>	Başlangıç seçenekleri	<a href="#">MQBEGIN</a>
<a href="#">MQCBD</a>	Geri çağırma tanımlayıcısı	<a href="#">MQCB</a>
<a href="#">MQCBO</a>	Create-bag seçenekleri	mqCreateBag
<a href="#">MQCHARV</a>	Değişken uzunluklu dizgi	<a href="#">MQINQMP</a>
<a href="#">MQCNO</a>	Bağlantı seçenekleri	<a href="#">MQCONN</a>
<a href="#">MQCSP</a>	Güvenlik parametreleri	<a href="#">MQCONN</a>
<a href="#">MQCTLO</a>	Geri Arama Seçenekleri	<a href="#">MQCTL</a>
<a href="#">MQDMPO</a>	İleti özelliği seçeneklerini sil	<a href="#">MQDLTMP</a>
<a href="#">MQGMO</a>	İleti alma seçenekleri	<a href="#">MQGet</a>
<a href="#">MQIMPO</a>	İleti özelliği sorma seçenekleri	<a href="#">MQINQMP</a>
<a href="#">Mqmd</a>	İleti tanımlayıcı	<a href="#">MQBUFMH</a> , <a href="#">MQMHBUF</a> , <a href="#">MQCB</a> , <a href="#">MQGET</a> , <a href="#">MQPUT</a> , <a href="#">MQPUT1</a>
<a href="#">MQMHBO</a>	İleti tanıtıcısı-arabellek seçenekleri	<a href="#">MQMHBUF</a>
<a href="#">MQOD</a>	Nesne tanımlayıcı	<a href="#">MQOPEN</a> , <a href="#">MQPUT1</a>
<a href="#">MQOR</a>	Nesne kaydı	<a href="#">MQOPEN</a> , <a href="#">MQPUT1</a>
<a href="#">MQPD</a>	Özellik tanımlayıcı	<a href="#">MQSETMP</a>
<a href="#">MQPMO</a>	Koyma iletisi seçenekleri	<a href="#">MQPUT</a> , <a href="#">MQPUT1</a>
<a href="#">MQPMR</a>	Koyma iletisi kaydı	<a href="#">MQPUT</a> , <a href="#">MQPUT1</a>
<a href="#">MQRR</a>	Yanıt kaydı	<a href="#">MQOPEN</a> , <a href="#">MQPUT</a> , <a href="#">MQPUT1</a>
<a href="#">MQSCO</a>	TLS yapılandırma seçenekleri	<a href="#">MQCONN</a>
<a href="#">MQSD</a>	Abonelik tanımlayıcısı	<a href="#">MQSUB</a>
<a href="#">MQSMPO</a>	İleti özelliği seçeneğini ayarla	<a href="#">MQSETMP</a>
<a href="#">MQSRO</a>	Abonelik isteği seçenekleri	<a href="#">MQSUBRQ</a>



Çizelge 464. MQI çağrılarında ya da çıkış işlevlerinde kullanılan yapı veri tipleri (devamı var)

Yapı	Açıklama	Kullanıldığı yerdeki çağrılar
<u>MQSTS</u>	Durum raporlama yapısı	<u>MQSTAT</u>

## İleti verilerinde kullanılan yapı veri tiplerinin özeti

Çizelge 465. İleti verilerinde kullanılan yapı veri tipleri

Yapı	Açıklama
<u>MQCIH</u>	CICS bilgi üstbilgisi
<u>MQCFH</u>	PCF üstbilgisi
<u>MQEPH</u>	Yerleşik PCF üstbilgisi
<u>MQDHD</u>	Dağıtım üstbilgisi
<u>MQDLH</u>	Teslim edilmemiş harf (teslim edilmemiş ileti) üstbilgisi
<u>MQIIH</u>	IMS bilgi üstbilgisi
<u>MQMDE</u>	İleti tanımlayıcı uzantısı
<u>MQRFH</u>	Kurallar ve biçimlendirme üstbilgisi
<u>MQRFH2</u>	Kurallar ve biçimlendirme üstbilgisi 2
<u>MQRMH</u>	Başvuru iletisi üstbilgisi
<u>MQTM</u>	Tetikleyici ileti
<u>MQTMC2</u>	Tetikleyici ileti (karakter biçimi 2)
<u>MQWIH</u>	İş bilgileri üstbilgisi
<u>MQXQH</u>	İletim kuyruğu üstbilgisi

**Not:** MQDXP yapısı (veri dönüştürme çıkış parametresi), ilişkili veri dönüştürme çağrılarıyla birlikte “Veri dönüştürme çıkışı” sayfa 891’inde açıklanmıştır.

## MQI yapılarının tutarlı bir şekilde eşlenmesine ilişkin kurallar

Programlama dilleri, yapılarla ilişkin destek düzeylerinde değişiklik gösterir ve MQI yapılarını her programlama dilinde tutarlı olarak eşlemek için belirli kurallar ve kurallar benimsenmiştir:

1. Yapılar doğal sınırlarıyla hizalanmalıdır.

- Çoğu MQI yapısı 4 baytlık hizalama gerektirir.
- IBM üzerinde, işaretçileri içeren yapılar 16 baytlık hizalama gerektirir; bunlar: MQCNO, MQOD, MQPMO.

2. Bir yapıdaki her alan doğal sınırında hizalanmalıdır.

- MQLONG ' ye eşit veri tiplerine sahip alanlar, 4 baytlık sınırlarda hizalanmalıdır.
- MQPTR ' ye eşit veri tiplerine sahip alanlar, IBM ürününde 16 baytlık sınırlara ve diğer ortamlarda 4 baytlık sınırlara hizalanmalıdır.
- Diğer alanlar 1 baytlık sınırlara hizalanır.

3. Bir yapının uzunluğu, yapının sınır hizalamasının katı olmalıdır.

- Çoğu MQI yapısının 4 byte 'ın katları olan uzunlukları vardır.
- IBM üzerinde, işaretçi içeren yapıların 16 baytlık katları olan uzunlukları vardır.

4. Gerektiğinde, önceki kurallara uyulmasını sağlamak için doldurma baytları ya da alanları eklenmelidir.

## Her yapı veri tipi tanımında kullanılan kurallar

Her yapı veri tipinin tanımı şunları içerir:

- Yapının amacına ve kullanımına genel bakış
- Programlama dilinden bağımsız bir biçimde, yapıdaki alanların açıklamaları
- Yapının desteklenen programlama dillerinin her birinde nasıl bildirildiğine ilişkin örnekler

Her yapı veri tipinin açıklaması aşağıdaki kısımları içerir:

### Yapı adı

Yapının adı ve ardından yapıdaki alanların özeti.

### Genel Bakış

Yapının amacının ve kullanımının kısa bir açıklaması.

### Alanlar

Alanların açıklamaları. Her alan için, alanın adını parantez () içindeki temel veri tipi izler. Metinde, alan adları italik yazı tipi kullanılarak gösterilir; örneğin, *Version*.

Alanın amacına ilişkin bir açıklama ve alanın alabileceği değerlerin bir listesi de vardır. Değişmezlerin adları büyük harfle gösterilir; örneğin, MQGMO\_STRUC\_ID. \* karakteri kullanılarak, önceki aynı olan sabit değerler kümesi gösterilir; örneğin: MQIA\_ \*.

Alanların açıklamalarında aşağıdaki terimler kullanılır:

### Giriş

Arama yaptığınızda alana bilgi girersiniz.

### output

Çağrı tamamlandığında ya da başarısız olduğunda kuyruk yöneticisi alandaki bilgileri döndürür.

### giriş/çıkış

Bir arama yaptığınızda alana bilgi girersiniz ve arama tamamlandığında ya da başarısız olduğunda kuyruk yöneticisi bilgileri değiştirir.

### Başlangıçtaki değerler

MQI ile verilen veri tanımlaması dosyalarındaki her alana ilişkin başlangıç değerlerini gösteren bir çizelge.

### C bildirim

C ' deki yapının tipik beyannamesi.

### COBOL bildirim

COBOL ' da yapının tipik açıklaması.

### PL/I bildirim

PL/I ' de yapının tipik açıklaması.

### High Level Assembler bildirim

System/390 çevirici dilinde yapının tipik bildirim.

### Visual Basic bildirim

Visual Basic 'de yapının tipik beyannamesi.



## C programlama

C programlama dilindeki MQI ' ı kullanmanıza yardımcı olacak bilgiler.

- [“Üstbilgi dosyaları” sayfa 263](#)
- [“İşlevler” sayfa 263](#)
- [“Tanımlanmamış veri tipine sahip parametreler” sayfa 263](#)
- [“Veri türleri” sayfa 264](#)
- [“İkili dizgiler işleniyor” sayfa 264](#)

- “Karakter dizilimlerini işleme” sayfa 264
- “Yapılara ilişkin ilk değerler” sayfa 264
- “Dinamik yapılar için ilk değerler” sayfa 265
- “C++ dilinden kullan” sayfa 265
- “Gösterim kuralları” sayfa 265

## Üstbilgi dosyaları

Çizelge 466. C üstbilgi dosyaları	
Dosya	İçerik
CMQC	Ana MQI için işlev prototipleri, veri tipleri ve adlandırılmış sabitler
CMQXC	Veri dönüştürme çıkışı için işlev prototipleri, veri tipleri ve adlandırılmış sabitler
CMQEC	Ana MQI için işlev prototipleri, veri tipleri ve adlandırılmış sabitler, veri dönüştürme çıkışı ve Arabirim Giriş Noktaları yapısı (CMQEC, CMQXC ve CMQC ' yi içerir)
CMQSTRC	MQI değişmez tanımlamalarını metin eşdeğerine dönüştüren işlevler.  <b>Uyarı:</b>  IBM MQ 9.1' den z/OS için geçerlidir. Bu üstbilgi dosyasını kullanan programlar LONGNAME derleyici seçeneğiyle derlenmelidir.

Uygulamaların taşınabilirliğini artırmak için, `#include` ön işlemci yönergesinde üstbilgi dosyasının adını küçük harfle kodlayın:

```
#include "cmqec.h"
```

## İşlevler

Bir işlevi her çağırdığında adres tarafından geçirilen tüm parametreleri belirtmenize gerek yoktur.

- *yalnızca giriş* ve MQHCONN, MQHOBJ ya da MQLONG tipinde parametreleri değere göre geçirin.
- Diğer tüm parametreleri adrese göre iletir.

Belirli bir parametrenin gerekli olmadığı durumlarda, parametre verilerinin adresi yerine, işlev çağrısında parametre olarak boş bir gösterge kullanın. Bunun mümkün olduğu parametreler, çağrı açıklamalarında tanımlanır.

İşlevin değeri olarak parametre döndürülmez; C terminolojisinde bu, tüm işlevlerin voiddeğerini döndürdüğü anlamına gelir.

İşlevin öznitelikleri MQENTRY makro değişkeniyle tanımlanır; bu makro değişkeninin değeri ortama bağlıdır.

## Tanımlanmamış veri tipine sahip parametreler

MQGET, MQPUT ve MQPUT1 işlevlerinde **Buffer** değiştirgesinin tanımlı olmayan bir veri tipi var. Bu parametre, uygulamanın ileti verilerini göndermek ve almak için kullanılır.

Bu sıralamaya ilişkin parametreler, C örneklerinde MQBYTE dizileri olarak gösterilir. Parametreleri bu şekilde bildirebilirsiniz, ancak genellikle bunları iletideki verilerin düzenini tanımlayan belirli bir yapı olarak bildirmek daha uygundur. Gerçek işlev değiştirgesini gösterge-to-void olarak bildirin ve işlev çağrısında parametre olarak herhangi bir veri türünün adresini belirtin.

## Veri türleri

C `typedef` deyimini kullanarak tüm veri tiplerini tanımlayın. Her veri tipi için, ilgili gösterge veri tipini de tanımlayın. İşaretçi veri tipinin adı, bir işaretçiyi göstermek için öneki P harfi olan temel veri ya da yapı veri tipinin adıdır. `MQPOINTER` makro değişkenini kullanarak göstergenin özniteliklerini tanımlayın; bu makro değişkeninin değeri ortama bağlıdır. Aşağıda işaretçi veri tiplerinin nasıl bildirileceği gösterilmektedir:

```
#define MQPOINTER *          /* depends on environment */
...
typedef MQLONG MQPOINTER PMQLONG; /* pointer to MQLONG */
typedef MQMD MQPOINTER PMQMD; /* pointer to MQMD */
```

## İkili dizgiler işleniyor

İkili veri dizgilerini `MQBYTE`n veri tiplerinden biri olarak bildirin.

Bu tipteki alanları kopyaladığınızda, karşılaştırdığınızda ya da ayarladığınızda, C işlevlerini **`memcpy`**, **`memcpya`** da **`memset`** kullanın; örneğin:

```
#include <string.h>
#include "cmqc.h"

MQMD MyMsgDesc;

memcpy(MyMsgDesc.MsgId,          /* set "MsgId" field to nulls */
       MQMI_NONE,              /* ...using named constant */
       sizeof(MyMsgDesc.MsgId));

memset(MyMsgDesc.CorrelId,      /* set "CorrelId" field to nulls */
       0x00,                  /* ...using a different method */
       sizeof(MQBYTE24));
```

**`strcpy`**, **`strcmp`**, **`strcpyya`** da **`strncmp`**dizgi işlevlerini kullanmayın; bunlar, `MQBYTE`n veri tipleriyle bildirilen veriler için doğru biçimde çalışmaz.

## Karakter dizilimlerini işleme

Kuyruk yöneticisi uygulamaya karakter verileri döndürdüğünde, kuyruk yöneticisi karakter verilerini her zaman alanın tanımlı uzunluğuna kadar boşluklarla doldurur. Kuyruk yöneticisi boş sonlandırılmış dizgiler *döndürmez*.

Bu nedenle, bu tür dizgileri kopyalarken, karşılaştırırken ya da birleştirirken **`strncpy`**, **`strncmpya`** da **`strncat`**dizgi işlevlerini kullanın.

Dizginin boş değerle (**`strcpy`**, **`strcmp`**, **`strcat`**) sonlandırılmasını gerektiren dizgi işlevlerini kullanmayın. Ayrıca, dizginin uzunluğunu saptamak için **`strlen`** işlevini kullanmayın; alanın uzunluğunu saptamak için **`sizeof`** işlevini kullanın.

## Yapılara ilişkin ilk değerler

Üstbilgi dosyaları, bu yapıların eşgörünümlerini bildirirken MQ yapılarına ilişkin ilk değerleri sağlamak için kullanabileceğiniz çeşitli makro değişkenlerini tanımlar.

Bu makro değişkenlerinin adları `MQxxx_DEFAULT` biçiminde olur; burada `MQxxx`, yapının adını gösterir. Bunlar aşağıdaki şekilde kullanılır:

```
MQMD MyMsgDesc = {MQMD_DEFAULT};
MQPMO MyPutOpts = {MQPMO_DEFAULT};
```

Bazı karakter alanları için (örneğin, çoğu yapıda oluşan *StrucId* alanları ya da `MQMD` ' de oluşan *Format* alanı), `MQI` geçerli olan belirli değerleri tanımlar. Geçerli değerlerin her biri için *iki* makro değişkeni sağlanır:

- Bir makro değişkeni, değeri, örtük boş değer eşleşmeleri dışında, uzunluğu tam olarak alanın tanımlı uzunluğu olan bir dizgi olarak tanımlar. Örneğin, MQMD ' deki *Format* alanı için aşağıdaki makro değişkeni sağlar (¬ tek bir boş karakteri temsil eder):

```
#define MQFMT_STRING "MQSTR¬¬"
```

Bu formu memcpy ve memcmp işlevleriyle kullanın.

- Diğer makro değişkeni değeri bir karakter dizisi olarak tanımlar; bu makro değişkeninin adı, \_ARRAY ile sonekli dizgi formunun adıdır. Örneğin:

```
#define MQFMT_STRING_ARRAY 'M','Q','S','T','R',' ','¬','¬','¬'
```

Bir yapı eşgörünümünü MQMD\_DEFAULT makro değişkeninin sağladıklarından farklı değerlerle bildirdiğinizde alanı kullanıma hazırlamak için bu formu kullanın. (Bu her zaman gerekli değildir; bazı ortamlarda değer dizgi biçimini her iki durumda da kullanabilirsiniz. Ancak, C++ programlama diliyle uyumluluk için bu gerektiğinden, bildirimler için dizi formunu kullanabilirsiniz.)

## Dinamik yapılar için ilk değerler

Bir yapının değişken sayıda eşgörünümü gerektiğinde, eşgörünümler genellikle calloc ya da malloc işlevleri kullanılarak dinamik olarak elde edilen ana bellekte yaratılır. Bu tür yapılardaki alanları kullanıma hazırlamak için aşağıdaki tekniği göz önünde bulundurun:

1. Yapıyı kullanıma hazırlamak için uygun MQxxx\_DEFAULT makro değişkenini kullanarak yapının bir eşgörünümünü bildirin. Bu eşgörünüm, diğer eşgörünümler için model olur:

```
MQMD Model = {MQMD_DEFAULT}; /* declare model instance */
```

Statik ya da auto anahtar sözcükleri, model eşgörünümüne gereken şekilde statik ya da dinamik geçerlik süresi vermek için bildirimde kodlanabilir.

2. Yapının dinamik bir eşgörünümü için depolama alanı elde etmek üzere calloc ya da malloc işlevlerini kullanın:

```
PMQMD Instance;  
Instance = malloc(sizeof(MQMD)); /* get storage for dynamic instance */
```

3. Model eşgörünümünü dinamik örneğe kopyalamak için memcpy işlevini kullanın:

```
memcpy(Instance,&Model,sizeof(MQMD)); /* initialize dynamic instance */
```

## C++ dilinden kullan

C++ programlama dili için, üstbilgi dosyaları yalnızca C++ derleyicisi kullandığınızda içerilen aşağıdaki ek deyimleri içerir:

```
#ifndef __cplusplus  
extern "C" {  
#endif  
  
/* rest of header file */  
  
#ifdef __cplusplus  
}  
#endif
```

## Gösterim kuralları

Bu bilgiler, işlevlerin nasıl çağrılacağını ve parametrelerin nasıl bildirileceğini gösterir.

Bazı durumlarda, parametreler deđişmez büyüklükte olmayan dizilerdir. Bunlar için, sayısal bir sabiti göstermek için küçük harf n kullanılır. Bu deđiştirgeye ilişkin bildirim kodladığınızda, n deđerini gereken sayısal deđerle deđiştirin.

## **COBOL**

Bu bölümde, COBOL programlama dilinden MQI ' yi kullanmanıza yardımcı olacak bilgiler yer alır.

### **High Level Assembler programlama**

System/390 Assembler programlama dilindeki MQI ' ı kullanmanıza yardımcı olacak bilgiler.

- “Makrolar” sayfa 266
- “Yapılar” sayfa 267
- “CMQVERA makrosu” sayfa 267
- “Gösterim kuralları” sayfa 267

## **Makrolar**

Adlandırılmış sabitler için iki makro ve her yapı için bir makro vardır. Bu dosyalar aşağıdaki tabloda özetlenmiştir.

<i>Çizelge 467. Çevirici makroları</i>	
<b>Dosya</b>	<b>İçerik</b>
CMQA	Ana MQI için adlandırılmış sabitler (eşittir)
CMQCIHA	CICS bilgi üstbilgisi yapısı
CMQCNOA	Bağlantı seçenekleri yapısı
CMQDLHA	Ölü harf üstbilgisi yapısı
CMQDXPA	Veri dönüştürme çıkış parametresi yapısı
CMQGMOA	İleti alma seçenekleri yapısı
CMQIIHA	IMS bilgi üstbilgisi yapısı
CMQMDA	İleti tanımlayıcı yapısı
CMQMDEA	İleti tanımlayıcı uzantısı yapısı
CMQODA	Nesne tanımlayıcı yapısı
CMQPMOA	Koyma iletisi seçenekleri yapısı
CMQRFHA	Kurallar ve biçimlendirme üstbilgi yapısı
CMQRFH2A	Kurallar ve biçimlendirme üstbilgi yapısı sürüm 2
CMQRMHA	Başvuru iletisi üstbilgi yapısı
CMQTMA	Tetikleyici ileti yapısı
CMQTMCA	Tetikleyici ileti yapısı (karakter biçimi) sürüm 2
CMQVERA	Yapı sürümü denetimi
CMQWIHA	İş bilgileri üstbilgi yapısı
CMQXA	Veri dönüştürme çıkışı için adlandırılmış sabitler
CMQXPA	API geçiş çıkış parametresi yapısı
CMQXQHA	İletim kuyruđu üstbilgi yapısı

## Yapılar

Yapılar, makronun işlemini denetlemek için çeşitli parametreleri olan makrolar tarafından oluşturulur. Bakınız [“Yapılar” sayfa 267](#)

### CMQVERA makrosu

Bu makro, yapı makrolarındaki DCLVER değiştirgesi için kullanılacak varsayılan değeri belirlemenizi sağlar.

CMQVERA tarafından belirtilen değer, yapı makrosu tarafından, DCLVER değiştirgesini yapı makrosunun çağrılmasından çıkarırsanız kullanılır. Varsayılan değer, DCLVER değiştirgesiyle CMQVERA makrosunun kodlanması ile belirlenir:

#### DCLVER=YRRLKTEKI

Varsayılan sürüm, geçerli (en son) sürüme ayarlanır.

#### DCLVER=BELIRTILDI

Varsayılan sürüm, VERSION değiştirgesiyle belirtilen sürüme ayarlanır.

DCLVER değiştirgesini belirtmeniz ve değer büyük harf olması gerekir. CMQVERA tarafından belirlenen değer, CMQVERA 'nın bir sonraki çağrılışına ya da yapıbiriminin sonuna kadar varsayılan değer olarak kalır. CMQVERA 'yı atlarsanız, varsayılan değer DCLVER=CURRENT olur.

### Gösterim kuralları

Diğer konular, çağrılarının nasıl çağrılacağını ve parametrelerin nasıl bildirileceğini gösterir. Bazı durumlarda, değiştirgeler, bir sayısal değişmezi göstermek için küçük bir n harfe sahip, değişmez büyüklüklü diziler ya da karakter dizileridir. Bu değiştirgeye ilişkin bildirim kodladığınızda, n değerini gereken sayısal değere değiştirin.

#### Yapılar

Yapılar, makronun işlemini denetlemek için çeşitli parametreleri olan makrolar tarafından oluşturulur.

**Not:** Zaman zaman IBM MQ yapılarının yeni sürümleri tanıtılır. Yeni sürümdeki ek alanlar, daha önce 256 bayttan küçük olan bir yapının 256 bayttan büyük olmasına neden olabilir. Bu nedenle, bir IBM MQ yapısını kopyalamak ya da bir IBM MQ yapısını boş değere ayarlamak üzere 256 bayttan büyük olabilecek yapılarla doğru şekilde çalışmak için çevirici yönergelerini yazın. Diğer bir seçenek olarak, yapının belirli bir sürümünü bildirmek için DCLVER makro değiştirgesini ya da CMQVERA makrosunu VERSION değiştirgesiyle birlikte kullanın.

- [“Yapının adının belirtilmesi” sayfa 267](#)
- [“Yapının biçimini belirtme” sayfa 268](#)
- [“Yapının sürümünü denetleme” sayfa 268](#)
- [“Bir yapının başka bir yapının içine gömülmüş olarak bildirilmesi” sayfa 268](#)
- [“Alanlar için başlangıç değerlerinin belirtilmesi” sayfa 268](#)
- [“Listelemeyi denetleme” sayfa 269](#)

### Yapının adının belirtilmesi

Bir yapının birden çok örneğini bildirmek için, makro yapıdaki her bir alanın adının başına kullanıcı tarafından belirtilen bir dizgi ve alt çizgi ekler.

Kullanılan dizgi, makronun çağrılmasında belirtilen etikettir. Etiket belirtilmezse, önceki oluşturmak için yapının adı kullanılır:

```
* Declare two object descriptors
      CMQODA ,          Prefix used="MQOD_" (the default)
MY_MQOD CMQODA ,          Prefix used="MY_MQOD_"
```

Bu kısımda gösterilen yapı bildirimleri varsayılan önceki kullanır.

## Yapının biçimini belirtme

Yapı bildirimleri, makro tarafından DSECT değiştirilmesiyle denetlenen iki biçimden birinde oluşturulabilir:

### DSECT = EVET

Yeni bir veri kısmı başlatmak için çevirici DSECT yönergesi kullanılır; yapı tanımı DSECT deyimini hemen izler. Makro çağrısındaki etiket, veri kısmının adı olarak kullanılır; etiket belirtilmezse, yapının adı kullanılır.

### DSECT = HAYIR

Çevirici DC yönergeleri, yapıyı yordamın yürürlükteki konumunda tanımlamak için kullanılır. Alanlar, makro çağrısında ilgili değiştirgelerin kodlanmasıyla belirtilebilecek değerlerle kullanıma hazırlanır. Makro çağrısında değer belirtilmeyen alanlar varsayılan değerlerle kullanıma hazırlanır.

Belirtilen değer büyük harfli olmalıdır. DSECT parametresi belirtilmezse, DSECT = NO varsayılır.

## Yapının sürümünü denetleme

Varsayılan olarak, makrolar her zaman her yapının en son sürümünü bildirir.

Yapıdaki *Version* alanı için bir değer belirtmek üzere VERSION makro parametresini kullanabilirsiniz, ancak bu değiştirge *Version* alanının ilk değerini tanımlar ve gerçekte bildirilen yapının sürümünü denetlemez. Bildirilen yapının sürümünü denetlemek için DCLVER değiştirgesini kullanın:

### DCLVER=YRRLKTEKI

Bildirilen sürüm güncel (en son) sürümdür.

### DCLVER=BELIRTILDI

Bildirilen sürüm, VERSION değiştirgesiyle belirtilen sürümdür. VERSION değiştirgesini atlarsanız, varsayılan değer sürüm 1 'dir.

VERSION değiştirgesini belirtirseniz, değer kendi kendini tanımlayan bir sayısal değişmez ya da gereken sürüm için belirtilen değişmez olmalıdır (örneğin, MQCNO\_VERSION\_3). Başka bir değer belirtirseniz, VERSION değeri geçerli bir değere çözülsün bile, yapı DCLVER=CURRENT belirtilmiş gibi bildirilir.

Belirtilen değer büyük harfli olmalıdır. DCLVER değiştirgesini atlarsanız, kullanılan değer MQDCLVER genel makro değişkeninden alınır. Bu değişkeni CMQVERA makrosunu kullanarak ayarlayabilirsiniz.

## Bir yapının başka bir yapının içine gömülmüş olarak bildirilmesi

Bir yapıyı başka bir yapının bileşeni olarak bildirmek için IÇIÇE değiştirgesini kullanın:

### NESTED=EVET

Yapı bildirimini başka bir yapının içine yerleştirilmiş.

### NESTED=HAYIR

Yapı bildirimini başka bir yapının içine yerleştirilmedi.

Belirtilen değer büyük harfli olmalıdır. IÇIÇE değiştirgesini atlarsanız, NESTED=NO varsayılır.

## Alanlar için başlangıç değerlerinin belirtilmesi

Bir yapıdaki bir alanı kullanıma hazırlamak için kullanılacak değeri, ilgili alanın adını (önek olmadan) makro çağrısında parametre olarak, gerekli değerle birlikte kodlayarak belirtin.

Örneğin, MQMT\_REQUEST ile kullanıma hazırlanan *MsgType* alanıyla bir ileti tanımlayıcı yapısını ve "MY\_REPLY\_TO\_QUEUE" dizgisiyle kullanıma hazırlanan *ReplyToQ* alanını bildirmek için aşağıdakileri kullanın:

```
MY_MQMD CMQMDA MSGTYPE=MQMT_REQUEST, X
          REPLYTOQ=MY_REPLY_TO_QUEUE
```

Makro çağrısında bir değer olarak adlandırılmış bir değişmez (eşit) belirtirseniz, adı belirtilen değişmezi tanımlamak için CMQA makrosunu kullanın. Karakter dizgisi değerlerini tek tırnak içine almayın.



## Listelemeyi denetleme

LIST deęiřtirgesini kullanarak çevirici listesinde yapı bildirimini görünüşünü denetleyin:

### LIST = EVET

Yapı bildirimi, çevirici listesinde görünür.

### LISTE = HAYIR

Yapı bildirimi, çevirici listesinde görünmüyor.

Belirtilen deęer büyük harfli olmalıdır. LIST parametresini çıkarırsanız, LIST = NO varsayılır.

## MQAIR-Kimlik doęrulama bilgileri kaydı

MQAIR yapısı, IBM MQ MQI client olarak çalışan bir uygulamanın istemci bağlantısı için kullanılacak bir kimlik doęrulamayı ilişkin bilgileri belirtmesini sağlar. Yapı, MQCONNX çağrısındaki bir giriş deęiřtirgesidir.

## Kullanılabilirlik

MQAIR yapısı ařaęıdaki istemciler için kullanılabilir:

-  AIX
-  Linux
-  Windows

## Karakter kümesi ve kodlama

MQAIR içindeki veriler, yerel kuyruk yöneticisinin karakter kümesi ve kodlamasında olmalıdır; bunlar **CodedCharSetId** kuyruk yöneticisi öznitelięi ve MQENC\_NATIVE ile verilir.

## Alanlar

**Not:** Ařaęıdaki tabloda, alanlar alfabetik olarak deęil, kullanıma göre gruplanır. Alt konular aynı sırayı izler.

Çizelge 468. MQAIR içindeki alanlar		
Alan adı ve açıklaması	Deęişmezin adı	Deęişmezin ilk deęeri (varsa)
<u>StrucId</u> (yapı tanıtıcısı)	MQAIR_STRUC_ID	'AIR-'
<u>Sürüm</u> (yapı sürümü numarası)	MQAIR_VERSION_1	1
<u>AuthInfoTip</u> (kimlik doęrulama bilgilerinin tipi)	MQAIT_CRL_LDAP	1
<u>AuthInfoConnName</u> (LDAP CRL sunucusunun bağlantı adı)	Yok	Boş deęerli dizilim ya da boşluklar
<u>LDAPUserNamePtr</u> (LDAP kullanıcı adının adresi)	Yok	Boş deęerli gösterge ya da boş deęerli byte
<u>LDAPUserNameGörelili Konum</u> (LDAP kullanıcı adının MQSCO ' nun başlangıcından görelili konumu)	Yok	0
<u>LDAPUserNameLength</u> (LDAP kullanıcı adının uzunluęu)	Yok	0
<u>LDAPPassword</u> (LDAP sunucusuna erişim parolası)	Yok	Boş deęerli dizilim ya da boşluklar
<b>Not:</b> Sürüm MQAIR_VERSION_2' den küçükse kalan alanlar yoksayıılır.		

Çizelge 468. MQAIR içindeki alanlar (devamı var)

Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
OCSPResponderURL (OCSP yanıtlayıcısıyla iletişim kurulabilecek URL)	Yok	Boş değerli dizilim ya da boşluklar
<p><b>Notlar:</b></p> <ol style="list-style-type: none"> <li>↪ simgesi, tek bir boş karakteri temsil eder.</li> <li>C programlama dilinde, makro değişkeniMQAIR_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlar için başlangıç değerleri sağlamak üzere bunu aşağıdaki şekilde kullanın:</li> </ol> <pre>MQAIR MyAIR = {MQAIR_DEFAULT};</pre>		

## Dil bildirimleri

### MQAIR için C bildirimini

```
typedef struct tagMQAIR MQAIR;
struct tagMQAIR {
    MQCHAR4    StrucId;           /* Structure identifier */
    MQLONG     Version;          /* Structure version number */
    MQLONG     AuthInfoType;     /* Type of authentication
                                information */
    MQCHAR264  AuthInfoConnName; /* Connection name of CRL LDAP
                                server */
    PMQCHAR    LDAPUserNamePtr;  /* Address of LDAP user name */
    MQLONG     LDAPUserNameOffset; /* Offset of LDAP user name from start
                                of MQAIR structure */
    MQLONG     LDAPUserNameLength; /* Length of LDAP user name */
    MQCHAR32   LDAPPASSWORD;     /* Password to access LDAP server */
    MQCHAR256  OCSPResponderURL; /* URL of OCSP responder */
};
```

### MQAIR için COBOL bildirimini

```
** MQAIR structure
10 MQAIR.
** Structure identifier
15 MQAIR-STRUCID PIC X(4).
** Structure version number
15 MQAIR-VERSION PIC S9(9) BINARY.
** Type of authentication information
15 MQAIR-AUTHINFOTYPE PIC S9(9) BINARY.
** Connection name of CRL LDAP server
15 MQAIR-AUTHINFOCONNNAME PIC X(264).
** Address of LDAP user name
15 MQAIR-LDAPUSERNAMEPTR POINTER.
** Offset of LDAP user name from start of MQAIR structure
15 MQAIR-LDAPUSERNAMEOFFSET PIC S9(9) BINARY.
** Length of LDAP user name
15 MQAIR-LDAPUSERNAMELENGTH PIC S9(9) BINARY.
** Password to access LDAP server
15 MQAIR-LDAPPASSWORD PIC X(32).
** URL of OCSP responder
15 MQAIR-OCSPRESPONDERURL PIC X(256).
```

### MQAIR için Visual Basic bildirimini

```
Type MQAIR
StrucId As String*4 'Structure identifier'
Version As Long 'Structure version number'
AuthInfoType As Long 'Type of authentication information'
AuthInfoConnName As String*264 'Connection name of CRL LDAP server'
```

LDAPUserNamePtr	As MQPTR	'Address of LDAP user name'
LDAPUserNameOffset	As Long	'Offset of LDAP user name from start' 'of MQAIR structure'
LDAPUserNameLength	As Long	'Length of LDAP user name'
LDAPPassword	As String*32	'Password to access LDAP server'
End Type		

### ***StrucId (MQCHAR4)***

Değer şu olmalıdır:

#### **MQAIR\_STRUCT\_ID**

Kimlik doğrulama bilgileri kaydına ilişkin tanıtıcı.

C programlama dili için, constant MQAIR\_STRUCT\_ID\_ARRAY de tanımlanır; bu değer MQAIR\_STRUCT\_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisidir.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQAIR\_STRUCT\_ID 'dir.

### ***Sürüm (MQUZE)***

MQAIR yapısının sürüm numarası.

Değer aşağıdakilerden biri olmalıdır:

#### **MQAIR\_VERSION\_1**

Version-1 kimlik doğrulama bilgileri kaydı.

#### **MQAIR\_VERSION\_2**

Version-2 kimlik doğrulama bilgileri kaydı.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

#### **MQAIR\_CURRENT\_SÜRÜM**

Kimlik doğrulama bilgileri kaydının geçerli sürümü.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQAIR\_VERSION\_1' dir.

### ***AuthInfoTip (MQUBE)***

Bu, kayıta bulunan kimlik doğrulama bilgilerinin tipidir.

Değer, aşağıdaki iki değştirgelerden biri olabilir:

#### **MQAIT\_CRL\_LDAP**

LDAP sunucusu kullanılarak sertifika iptal denetimi.

#### **MQAIT\_OCSP**

OCSP kullanılarak sertifika iptal denetimi.

Değer geçerli değilse, çağrı neden kodu MQRC\_AUTH\_INFO\_TYPE\_ERROR neden ile başarısız olur.

Bu bir giriş alanıdır. Bu alanın ilk değeri MQAIT\_CRL\_LDAP ' dir.

### ***AuthInfoConnName (MQCHAR264)***

Bu ad, LDAP sunucusunun çalışmakta olduğu anasistemin adı ya da ağ adresi. Bunu, parantez içine alınmış isteğe bağlı bir kapı numarası takip edebilir. Varsayılan kapı numarası 389'dur.

Değer, alanın uzunluğundan kısaysa, değeri boş değerli bir karakterle sonlandırın ya da alanın uzunluğuna göre boşluklarla doldurur. Değer geçersiz ise, çağrı neden kodu MQRC\_AUTH\_INFO\_CONN\_NAME\_ERROR neden ile başarısız olur.

Bu bir giriş alanıdır. Bu alanın uzunluğu MQ\_AUTH\_INFO\_CONN\_NAME\_LENGTH tarafından verilir. Bu alanın ilk değeri, C içindeki boş değerli dizidir ve diğer programlama dillerindeki boşluk karakteridir.

### ***LDAPUserNamePtr (PMQCHAR)***

Bu, LDAP kullanıcı adıdır.

Bu, LDAP CRL sunucusuna erişmeyi deneyen kullanıcının Ayırt Edici Adı ile oluşur. If the value is shorter than the length specified by *LDAPUserNameLength*, terminate the value with a null character, or pad it with blanks to the length *LDAPUserNameLength*. *LDAPUserNameLength* sıfırda, alan yoksayıdır.

LDAP kullanıcı adını aşağıdaki iki yoldan birini kullanarak temin edebilirsiniz:

- *LDAPUserNamePtr* işaretçi alanını kullanarak

Bu durumda uygulama, MQAIR yapısından ayrı bir dizgi bildirebilir ve *LDAPUserNamePtr*, dizginin adresine ayarlıdır.

Farklı ortamlara (örneğin, C programlama dili gibi) taşınabilir bir şekilde işaretçi veri tipini destekleyen programlama dilleri için *LDAPUserNamePtr* 'i kullanmayı düşünün.

- By using the offset field *LDAPUserNameOffset*

Bu durumda, uygulamanın MQSCO yapısını içeren bir bileşik yapı bildirmesi gerekir; bunu izleyen MQAIR kayıtları dizisi ve ardından LDAP kullanıcı adı dizgileri gelir ve *LDAPUserNameOffset* olarak, MQAIR yapısının başlangıcındaki uygun ad dizgisinin görelî konuta ayarlanır. Bu değerin doğru olduğundan ve bir MQlong (en kısıtlayıcı programlama dili COBOL) içinde konaklayabilecek bir değere sahip olduğundan emin olun (geçerli aralık -999 999 999 ile +999 999 999 aralığında).

İşaretçi veri tipini desteklemeyen programlama dilleri için *LDAPUserNameOffset*, ya da işaretçi veri tipini farklı ortamlara (örneğin, COBOL programlama dili gibi) portatif olmayabilecek bir şekilde uygulamayı düşünün.

Hangi teknik seçiliyse, *LDAPUserNamePtr* ve *LDAPUserNameOffset* 'den yalnızca birini kullanın; Her ikisi de sıfır dışında bir durumda, çağrı neden kodu MQRC\_LDAP\_USER\_NAME\_ERROR ile başarısız olur.

Bu bir giriş alanıdır. Bu alanın ilk değeri, göstergeleri destekleyen programlama dillerindeki boş değerli göstergedir ve tersi durumda, boş değerli byte dizilimidir.

**Not:** Programlama dilinin işaretçi veri tipini desteklemediği altyapılarda, bu alan uygun uzunluğun byte dizgisi olarak bildirilir.

### ***LDAPUserNameGörelî Konumu (MQUZE)***

Bu, MQAIR yapısının başlangıcındaki LDAP kullanıcı adının byte cinsinden görelî konudur.

Görelî konum pozitif ya da negatif olabilir. *LDAPUserNameLength* sıfırda, alan yoksayıdır.

LDAP kullanıcı adını belirtmek için *LDAPUserNamePtr* ya da *LDAPUserNameOffset* seçeneğini kullanabilirsiniz, ancak her ikisi de değil; ayrıntılar için *LDAPUserNamePtr* alanının açıklamasına bakın.

Bu bir giriş alanıdır. Bu alanın ilk değeri 0 'tır.

### ***LDAPUserNameUzunluk (MQUZE)***

Bu, *LDAPUserNamePtr* ya da *LDAPUserNameOffset* alanı tarafından adreslenen LDAP kullanıcı adının byte cinsinden uzunluğudur. Değer, MQ\_DISTINGUISH\_NAME\_LENGTH ile sıfır aralığında olmalıdır. Değer geçerli değilse, çağrı neden kodu MQRC\_LDAP\_USER\_NAME\_LENGTH\_ERR neden kodlarıyla başarısız olur.

İlgili LDAP sunucusu için bir kullanıcı adı gerekmiyorsa, bu alanı sıfır olarak ayarlayın.

Bu bir giriş alanıdır. Bu alanın ilk değeri 0 'tır.

### ***LDAPPassword (MQCHAR32)***

Bu, LDAP CRL sunucusuna erişmek için gerekli olan paroladır. Değer, alanın uzunluğundan kısaysa, değeri boş değerli bir karakterle sonlandırın ya da alanın uzunluğuna göre boşluklarla doldurur.

LDAP sunucusu parola gerektirmiyorsa ya da LDAP kullanıcı adını atarsanız, *LDAPPassword* boş değerli ya da boş olmalıdır. LDAP kullanıcı adını atarsanız ve *LDAPPassword* boş değerli ya da boş değilse, çağrı neden kodu MQRC\_LDAP\_PASSWORD\_ERROR neden ile başarısız olur.

Bu bir giriş alanıdır. Bu alanın uzunluğu MQ\_LDAP\_PASSWORD\_LENGTH tarafından verilmektedir. Bu alanın ilk değeri, C içindeki boş değerli dizidir ve diğer programlama dillerindeki boşluk karakteridir.

## OCSPResponderURL (MQCHAR256)

OCSP yanıtlayıcıya ilişkin bağlantı ayrıntılarını temsil eden bir MQAIR yapısı için bu alan, yanıt verenin iletişim kurabileceği URL 'yi içerir.

Bu alanın değeri bir HTTP URL 'dir. Bu alan, AuthorityInfoAccess (AIA) sertifika uzantısındaki bir URL 'nin önceliğini alır.

Aşağıdaki deyimler doğru değilse, değer yoksayılr:

- MQAIR yapısı Sürüm 2 ya da üsttür (Sürüm alanı MQAIR\_VERSION\_2 ya da üstü olarak ayarlıdır).
- AuthInfoTip alanı MQAIT\_OCSP olarak ayarlıdır.

Alan, doğru biçimde bir HTTP URL adresi içermiyorsa (ve yoksayılmamışsa), MQCONNX çağrısı hata kodu MQRC\_OCSP\_URL\_ERROR ile başarısız olur.

Bu alan büyük ve küçük harfe duyarlıdır. It must start with the string http:// in lowercase. URL 'nin geri kalanı, OCSP sunucusu uygulamasına bağlı olarak büyük/küçük harf duyarlı olabilir.

Bu alan veri dönüştürmeye tabi değildir.

## MQBMHO-İleti işleme seçenekleri için arabellek

MQBMHO yapısı, uygulamaların, ileti tanıtıcılarının arabelleklerden nasıl üretildiğini denetleyen seçenekleri belirtmesine olanak sağlar. Yapı, MQBUFMH çağrısındaki bir giriş değiştirgesidir.

## Karakter kümesi ve kodlama

MQBMHO içindeki veriler, uygulamanın karakter kümesinde ve uygulamanın kodlamasında olmalıdır (MQENC\_NATIVE).

## Alanlar

**Not:** Aşağıdaki tabloda, alanlar alfabetik olarak değil, kullanıma göre gruplanır. Alt konular aynı sırayı izler.

Çizelge 469. MQBMHO içindeki alanlar		
Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
StrucId (yapı tanıtıcısı)	MQBMHO_STRUC_ID	'BMHO'
Sürüm (yapı sürümü numarası)	MQBMHO_VERSION_1	1
Seçenekler (MQBMHO işlemini denetleyen seçenekler)	MQBMHO_NONE	0

**Notlar:**

1. C programlama dilinde, makro değişkeniMQBMHO\_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlar için başlangıç değerleri sağlamak üzere bunu aşağıdaki şekilde kullanın:

```
MQBMHO MyBMHO = {MQBMHO_DEFAULT};
```

## Dil bildirimleri

MQBMHO için C bildirimi

```
typedef struct tagMQBMHO MQBMHO;
struct tagMQBMHO {
    MQCHAR4 StrucId;          /* Structure identifier */
    MQLONG  Version;         /* Structure version number */
    MQLONG  Options;        /* Options that control the action of
```

```
}; MQBUFMH */
```

### MQBMHO için COBOL bildirimi

```
** MQBMHO structure
10 MQBMHO.
** Structure identifier
15 MQBMHO-STRUCID PIC X(4).
** Structure version number
15 MQBMHO-VERSION PIC S9(9) BINARY.
** Options that control the action of MQBUFMH
15 MQBMHO-OPTIONS PIC S9(9) BINARY.
```

### MQBMHO için PL/I bildirimi

```
Dcl
1 MQBMHO based,
3 StrucId char(4), /* Structure identifier */
3 Version fixed bin(31), /* Structure version number */
3 Options fixed bin(31), /* Options that control the action
of MQBUFMH */
```

### MQBMHO için High Level Assembler bildirimi

```
MQBMHO DSECT
MQBMHO_STRUCID DS CL4 Structure identifier
MQBMHO_VERSION DS F Structure version number
MQBMHO_OPTIONS DS F Options that control the
* action of MQBUFMH
MQBMHO_LENGTH EQU *-MQBMHO
MQBMHO_AREA DS CL(MQBMHO_LENGTH)
```

### **StrucId (MQCHAR4)**

Arabellek-ileti tanıtıcısı yapısı- StrucId alanı

Bu, yapı tanıtıcısıdır. Değer şu olmalıdır:

#### **MQBMHO\_STRUC\_ID**

İleti tanıtıcısı yapısına arabellek için tanıtıcı.

C programlama dili için, MQBMHO\_STRUC\_ID\_ARRAY değişmezi de tanımlanır; bu, MQBMHO\_STRUC\_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisidir.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQBMHO\_STRUC\_ID 'dir.

### **Sürüm (MQUZE)**

Arabellek-ileti tanıtıcısı yapısı-Sürüm alanı

Bu, yapı sürüm numarasıdır. Değer şu olmalıdır:

#### **MQBMHO\_VERSION\_1**

Arabellek için ileti tanıtıcısı yapısına ilişkin sürüm numarası.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

#### **MQBMHO\_CURRENT\_VERSION**

Arabellek, ileti tanıtıcısı yapısına ilişkin yürürlükteki sürüm.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQBMHO\_VERSION\_1' dir.

### **Seçenekler (MQUZE)**

Arabellek-ileti tanıtıcısı yapısı-Seçenekler alanı

Değer şu şekilde olabilir:

## MQBMO\_DELETE\_PROPERTIES

İleti tutamaçına eklenen özellikler arabellekten silinir. Arama başarısız olursa özellikler silinmez.

Varsayılan seçenekler: Açıklanmanız gerekmiyorsa, aşağıdaki seçeneği kullanın:

## MQBMO\_YOK

Seçenek belirtilmedi.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQBMO\_DELETE\_PROPERTIES olarak değerlendirilmektedir.

## V 9.2.4 MQBNO-Dengeleme seçenekleri

Aşağıdaki tablo, yapıdaki alanları özetler.

### Alanlar

**Not:** Aşağıdaki tabloda, alanlar alfabetik olarak değil, kullanıma göre gruplanır. Alt konular aynı sırayı izler.

Çizelge 470. MQBNO içindeki alanlar		
Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
<u>StrucId</u> (yapı tanıtıcısı)	MQBNO_STRUC_ID	'BNO↵'
<u>Sürüm</u> (yapı sürümü numarası)	MQBNO_VERSION_1	1
<u>ApplicationType</u> (yapıda ayarlanan dengeleme seçeneğinin tipi)	MQBNO_VALTYPE_SIMPLE	0
<u>Zamanaşımı</u> (yeniden dengeleme, uygulama etkinliğini kesintiye uğratabilir)	MQBNO_TIMEOUT_AS_DEFAULT	0
<u>BalanceOptions</u> (düzenleyen uygulama tarafından ayarlanan dengeleme seçenekleri)	MQBNO_OPTIONS_NONE	0

**Notlar:**

- ↵ simgesi, tek bir boş karakteri temsil eder.
- C programlama dilinde, MQBNO\_DEFAULT makro değişkeni çizelgede listelenen değerleri içerir. Yapıdaki alanlar için başlangıç değerleri sağlamak üzere bunu aşağıdaki şekilde kullanın:

```
MQBNO MyBNO = {MQBNO_DEFAULT};
```

### Dil bildirimleri

MQBNO için C bildirimi

```
typedef struct tagMQBNO MQBNO;
struct tagMQBNO {
    MCHAR4    StrucId;          /* Structure identifier */
    MQLONG    Version;         /* Structure version number */
    MQLONG    Type;           /* Type of balancing options set in the
                             structure */
    MQLONG    Timeout;        /* Timeout after which re-balancing might
                             interrupt application activity */
    MQLONG    BalanceOptions; /* Balancing options set by the issuing
                             application */
};
```

## MQBNO için COBOL bildirimi

```
** MQBNO structure
10 MQBNO.
** Structure identifier
15 MQBNO-STRUCID PIC X(4).
** Structure version number
15 MQBNO-VERSION PIC S9(9) BINARY.
** Type of balancing options set in the structure
15 MQBNO-TYPE PIC S9(9) BINARY.
** Timeout after which re-balancing might interrupt application activity
15 MQBNO-TIMEOUT PIC S9(9) BINARY.
** Balancing options set by the issuing application
15 MQBNO-BALANCEOPTIONS PIC S9(9) BINARY.
```

## MQBNO için PL/I bildirimi

```
dcl
1 MQBNO based,
3 StrucId char(4), /* Structure identifier */
3 Version fixed bin(31), /* Structure version number */
3 Type fixed bin(31), /* Type of balancing options set in the
structure*/
3 Timeout fixed bin(31), /* Timeout after which re-balancing might
interrupt application activity */
3 BalanceOptions fixed bin(31), /* Balancing options set by the issuing
application*/
```

## İlgili başvurular

“MQCNO-Bağlantı seçenekleri” sayfa 314

MQCNO yapısı, uygulamanın kuyruk yöneticisine yönelik bağlantıyla ilgili seçenekleri belirtmesine olanak sağlar. Yapı, MQCONNX çağrısındaki bir giriş/çıkış değiştirgesidir.

### V 9.2.4 StrucId (MQCHAR4)

StrucId her zaman bir giriş alanıdır. Başlangıç değeri BNO ' dur.

Değer şu olmalıdır:

#### BNO

Dengeleme seçenekleri yapısına ilişkin tanıttıcı.

C programlama dili için, sabit MQBNO\_STRUC\_ID\_ARRAY değişmezi de tanımlanır; bu değişmez BNO ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisidir.

**StrucId** ya da MQRC\_BNO\_ERROR için geçerli bir değer sağlamanız gerekir.

### V 9.2.4 Sürüm (MQUZE)

Sürüm her zaman bir giriş alanıdır. İlk değeri MQBNO\_VERSION\_1' dir.

Değer şu olmalıdır:

#### MQBNO\_VERSION\_1

Version-1 dengeleme-seçenekler yapısı.

**Version** ya da MQRC\_BNO\_ERROR için geçerli bir değer sağlamanız gerekir.

### V 9.2.4 ApplicationType (MQUSEL)

Yapıdaki dengeleme seçeneğinin tipi.

Olası değerler şunlardır:

#### MQBNO\_BALTYPE\_SIMPLE

Basit dengeleme; Tek tip kümelerde uygulama yeniden dengelemeinde açıklananlara ek olarak belirli kurallar uygulanmaz.



## **MQBNO\_BALTYPE\_REQREP**

İstek-Yanıt dengeleme; her bir MQPUT çağrısından sonra, bir yanıt iletisi için eşleşen bir MQGET çağrısının beklenmesi beklenir. Böyle bir ileti alınana kadar dengeleme gecikmeli ya da istek iletisi EXPIRY aşılmıştır.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQBNO\_BALTYPE\_SIMPLE değerine sahip.

Yalnızca **ApplicationType** alanı için bir değer sağlamalısınız ya da MQRC\_BNO\_ERROR geri döndürülür.

**Not:** An additional value for this field of MQBNO\_BALTYPE\_RA\_MANAGED is reserved for use by the IBM MQ Resource Adapter for JEE environments. Bir uygulamanın bu değeri doğrudan sağlamasına ilişkin bir hata olmakla birlikte, örneğin, uygulama durumu sorgulanırken raporlanabilir.

## **V 9.2.4 Zamanaşımı (MQUZE)**

Yeniden dengelemeyi yeniden dengeledikten sonra **Timeout** , uygulama etkinliğini kesintiye uğratabilir.

Olası değerler şunlardır:

### **MQBNO\_TIMEOUT\_AS\_VARSAYILAN**

Set default timeout değeri.

### **MQBNO\_TIMEOUT\_IMMEDIATE**

Anında zamanaşımı oluşur.

### **MQBNO\_TIMEOUT\_HIR**

Zamanaşımı oluşmaz.

Bu alanın ilk değeri MQBNO\_TIMEOUT\_AS\_VARSAYILANDIR.

Yalnızca tanımlı değerlerden bir değer ya da **Timeout** alanı için 0-999999999 saniye,returnedalanı için ya da MQRC\_BNO\_ERROR değeri döndürülmelidir.

## **V 9.2.4 BalanceOptions (MQLONG)**

Verme uygulaması tarafından ayarlanan dengeleme seçenekleri.

Olası değerler şunlardır:

### **MQBNO\_OPTIONS\_NONE**

Hiçbir seçenek ayarlanmadı

### **MQBNO\_OPTIONS\_IGNORE\_TRANS**

Bu seçeneğin ayarlanması, bir hareketin ortasında bile uygulamaların yeniden dengelenmesine olanak sağlar.

Bu alanın ilk değeri MQBNO\_OPTIONS\_NONE olur.

You can provide any combination of the defined values using the logical or character for the **BalanceOptions** field. Geçerli olmayan değerler, MQRC\_BNO\_ERROR ' in döndürülmesine neden olur.

## **MQBO-Başlangıç seçenekleri**

MQBO yapısı, uygulamanın bir iş birimi yaratılmasıyla ilgili seçenekleri belirtmesine olanak sağlar. Yapı, MQBEGIN çağrısında bir giriş/çıkış değiştirgesidir.

## **Kullanılabilirlik**

MQBO yapısı aşağıdaki altyapılarda bulunur:

-  AIX
-  IBM i
-  Linux
-  Windows

IBM MQ MQI clients için MQBO yapısı kullanılmıyor.

## Karakter kümesi ve kodlama

MQBO 'daki veriler, **CodedCharSetId** kuyruk yöneticisi özneliği ve MQENC\_NATIVE tarafından verilen yerel kuyruk yöneticisinin kodlaması tarafından belirtilen karakter kümesinde olmalıdır. Ancak, uygulama bir MQ MQI istemcisi olarak çalışıyorsa, yapı istemcinin karakter kümesinde ve kodlamasında olmalıdır.

## Alanlar

**Not:** Aşağıdaki tabloda, alanlar alfabetik olarak değil, kullanıma göre gruplanır. Alt konular aynı sırayı izler.

Çizelge 471. MQBO için MQBO 'daki Alanlar		
Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
<u>StrucId</u> (yapı tanıtıcısı)	MQBO_STRUC_ID	'BO↯↯'
<u>Sürüm</u> (yapı sürümü numarası)	MQBO_VERSION_1	1
<u>Seçenekler</u> (MQBEGIN işlemini denetleyen seçenekler)	MQBO_NONE	0

**Notlar:**

- ↯ simgesi, tek bir boş karakteri temsil eder.
- C programlama dilinde, makro değişkeni MQBO\_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlar için başlangıç değerleri sağlamak üzere bunu aşağıdaki şekilde kullanın:

```
MQBO MyBO = {MQBO_DEFAULT};
```

## Dil bildirimleri

MQBO için C bildirimini

```
typedef struct tagMQBO MQBO;
struct tagMQBO {
    MQCHAR4 StrucId; /* Structure identifier */
    MQLONG Version; /* Structure version number */
    MQLONG Options; /* Options that control the action of MQBEGIN */
};
```

MQBO için COBOL bildirimini

```
** MQBO structure
10 MQBO.
** Structure identifier
15 MQBO-STRUCID PIC X(4).
** Structure version number
15 MQBO-VERSION PIC S9(9) BINARY.
** Options that control the action of MQBEGIN
15 MQBO-OPTIONS PIC S9(9) BINARY.
```

MQBO için PL/I bildirimini

```
dcl
1 MQBO based,
3 StrucId char(4), /* Structure identifier */
3 Version fixed bin(31), /* Structure version number */
3 Options fixed bin(31); /* Options that control the action of
MQBEGIN */
```

## MQBO için Visual Basic bildirimi

```
Type MQBO
  StrucId As String*4 'Structure identifier'
  Version As Long     'Structure version number'
  Options As Long     'Options that control the action of MQBEGIN'
End Type
```

### **StrucId (MQCHAR4)**

Bu alan her zaman bir giriş alanıdır. Başlangıç değeri MQBO\_STRUC\_ID ' dir.

Değer şu olmalıdır:

#### **MQBO\_STRUC\_ID**

Başlangıç seçenekleri yapısına ilişkin tanıttıcı.

C programlama dili için, aynı zamanda MQBO\_STRUC\_ID\_ARRAY değişmezi de tanımlanır; bu, MQBO\_STRUC\_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisi olur.

### **Sürüm (MQUZE)**

Bu alan her zaman bir giriş alanıdır. Başlangıç değeri MQBO\_VERSION\_1 ' dir.

Değer şu olmalıdır:

#### **MQBO\_VERSION\_1**

Başlangıç seçenekleri yapısına ilişkin sürüm numarası.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

#### **MQBO\_CURRENT\_VERSION**

Başlangıç seçenekleri yapısının yürürlükteki sürümü.

### **Seçenekler (MQUZE)**

Bu alan her zaman bir giriş alanıdır. İlk değeri MQBO\_NONE olur.

Değer şu olmalıdır:

#### **MQBO\_NONE**

Seçenek belirtilmedi.

## **MQCBC-Geri çağırma bağlamı**

MQCBC yapısı, bir geri çağırma işlevine geçirilen bağlam bilgilerini belirtmek için kullanılır. Yapı, bir ileti tüketicisi yordamına yapılan çağrıdaki bir giriş/çıkış değiştirgesidir.

### **Kullanılabilirlik**

MQCBC yapısı aşağıdaki altyapılarda bulunur:

-  AIX
-  IBM i
-  Linux
-  Windows
-  z/OS

ve bu sistemlere bağlı IBM MQ MQI clients için.

### **Sürüm**

MQCBC ' nin yürürlükteki sürümü MQCBC\_VERSION\_2.

## Karakter kümesi ve kodlama

MQCBC içindeki veriler, **CodedCharSetId** kuyruk yöneticisi özneliği ve MQENC\_NATIVE tarafından verilen yerel kuyruk yöneticisinin kodlaması tarafından belirtilen karakter kümesinde olmalıdır. Ancak, uygulama bir MQ MQI istemcisi olarak çalışıyorsa, yapı istemcinin karakter kümesinde ve kodlamasında olur.

## Alanlar

**MQCBC** yapısı için başlangıç değeri yoktur. Yapı, bir geri çağırma yordamına parametre olarak geçirilir. Kuyruk yöneticisi yapıyı kullanıma hazırlar; uygulamalar yapıyı hiçbir zaman kullanıma hazırlamaz.

### Notlar:

- Aşağıdaki tabloda, alanlar alfabetik olarak değil, kullanıma göre gruplanır. Alt konular aynı sırayı izler.
- MQCBC yapısı için başlangıç değeri yok. Yapı, bir geri çağırma yordamına parametre olarak geçirilir. Kuyruk yöneticisi yapıyı kullanıma hazırlar; uygulamalar yapıyı hiçbir zaman kullanıma hazırlamaz.

Çizelge 472. MQCBC içindeki alanlar	
Alan	Açıklama
<u>StrucID</u>	Yapı tanıtıcısı
<u>Sürüm</u>	Yapı sürümü numarası
<u>CallType</u>	İşlev neden çağırıldı
<u>Hobj</u>	Nesne tanıtıcısı
<u>CallbackArea</u>	Geri çağırma işlevinin kullanılacağı alan
<u>ConnectionArea</u>	Geri çağırma işlevinin kullanılacağı alan
<u>CompCode</u>	Tamamlanma kodu
<u>Neden</u>	Neden Kodu
<u>STATE</u>	Geçerli tüketicinin durumunun göstergesi
<u>DataLength</u>	İleti uzunluğu
<u>BufferLength</u>	İleti arabelleğinin bayt cinsinden uzunluğu
<u>İşaretler</u>	Genel işaretler
<b>Not:</b> Sürüm MQCBC_VERSION_2 değerinden küçükse, geri kalan alan yoksayılır	
<u>ReconnectDelay</u>	Yeniden bağlanma girişiminden önceki milisaniye sayısı

## Dil bildirimleri

MQCBC için C bildirimi

```
typedef struct tagMQCBC MQCBC;
struct tagMQCBC {
    MQCHAR4    StrucId;                /* Structure identifier */
    MQLONG     Version;                /* Structure version number */
    MQLONG     CallType;               /* Why Function was called */
    MQHOBj     Hobj;                  /* Object Handle */
    MQPTR      CallbackArea;           /* Callback data passed to the function */
    MQPTR      ConnectionArea;         /* MQCTL data area passed to the function */
    MQLONG     CompCode;               /* Completion Code */
    MQLONG     Reason;                 /* Reason Code */
    MQLONG     State;                  /* Consumer State */
    MQLONG     DataLength;             /* Message Data Length */
    MQLONG     BufferLength;           /* Buffer Length */
    MQLONG     Flags;                  /* Flags containing information about
```

```

                                this consumer */
/* Ver:1 */
MQLONG   ReconnectDelay;      /* Number of milliseconds before */
/* Ver:2 */ };                /* reconnect attempt */

```

## MQCBC için COBOL bildirimi

```

** MQCBC structure
 10 MQCBC.
** Structure Identifier
 15 MQCBC-STRUCID                PIC X(4).
** Structure Version
 15 MQCBC-VERSION                PIC S9(9) BINARY.
** Call Type
 15 MQCBC-CALLTYPE              PIC S9(9) BINARY.
** Object Handle
 15 MQCBC-HOBJ                  PIC S9(9) BINARY.
** Callback User Area
 15 MQCBC-CALLBACKAREA          POINTER
** Connection Area
 15 MQCBC-CONNECTIONAREA        POINTER
** Completion Code
 15 MQCBC-COMPCODE              PIC S9(9) BINARY.
** Reason Code
 15 MQCBC-REASON                PIC S9(9) BINARY.
** Consumer State
 15 MQCBC-STATE                 PIC S9(9) BINARY.
** Data Length
 15 MQCBC-DATALENGTH            PIC S9(9) BINARY.
** Buffer Length
 15 MQCBC-BUFFERLENGTH          PIC S9(9) BINARY.
** Flags
 15 MQCBC-FLAGS                 PIC S9(9) BINARY.
** Ver:1 **
** Number of milliseconds before reconnect attempt
 15 MQCBC-RECONNECTDELAY        PIC S9(9) BINARY.
** Ver:2 **

```

## MQCBC için PL/I bildirimi

```

dcl
 1 MQCBC based,
 3 StrucId          char(4),          /* Structure identifier */
 3 Version          fixed bin(31),    /* Structure version */
 3 CallType        fixed bin(31),    /* Callback type */
 3 Hobj            fixed bin(31),    /* Object Handle */
 3 CallbackArea    pointer,          /* User area passed to the function */
 3 ConnectionArea  pointer,          /* Connection User Area */
 3 CompCode        fixed bin(31);    /* Completion Code */
 3 Reason          fixed bin(31);    /* Reason Code */
 3 State           fixed bin(31);    /* Consumer State */
 3 DataLength      fixed bin(31);    /* Message Data Length */
 3 BufferLength     fixed bin(31);    /* Message Buffer length */
 3 Flags           fixed bin(31);    /* Consumer Flags */
/* Ver:1 */
 3 ReconnectDelay  fixed bin(31);    /* Number of milliseconds before */
/* Ver:2 */                /* reconnect attempt */

```

## MQCBC için High Level Assembler (Üst Düzey Çevirici) bildirimi

MQCBC	DSECT	
MQCBC	DS 0F	Force fullword alignment
MQCBC_STRUCID	DS CL4	Structure identifier
MQCBC_VERSION	DS F	Structure version number
MQCBC_CALLTYPE	DS F	Why Function was called
MQCBC_HOBJ	DS F	Object Handle
MQCBC_CALLBACKAREA	DS A	Callback data passed to the function
MQCBC_CONNECTIONAREA	DS A	MQCTL Data area passed to the function
MQCBC_COMPCODE	DS F	Completion Code
MQCBC_REASON	DS F	Reason Code
MQCBC_STATE	DS F	Consumer State
MQCBC_DATALENGTH	DS F	Message Data Length
MQCBC_BUFFERLENGTH	DS F	Buffer Length
MQCBC_FLAGS	DS F	Flags containing information about this consumer
MQCBC_RECONNECTDELAY	DS F	Number of milliseconds before reconnect

MQCBC_LENGTH	EQU *-MQCBC
MQCBC_AREA	ORG MQCBC DS CL(MQCBC_LENGTH)

### **StrucId (MQCHAR4)**

Bu alandaki değer yapı tanıtıcısıdır.

Değer şu olmalıdır:

#### **MQCBC\_STRUC\_ID**

Geri çağırma bağlamı yapısına ilişkin tanıtıcı.

C programlama dili için, sabit MQCBC\_STRUC\_ID\_ARRAY değişmezi de tanımlanır; bu, MQCBC\_STRUC\_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisi olur.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQCBC\_STRUC\_ID 'dir.

### **Sürüm (MQUZE)**

Bu alandaki değer, yapı sürüm numarasıdır.

Değer şu olmalıdır:

#### **MQCBC\_VERSION\_1**

Version-1 geri bildirme bağlamı yapısı.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

#### **MQCBC\_CURRENT\_VERSION**

Geri çağırma bağlamı yapısının yürürlükteki sürümü.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQCBC\_VERSION\_1' dir.

Geri bildirme işlevi her zaman yapının en son sürümünü geçirilir.

### **CallType (MQUZE)**

Bu işlevin neden çağırıldığı hakkında bilgi içeren alan; aşağıdaki değerler tanımlanmıştır.

İleti teslim çağırısı tipleri: Bu çağrı tipleri bir iletiyle ilgili bilgi içerir. **DataLength** ve **BufferLength** parametreleri bu çağrı tipleri için geçerlidir.

#### **MQCBCT\_MSG\_REMOVED**

İleti tüketici işlevi, nesne tanıtıcısından yok edici bir şekilde kaldırılmış bir iletiyle çağırıldı.

*CompCode* değeri MQCC\_UYARI ise, *Reason* alanının değeri MQRC\_TRUNCATED\_MSG\_ACCEPTED ya da bir veri dönüştürme sorununu belirten kodlardan biri olur.

#### **MQCBCT\_MSG\_NOT\_REMOVED**

İleti tüketici işlevi, nesne tanıtıcısından henüz yok edici bir şekilde kaldırılmamış bir iletiyle çağırıldı.

İleti, *MsgToken* komutunu kullanarak, nesne tanıtıcısından yok edici bir şekilde kaldırılabilir.

İleti şu nedenle kaldırılmamış olabilir:

- MQGMO seçenekleri için bir göz atma işlemi istendi, MQGMO\_BROWSE\_\*
- İleti, kullanılabilir arabellekten daha büyük ve MQGMO seçenekleri MQGMO\_ACCEPT\_TRUNCATED\_MSG belirtmiyor

*CompCode* ' in değeri MQCC\_UYARI ise, *Reason* alanının değeri MQRC\_TRUNCATED\_MSG\_FAILED ya da bir veri dönüştürme sorununu belirten kodlardan biri.

Geri çağırma denetim çağırısı tipleri: Bu çağrı tipleri, geri bildirimden denetime ilişkin bilgileri içerir ve bir iletiyle ilgili ayrıntıları içermez. Bu çağrı tipleri, MQCBD yapısındaki Seçenekler ' i kullanarak istenir.

**DataLength** ve **BufferLength** parametreleri bu çağrı tipleri için geçerli değildir.

#### **MQCBCT\_REGISTER\_CALL**

Bu çağrı tipinin amacı, geri bildirme işlevinin bazı ilk ayarları gerçekleştirilmesine izin vermesidir.

Geri bildirme işlevi çağrıldığında, geri bildirme kaydedildikten hemen sonra, bu durumda MÖCB çağrısından *Operation* alanı için bir değer kullanılarak bir MÖCB çağrısının geri döndürülmesi gerekir.

Bu arama tipi hem ileti tüketicileri hem de olay işleyicileri için kullanılır.

İstenirse, bu, geri çağırma işlevinin ilk çağrısıdır.

*Reason* alanının değeri MÖRC\_NONE olur.

### **MÖCBCT\_START\_CALL**

Bu çağrı tipinin amacı, geri bildirme işlevinin başlatıldığı zaman bazı ayarları gerçekleştirmesine izin vermesidir; örneğin, daha önce durdurulduğunda temizlenen kaynakları yeniden yürürlüğe almak.

Geri bildirme işlevi, bağlantı MÖOP\_START ya da MÖOP\_START\_WAT kullanılarak başlatıldığında çağrılır.

Bir geri bildirme işlevi başka bir geri çağırma işlevinde kaydedilirse, bu çağrı tipi geri çağırma döndürdüğünde çağrılır.

Bu arama tipi yalnızca ileti kullanıcıları için kullanılır.

*Reason* alanının değeri MÖRC\_NONE olur.

### **MÖCBCT\_STOP\_CALL**

Bu çağrı tipinin amacı, geri bildirme işlevinin bir süre durdurulduğunda temizleme işlemini gerçekleştirmesine izin vermesidir; örneğin, iletilerin tüketilmesi sırasında edinilen ek kaynakları temizlemesidir.

Geri çağırma işlevi, MÖOP\_STOP için *Operation* alanı için bir değer kullanılarak bir MÖCTL çağrısı yayınlandığında çağrılır.

Bu arama tipi yalnızca ileti kullanıcıları için kullanılır.

*Reason* alanının değeri, durmanın nedenini gösterecek şekilde ayarlanır.

### **MÖCBCT\_DEREGISTIER\_CALL**

Bu çağrı tipinin amacı, geri bildirme işlevinin tüketmek işleminin sonunda son temizlemeyi gerçekleştirmesine izin vermesidir. Geri bildirme işlevi aşağıdaki durumlarda çağrılır:

- Geri bildirme işlevi, MÖOP\_DEREGISTER ile bir MÖCB çağrısı kullanılarak kayıttan kaldırılır.
- Kuyruk kapatıldı, örtülü bir kayıttan çıkarıcı neden oluyor. Bu örnekte, geri bildirme işlevi nesne tanıtıcısı olarak MÖHO\_UNUABLE\_HOBJ ' i iletir.
- MÖDISC çağrısı tamamlandı-örtülü kapanmaya neden oluyor ve bu nedenle bir kayıt kaldırma işlemi tamamlandı. Bu durumda, bağlantının bağlantısı kesilir ve devam eden hareket henüz kesinleştirilmedi.

Bu eylemlerden herhangi biri geri bildirme işlevinin kendisi içinde alınırsa, geri çağırma döndürdüktan sonra işlem çağrılır.

Bu arama tipi hem ileti tüketicileri hem de olay işleyicileri için kullanılır.

İstenirse, bu, geri bildirme işlevinin son çağrısıdır.

*Reason* alanının değeri, durmanın nedenini gösterecek şekilde ayarlanır.

### **MÖCBCT\_EVENT\_CALL**

#### **Olay işleyici işlevi**

Kuyruk yöneticisi ya da bağlantı durduğunda ya da sustururken, olay işleyici işlevi bir ileti olmadan çağrılır.

Bu arama, tüm geri bildirme işlevleri için uygun işlemi yapmak için kullanılabilir.

#### **Message consumer işlevi**

Nesne tanıtıcısı için belirli bir hata (*CompCode* = MÖCC\_FAILED) saptandığında ileti tüketici işlevi çağrılmadan çağrılır; örneğin, *Reason* kodu = MÖRC\_GET\_INHIBITED.

*Reason* alanının değeri, çağırmanın nedenini gösterecek şekilde ayarlanır.

## **MQCBCT\_MC\_EVENT\_CALL**

The event handler function has been invoked for multicast events; The event handler is sent IBM MQ Multicast events instead of 'normal' IBM MQ events.

MQCBCT\_MC\_EVENT\_CALL ile ilgili ek bilgi için [Multicast exception raporlaması](#) başlıklı konuya bakın.

## **Hobj (MQHOBj)**

Bu, ileti tüketicisine yapılan çağrılar için nesne tanıtıcısıdır.

Bir olay işleyici için bu değer MQHO\_NONE olur.

Bir ileti kuyruktan kaldırılmamışsa iletiyi almak için, uygulama bu tanıtıcıyı ve İleti Alma Seçenekleri bloğunda ileti simgesini kullanabilir.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQHO\_UNUABLE\_HOBJ değeridir.

## **CallbackArea (MQPTR)**

Bu alan, geri bildirme işlevi için kullanılabilir.

Kuyruk yöneticisi, bu alanın içeriğine dayalı olarak hiçbir karar vermez ve geri bildirme işlevini tanımlamak için kullanılan MQCBB çağrısındaki bir parametre olan MQCBD yapısındaki [CallbackArea](#) alanından değişmez olarak geçirilir.

*CallbackArea* üzerinde yapılan değişiklikler, bir *HObj* için geri bildirme işlevine ilişkin çağrılar boyunca korunur. Bu alan, diğer çekme noktalarına ilişkin geri çağrı işlevleriyle paylaşılmaz.

Bu, geri bildirme işlevi için bir giriş/çıkış alanıdır. Bu alanın başlangıç değeri, boş değerli ya da boş değerli bayttır.

## **ConnectionArea (MQPTR)**

Bu alan, geri bildirme işlevi için kullanılabilir.

Kuyruk yöneticisi, bu alanın içeriğine dayalı olarak hiçbir karar vermez ve geri bildirme işlevini denetlemek için kullanılan MQCTL çağrısındaki bir parametre olan MQCTLO yapısındaki [ConnectionArea](#) alanından değiştirilmeden geçirilir.

Geri bildirme işlevleri tarafından bu alanda yapılan değişiklikler, geri bildirme işlevi çağrıları boyunca korunur. Bu alan, tüm geri bildirme işlevleri tarafından paylaşılacak bilgileri geçirmek için kullanılabilir. *CallbackArea*' in tersine, bu alan, bağlantı tanıtıcısı için tüm geri çağırılarda ortaktır.

Bu bir giriş ve çıkış alanıdır. Bu alanın başlangıç değeri, boş değerli ya da boş değerli bayttır.

## **CompCode (MQUZN)**

Bu alan tamamlama kodudur. Bu ileti, iletiyi alan herhangi bir sorun olup olmadığını gösterir.

Değer aşağıdakilerden biridir:

### **MQCC\_OK**

Başarıyla tamamlandı

### **MQCC\_UYARI**

Uyarı (kısmi tamamlama)

### **MQCC\_FAILED**

Arama başarısız oldu

Bu bir giriş alanıdır. Bu alanın ilk değeri MQCC\_OK olur.

## **Neden (MQUZE)**

Bu, *CompCode*' yi niteleyen bir neden kodudur.



Bu bir giriş alanıdır. Bu alanın ilk değeri MQRC\_NONE olur.

### **Eyalet (MQUZE)**

Mevcut tüketicinin durumuna ilişkin bir gösterge. Tüketici işlevine sıfır olmayan bir neden kodu geçirildiğinde, bu alan bir uygulamaya değer olarak en çok değer sağlar.

Bu alanı, her neden kodu için davranış kodlarına gerek kalmadığından uygulama programlamasını kolaylaştırmak için kullanabilirsiniz.

Bu bir giriş alanıdır. Bu alanın ilk değeri MQCS\_NONE olur.

Çizelge 473.		
Durum	Kuyruk yöneticisi işlemi	Değişmezin değeri
<i>MQCS_NONE</i> Bu neden kodu, ek neden bilgisi olmayan olağan çağrışı gösterir.	Hayır, bu normal bir işlemdir.	0
<i>MQCS_SUSPENDED_TEMPORARY</i> Bu neden kodları geçici durumları gösterir.	Geri çağırma yordamı, koşulu raporlamak ve sonra askıya almak için çağrılır. Bir süre geçtikten sonra, sistem işlemi yeniden deneyebilir ve bu da aynı durumun yeniden yükseltilmesine neden olabilir.	1
<i>MQCS_SUSPENDED_USER_ACTION</i> Bu neden kodları, geri bildirmenin koşulu çözmek için harekete geçmeleri gereken koşulları gösterir.	Tüketici askıya alınır ve geri çağırma yordamı durumu raporlamak için çağrılır. Geri çağırma yordamı, mümkünse koşulu çözmeli ve bağlantıyı devam ettirmeli ya da kapatılmalıdır.	2
<i>MQCS_SUSPENDED</i> Bu neden kodları, daha fazla ileti geri bildirmesini önleyen hataları gösterir.	Kuyruk yöneticisi geri çağırma işlevini otomatik olarak askıya alır. Geri çağırma işlevi sürdürülürse, yine aynı neden kodu alma olasılığı vardır.	3
<i>MQCS_STOPPED</i> Bu neden kodları, ileti tüketiminin sona ermesini gösterir.	Kural dışı durum işleyicisine ve belirtilen MQCBDO_STOP_CALL belirtilen çağrılara teslim edildi. Başka ileti tüketilemez.	4

### **DataLength (MQUZE)**

Bu, iletteki uygulama verilerinin bayt cinsinden uzunluğudur. Değer sıfırsa, iletinin uygulama verisi içermediği anlamına gelir.

DataLength alanı, iletinin uzunluğunu içerir, ancak tüketiciye aktarılan ileti verilerinin uzunluğunun tam olarak uzunluğunu içermez. Bu, iletinin kısaltılması olabilir. Tüketiciye ne kadar veri aktarıldığını belirlemek için MQGMO 'daki ReturnedLength alanını kullanın.

Neden kodu, iletinin kesildiğini gösteriyorsa, gerçek iletinin ne kadar büyük olduğunu belirlemek için DataLength alanını kullanabilirsiniz. Bu, ileti verilerini yerleştirmek için gereken arabelleğin büyüklüğünü belirlemenize olanak tanır ve daha sonra, MaxMsgUzunluğu 'i uygun bir değerle güncellemek için bir MQCB çağrısı yayımlayabilmenize olanak tanır.

MQGMO\_CONVERT seçeneği belirtilirse, dönüştürülen ileti, DataLength için döndürülen değerden büyük olabilir. In such cases, the application probably needs to issue an MQCB call to update the MaxMsgUzunluğu to be greater than the value returned by the queue manager for DataLength.

İleti kesme sorunlarını önlemek için, MaxMsgLength değerini MQCBD\_FULL\_MSG\_LENGTH olarak belirtin. Bu, kuyruk yöneticisinin veri dönüştürmesinden sonra tam ileti uzunluğu için bir arabellek

ayırmasına neden olur. Ancak, bu seçenek belirtilse bile, isteği doğru işlemek için yeterli depolama alanının kullanılmaması yine de mümkün olur. Uygulamalar her zaman döndürülen neden kodunu denetlemelidir. Örneğin, iletiyi dönüştürmek için yeterli bellek ayırmak mümkün değilse, iletiler uygulamaya dönüştürülmemiş olarak döndürülür.

Bu, ileti tüketicisi işlevine ilişkin bir giriş alanıdır; bir olay işleyici işleviyle ilgili değildir.

### **BufferLength (MQUZN)**

Bu alan, bu işleve aktarılan ileti arabelleğindeki bayt cinsinden uzunluktur.

Arabellek, tüketici için tanımlanan MaxMsgUzunluğu değerinden ve MQGMO 'daki ReturnedLength değerine göre daha büyük olabilir.

Gerçek ileti uzunluğu DataLength alanında sağlanır.

Uygulama, geri bildirme işlevi süresi boyunca tüm arabelleği kendi amaçları için kullanabilir.

Bu, ileti tüketicisi işlevine ilişkin bir giriş alanıdır; bir kural dışı durum işleyici işleviyle ilgili değildir.

### **İşaretler (MQUZE)**

Bu tüketiciyle ilgili bilgileri içeren işaretler.

Aşağıdaki seçenek tanımlıdır:

#### **MQCBCF\_READA\_BUFFER\_BOŞ**

MQCO\_QUIESCE seçeneği kullanılarak önceki bir MQCLOSE çağrısı bir MQRC\_READ\_AHEAD\_MSGS neden koduyla başarısız olursa, bu işaret döndürülebilmektedir.

Bu kod, son okuma öncesinde okunan iletinin yayınlanmakta olduğunu ve arabelleğin artık boş olduğunu gösterir. Uygulama, MQCO\_QUIESCE) seçeneğini kullanarak başka bir MQCLOSE çağrısı yayınlarsa başarılı olur.

Bir uygulamanın, yürürlükteki seçim ölçütleriyle eşleşmeyen okuma tamamlama arabelleğindeki iletiler olabileceğinden, bir uygulamanın bu işaret kümesiyle bir ileti verilmesine garanti verilmediğini unutmayın. Bu örnekte, tüketici işlevi, MQRC\_HOBJ\_QUIESCED neden koduyla çağrılır.

İleriye okuma arabelleği tamamen boşsa, tüketici MQCBCF\_READA\_BUFFER\_BOMPTY işaretiyle ve MQRC\_HOBJ\_QUIESCED\_NO\_MSGS neden kodlarıyla çağrılır.

Bu, ileti tüketicisi işlevine ilişkin bir giriş alanıdır; bir olay işleyici işleviyle ilgili değildir.

### **ReconnectDelay (MQUBE)**

ReconnectDelay (Yeniden Yapılandırma Gecikmesi), kuyruk yöneticisinin yeniden bağlanmayı denemeden önce ne kadar bekleyeceğini belirtir. Bu alan, gecikmeyi değiştirmek ya da tümüyle yeniden bağlantıyı durdurmak için bir olay işleyiciyle değiştirilebilir.

Use the ReconnectDelay field only if the value of the Neden field in the Callback Context is MQRC\_RECONNECTING.

On entry to the event handler the value of ReconnectDelay is the number of milliseconds the queue manager is going to wait before making a reconnection attempt. [Çizelge 474 sayfa 286](#) , olay işleyicisinden dönüşte kuyruk yöneticisinin davranışını değiştirmek için ayarlayabileceğiniz değerleri listeler.

<i>Çizelge 474. ReconnectDelay değerleri</i>		
<b>Ad</b>	<b>Değer</b>	<b>Tanım</b>
MQRD_NO_RECONNECT	-1	Başka bir yeniden bağlantı kurma girişiminde bulunmaz. Uygulamaya bir hata döndürüldü.
MQRD_NO_DELAY	0	Hemen yeniden bağlanmayı deneyin.
<i>Milliseconds</i>	>0	Bağlantıyı yeniden denemeden önce bu süre (milisaniye) beklenir.

## MQCBD-Geri çağırma tanımlayıcısı

MQCBD yapısı, bir geri çağırma işlevini ve kuyruk yöneticisi tarafından kullanımını denetleyen seçenekleri belirtmek için kullanılır. Yapı, MQCB çağırısındaki bir giriş değiştirgesidir.

### Kullanılabilirlik

MQCBD yapısı aşağıdaki altyapılarda kullanılabilir:

-  AIX
-  IBM i
-  Linux
-  Windows
-  z/OS

ve bu sistemlere bağlı IBM MQ MQI clients için.

### Sürüm

MQCBD ' nin yürürlükteki sürümü MQCBD\_VERSION\_1.

### Karakter kümesi ve kodlama

MQCBD içindeki veriler, **CodedCharSetId** kuyruk yöneticisi özneliği tarafından belirtilen karakter kümesinde ve MQENC\_NATIVE tarafından verilen yerel kuyruk yöneticisinin kodlamasında olmalıdır. Ancak, uygulama bir MQ MQI istemcisi olarak çalışıyorsa, yapı istemcinin karakter kümesinde ve kodlamasında olmalıdır.

### Alanlar

**Not:** Aşağıdaki tabloda, alanlar alfabetik olarak değil, kullanıma göre gruplanır. Alt konular aynı sırayı izler.

Çizelge 475. MQCBD ' deki Alanlar		
Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
<u>StrucID</u> (yapı tanıtıcısı)	MQCBD_STRUC_ID	' CBD- '
<u>Sürüm</u> (yapı sürümü numarası)	MQCBD_VERSION_1	1
<u>CallbackType</u> (geri çağırma işlevinin tipi)	MQCBT_MESSAGE_CONSUMER	1
<u>Seçenekler</u> (ileti tüketimini denetleyen seçenekler)	MQCBDO_NONE	0
<u>CallbackArea</u> (geri çağırma işlevinin kullanacağı alan)	Yok	Boş değerli gösterge ya da boş değerli boşluklar
<u>CallbackFunction</u> (işlevin bir API çağırısı olarak çağırılıp çağırılmayacağı)	Yok	Boş değerli gösterge ya da boş değerli boşluklar
<u>CallbackName</u> (işlevin dinamik olarak bağlı bir program olarak çağırılıp çağırılmayacağı)	Yok	Boş değerli dizilim ya da boşluklar
<u>MaxMsgLength</u> (okunabilen en uzun iletinin uzunluğu)	MQCBD_FULL_MSG_LENGTH	-1

Çizelge 475. MQCBD 'deki Alanlar (devamı var)

Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
<b>Notlar:</b>		
<p>1. ~ simgesi, tek bir boş karakteri temsil eder.</p> <p>2. Boş değerli dizilim ya da boşluklar değeri, C programlama dilindeki boş değerli diziyi ve diğer programlama dillerindeki boş karakterleri gösterir.</p> <p>3. C programlama dilinde, makro değişkeniMQCBD_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlar için başlangıç değerleri sağlamak üzere bunu aşağıdaki şekilde kullanın:</p>		
<pre>MQCBD MyCBD = {MQCBD_DEFAULT};</pre>		

## Dil bildirimleri

### MQCBD için C bildirimi

```
typedef struct tagMQCBD MQCBD;
struct tagMQCBD {
    MQCHAR4    StructId;           /* Structure identifier */
    MQLONG     Version;           /* Structure version number */
    MQLONG     CallbackType;      /* Callback function type */
    MQLONG     Options;           /* Options controlling message
    consumption */
    MQPTR      CallbackArea;      /* User data passed to the function */
    MQPTR      CallbackFunction;  /* Callback function pointer */
    MQCHAR128  CallbackName;     /* Callback name */
    MQLONG     MaxMsgLength;     /* Maximum message length */
};
```

### MQCBD için COBOL bildirimi

```
** MQCBCD structure
10  MQCBD.
** Structure Identifier
15  MQCBD-STRUCID                PIC X(4).
** Structure Version
15  MQCBD-VERSION                PIC S9(9) BINARY.
** Callback Type
15  MQCBD-CALLBACKTYPE          PIC S9(9) BINARY.
** Options
15  MQCBD-OPTIONS                PIC S9(9) BINARY.
** Callback User Area
15  MQCBD-CALLBACKAREA          POINTER
** Callback Function Pointer
15  MQCBD-CALLBACKFUNCTION      FUNCTION-POINTER
** Callback Program Name
15  MQCBD-CALLBACKNAME          PIC X(128)
** Maximum Message Length
15  MQCDB-MAXMSGLNGTH           PIC S9(9) BINARY.
```

### MQCBD için PL/I bildirimi

```
dcl
1  MQCBD based,
3  StructId          char(4),          /* Structure identifier*/
3  Version           fixed bin(31),    /* Structure version*/
3  CallbackType     fixed bin(31),    /* Callback function type */
3  Options          fixed bin(31),    /* Options */
3  CallbackArea     pointer,          /* User area passed to the function */
3  CallbackFunction pointer,          /* Callback Function Pointer */
3  CallbackName     char(128),        /* Callback Program Name */
3  MaxMsgLength     fixed bin(31);    /* Maximum Message Length */
```

## **StrucId (MQCHAR4)**

Geri çağırma tanımlayıcısı yapısı- StrucId alanı

Bu, yapı tanıtıcısıdır; değer şöyle olmalıdır:

### **MQCBD\_STRUC\_ID**

Geri çağırma tanımlayıcısı yapısına ilişkin tanıtıcı.

C programlama dili için, MQCBD\_STRUC\_ID\_ARRAY değişmezi de tanımlanır; bu, MQCBD\_STRUC\_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisidir.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQCBD\_STRUC\_ID 'dir.

## **Sürüm (MQUZE)**

Geri çağırma tanımlayıcısı yapısı-Sürüm alanı

Bu, yapı sürüm numarasıdır; değer şu olmalıdır:

### **MQCBD\_VERSION\_1**

Version-1 geri bildirme tanımlayıcısı yapısı.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

### **MQCBD\_CURRENT\_VERSION**

Geri çağırma tanımlayıcısı yapısının yürürlükteki sürümü.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQCBD\_VERSION\_1' dir.

## **CallbackType (MQUZE)**

Geri çağırma tanımlayıcı yapısı- CallbackType alanı

Bu, geri bildirme işlevinin tipidir. Değer aşağıdakilerden biri olmalıdır:

### **MQCBT\_MESSAGE\_CONSUMER**

Bu geri çağırma ileti tüketici işlevi olarak tanımlar.

Bir ileti tüketici geri bildirme işlevi, bir ileti, belirtilen seçim ölçütleriyle toplantı sırasında, bir nesne tanıtıcısında kullanılabilir olduğunda ve bağlantı başlatılmış olduğunda çağrılır.

### **MQCBT\_EVENT\_HANDLER**

Bu geri çağırma zamanuyumsuz olay yordamı olarak tanımlar; bir tanıtıcı için iletileri tüketmek için yönlendirilmez.

Olay işleyicisini tanımlayan MQCB çağrısında *Hobj* gerekli değil ve belirtilirse yoksayılır.

Olay işleyici, tüm ileti tüketicisi ortamını etkileyen koşullar için çağrılır. Bir olay, kuyruk yöneticisi ya da bağlantı durdurma ya da susturma gibi bir olay gerçekleştiğinde, tüketici işlevi ileti olmadan çağrılır. Tek bir ileti tüketicisine özgü koşullar için çağrılmaz; örneğin, MQRC\_GET\_INHIMATED.

Aşağıdaki ortamlar dışında, olaylar, bağlantının başlatılıp başlatılmadığından bağımsız olarak, uygulamaya teslim edilir:

- z/OS üzerinde CICS ortamı
- iş parçacıklı uygulamalar

If the caller does not pass one of these values, the call fails with a *Reason* code of MQRC\_CALLBACK\_TYPE\_ERROR

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQCBT\_MESSAGE\_CONSUMER olur.

## **Seçenekler (MQUZE)**

Geri çağırma tanımlayıcısı yapısı-Seçenekler alanı

Bu seçeneklerden birini ya da birkaçını belirleyebilirsiniz. birden çok seçenek belirtmek için, değerleri bir araya ekleyin (aynı değişmezi bir kereden fazla eklemeyin) ya da bit ya da işlemi kullanarak değerleri birleştirin (programlama dili, bit işlemleri destekliyorsa).

### **MQCBDO\_FAIL\_IF QUIESCING**

Kuyruk yöneticisi susturma durumdaysa, MQCB çağrısı başarısız olur.

z/OS' ta bu seçenek, bağlantı ( CICS ya da IMS uygulaması için) susturulmuş durumdaysa, MQCB çağrısını başarısız olarak zorlar.

MQCB çağrısında geçirilen MQGMO seçeneklerinde MQGMO\_FAIL\_IF QUIESCING değerini belirtin; bu seçenek, susturma sırasında ileti tüketicilerine bildirilmesine neden olur.

**Denetim seçenekleri:** Aşağıdaki seçenekler, tüketici değişikliklerinin durumu değiştiğinde, bir ileti olmadan geri çağırma işlevinin çağrılıp çağrılmayacağını denetler:

### **MQCBDO\_REGISTER\_CALL**

Geri bildirme işlevi, MQCBCT\_REGISTER\_CALL çağrı tipiyle çağrılır.

### **MQCBDO\_START\_CALL**

Geri bildirme işlevi, MQCBCT\_START\_CALL çağrı tipiyle çağrılır.

### **MQCBDO\_STOP\_CALL**

Geri bildirme işlevi, MQCBCT\_STOP\_CALL çağrısıyla çağrılır.

### **MQCBDO\_DEREGISTIER\_CALL**

Geri bildirme işlevi, MQCBCT\_DEREGISTER\_CALL çağrı tipiyle çağrılır.

### **MQCBDO\_EVENT\_CALL**

Geri bildirme işlevi, MQCBCT\_EVENT\_CALL çağrısıyla çağrılır.

### **MQCBDO\_MC\_EVENT\_CALL**

Geri bildirme işlevi, MQCBCT\_\_MC\_EVENT\_CALL çağrısıyla çağrılır.

Bu çağrı tipleriyle ilgili ek ayrıntılar için [CallType](#) (CallType) konusuna bakın.

**Varsayılan seçenek:** Açıklanan seçeneklerden herhangi birine gereksiniminiz yoksa, aşağıdaki seçeneği kullanın:

### **MQCBDO\_NONE**

Başka bir seçenek belirtilmemeyi belirtmek için bu değeri kullanın; tüm seçenekler varsayılan değerlerini kabul eder.

MQCBDO\_NONE, program belgelerine yardımcı olmak için tanımlıdır; bu seçeneğin diğeriyle birlikte kullanılması amaçlanmaz, ancak değeri sıfır olduğu için, bu tür bir kullanım saptanamaz.

Bu bir giriş alanıdır. *Options* alanının ilk değeri MQCBDO\_NONE olur.

### **CallbackArea (MQPTR)**

Geri çağırma tanımlayıcı yapısı- CallbackArea alanı

Bu alan, geri bildirme işlevi için kullanılabilir bir alandır.

Kuyruk yöneticisi, bu alanın içeriğine dayalı olarak karar vermez ve geri bildirme işlevi bildiriminde bir parametre olan MQCBC yapısındaki [CallbackArea](#) alanından değiştirilmeden geçirilir.

Değer yalnızca MQOP\_REGISTER değerinin MQOP\_REGISTER değerini içeren bir *Operation* üzerinde kullanılır; bu değer, tanımlı geri çağırma olmadan, önceki bir tanımlamayı değiştirmez.

Bu, geri bildirme işlevi için bir giriş ve çıkış alanıdır. Bu alanın başlangıç değeri, boş değerli ya da boş değerli bayttır.

### **CallbackFunction (MQPTR)**

Geri çağırma tanımlayıcı yapısı- CallbackFunction alanı

Geri bildirme işlevi, işlev çağrısı olarak çağrılır.

Geri bildirme işlevine ilişkin bir gösterge belirtmek için bu alanı kullanın.

*CallbackFunction* ya da *CallbackName* değerini belirtmeniz gerekir. Her ikisini birden belirtiyorsanız, MQRC\_CALLBACK\_ROUTINE\_ERROR neden kodu döndürülür.

*CallbackName* ya da *CallbackFunction* ayarlandıysa, çağrı neden kodu MQRC\_CALLBACK\_ROUTINE\_ERROR neden ile başarısız olur.

Bu seçenek, şu ortamda desteklenmez: İşlev göstergesi başvurularını desteklemeyen programlama dilleri ve derleyiciler. Böyle durumlarda arama, MQRC\_CALLBACK\_ROUTINE\_ERROR neden kodunda başarısız olur.

**z/OS** z/OS işletim sisteminde, işlevin OS bağı kuralları ile çağrılması beklenmelidir. Örneğin, C programlama dilinde şunları belirtin:

```
#pragma linkage(MQCB_FUNCTION,OS)
```

Bu bir giriş alanıdır. Bu alanın başlangıç değeri, boş değerli ya da boş değerli bayttır.

**Not:** When using CICS with IBM WebSphere MQ 7.0.1, asynchronous consumption is supported if:

- Apar PK66866 , CICS TS 3.2' ye uygulanır.
- Apar PK89844 , CICS TS 4.1' e uygulanmaktadır.

### **CallbackName (MQCHAR128)**

Geri çağırma tanımlayıcı yapısı- CallbackName alanı

Geri bildirme işlevi dinamik olarak bağlantılı bir program olarak çağrılır.

*CallbackFunction* ya da *CallbackName* değerini belirtmeniz gerekir. Her ikisini birden belirtiyorsanız, MQRC\_CALLBACK\_ROUTINE\_ERROR neden kodu döndürülür.

*CallbackName* ya da *CallbackFunction* ayarlanmazsa, çağrı neden kodu MQRC\_CALLBACK\_ROUTINE\_ERROR neden ile başarısız olur.

Kullanılacak ilk geri çağrı yordamı kayıtlı olduğunda modül yüklenir ve son geri bildirme yordamı derejisterleri kullanacak şekilde geri yüklendiğinde yüklenir.

Aşağıdaki metinde belirtilenler dışında, ad alanda sola yaslanmış, boşluk içermeyen bir ad; adın kendisi, alanın uzunluğuna kadar boşluklarla dolduruludur. Aşağıdaki açıklamalarda, köşeli ayraçlar ([]) isteğe bağlı bilgileri gösterir:

#### **IBM i**

Geri çağırma adı, aşağıdaki biçimlerden biri olabilir:

- Kitaplık "/" Program
- Library "/" ServiceProgram ("FunctionName")

Örneğin, MyLibrary/MyProgram(MyFunction).

Kitaplık adı \*LIBL olabilir. Kitaplık ve program adları en çok 10 karakterle sınırlanmıştır.

#### **AIX and Linux**

Geri çağırma adı, dinamik olarak yüklenebilir bir modülün ya da kitaplığın adı, o kitaplıkta bulunan bir işlevin adına sahip olan bir kitaplık adıdır. İşlev adının ayraç içine alınması gerekir. Kitaplık adı isteğe bağlı olarak bir dizin yolu ile önek olarak önek olarak kullanılabilir:

```
[path]library(function)
```

Yol belirlenmezse, sistem arama yolu kullanılır.

Ad en çok 128 karakterle sınırlanmıştır.

#### **Windows**

Geri çağırma adı, bir dinamik bağlantı kitaplığının adıdır, o kitaplıkta bulunan bir işlevin adına sahip olan bir alt düzeltmenin adıdır. İşlev adı parantez içine alınmalıdır. Kitaplık adı isteğe bağlı olarak bir dizin yolu ve sürücü öneki ile önek olarak kullanılabilir:

```
[d:][path]library(function)
```

Sürücü ve yol belirlenmezse, sistem arama yolu kullanılır.

Ad en çok 128 karakterle sınırlanmıştır.

### **z/OS**

Geri çağırma adı, LINK ya da LOAD makrosu için EP parametresindeki belirtim için geçerli olan bir yükleme modülünün adıdır.

Ad en çok 8 karakterden oluşan bir değer ile sınırlanmıştır.

### **z/OS CICS**

Geri bildirme adı, EXEC CICS LINK komut makrosu PROGRAM parametresindeki belirtim için geçerli olan bir yükleme modülünün adıdır.

Ad en çok 8 karakterden oluşan bir değer ile sınırlanmıştır.

Program, kurulu PROGRAM tanımının REMOTEYTEM seçeneği kullanılarak ya da dinamik yönlendirme programı tarafından uzak olarak tanımlanabilir.

Program IBM MQ API çağrılarını kullanacaksa, uzak CICS bölgesi IBM MQ ' e bağlanmalıdır. Ancak, MQCBC yapısındaki Hobj alanının uzak bir sistemde geçerli olmadığı unutulmamaktadır.

*CallbackName* yükleme girişimi sırasında bir hata oluşursa, uygulamaya aşağıdaki hata kodlarından biri döndürülür:

- MQRC\_MODULE\_NOT\_FOUND
- MQRC\_MODULE\_INVALID
- MQRC\_MODULE\_ENTRY\_NOT\_FOUND

Yükleme girişiminde bulunan modülün adını ve işletim sisteminden hatalı neden kodunu içeren hata günlüğüne bir ileti de yazılır.

Bu bir giriş alanıdır. Bu alanın başlangıç değeri, boş değerli bir dizilim ya da boşlukdur.

### **MaxMsgUzunluğu (MQUZE)**

Bu, tanıtıcıdan okunabilen ve geri çağırma yordamında okunabilen en uzun iletinin bayt cinsinden uzunluğudur. Geri Çağırma tanımlayıcı yapısı- MaxMsgUzunluk alanı

If a message has a longer length, the callback routine receives *MaxMsgLength* bytes of the message, and reason code:

- MQRC\_TRUNCATED\_MSG\_BASARISIZ oldu ya da
- MQGMO\_ACCEPT\_TRUNCATED\_MSG belirtilirse MQRC\_TRUNCATED\_MSG\_KABUL edilir.

Gerçek ileti uzunluğu, MQCBC yapısındaki DataLength alanında sağlanır.

Aşağıdaki özel değer tanımlıdır:

### **MQCBD\_FULL\_MSG\_LENGTH**

Arabellek uzunluğu sistem tarafından, iletilerin kesilmeden döndürülmesi için ayarlanıyor.

İletiyi almak için arabellek ayırmak için yeterli bellek yoksa, sistem geri bildirme işlevini MQRC\_STORAGE\_NOT\_AVAM neden koduyla çağırılır.

Örneğin, veri dönüştürme isteğinde bulunsanız ve ileti verilerini dönüştürmek için yeterli bellek yoksa, dönüştürülemez ileti geri bildirme işlevine geçilir.

Bu bir giriş alanıdır. *MaxMsgLength* alanının ilk değeri MQCBD\_FULL\_MSG\_LENGTH değerine sahip.

### **MQCHARV-Değişken Uzunluklu Dizgi**

Değişken uzunluklu bir dizgiyi tanımlamak için MQCHARV yapısını kullanın.

### **Kullanılabilirlik**

MQCHARV yapısı aşağıdaki altyapılarda bulunur:

-  AIX



-  IBM i
-  Linux
-  Windows

ve bu sistemlere bağı IBM MQ MQI clients için.

## Karakter kümesi ve kodlama

MQCHARV içindeki veriler, MQENC\_NATIVE tarafından verilen yerel kuyruk yöneticisinin kodlamasında ve yapı içindeki VSCCSID alanının karakter kümesinde olmalıdır. Uygulama bir MQ istemcisi olarak çalışıyorsa, yapı istemcinin kodlamasında olmalıdır. Bazı karakter kümelerinin kodlamaya bağı bir gösterimi vardır. VSCCSID bu karakter kümelerinden biriye, kullanılan kodlama, MQCHARV ' deki diğr alanlarla aynı kodlamadır. VSCCSID ile tanıtilan karakter takımı çift baytlık karakter takımı (DBCS) olabilir.

## Kullanım

MQCHARV yapısı, bunu içeren yapıyla bitişik olmayabilecek verileri adresler. Bu verileri ele almak için, işaretçi veri tipiyle bildirilen alanlar kullanılabilir. COBOL ' un tüm ortamlarda işaretçi veri tipini desteklemediğini unutmayın. Bu nedenle, veriler MQCHARV ' yi içeren yapının başlangıcındaki verilerin görel konumunu içeren alanlar kullanılarak da ele alınabilir.

## Alanlar

**Not:** Aşağıdaki tabloda, alanlar alfabetik olarak değil, kullanıma göre gruplanır. Alt konular aynı sırayı izler.

Çizelge 476. MQCHARV içindeki alanlar		
Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
<u>VSPtr</u> (değişken uzunluklu dizgiyi gösteren gösterge)	Yok	Boş değeri gösterge ya da boş değeri byte 'lar.
<u>VOffset</u> (bu MQCHARV yapısını içeren yapının başlangıcındaki değişken uzunluklu dizginin bayt cinsinden görel konumu)	Yok	0
<u>VSBufSize</u> (VSPtr ya da VOffset alanı tarafından adreslenen arabelleğin bayt cinsinden boyutu)	MQVS_USE_VSLENGTH	0
<u>VSLkalınlık</u> (VSPtr ya da VOffset alanı tarafından adreslenen değişken uzunluklu dizginin bayt cinsinden uzunluğu)	Yok	0
<u>VSCCSID</u> (VSPtr ya da VOffset alanı tarafından adreslenen değişken uzunluklu dizginin karakter takımı tanıtıcısı)	MQCCSI_APPL	-3
<p><b>Not:</b> C programlama dilinde, MQCHARV_DEFAULT makro değişkeni çizelgede listelenen değeri içerir. Yapıdaki alanlar için başlangıç değeri sağlamak üzere aşağıdaki şekilde kullanılabilir:</p> <pre>MQCHARV MyVarStr = {MQCHARV_DEFAULT};</pre>		

## Dil bildirimleri

### MQCHARV için C bildirimi

```
typedef struct tagMQCHARV MQCHARV;
struct tagMQCHARV {
    MQPTR      VSPtr;                /* Address of variable length string */
    MQLONG     VSOffset;            /* Offset of variable length string */
    MQLONG     VSBufSize;          /* Size of buffer */
    MQLONG     VSLength;           /* Length of variable length string */
    MQLONG     VSCCSID;            /* CCSID of variable length string */
};
```

### MQCHARV için COBOL bildirimi

```
** MQCHARV structure
10 MQCHARV.
** Address of variable length string
15 MQCHARV-VSPTR      POINTER.
** Offset of variable length string
15 MQCHARV-VSOFFSET  PIC S9(9) BINARY.
** Size of buffer
15 MQCHARV-VSBUFSIZE PIC S9(9) BINARY.
** Length of variable length string
15 MQCHARV-VSLENGTH  PIC S9(9) BINARY.
** CCSID of variable length string
15 MQCHARV-VSCCSID   PIC S9(9) BINARY.
```

**Not:** Bir COBOL uygulamasını ortamlar arasında aktarmak istiyorsanız, işaretçi veri tipinin tüm amaçlanan ortamlarda kullanılabilir olup olmadığını öğrenmeniz gerekir. Değilse, uygulamanın işaretçi alanları yerine kayma alanlarını kullanarak verileri ele alması gerekir. Göstergelerin desteklenmediği ortamlarda, işaretçi alanlarını uygun uzunlukta bayt dizgileri olarak bildirebilirsiniz; ilk değer tümüyle boş değerli byte dizgisi olur. Göreli konum alanlarını kullanıyorsanız, bu başlangıç değerini değiştirmeyin. Sağlanan kopya kitaplarını değiştirmeden bunu yapmanın bir yolu, aşağıdakileri kullanmaktır:

```
COPY CMQCHRVV REPLACING POINTER BY ==BINARY PIC S9(9)==.
```

Burada CMQCHRVV , kullanılacak kopya defteriyle değiştirilebilir.

### MQCHARV için PL/I bildirimi

```
dcl
1 MQCHARV based,
3 VSPtr      pointer,          /* Address of variable length string */
3 VSOffset   fixed bin(31),    /* Offset of variable length string */
3 VSBufSize  fixed bin(31),    /* Size of buffer */
3 VSLength   fixed bin(31),    /* Length of variable length string */
3 VSCCSID    fixed bin(31);    /* CCSID of variable length string */
```

### MQCHARV için High Level Assembler bildirimi

```
MQCHARV          DSECT
MQCHARV_VSPTR    DS   F      Address of variable length string
MQCHARV_VSOFFSET DS   F      Offset of variable length string
MQCHARV_VSBUFSIZE DS   F      Size of buffer
MQCHARV_VSLENGTH DS   F      Length of variable length string
MQCHARV_VSCCSID DS   F      CCSID of variable length string
*
MQCHARV_LENGTH   EQU  *-MQCHARV
MQCHARV_AREA     DS   CL(MQCHARV_LENGTH)
```

### VSPtr (MQPTR)

Bu, değişken uzunluklu dizilimin göstergesini gösterir.

Değişken uzunluklu dizgiyi belirtmek için, VSPtr ya da VSOffset alanını kullanabilirsiniz, ancak her ikisini birden belirleyemez.

Bu alanın başlangıç değeri, boş değerli ya da boş değerli bayttır.

### **VSOffset (MQUZE)**

Görelî konum pozitif ya da negatif olabilir. Değişken uzunluklu dizgiyi belirtmek için, VSPtr ya da VSOffset alanını kullanabilirsiniz, ancak her ikisini birden belirleyemez. MQCHARV ' nin başlangıcındaki değişken uzunluklu dizginin bayt cinsinden görelî konumu ya da onu içeren yapı.

MQCHARV yapısı başka bir yapı içine yerleştirildiğinde, bu değer, bu MQCHARV yapısını içeren yapının başlangıcındaki değişken uzunluktaki dizginin bayt cinsinden görelî konudur. MQCHARV yapısı başka bir yapı içine yerleştirilmediyse, örneğin, bir işlev çağrısında parametre olarak belirtilirse, görelî konum MQCHARV yapısının başlangıcıyla görelî olur.

Bu alanın ilk değeri 0 'tır.

### **VSBufSize (MQUZE)**

Bu, VSPtr ya da VSOffset alanı tarafından adreslenen arabelleğin bayt cinsinden büyüklüğüdür.

Bir işlev çağrısında bir çıkış alanı olarak MQCHARV yapısı kullanıldığında, bu alan, sağlanan arabelleğin uzunluğuna uygun olarak kullanıma hazırlanmalıdır. VSLength değeri, VSBufSize değerinden büyükse, arabelleğindeki araya yalnızca VSBufSize byte veri döndürülebilir.

Bu değer, sıfıra eşit ya da bu değere eşit bir değer ya da tanınan şu özel değer olmalıdır:

#### **MQVS\_USE\_VSLENGTH**

Belirtildiğinde, arabelleğin uzunluğu, MQCHARV yapısındaki VSLTOL (VSLU; Güç Gücü) alanından alınır. Bu değeri, yapıyı bir çıkış alanı olarak kullanırken ve bir arabellek sağlandığı sırada kullanmayın.

Bu, bu alanın ilk değeridir.

### **Güç Gücü (MQUZA)**

VSPtr ya da VSOffset alanı tarafından adreslenen değişken uzunluktaki dizilimin bayt cinsinden uzunluğu.

Bu alanın ilk değeri 0 'tır. Değer sıfırdan büyük ya da sıfıra eşit olmalı ya da şu özel değer tanınmalıdır:

#### **MQVS\_NULL\_TERMINATED**

MQVS\_NULL\_TERMINATED belirtilmediyse, dizginin bir parçası olarak VSLength byte 'ları içerilir. Boş değerli karakterler varsa, diziyi sınırlamazlar.

MQVS\_NULL\_TERMINATED belirtilirse, dizgi dizgide saptanan ilk boş değerle sınırlanır. Boş değer, o dizginin bir parçası olarak içerilmedi.

**Not:** MQVS\_NULL\_TERMINATED belirtildiyse, bir dizgiyi sonlandırmak için kullanılan boş karakter, VSCCSID tarafından belirtilen kod kümesinden boş değerde olur.

Örneğin, UTF-16 'da (CCSID' ler 1200, 13488 ve 17584), bu, boş değer içeren iki baytlık Unicode kodlamasıdır. Bu kodlama, 16 bitlik tüm sıfırların sayısıdır. UTF-16 'da, karakterlerin bir parçası olan tüm sıfıra ayarlanmış tek byte 'ları bulmak yaygındır (örneğin, 7 bit ASCII karakterleri), ancak çift byte sınırında iki 'sıfır ' byte bulunduğunda dizgiler boş olarak sonlandırılır. Geçerli karakterlerin her bir parçasıysa, tek bir sınır üzerinde iki 'sıfır' byte elde etmek mümkündür. Örneğin, x '01' x '00 x' 00 '00' x '30', geçerli iki Unicode karakteri temsil eder ve boş değer olarak dizgiyi sonlandırmaz.

### **VSCCSID (MQLONG)**

Bu, VSPtr ya da VSOffset alanı tarafından adreslenen değişken uzunluklu dizginin karakter kümesi tanıtıcısıdır.

Bu alanın ilk değeri MQCCSI\_APPL' dir; bu değer, MQ tarafından yürürlükteki işlemin gerçek karakter kümesi tanıtıcısı olarak değiştirilmesi gerektiğini belirtmek için tanımlanır. Sonuç olarak, MQCCSI\_APPL değişiminin değeri hiçbir zaman değişken uzunluklu bir dizgiyle ilişkilendirilmez.

Bu alanın ilk değeri, derleme biriminize ilişkin MQCCSI\_APPL değişimi için farklı bir değer tanımlanarak değiştirilebilir. Bunu nasıl yapacağınız, uygulamanızın programlama diline bağlıdır.

**z/OS** z/OS sistemlerinde, *MQCCSI\_APPL* tarafından kullanılan CCSID varsayılan uygulaması aşağıdaki gibi tanımlanır:

- DLL arabirimini kullanan toplu LE uygulamaları için varsayılan değer, **MQCONN** ' un yayınlandığı sırada yürürlükteki ülke değeriyle ilişkilendirilen CODESET değeridir (varsayılan değer 1047 'dir).
- Toplu iş MQ sınırlı kod öbeklerinden biriyle bağ tanımlanan toplu LE uygulamaları için varsayılan değer, **MQCONN** sonrasında yayınlanan ilk MQI çağrısı sırasında yürürlükteki ülke değeriyle ilişkilendirilmiş CODESET değeridir (varsayılan değer 1047 'dir).
- z/OS UNIX System Services iş parçacığında çalışan LE dışı toplu uygulamalar için varsayılan değer, **MQCONN** sonrasında yayınlanan ilk MQI çağrısı sırasında THLICCSID değeridir (varsayılan değer 1047 'dir).
- Diğer toplu iş uygulamaları için varsayılan değer, kuyruk yöneticisinin CCSID değeridir.

## **MQCCSI\_APPL ' in yeniden tanımlanması**

Aşağıdaki örneklerde, çeşitli programlama dillerinde MQCCSI\_APPL değerini nasıl geçersiz kılabileceğiniz gösterilmiştir. MQCCSI\_APPL değerini değiştirerek, her değişken uzunluklu dizgi için VSCCSID ' yi ayrı olarak ayarlama gereksinimini kaldırabilirsiniz. Bu örneklerde CCSID 1208 olarak ayarlanır; bu değeri, gerek duyduğunuz değerle değiştirin. Bu değer, belirli bir MQCHARV eşgörünümünde VSCCSID ' yi ayarlayarak geçersiz kılabileceğiniz varsayılan değer olur.

C kullanımı

```
#define MQCCSI_APPL 1208
#include <cmqc.h>
```

COBOL kullanımı

```
COPY CMQXYZV REPLACING -3 BY 1208.
```

PL/I kullanımı

```
%MQCCSI_APPL = '1208';
%include syslib(cmqp);
```

High Level Assembler kullanımı

```
MQCCSI_APPL EQU 1208
CMQA LIST=NO
```

## **MQCIH- CICS bridge üstbilgisi**

MQCIH yapısı, CICS bridge üzerinden CICS ' e gönderilen bir iletiye ilişkin üstbilgi bilgilerini açıklar.

Desteklenen herhangi bir IBM MQ altyapısı için MQCIH yapısını içeren bir ileti yaratabilir ve iletebilirsiniz, ancak CICS bridge yalnızca bir IBM MQ for z/OS kuyruk yöneticisi tarafından kullanılabilir. Bu nedenle, iletinin z/OS dışı bir kuyruk yöneticisinden CICS ' e ulaşabilmesi için kuyruk yöneticisi ağınızın, iletinin yöneltilebileceği en az bir z/OS kuyruk yöneticisi içermesi gerekir.

IBM MQ 9.0 tarafından desteklenen tüm CICS sürümleri ve daha sonra köprüünün CICS tarafından sağlanan sürümü kullanılır. IBM MQ CICS bağdaştırıcısının ve IBM MQ CICS bridge bileşenlerinin yapılandırılmasıyla ilgili daha fazla bilgi için CICS belgelerinin [MQ bağlantılarının yapılandırılması](#) bölümüne bakın.

## **Kullanılabilirlik**

MQCIH yapısı aşağıdaki altyapılarda bulunur:

-  AIX
-  Linux
-  Windows
-  z/OS

ve bu sistemlere bağı IBM MQ MQI clients için.

## Biçim Adı

MQFMT\_CICS

## Sürüm

Yürürlükteki MQCIH sürümü: MQCIH\_VERSION\_2. Yapının yalnızca daha yeni sürümünde var olan alanlar, izleyen tanımlarda bu şekilde tanımlanır.

Desteklenen programlama dilleri için sağlanan üstbilgi, COPY ve INCLUDE dosyaları, *Version* alanının ilk değeri MQCIH\_VERSION\_2olarak ayarlanmış MQCIH ' nin en son sürümünü içerir.

## Karakter kümesi ve kodlama

MQCIH yapısı ve uygulama iletisi verileri için kullanılan karakter kümesi ve kodlama için özel koşullar geçerlidir:

- CICS bridge kuyruğunun sahibi olan kuyruk yöneticisine bağlanan uygulamaların, kuyruk yöneticisinin karakter kümesi ve kodlamasında bulunan bir MQCIH yapısı sağlaması gerekir. Bunun nedeni, bu durumda MQCIH yapısının veri dönüştürmesinin gerçekleştirilmemiş olmasıdır.
- Diğer kuyruk yöneticilerine bağlanan uygulamalar, desteklenen karakter kümelerinde ve kodlamasında bulunan bir MQCIH yapısı sağlayabilir; CICS bridge kuyruğunun sahibi olan kuyruk yöneticisine bağı alıcı ileti kanalı aracısı, MQCIH yapısını dönüştürür.
- MQCIH yapısını izleyen uygulama iletisi verileri, MQCIH yapısıyla aynı karakter kümesinde ve kodlamada olmalıdır. MQCIH yapısındaki *CodedCharSetId* ve *Encoding* alanlarını, uygulama ileti verilerinin karakter kümesini ve kodlamasını belirtmek için kullanamazsınız.

Veriler, kuyruk yöneticisi tarafından desteklenen yerleşik biçimlerden biri değilse, uygulama ileti verilerini dönüştürmek için bir veri dönüştürme çıkışı sağlamanız gerekir.

## Alanlar

**Not:** Aşağıdaki tabloda, alanlar alfabetik olarak değil, kullanıma göre gruplanır. Alt konular aynı sırayı izler.

Çizelge 477. MQCIH için MQCIH içindeki alanlar		
Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
<u>StrucId</u> (yapı tanıtıcısı)	MQCIH_STRUC_ID	'CIH~'
<u>Sürüm</u> (yapı sürümü numarası)	MQCIH_VERSION_2	2
<u>StrucLength</u> (MQCIH yapısının uzunluğu)	MQCIH_LENGTH_2	180
<u>Kodlama</u> (ayrılmış)	Yok	0
<u>CodedCharSetId</u> (ayrılmış)	Yok	0
<u>Biçim</u> (MQCIH ' yi izleyen verilerinMQ biçimi adı)	MQFMT_NONE	Boşluklar
<u>İşaretler</u> (işaretler)	MQCIH_YOK	0

Çizelge 477. MÇCIH için MÇCIH içindeki alanlar (devamı var)

Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
<u>ReturnCode</u> (köprüden dönüş kodu)	MÇCRC_OK	0
<u>CompCode</u> (MQ tamamlama kodu ya da CICS EIBRESP)	MÇCC_OK	0
<u>Neden</u> (MQ neden ya da geribildirim kodu ya da CICS EIBRESP2)	MÇRC_NONE	0
<u>UOWControl</u> (iş birimi denetimi)	YALNIZCA MÇCUOWC_ONLY	273
<u>GetWaitInterval</u> (Köprü görevi tarafından verilen MQGET çağrısı için bekleme aralığı)	MÇCGWI_DEFAULT	-2
<u>LinkType</u> (link tipi)	MÇCLT_PROGRAM	1
<u>OutputDataLength</u> (çıkış COMMAREA veri uzunluğu)	MÇCODL_AS_INPUT	-1
<u>FacilityKeepTime</u> (köprü tesisini serbest bırakma süresi)	Yok	0
<u>ADSDescriptor</u> (ADS tanımlayıcısını gönder/al)	MÇCADSD_NONE	0
<u>ConversationalTask</u> (görevin etkileşimli olup olmayabileceği)	MÇCCT_NO	0
<u>TaskEndStatus</u> (görevin sonundaki durum)	MÇCTES_NOSYNC	0
<u>Tesis</u> (köprü olanağı simgesi)	MÇCFAC_NONE	Boş Değerler
<u>İşlev</u> (MQ çağrı adı ya da CICS EIBFN işlevi)	MÇCFUNC_NONE	Boşluklar
<u>AbendCode</u> (olağandışı bitiş kodu)	Yok	Boşluklar
<u>Authenticator</u> (parola ya da geçiş bileti)	Yok	Boşluklar
<u>Reserved1</u> (ayrılmış)	Yok	Boşluklar
<u>ReplyToBiçim</u> (MQ yanıt iletisinin biçim adı)	MÇFMT_NONE	Boşluklar
<u>RemoteSysId</u> (kullanılacak uzak CICS sistem tanıtıcısı)	Yok	Boşluklar
<u>RemoteTransTnt</u> (kullanılacak CICS RTRANSID)	Yok	Boşluklar
<u>TransactionId</u> (eklenecek işlem)	Yok	Boşluklar
<u>FacilityLike</u> (uçbirim öykünülen öznelikleri)	Yok	Boşluklar
<u>AttentionId</u> (AID tuşu)	Yok	Boşluklar
<u>StartCode</u> (hareket başlatma kodu)	MÇCSC_NONE	Boşluklar
<u>CancelCode</u> (hareket kodunu olağandışı sona erdir)	Yok	Boşluklar
<u>NextTransactionId</u> (eklenecek sonraki işlem)	Yok	Boşluklar
<u>Reserved2</u> (ayrılmış)	Yok	Boşluklar
<u>Reserved3</u> (ayrılmış)	Yok	Boşluklar
<b>Not:</b> <i>Version</i> , MÇCIH_VERSION_2' den küçükse kalan alanlar yoktur.		

Çizelge 477. MÇCIH için MÇCIH içindeki alanlar (devamı var)

Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
<u>CursorPosition</u> (imleç konumu)	Yok	0
<u>ErrorOffset</u> (iletideki hatanın görelî konumu)	Yok	0
<u>InputItem</u> (giriş öğesi)	Yok	0
<u>Reserved4</u> (ayrılmış)	Yok	0

**Notlar:**

1. – simgesi, tek bir boş karakteri temsil eder.
2. C programlama dilinde, makro değişkeni MÇCIH\_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlar için başlangıç değerleri sağlamak üzere bunu aşağıdaki şekilde kullanın:

```
MÇCIH MyCIH = {MÇCIH_DEFAULT};
```

**Dil bildirimleri**

MÇCIH için C bildirimi

```
typedef struct tagMÇCIH MÇCIH;
struct tagMÇCIH {
    MÇCHAR4  StructId;           /* Structure identifier */
    MÇLONG   Version;           /* Structure version number */
    MÇLONG   StructLength;      /* Length of MÇCIH structure */
    MÇLONG   Encoding;          /* Reserved */
    MÇLONG   CodedCharSetId;    /* Reserved */
    MÇCHAR8  Format;             /* MQ format name of data that follows
                                MÇCIH */
    MÇLONG   Flags;              /* Flags */
    MÇLONG   ReturnCode;        /* Return code from bridge */
    MÇLONG   CompCode;          /* MQ completion code or CICS EIBRESP */
    MÇLONG   Reason;            /* MQ reason or feedback code, or CICS
                                EIBRESP2 */
    MÇLONG   UOWControl;        /* Unit-of-work control */
    MÇLONG   GetWaitInterval;   /* Wait interval for MÇGET call issued
                                by bridge task */
    MÇLONG   LinkType;          /* Link type */
    MÇLONG   OutputDataLength;   /* Output COMMAREA data length */
    MÇLONG   FacilityKeepTime;  /* Bridge facility release time */
    MÇLONG   ADSDescriptor;     /* Send/receive ADS descriptor */
    MÇLONG   ConversationalTask; /* Whether task can be conversational */
    MÇLONG   TaskEndStatus;     /* Status at end of task */
    MÇBYTE8  Facility;          /* Bridge facility token */
    MÇCHAR4  Function;          /* MQ call name or CICS EIBFN
                                function */
    MÇCHAR4  AbendCode;         /* Abend code */
    MÇCHAR8  Authenticator;     /* Password or passticket */
    MÇCHAR8  Reserved1;         /* Reserved */
    MÇCHAR8  ReplyToFormat;     /* MQ format name of reply message */
    MÇCHAR4  RemoteSysId;       /* Reserved */
    MÇCHAR4  RemoteTransId;     /* Reserved */
    MÇCHAR4  TransactionId;     /* Transaction to attach */
    MÇCHAR4  FacilityLike;      /* Terminal emulated attributes */
    MÇCHAR4  AttentionId;       /* AID key */
    MÇCHAR4  StartCode;         /* Transaction start code */
    MÇCHAR4  CancelCode;        /* Abend transaction code */
    MÇCHAR4  NextTransactionId; /* Next transaction to attach */
    MÇCHAR8  Reserved2;         /* Reserved */
    MÇCHAR8  Reserved3;         /* Reserved */
    MÇLONG   CursorPosition;    /* Cursor position */
    MÇLONG   ErrorOffset;       /* Offset of error in message */
    MÇLONG   InputItem;         /* Reserved */
    MÇLONG   Reserved4;         /* Reserved */
};
```

## MQCIH için COBOL bildirimi

```
** MQCIH structure
10 MQCIH.
** Structure identifier
15 MQCIH-STRUCID PIC X(4).
** Structure version number
15 MQCIH-VERSION PIC S9(9) BINARY.
** Length of MQCIH structure
15 MQCIH-STRUCLength PIC S9(9) BINARY.
** Reserved
15 MQCIH-ENCODING PIC S9(9) BINARY.
** Reserved
15 MQCIH-CODEDCHARSETID PIC S9(9) BINARY.
** MQ format name of data that follows MQCIH
15 MQCIH-FORMAT PIC X(8).
** Flags
15 MQCIH-FLAGS PIC S9(9) BINARY.
** Return code from bridge
15 MQCIH-RETURNCode PIC S9(9) BINARY.
** MQ completion code or CICS EIBRESP
15 MQCIH-COMPCODE PIC S9(9) BINARY.
** MQ reason or feedback code, or CICS EIBRESP2
15 MQCIH-REASON PIC S9(9) BINARY.
** Unit-of-work control
15 MQCIH-UOWCONTROL PIC S9(9) BINARY.
** Wait interval for MQGET call issued by bridge task
15 MQCIH-GETWAITINTERVAL PIC S9(9) BINARY.
** Link type
15 MQCIH-LINKTYPE PIC S9(9) BINARY.
** Output COMMAREA data length
15 MQCIH-OUTPUTDATALENGTH PIC S9(9) BINARY.
** Bridge facility release time
15 MQCIH-FACILITYKEEPTIME PIC S9(9) BINARY.
** Send/receive ADS descriptor
15 MQCIH-ADSDESCRIPTOR PIC S9(9) BINARY.
** Whether task can be conversational
15 MQCIH-CONVERSATIONALTASK PIC S9(9) BINARY.
** Status at end of task
15 MQCIH-TASKENDSTATUS PIC S9(9) BINARY.
** Bridge facility token
15 MQCIH-FACILITY PIC X(8).
** MQ call name or CICS EIBFN function
15 MQCIH-FUNCTION PIC X(4).
** Abend code
15 MQCIH-ABENDCODE PIC X(4).
** Password or passticket
15 MQCIH-AUTHENTICATOR PIC X(8).
** Reserved
15 MQCIH-RESERVED1 PIC X(8).
** MQ format name of reply message
15 MQCIH-REPLYTOFORMAT PIC X(8).
** Reserved
15 MQCIH-REMOTESYSID PIC X(4).
** Reserved
15 MQCIH-REMOtetransid PIC X(4).
** Transaction to attach
15 MQCIH-TRANSACTIONID PIC X(4).
** Terminal emulated attributes
15 MQCIH-FACILITYLIKE PIC X(4).
** AID key
15 MQCIH-ATTENTIONID PIC X(4).
** Transaction start code
15 MQCIH-STARTCODE PIC X(4).
** Abend transaction code
15 MQCIH-CANCELCode PIC X(4).
** Next transaction to attach
15 MQCIH-NEXTTRANSACTIONID PIC X(4).
** Reserved
15 MQCIH-RESERVED2 PIC X(8).
** Reserved
15 MQCIH-RESERVED3 PIC X(8).
** Cursor position
15 MQCIH-CURSORPOSITION PIC S9(9) BINARY.
** Offset of error in message
15 MQCIH-ERROROFFSET PIC S9(9) BINARY.
** Reserved
15 MQCIH-INPUTITEM PIC S9(9) BINARY.
```



\*\* Reserved  
15 MQCIH-RESERVED4 PIC S9(9) BINARY.

## MQCIH için PL/I bildirimi

```

dcl
  1 MQCIH based,
  3 StrucId          char(4),          /* Structure identifier */
  3 Version          fixed bin(31),    /* Structure version number */
  3 StrucLength      fixed bin(31),    /* Length of MQCIH structure */
  3 Encoding         fixed bin(31),    /* Reserved */
  3 CodedCharSetId   fixed bin(31),    /* Reserved */
  3 Format            char(8),          /* MQ format name of data that
                                     follows MQCIH */
  3 Flags            fixed bin(31),    /* Flags */
  3 ReturnCode       fixed bin(31),    /* Return code from bridge */
  3 CompCode         fixed bin(31),    /* MQ completion code or CICS
                                     EIBRESP */
  3 Reason           fixed bin(31),    /* MQ reason or feedback code, or
                                     CICS EIBRESP2 */
  3 UOWControl       fixed bin(31),    /* Unit-of-work control */
  3 GetWaitInterval fixed bin(31),    /* Wait interval for MQGET call
                                     issued by bridge task */
  3 LinkType         fixed bin(31),    /* Link type */
  3 OutputDataLength fixed bin(31),    /* Output COMMAREA data length */
  3 FacilityKeepTime fixed bin(31),    /* Bridge facility release time */
  3 ADSDescriptor    fixed bin(31),    /* Send/receive ADS descriptor */
  3 ConversationalTask fixed bin(31), /* Whether task can be
                                     conversational */
  3 TaskEndStatus    fixed bin(31),    /* Status at end of task */
  3 Facility         char(8),          /* Bridge facility token */
  3 Function         char(4),          /* MQ call name or CICS EIBFN
                                     function */
  3 AbendCode        char(4),          /* Abend code */
  3 Authenticator    char(8),          /* Password or passticket */
  3 Reserved1        char(8),          /* Reserved */
  3 ReplyToFormat    char(8),          /* MQ format name of reply
                                     message */
  3 RemoteSysId      char(4),          /* Reserved */
  3 RemoteTransId    char(4),          /* Reserved */
  3 TransactionId    char(4),          /* Transaction to attach */
  3 FacilityLike     char(4),          /* Terminal emulated attributes */
  3 AttentionId      char(4),          /* AID key */
  3 StartCode        char(4),          /* Transaction start code */
  3 CancelCode       char(4),          /* Abend transaction code */
  3 NextTransactionId char(4),          /* Next transaction to attach */
  3 Reserved2        char(8),          /* Reserved */
  3 Reserved3        char(8),          /* Reserved */
  3 CursorPosition   fixed bin(31),    /* Cursor position */
  3 ErrorOffset      fixed bin(31),    /* Offset of error in message */
  3 InputItem        fixed bin(31),    /* Reserved */
  3 Reserved4        fixed bin(31);    /* Reserved */

```

## MQCIH için High Level Assembler bildirimi

MQCIH	DSECT	
MQCIH_STRUCID	DS CL4	Structure identifier
MQCIH_VERSION	DS F	Structure version number
MQCIH_STRUCLNGTH	DS F	Length of MQCIH structure
MQCIH_ENCODING	DS F	Reserved
MQCIH_CODEDCHARSETID	DS F	Reserved
MQCIH_FORMAT	DS CL8	MQ format name of data that follows MQCIH
*		
MQCIH_FLAGS	DS F	Flags
MQCIH_RETURNCODE	DS F	Return code from bridge
MQCIH_COMPCODE	DS F	MQ completion code or CICS EIBRESP
MQCIH_REASON	DS F	MQ reason or feedback code, or CICS EIBRESP2
*		
MQCIH_UOWCONTROL	DS F	Unit-of-work control
MQCIH_GETWAITINTERVAL	DS F	Wait interval for MQGET call issued by bridge task
*		
MQCIH_LINKTYPE	DS F	Link type
MQCIH_OUTPUTDATALENGTH	DS F	Output COMMAREA data length
MQCIH_FACILITYKEEPTIME	DS F	Bridge facility release time
MQCIH_ADSDSCRIPTOR	DS F	Send/receive ADS descriptor
MQCIH_CONVERSATIONALTASK	DS F	Whether task can be conversational
MQCIH_TASKENDSTATUS	DS F	Status at end of task

MQCIH_FACILITY	DS	XL8	Bridge facility token
MQCIH_FUNCTION	DS	CL4	MQ call name or CICS EIBFN function
MQCIH_ABENDCODE	DS	CL4	Abend code
MQCIH_AUTHENTICATOR	DS	CL8	Password or passticket
MQCIH_RESERVED1	DS	CL8	Reserved
MQCIH_REPLYTOFORMAT	DS	CL8	MQ format name of reply message
MQCIH_REMOTESYSID	DS	CL4	Reserved
MQCIH_REMOTETRANSID	DS	CL4	Reserved
MQCIH_TRANSACTIONID	DS	CL4	Transaction to attach
MQCIH_FACILITYLIKE	DS	CL4	Terminal emulated attributes
MQCIH_ATTENTIONID	DS	CL4	AID key
MQCIH_STARTCODE	DS	CL4	Transaction start code
MQCIH_CANCELCODE	DS	CL4	Abend transaction code
MQCIH_NEXTTRANSACTIONID	DS	CL4	Next transaction to attach
MQCIH_RESERVED2	DS	CL8	Reserved
MQCIH_RESERVED3	DS	CL8	Reserved
MQCIH_CURSORPOSITION	DS	F	Cursor position
MQCIH_ERROROFFSET	DS	F	Offset of error in message
MQCIH_INPUTITEM	DS	F	Reserved
MQCIH_RESERVED4	DS	F	Reserved
*			
MQCIH_LENGTH	EQU	*-MQCIH	
	ORG	MQCIH	
MQCIH_AREA	DS	CL(MQCIH_LENGTH)	

## MQCIH için Visual Basic bildirimi

```

Type MQCIH
  StrucId          As String*4 'Structure identifier'
  Version         As Long      'Structure version number'
  StrucLength     As Long      'Length of MQCIH structure'
  Encoding        As Long      'Reserved'
  CodedCharSetId As Long      'Reserved'
  Format          As String*8  'MQ format name of data that follows'
                    'MQCIH'

  Flags          As Long      'Flags'
  ReturnCode     As Long      'Return code from bridge'
  CompCode      As Long      'MQ completion code or CICS EIBRESP'
  Reason        As Long      'MQ reason or feedback code, or CICS'
                    'EIBRESP2'

  UOWControl     As Long      'Unit-of-work control'
  GetWaitInterval As Long      'Wait interval for MQGET call issued'
                    'by bridge task'

  LinkType       As Long      'Link type'
  OutputDataLength As Long      'Output COMMAREA data length'
  FacilityKeepTime As Long      'Bridge facility release time'
  ADSDescriptor  As Long      'Send/receive ADS descriptor'
  ConversationalTask As Long      'Whether task can be conversational'
  TaskEndStatus  As Long      'Status at end of task'
  Facility       As MQBYTE8   'Bridge facility token'
  Function       As String*4   'MQ call name or CICS EIBFN function'
  AbendCode     As String*4   'Abend code'
  Authenticator As String*8   'Password or passticket'
  Reserved1     As String*8   'Reserved'
  ReplyToFormat As String*8   'MQ format name of reply message'
  RemoteSysId   As String*4   'Reserved'
  RemoteTransId As String*4   'Reserved'
  TransactionId As String*4   'Transaction to attach'
  FacilityLike  As String*4   'Terminal emulated attributes'
  AttentionId   As String*4   'AID key'
  StartCode     As String*4   'Transaction start code'
  CancelCode    As String*4   'Abend transaction code'
  NextTransactionId As String*4 'Next transaction to attach'
  Reserved2     As String*8   'Reserved'
  Reserved3     As String*8   'Reserved'
  CursorPosition As Long      'Cursor position'
  ErrorOffset   As Long      'Offset of error in message'
  InputItem     As Long      'Reserved'
  Reserved4     As Long      'Reserved'
End Type

```

## Kullanım

Uygulama Çizelge 477 sayfa 297’inde gösterilen ilk değerlerle aynı değerleri gerektiriyorsa ve köprü AUTH=LOCAL ya da AUTH=TANIMLA ile çalışıyorsa, iletiden MQCIH yapısını atlayabilirsiniz. Diğer tüm durumlarda, yapı mevcut olmalıdır.

Köprü version-1 ya da version-2 MÇCIH yapısını kabul eder, ancak 3270 hareketleri için version-2 yapısı kullanmanız gerekir.

Uygulama, istek alanları olarak belgelenen alanların, köprüye gönderilen iletide uygun değerlere sahip olmasını sağlamalıdır; bu alanlar köprüye giriştir.

Yanıt alanları olarak belgelenen alanlar, köprünün uygulamaya gönderdiği yanıt iletisinde CICS bridge tarafından ayarlanır. *ReturnCode*, *Function*, *CompCode*, *Reason* ve *AbendCode* alanlarında hata bilgileri döndürülür, ancak bunların tümü her durumda ayarlanmaz. Aşağıdaki tabloda, farklı *ReturnCode* değerleri için hangi alanların ayarlandığı gösterilmektedir.

<i>Çizelge 478. MÇCIH için MÇCIH yapısındaki hata bilgisi alanlarının içeriği</i>				
<b>ReturnCode</b>	<b>Function</b>	<b>CompCode</b>	<b>Reason</b>	<b>AbendCode</b>
MÇCRC_OK	-	-	-	-
MÇCRC_BRIDGE_HATA	-	-	MÇFB_CICS_*	-
MÇCRC_MÇ_API_ERROR MÇCRC_BRIDGE_TIMEOUT	MÇ çağrı adı	MÇ <i>CompCode</i>	MÇ <i>Reason</i>	-
MÇCRC_CICS_EXEC_ERROR MÇCRC_SECURITY_ERROR MÇCRC_PROGRAM_NOT_KULLANILABİLİR MÇCRC_TRANSID_NOT_VAR	CICS EIBFN	CICS EIBRESP	CICS EIBRESP2	-
MÇCRC_BRIDGE_ABEND MÇCRC_APPLICATION_ABEND	-	-	-	CICS KODU

### **StrucId (MÇCHAR4)**

Bu alan, ilk MÇCIH\_STRUC\_ID değeri olan bir istek alanıdır.

Değer şu olmalıdır:

#### **MÇCIH\_STRUC\_ID**

CICS bilgi üstbilgisi yapısına ilişkin tanıtıcı.

C programlama dili için, MÇCIH\_STRUC\_ID\_ARRAY değişmezi de tanımlanır; bu, MÇCIH\_STRUC\_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisidir.

### **Sürüm (MÇUZE)**

Bu alan bir istek alanıdır. İlk değeri MÇCIH\_VERSION\_2' dir.

Değer aşağıdakilerden biri olmalıdır:

#### **MÇCIH\_VERSION\_1**

Version-1 CICS bilgi üstbilgisi yapısı.

#### **MÇCIH\_VERSION\_2**

Version-2 CICS bilgi üstbilgisi yapısı.

Yalnızca yapının daha yeni sürümünde var olan alanlar, alanların açıklamalarında olduğu gibi tanımlanır.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

#### **MÇCIH\_CURRENT\_VERSION**

CICS bilgi üstbilgisi yapısının yürürlükteki sürümü.

### **StrucLength (MÇUZE)**

Bu alan, başlangıç değeri olan MÇCIH\_LENGTH\_2 olan bir istek alanıdır.

Değer aşağıdakilerden biri olmalıdır:

#### **MÇCIH\_LENGTH\_1**

version-1 CICS bilgi üstbilgisi yapısının uzunluğu.

## **MQCIH\_LENGTH\_2**

version-2 CICS bilgi üstbilgisi yapısının uzunluğu.

Aşağıdaki değişmez, yürürlükteki sürümün uzunluğunu belirtir:

## **MQCIH\_CURRENT\_LENGTH**

CICS bilgi üstbilgisi yapısının geçerli sürümünün uzunluğu.

## **Kodlama (MQUZE)**

Bu alan, ayrılmış bir alandır; değeri anlamlı değildir. Başlangıç değeri 0 'tır.

MQCIH yapısını izleyen desteklenen yapılara ilişkin kodlama, MQCIH yapısının kendi kodlaması ile aynıdır ve yukarıdaki herhangi bir IBM MQ üstbilgisinden alınır.

## **CodedCharSetId (MQHOMER)**

CodedCharSetId , ayrılmış bir alandır; değeri anlamlı değil. Bu alanın ilk değeri 0 'tır.

MQCIH yapısını izleyen desteklenen yapılara ilişkin Karakter Kümesi Tanıtıcısı, MQCIH yapısının kendisinin karakter kümesi tanıtıcısıyla aynıdır ve yukarıdaki herhangi bir IBM MQ üstbilgisinden alınır.

## **Biçim (MQCHAR8)**

Bu alanda, MQCIH yapısından sonra gelen verilerin IBM MQ biçim adı gösterilir.

MQPUT ya da MQPUT1 çağrısında, uygulama bu alanı veri için uygun değere ayarlamalıdır. Bu alanın kodlanması için kurallar, MQMD ' de *Format* alanının kodlanması için kurullarla aynıdır.

*ReplyToFormat* alanında MQFMT\_NONE değeri varsa, yanıt iletisi için bu biçim adı da kullanılır.

- DPL istekleri için *Format* , COMMAREA ' nın biçim adı olmalıdır.
- For 3270 requests, *Format* must be CSQCBDCI, and the bridge sets the format to CSQCBDCO for Reply messages.

Bu biçimlere ilişkin veri dönüştürme çıkışları, çalıştırılacağı kuyruk yöneticisine kurulmalıdır.

İstek iletisi bir hata yanıtı iletisi oluşturduysa, hata yanıt iletisinin biçim adı MQFMT\_STRING olur.

Bu alan bir istek alanıdır. Bu alanın uzunluğu MQ\_FORMAT\_LENGTH tarafından verilir. Bu alanın ilk değeri MQFMT\_NONE olur.

## **İşaretler (MQUZE)**

Bu alan bir istek alanıdır. Bu alanın ilk değeri MQCIH\_NONE olur.

Değer şu olmalıdır:

### **MQCIH\_NONE**

Bayrak yok.

### **MQCIH\_PASS\_EXPECTION**

Yanıt iletisi şunları içerir:

- İstek iletisine göre aynı süre bitimi rapor seçenekleri.
- Köprünün işleme süresi için ayarlama yapılmasıyla, istek iletisinden kalan süre bitimi (süre bitimi).

Bu değeri çıkarırsanız, süre bitim süresi *sınırsız* olarak ayarlanır.

### **MQCIH\_REPLY\_WITHOUT\_NULLS**

Bir CICS DPL programı isteğinin yanıt iletisi uzunluğu, DPL programı tarafından döndürülen COMMAREA 'nın sonunda sondaki boş değerleri (X'00') dışlamak için ayarlanır. Bu değer ayarlanmazsa, boş değerler önemli olabilir ve tam COMMAREA döndürülür.

### **MQCIH\_SYNC\_ON\_RETURN**

DPL istekleri için CICS bağlantısı, SYNCONRETURN seçeneğini kullanır; program, başka bir CICS bölgesine gönderilmişse, program tamamlandığında CICS ' un bir eşitleme noktasını almasına neden

olur. The bridge does not specify to which CICS region to ship the request; that is controlled by the CICS program definition or workload balancing facilities.

### **ReturnCode (MQLONG)**

Bu alanın değeri, köprü tarafından gerçekleştirilen işlemlerin sonucunu açıklayan CICS bridge dönüş kodudur. Bu alan, ilk değeri MQCRC\_OK olan bir yanıt alanıdır.

*Function, CompCode, Reasonve AbendCode* alanları ek bilgi içerebilir (bkz. [Çizelge 478 sayfa 303](#)). Değer aşağıdakilerden biridir:

#### **MQCRC\_APPLICATION\_ABEND**

(5, X'005 ') Uygulama olağandışı bitti.

#### **MQCRC\_BRIDGE\_ABEND**

(4, X'004 ') CICS bridge olağandışı bitti.

#### **MQCRC\_BRIDGE\_HATA**

(3, X'003 ') CICS bridge bir hata algıladı.

#### **MQCRC\_BRIDGE\_TIMEOUT**

(8, X'008 ') Yürürlükteki iş birimi içindeki ikinci ya da sonraki ileti, belirtilen süre içinde alınmadı.

#### **MQCRC\_CICS\_EXEC\_ERROR (MQCRC\_CICS\_HATASI)**

(1, X'001 ') EXEC CICS deyimi bir hata saptadı.

#### **MQCRC\_MQ\_API\_ERROR**

(2, X'002 ') MQ çağırısı bir hata saptadı.

#### **MQCRC\_OK**

(0, X'000 ') Hata yok.

#### **MQCRC\_PROGRAM\_YOK**

(7, X'007 ') Program kullanılmıyor.

#### **MQCRC\_SECURITY\_HATA**

(6, X'006 ') Güvenlik hatası oluştu.

#### **MQCRC\_TRANSID\_NOT\_KULLANILAMIYOR**

(9, X'009 ') İşlem kullanılmıyor.

### **CompCode (MQLONG)**

Bu alan bir yanıt alanıdır. İlk değeri MQCC\_OK

Bu alanda döndürülen değer *ReturnCode* ; bkz. [Çizelge 478 sayfa 303](#).

### **Neden (MQLONG)**

Bu alan bir yanıt alanıdır. İlk değeri MQRC\_NONE.

Bu alanda döndürülen değer *ReturnCode* ; bkz. [Çizelge 478 sayfa 303](#).

### **UOWControl (MQLONG)**

Bu alan, CICS bridgetarafından gerçekleştirilen iş birimi işlemlerini denetleyen bir istek alanıdır. Bu alanın ilk değeri MQCUOWC\_ONLY değeridir.

Köprüyü tek bir işlem çalıştırmasını ya da bir iş birimi içinde bir ya da daha fazla program çalıştırmasını isteyebilirsiniz. Bu alan, CICS bridge ' in bir iş birimi başlattığı, yürürlükteki iş birimi içinde istenen işlevi gerçekleştirip gerçekleştirmeyeceğini ya da kesinleştirerek ya da yedeklemesiyle iş birimini sona erdirip bitirmediğini gösterir. Veri iletim akışlarını eniyilemek için çeşitli birleşimler desteklenir.

Değer aşağıdakilerden biri olmalıdır:

#### **YALNIZCIK**

İş birimini başlatın, işlev gerçekleştirin, sonra iş birimini kesinleştirin.

#### **MQCUOWC\_CONTINY**

Yürürlükteki iş birimine ilişkin ek veriler (yalnızca 3270).

**İLK MQCUOWC\_FIRST**

İş birimini başlatın ve işlev gerçekleştirin.

**MQCUOWC\_ORTA**

Yürürlükteki iş birimi içinde işlev gerçekleştir

**MQCUOWC\_SON**

İşlev gerçekleştirin ve iş birimini kesinleştirin.

**MQCUOWC\_COMMIT**

İş birimini kesinleştirin (yalnızca DPL).

**MQCUOWC\_BACKOUT**

Çalışma birimini yedekle (yalnızca DPL).

**GetWaitAralığı (MQUZE)**

Bu alan bir istek alanıdır. Başlangıç değeri MQCGWI\_VARSAYILANDIR.

Bu alan yalnızca, *UOWControl* MQCUOWC\_FIRST değerine sahip olduğunda geçerlidir. Bu ileti, gönderme uygulamasının, köprü tarafından yayınlanan MQGET çağrılarının, bu ileti tarafından başlatılan iş birimine ilişkin ikinci ve sonraki istek iletilerini bekleyeceği yaklaşık süreyi milisaniye olarak belirlemesini sağlar. Bu olanak, köprü tarafından kullanılan varsayılan bekleme aralığını geçersiz kılar. Aşağıdaki özel değerleri kullanabilirsiniz:

**MQCGWI\_VARSAYILAN**

Varsayılan bekleme aralığı.

This value causes the CICS bridge to wait for the time specified when the bridge was started.

**MQWI\_UNSNMA**

Sınırsız bekleme aralığı.

**LinkType (MQUZE)**

Bu alan bir istek alanıdır. İlk değeri MQCLT\_PROGRAR değeridir.

Bu değer, köprü'nün bağlanmayı denediği nesnenin tipini belirtir. Bu değer aşağıdaki değerlerden biri olmalıdır:

**MQCLT\_PROGRAM**

DPL programı.

**MQCLT\_TRANSACTION**

3270 hareketi.

**OutputDataUzunluk (MQUZE)**

Bu alan, yalnızca DPL programları için kullanılan bir istek alanıdır. İlk değeri MQCODL\_AS\_INPUT değeridir.

Bu değer, yanıt iletilinde istemciye döndürülebilmek için kullanıcı verilerinin uzunluğudur. Bu uzunluk, 8 byte 'lık program adını içerir. Bağlantılı programa geçirilen COMMAREA uzunluğu, bu alanın üst sınırı ve istek iletilinde kullanıcı verilerinin uzunluğunun (eksi 8) uzunluğudur.

**Not:** Bir iletteki kullanıcı verilerinin uzunluğu, MQCIH yapısı hariç olmak üzere iletinin uzunluğudur.

İstek iletilinde kullanıcı verilerinin uzunluğu *OutputDataLength*' dan küçük olursa, LINK komutunun DATALENGTH seçeneği kullanılır ve LINK , başka bir CICS bölgesine verimli bir şekilde işlev sevk edilir.

Aşağıdaki özel değeri kullanabilirsiniz:

**MQCODL\_AS\_INPUT**

Çıkış uzunluğu giriş uzunluğuna göre aynı.

Bağlantılı programa geçirilen COMMAREA ' nın yeterli boyutta olduğundan emin olmak için, yanıt istenmese de bu değer gerekebilir.

### **FacilityKeepSaat (MQUZE)**

FacilityKeepSüre, kullanıcı işlemi sona erdikten sonra köprü tesisinin alıkonacağı sürenin saniye cinsinden uzunluğudur.

Sözde etkileşimli işlemler için, sözde bir dönüştürücü işleminin beklenen süresine karşılık gelen bir değer belirtin; bir sözde sohbetin son işlemi için sıfır belirtin ve diğer işlem tipleri için sıfır belirtin.

Bu alan, yalnızca 3270 işlemleri için kullanılan bir istek alanıdır. Bu alanın ilk değeri 0 'tır.

### **ADSDescriptor (MQUZE)**

Bu alan, SEND ve RECEIVE BMS isteklerinde ADS tanımlayıcılarının gönderilip gönderilmeyeceğini belirleyen bir göstergedir.

Aşağıdaki değerler tanımlanır:

#### **MQCADSD\_NONE**

ADS tanımlayıcıları göndermeyin ya da göndermeyin.

#### **MQCADSD\_SEND**

ADS tanımlayıcıları gönderin.

#### **MQCADSD\_RECV**

ADS tanımlayıcıları alın.

#### **MQCADSD\_MSGSAYI**

ADS açıklayıcıları için ileti biçimini kullanın.

Bu, ADS tanımlayıcısının uzun biçimini kullanarak ADS tanımlayıcılarını gönderir ya da alır. Uzun biçimli alanlar, 4 baytlık sınırlar üzerinde hizalanmış alanlar içerir.

*ADSDescriptor* alanını aşağıdaki gibi ayarlayın:

- ADS tanımlayıcıları kullanmayorsanız, alanı MQCADSD\_NONE olarak ayarlayın.
- Her ortamda *aynı* CCSID değeri ile ADS tanımlayıcıları kullanıyorsanız, alanı MQCADSD\_SEND ve MQCADSD\_RECV toplamıyla ayarlayın.
- Her ortamda *farklı* CCSID ' lerle ADS tanımlayıcıları kullanıyorsanız, alanı MQCADSD\_SEND, MQCADSD\_RECV ve MQCADSD\_MSGFORBIT toplayacak şekilde ayarlayın.

Bu, yalnızca 3270 işlemleri için kullanılan bir istek alanıdır. Bu alanın ilk değeri MQCADSD\_NONE olur.

### **ConversationalTask (MQUSED)**

Bu alan, daha fazla bilgi için ya da görevi durdurup olağandışı bir ileti yayınlamasına izin verilip verilmeyeceğini belirleyen bir göstergedir.

Değer, aşağıdaki seçeneklerden biri olmalıdır:

#### **MQCCT\_YES**

Bu görev birbirine yakındır.

#### **MQCCT\_NO**

Bu görev birbirine yakınmacı değil.

Bu alan, yalnızca 3270 işlemleri için kullanılan bir istek alanıdır. Bu alanın ilk değeri MQCCT\_NO olur.

### **TaskEndDurumu (MQUZE)**

Bu alan, görev sonunda kullanıcı hareketinin durumunu gösteren bir yanıt alanıdır. Bu alan yalnızca 3270 hareketleri için kullanılır ve ilk değeri MQCTES\_NOSYNC 'dir.

Aşağıdaki değerlerden biri döndürülür:

#### **MQCTES\_NOSYNC**

Eşitlenmedi.

Kullanıcı hareketi henüz tamamlanmamış ve eşitlenmiş değil. MQMD 'deki *MsgType* alanı, bu vakadaki MQMT\_REQUEST' tır.

## **MQCTES\_COMMIT**

İşlerin kesinleştirilmesinin birimi.

Kullanıcı işlemi henüz tamamlanmadı, ancak ilk iş birimini eşitlemiştir. MQMD 'deki *MsgType* alanı, bu vakada MQMT\_DATAGRAM' tır.

## **MQCTES\_BACKUT**

Çalışma birimi geri döndü.

Kullanıcı işlemi henüz tamamlanmadı. Yürürlükteki iş birimi geriletilir. MQMD 'deki *MsgType* alanı, bu vakada MQMT\_DATAGRAM' tır.

## **MQCTES\_ENDTASK**

Görevi sona erdirin.

Kullanıcı işlemi sona erdi (ya da olağandışı sona erdirildi). Bu vakada MQMD 'deki *MsgType* alanı MQMT\_REPLY' dir.

## **Tesis (MQBYTE8)**

Bu alanda, 8 baytlık köprü olanağı simgesi gösterilir.

Bir köprü olanağı simgesi, bir sözde etkileşimde birden çok işlemin aynı köprü olanağını (sanal 3270 uçbirimi) kullanmasına olanak sağlar. Bir sözde sohbetteki ilk ya da yalnızca ileti içinde MQCFAC\_NONE değerini ayarlayın. Bu değer, CICS ' a bu ileti için yeni bir köprü olanağı ayırmayı bildirir. Giriş iletilerinde sıfır olmayan bir *FacilityKeepTime* belirtildiğinde yanıt iletilerinde bir köprü olanağı simgesi döndürülür. Bir sözde etkileşim içindeki sonraki giriş iletileri, aynı köprü olanağı simgesini kullanmalıdır.

Aşağıdaki özel değer tanımlıdır:

## **MQCFAC\_NONE**

Herhangi bir tesis simgesi belirtilmedi.

C programlama dili için, sabit MQCFACE\_NONE\_DIZISI de tanımlanır ve MQCFAC\_NONE ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisidir.

Bu alan hem bir istek, hem de yalnızca 3270 işlemleri için kullanılan bir yanıt alanıdır. Bu alanın uzunluğu MQ\_FACILITY\_LENGTH tarafından verilir. Bu alanın ilk değeri MQCFAC\_NONE olur.

## **İşlev (MQCHAR4)**

Bu alan bir yanıt alanıdır. Bu alanın uzunluğu MQ\_FUNCTION\_LENGTH tarafından verilir. Bu alanın ilk değeri MQCFUNC\_NONE.

Bu alanda döndürülen değer *ReturnCode* ; bkz. [Çizelge 478 sayfa 303](#). *Function* bir IBM MQ arama adı içerdiğinde aşağıdaki değerler kullanılabilir:

## **MQCFUNC\_MQCONN**

MQCONN çağrısı.

## **MQCFUNC\_MQGET**

MQGET çağrısı.

## **MQCFUNC\_MQINQ**

MQINQ çağrısı.

## **MQCFUNC\_MQOPEN**

MQOPEN çağrısı.

## **MQCFUNC\_MQPUT**

MQPUT çağrısı.

## **MQCFUNC\_MQPUT1**

MQPUT1 çağrısı.

## **MQCFUNC\_NONE**

Telefon yok.



Her durumda, C programlama dili için, MQCFUNC\_\*\_ARRAY değişmezleri de tanımlanır; bu değişmezler, karşılık gelen MQCFUNC\_\* değişmezleriyle aynı değerlere sahiptir, ancak dizgiler yerine karakter dizileridir.

### **AbendCode (MQCHAR4)**

AbendCode , bir yanıt alanıdır. Bu alanın uzunluğu MQ\_ABEND\_CODE\_LENGTH tarafından verilir. Bu alanın ilk değeri 4 boş karakterdir.

Bu alanda döndürülen değer, yalnızca *ReturnCode* alanının MQCRC\_APPLICATION\_ABEND ya da MQCRC\_BRIDGE\_ABEND değerine sahip olması durumunda önemlidir. Yapılırsa, *AbendCode* , CICS ABCODE değerini içerir.

### **Kimlik doğrulayıcı (MQCHAR8)**

Bu alanın değeri, parola ya da passticket 'dir.

If user-identifier authentication is active for the CICS bridge, *Authenticator* is used with the user identifier in the MQMD identity context to authenticate the sender of the message.

Bu bir istek alanıdır. Bu alanın uzunluğu MQ\_AUTHENTICATOR\_LENGTH tarafından verilir. Bu alanın ilk değeri 8 boşluktur.

### **Reserved1 (MQCHAR8)**

Bu alan ayrılmış bir alandır. Değer 8 boşluk olmalıdır.

### **ReplyToBiçimi (MQCHAR8)**

Bu alanın değeri, yürürlükteki iletiye yanıt olarak gönderilen yanıt iletisinin IBM MQ biçimi adıdır.

Bu alanın kodlanması için kurallar, MQMD ' de *Format* alanının kodlanması için bu kurallarla aynıdır.

Bu alan, yalnızca DPL programları için kullanılan bir istek alanıdır. Bu alanın uzunluğu MQ\_FORMAT\_LENGTH tarafından verilir. Bu alanın ilk değeri MQFMT\_NONE olur.

### **RemoteSysTanıtıcısı (MQCHAR4)**

Bu alanda, isteğin işlenmesine ilişkin CICS sisteminin CICS sistem tanıtıcısı gösterilir.

Bu alan boş bırakılırsa, CICS sistem isteği, köprü izleyicisi ile aynı CICS sisteminde işlenir. Kullanılan SYSID yanıt iletisinde döndürülür.

3270 sözde sohbetinde, sohbetteki sonraki tüm iletiler, ilk yanıtta döndürülen uzak SYSID ' yi belirtmelidir. Belirtirse, SYSNID aşağıdaki gibi olmalıdır:

- Aktif olun.
- IBM MQ İstek kuyruğuna erişiminiz var.
- Köprü izleyicisinin CICS sisteminden CICS ISC bağlantıları tarafından erişilebilir.

### **RemoteTransTanıtıcısı (MQCHAR4)**

Bu alan, isteğe bağlı bir İstek alanıdır. Bu alanın uzunluğu MQ\_TRANSACTION\_ID\_LENGTH tarafından verilir.

Belirtirse, alan, CICS START ' nin RTRANSID değeri olarak kullanılır.

### **TransactionId (MQCHAR4)**

Bu alan bir istek alanıdır. Uzunluğu MQ\_TRANSACTION\_ID\_LENGTH tarafından verilir. Bu alanın ilk değeri dört boşluktur.

If *LinkType* has the value MQCLT\_TRANSACTION, *TransactionId* is the transaction identifier of the user transaction to be run; specify a nonblank value in this case.

If *LinkType* has the value MQCLT\_PROGRAM, *TransactionId* is the transaction code under which all programs within the unit of work are to be run. Boş bir değer belirtirseniz, CICS DPL köprüsü varsayılan hareket kodu (CKBP) kullanılır. If the value is nonblank, you must have defined it to CICS as a local

transaction with an initial program that is CSQCBP00. Bu alan yalnızca *UOWControl* , MQCUOWC\_FIRST ya da MQCUOWC\_ONLY değerine sahip olduğunda geçerlidir.

### **FacilityLike (MQCHAR4)**

FacilityLike , köprü tesisine ilişkin model olarak kullanılacak kurulu bir uçbirimin adıdır.

A value of blanks means that *FacilityLike* is taken from the bridge transaction profile definition, or a default value is used.

Bu alan, yalnızca 3270 işlemleri için kullanılan bir istek alanıdır. Bu alanın uzunluğu MQ\_FACILITY\_LIKE\_LENGTH tarafından verilir. Bu alanın ilk değeri dört boşluktur.

### **AttentionId (MQCHAR4)**

Bu alandaki değer, işlem başlatıldığında AID tuşunun ilk değerini belirler. 1 byte 'lık bir değer, sola hizalanmış.

AttentionId , yalnızca 3270 işlemleri için kullanılan bir istek alanıdır. Bu alanın uzunluğu MQ\_ATTEN\_ID\_LENGTH tarafından verilir. Bu alanın ilk değeri dört boşluktur.

### **StartCode (MQCHAR4)**

Bu alanın değeri, köprü'nün bir uçbirim hareketi öykünmesini mi, yoksa START ile başlatılan bir işlemi mi taklit ettiğini belirten bir göstergedir.

Değer aşağıdakilerden biri olmalıdır:

#### **MQCSC\_START**

Başlatma.

#### **MQCSC\_STARTDATA**

Verileri başlatın.

#### **MQCSC\_TERMINPUT**

Terminal girişi.

#### **MQCSC\_NONE**

Yok.

Tüm durumlarda, C programlama dili için, MQCSC\_\*\_ARRAY değişmezleri de tanımlanır; bu değişmezler, karşılık gelen MQCSC\_\* değişmezleriyle aynı değerlere sahiptir, ancak dizgiler yerine karakter dizileridir.

Köprüden gelen yanıtta, bu alan, *NextTransactionId* alanında bulunan bir sonraki işlem tanıtıcısına uygun başlangıç koduna ayarlanır. Yanıtta aşağıdaki başlangıç kodları olabilir:

- MQCSC\_START
- MQCSC\_STARTDATA
- MQCSC\_TERMINPUT

CICS Transaction Server 1.2 için, bu alan yalnızca bir istek alanıdır; yanıtta değeri tanımsız olur.

CICS Transaction Server 1.3 ve sonraki yayınlar için bu alan hem bir istek, hem de bir yanıt alanıdır.

Bu alan yalnızca 3270 hareketleri için kullanılır. Bu alanın uzunluğu MQ\_START\_CODE\_LENGTH tarafından verilir. Bu alanın ilk değeri MQCSC\_NONE olur.

### **CancelCode (MQCHAR4)**

Bu alandaki değer, hareketi sonlandırmak için kullanılacak olağandışı bitiş kodudur (olağan durumda, daha fazla veri isteyen bir etkileşimli işlem). Ters durumda, bu alan boşluk karakterine ayarlanır.

Bu alan, yalnızca 3270 işlemleri için kullanılan bir istek alanıdır. Bu alanın uzunluğu MQ\_CANCEL\_CODE\_LENGTH tarafından verilir. Bu alanın ilk değeri dört boşluktur.

### **NextTransactionTanıtıcısı (MQCHAR4)**

Bu değer, kullanıcı hareketi tarafından döndürülen sonraki işlemin adıdır (genellikle EXEC CICS RETURN TRANSID tarafından verilir). Sonraki hareket yoksa, bu alan boşluk karakterine ayarlanır.

Bu alan, yalnızca 3270 işlemleri için kullanılan bir yanıt alanıdır. Bu alanın uzunluğu MQ\_TRANSACTION\_ID\_LENGTH tarafından verilir. Bu alanın ilk değeri dört boşluktur.

### **Reserved2 (MQCHAR8)**

Bu alan ayrılmış bir alandır. Değer 8 boşluk olmalıdır.

### **Reserved3 (MQCHAR8)**

Bu alan ayrılmış bir alandır. Değer 8 boşluk olmalıdır.

### **CursorPosition (MQUZE)**

Bu alandaki değer, hareket başlatıldığında ilk imleç konumunu gösterir. Etkileşimli işlemler için, imleç konumu RECEIVE vektöründe yer alıyor.

Bu alan, yalnızca 3270 işlemleri için kullanılan bir istek alanıdır. Bu alanın ilk değeri 0 'tır. *Version* , MQCIH\_VERSION\_2değerinden küçükse bu alan yoktur.

### **ErrorOffset (MQUZE)**

ErrorOffset alanı, köprü çıkışı tarafından saptanan geçersiz verilerin konumunu gösterir. Bu alan, iletinin başlangıcından, geçersiz verilerin yerine görelî konum sağlar.

ErrorOffset , yalnızca 3270 işlemleri için kullanılan bir yanıt alanıdır. Bu alanın ilk değeri 0 'tır. *Version* , MQCIH\_VERSION\_2değerinden küçükse bu alan yoktur.

### **InputItem (MQUZE)**

Bu alan ayrılmış bir alandır. Değer 0 olmalıdır.

*Version* , MQCIH\_VERSION\_2değerinden küçükse bu alan yoktur.

### **Reserved4 (MQUZE)**

Bu alan ayrılmış bir alandır. Değer 0 olmalıdır.

*Version* , MQCIH\_VERSION\_2değerinden küçükse bu alan yoktur.

## **MQCMHO-İleti tanıtıcı seçenekleri yarat**

**MQCMHO** yapısı, uygulamaların ileti tanıtıcılarının nasıl yaratılacağını denetleyen seçenekleri belirtmesine olanak sağlar. Yapı, **MQCRTMH** çağrısındaki bir giriş parametresidir.

### **Kullanılabilirlik**

**MQCMHO** yapısı aşağıdaki platformlarda bulunur:

-  AIX
-  IBM i
-  Linux
-  Windows
-  z/OS

ve IBM MQ istemcileriyle birlikte kullanın.

## Karakter kümesi ve kodlama

**MQCMHO** içindeki veriler, uygulamanın karakter kümesinde ve uygulamanın kodlamasında olmalıdır (**MQENC\_NATIVE**).

## Alanlar

**Not:** Aşağıdaki tabloda, alanlar alfabetik olarak değil, kullanıma göre gruplanır. Alt konular aynı sırayı izler.

Çizelge 479. MQCMHO içindeki alanlar		
Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
StrucId (yapı tanıtıcısı)	MQCMHO_STRUC_ID	'CMHO'
Sürüm (yapı sürümü numarası)	MQCMHO_VERSION_1	1
Seçenekler (seçenekler)	MQCMHO_DEFAULT_VAL IDATION	0

**Notlar:**

1. C programlama dilinde, makro değişkeni **MQCMHO\_DEFAULT**, çizelgede listelenen değerleri içerir. Yapıdaki alanlar için başlangıç değerleri sağlamak üzere aşağıdaki şekilde kullanılabilir:

```
MQCMHO MyCMHO = {MQCMHO_DEFAULT};
```

## Dil bildirimleri

MQCMHO için C bildirimini

```
struct tagMQCMHO {
    MQCHAR4   StrucId;           /* Structure identifier */
    MQLONG    Version;          /* Structure version number */
    MQLONG    Options;         /* Options that control the action of MQCRTMH */
};
```

MQCMHO için COBOL bildirimini

```
** MQCMHO structure
10 MQCMHO.
** Structure identifier
15 MQCMHO-STRUCID PIC X(4).
** Structure version number
15 MQCMHO-VERSION PIC S9(9) BINARY.
** Options that control the action of MQCRTMH
15 MQCMHO-OPTIONS PIC S9(9) BINARY.
```

MQCMHO için PL/I bildirimini

```
dcl
1 MQCMHO based,
3 StrucId char(4), /* Structure identifier */
3 Version fixed bin(31), /* Structure version number */
3 Options fixed bin(31), /* Options that control the action of MQCRTMH */
```

MQCMHO için High Level Assembler bildirimini

```
MQCMHO DSECT
MQCMHO_STRUCID DS CL4 Structure identifier
MQCMHO_VERSION DS F Structure version number
MQCMHO_OPTIONS DS F Options that control the action of
```

*			MQCRTMH
MQCMHO_LENGTH	EQU	*	MQCMHO
MQCMHO_AREA	DS		CL(MQCMHO_LENGTH)

### **StrucId (MQCHAR4)**

Bu alan her zaman bir giriş alanıdır. Başlangıç değeri MQCMHO\_STRUC\_ID 'dir.

Bu, yapı tanıtıcısıdır; değer şöyle olmalıdır:

#### **MQCMHO\_STRUC\_ID**

İleti tanıtıcısı seçenekleri yapısının yaratılmasına ilişkin tanıtıcı.

For the C programming language, the constant **MQCMHO\_STRUC\_ID\_ARRAY** is also defined; this has the same value as **MQCMHO\_STRUC\_ID**, but is an array of characters instead of a string.

### **Sürüm (MQUZE)**

Bu alan her zaman bir giriş alanıdır. İlk değeri MQCMHO\_VERSION\_1' dir.

Bu, yapı sürüm numarasıdır; değer şu olmalıdır:

#### **MQCMHO\_VERSION\_1**

Version-1 , ileti tanıtıcısı seçenekleri yapısını yaratır.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

#### **MQCMHO\_CURRENT\_VERSION**

İleti tanıtıcısı yaratma seçenekleri yapısının yürürlükteki sürümü.

### **Seçenekler (MQUZE)**

Bu alan her zaman bir giriş alanıdır. İlk değeri MQCMHO\_DEFAULT\_VALIDATION değeridir.

Aşağıdaki seçeneklerden biri belirlenebilir:

#### **MQCMHO\_VALIDATE**

Bu ileti tanıtıcısında bir özellik ayarlamak için **MQSETMP** çağrıldığında, özellik adının doğrulanmasını sağlamak için bu özellik adının geçerliliği denetlenir:

- geçersiz karakter içeriyor.
- JMS ya da usr 'i başlamıyor.JMS dışında, aşağıdaki durumlar dışında:

- JMSCorrelationID
- JMSReplyTo
- JMSType
- JMSXGroupID
- JMSXGroupSeq

Bu adlar JMS özellikleri için ayrılmıştır.

- aşağıdaki anahtar sözcüklerden biri, büyük ya da küçük harflerin herhangi bir karışımında yer almaz:
  - VE
  - Arasında
  - Escape
  - YANLIŞ
  - IN

- BUDUR
- Benzer
- DEĞİL
- BOŞ DEĞERLİ
- OR
- DOĞRU

- Vücuda başlamıyor. ya da Kök. ( Root.MQMDdışında).

Özellik MQ-defind (mq. \*) ise ve ad tanındı, özellik tanımlayıcı alanları, özellik için doğru değerlere ayarlanır. Özellik tanınmadıysa, özellik tanımlayıcısının *Support* alanı **MQPD\_OPTIONAL** olarak ayarlanır.

### **MQCMHO\_DEFAULT\_VALIDATION**

Bu değer, özellik adlarına ilişkin varsayılan geçerlilik denetimi düzeyinin ortaya çıkacağını belirtir.

Varsayılan geçerlilik denetimi düzeyi, **MQCMHO\_VALIDATE** ile belirtilen düzeyle eşdeğerdir.

Bu değer, varsayılan değerdir.

### **MQCMHO\_NO\_VALIDATION**

Özellik adı üzerinde doğrulama gerçekleştirilmez. **MQCMHO\_VALIDATE**' in açıklamasına bakın.

**Varsayılan seçenek:** Tanımlanan yukarıdaki seçeneklerden hiçbiri gerekmiyorsa, aşağıdaki seçenek kullanılabilir:

### **MQCMHO\_NONE**

Tüm seçenekler varsayılan değerlerini varsayar. Başka bir seçenek belirtilmemeyi belirtmek için bu değeri kullanın. **MQCMHO\_NONE** , program belgelerine yardımcı olur; bu seçeneğin başka biriyle birlikte kullanılması amaçlanmaz, ancak değeri sıfır olduğu için bu tür bir kullanım algılanamaz.

## **MQCNO-Bağlantı seçenekleri**

MQCNO yapısı, uygulamanın kuyruk yöneticisine yönelik bağlantıyla ilgili seçenekleri belirtmesine olanak sağlar. Yapı, MQCONNX çağrısındaki bir giriş/çıkış deęiştirgesidir.

Paylaşılan tanıtıcıları ve MQCONNX çağrısıyla ilgili daha fazla bilgi için bkz. [MQCONNX ile paylaşılan \(iş parçacığından bağımsız\) bağlantılar](#).

### **Kullanılabilirlik**

MQCNO\_VERSION\_4 dışındaki MQCNO yapısının tüm sürümleri aşağıdaki altyapılarda bulunur:

-  AIX
-  IBM i
-  Linux
-  Windows

ve bu sistemlere baęlı IBM MQ MQI clients için.

### **Sürüm**

Desteklenen programlama dilleri için sağlanan üstbilgi, COPY ve INCLUDE dosyaları, MQCNO ' nun en son sürümünü içerir, ancak *Version* alanının ilk değeri MQCNO\_VERSION\_1 olarak ayarlanmıştır. version-1 yapısında var olmayan alanları kullanmak için uygulamanın *Version* alanını gerekli sürüm numarasına ayarlaması gerekir.

## Karakter kümesi ve kodlama

MQCNO içindeki veriler, **CodedCharSetId** kuyruk yöneticisi özneliği tarafından belirtilen karakter kümesinde ve MQENC\_NATIVE tarafından verilen yerel kuyruk yöneticisinin kodlamasında yer almalıdır. Ancak, uygulama IBM MQ MQI clientolarak çalışıyorsa, yapı istemcinin karakter kümesinde ve kodlamasında olmalıdır.

## Alanlar

**Not:** Aşağıdaki tabloda, alanlar alfabetik olarak değil, kullanıma göre gruplanır. Alt konular aynı sırayı izler.

Çizelge 480. MQCNO içindeki alanlar		
Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
<u>StrucId</u> (yapı tanıtıcısı)	MQCNO_STRUC_ID	'CNO~'
<u>Sürüm</u> (yapı sürümü numarası)	MQCNO_VERSION_1	1
<u>Seçenekler</u> (MQCONNX işlemini denetleyen seçenekler)	MQCNO_NONE	0
<b>Not:</b> <i>Version</i> , MQCNO_VERSION_2' den küçükse kalan alanlar yoksayılr.		
<u>ClientConnGörelı Konum</u> (istemci bağlantısı için MQCD yapısının görelı konumu)	Yok	0
<u>ClientConnPtr</u> (istemci bağlantısı için MQCD yapısının adresi)	Yok	Boş değeri gösterge ya da boş değeri byte
<b>Not:</b> <i>Version</i> , MQCNO_VERSION_3' den küçükse kalan alanlar yoksayılr.		
<u>ConnTag</u> (kuyruk yöneticisi bağlantı etiketi)	MQCT_NONE	Boş Değerler
<b>Not:</b> <i>Version</i> , MQCNO_VERSION_4' den küçükse kalan alanlar yoksayılr.		
<u>SSLConfigPtr</u> (istemci bağlantısı için MQSCO yapısının adresi)	Yok	Boş değeri gösterge ya da boş değeri byte
<u>SSLConfigOffset</u> (istemci bağlantısı için MQSCO yapısının görelı konumu)	Yok	0
<b>Not:</b> <i>Version</i> , MQCNO_VERSION_5' den küçükse kalan alanlar yoksayılr.		
<u>ConnectionId</u> (benzersiz bağlantı tanıtıcısı)	Yok	Boş değeri gösterge ya da boş değeri byte
<u>SecurityParmsGörelı Konumu</u> (güvenlik değıştirgeleri için MQSCO yapısının görelı konumu)	Yok	Boş değeri gösterge ya da boş değeri byte
<u>SecurityParmsPtr</u> (güvenlik değıştirgeleri için MQSCO yapısının adresi)	Yok	Boş değeri gösterge ya da boş değeri byte
<b>Not:</b> <i>Version</i> , MQCNO_VERSION_6' dan küçükse kalan alanlar yoksayılr.		
<u>Ayrılmış</u> (ayrılmış alan)	Yok	Yapıyı 64 bitlik bir sınıra çıkarmak için ayrılmış alan.

Çizelge 480. MQCNO içindeki alanlar (devamı var)

Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
<u>CCDTUrlLength</u> (CCDT URL uzunluğu)	Yok	<u>CCDTUrlPtr</u> ya da <u>CCDTUrlOffset</u> ile tanımlanan dizginin uzunluğu
<u>CCDTUrlPtr</u> (CCDT URL göstergesi)	Yok	Bağlantı için kullanılacak istemci bağlantı kanalı çizelgesinin yerini saptamak için, URL içeren bir dizgiyi gösteren gösterge.
<u>CCDTUrlOffset</u> (CCDT URL görelî konumu)	Yok	Bağlantı için kullanılacak istemci bağlantı kanalı çizelgesinin konumunu tanımlayan bir URL ' yi içeren bir dizgiden bayt cinsinden görelî konum.
<p>► V9.2.0 ► V9.2.0</p> <p><b>Not:</b> <i>Version</i> , MQCNO_VERSION_7' den küçükse kalan alanlar yoksayılr.</p>		
► V9.2.0 <u>AppName</u> (uygulama tarafından ayarlanan ad)	Yok	Kuyruk yöneticisine yönelik bağlantıyı tanımlamak için uygulama tarafından ayarlanan ad
► V9.2.0 <u>Reserved2</u> (ayrılmış alan)	Yok	Yapıyı 64 bitlik bir sınıra çıkarmak için ayrılmış alan.
<p>► V9.2.4 ► V9.2.4</p> <p><b>Not:</b> <i>Version</i> , MQCNO_VERSION_8' den küçükse kalan alanlar yoksayılr.</p>		
► V9.2.4 <u>BalanceParms</u>   Görelî Konum	Yok	MQBNO yapısına byte olarak görelî konum
► V9.2.4 <u>BalanceParmsPtr</u>	Yok	MQBNO yapısının yerini gösteren gösterge
<p><b>Notlar:</b></p> <ol style="list-style-type: none"> <li>1. ~ simgesi, tek bir boş karakteri temsil eder.</li> <li>2. C programlama dilinde, makro değişkeniMQCNO_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlar için başlangıç değerleri sağlamak üzere bunu aşağıdaki şekilde kullanın:</li> </ol> <pre>MQCNO MyCNO = {MQCNO_DEFAULT};</pre>		



## Dil bildirimleri

### MQCNO için C bildirimi

#### V 9.2.0

```
typedef struct tagMQCNO MQCNO;
struct tagMQCNO {
    MQCHAR4    StructId;           /* Structure identifier */
    MQLONG     Version;           /* Structure version number */
    MQLONG     Options;           /* Options that control the action of
    MQCONNX */

    MQLONG     ClientConnOffset; /* Offset of MQCD structure for client
    connection */
    MQPTR      ClientConnPtr;     /* Address of MQCD structure for client
    connection */
    MQBYTE128  ConnTag;           /* Queue manager connection tag */
    PMQSCO     SSLConfigPtr;      /* Address of MQSCO structure for client
    connection */
    MQLONG     SSLConfigOffset;   /* Offset of MQSCO structure for client
    connection */
    MQBYTE24   ConnectionId;      /* Unique connection identifier */
    MQLONG     SecurityParmsOffset /* Security fields */
    PMQCSP     SecurityParmsPtr   /* Security parameters */
    MQLONG     CCDTUrlLength      /* Length of string identified by Ptr or offset */
    MQLONG     CCDTUrlOffset      /* Offset in bytes to URL of client connection channel */
    PMQURL     CCDTUrlPtr         /* Address to string containing URL */
    MQBYTE4    Reserved           /* Reserved field to pad out to 64 bit boundary */
    MQCHAR28   ApplName           /* Name set by the application to identify the connection to
    the queue manager */
    MQBYTE4    Reserved2         /* Reserved field to pad out to 64 bit boundary */
};
```

#### V 9.2.4

```
typedef struct tagMQCNO MQCNO;
struct tagMQCNO {
    MQCHAR4    StructId;           /* Structure identifier */
    MQLONG     Version;           /* Structure version number */
    MQLONG     Options;           /* Options that control the action of
    MQCONNX */

    MQLONG     ClientConnOffset; /* Offset of MQCD structure for client
    connection */
    MQPTR      ClientConnPtr;     /* Address of MQCD structure for client
    connection */
    MQBYTE128  ConnTag;           /* Queue manager connection tag */
    PMQSCO     SSLConfigPtr;      /* Address of MQSCO structure for client
    connection */
    MQLONG     SSLConfigOffset;   /* Offset of MQSCO structure for client
    connection */
    MQBYTE24   ConnectionId;      /* Unique connection identifier */
    MQLONG     SecurityParmsOffset /* Security fields */
    PMQCSP     SecurityParmsPtr   /* Security parameters */
    MQLONG     CCDTUrlLength      /* Length of string identified by Ptr or offset */
    MQLONG     CCDTUrlOffset      /* Offset in bytes to URL of client connection channel */
    PMQURL     CCDTUrlPtr         /* Address of string containing URL */
    MQBYTE4    Reserved           /* Reserved field to pad out to 64 bit boundary */
    MQCHAR28   ApplName           /* Name set by the application to identify the connection to
    the queue manager */
    MQBYTE4    Reserved2         /* Reserved field to pad out to 64 bit boundary */
    MQLONG     BalanceParmsOffset /* Offset of the MQBMO structure */
    PMQBMO     BalanceParmsPtr    /* Address of the location of the MQBMO structure */
};
```

### MQCNO için COBOL bildirimi

#### V 9.2.0

```
** MQCNO structure
10 MQCNO.
** Structure identifier
15 MQCNO-STRUCID PIC X(4).
** Structure version number
15 MQCNO-VERSION PIC S9(9) BINARY.
** Options that control the action of MQCONNX
15 MQCNO-OPTIONS PIC S9(9) BINARY.
** Offset of MQCD structure for client connection
15 MQCNO-CLIENTCONNOFFSET PIC S9(9) BINARY.
** Address of MQCD structure for client connection
```

```

15 MQCNO-CLIENTCONNPTR    POINTER.
** Queue manager connection tag
15 MQCNO-CONNTAG          PIC X(128).
** Address of MQSCO structure for client connection
15 MQCNO-SSLCONFIGPTR     POINTER.
** Offset of MQSCO structure for client connection
15 MQCNO-SSLCONFIGOFFSET  PIC S9(9) BINARY.
** Unique connection identifier
15 MQCNO-CONNECTIONID     PIC X(24).
** Offset of MQCSP structure for security parameters
15 MQCNO-SECURITYPARMSOFFSET PIC S9(9) BINARY.
** Address of MQCSP structure for security parameters
15 MQCNO-SECURITYPARMSPTR POINTER.
** Length of string identified by CCDTURLPtr or CCDTURLOffset
15 MQCNO-CCDTURLLENGTH
** Pointer to a string which contains a URL, to identify the location of the client
connection channel
15 MQCNO-CCDTURLPTR
** Offset in bytes from a string which contains a URL that identifies the location of the
client connection channel table
15 MQCNO-CCDTURLOFFSET
** Reserved field to pad to 64 bit boundary
15 MQCNO-RESERVED
** Name set by the application to identify the connection to the queue manager
15 MQCNO-APPLNAME
** Reserved field to pad to 64 bit boundary
15 MQCNO-RESERVED2

```

#### V 9.2.4

```

** MQCNO structure
10 MQCNO.
** Structure identifier
15 MQCNO-STRUCID          PIC X(4).
** Structure version number
15 MQCNO-VERSION          PIC S9(9) BINARY.
** Options that control the action of MQCONN
15 MQCNO-OPTIONS          PIC S9(9) BINARY.
** Offset of MQCD structure for client connection
15 MQCNO-CLIENTCONNOFFSET PIC S9(9) BINARY.
** Address of MQCD structure for client connection
15 MQCNO-CLIENTCONNPTR   POINTER.
** Queue manager connection tag
15 MQCNO-CONNTAG          PIC X(128).
** Address of MQSCO structure for client connection
15 MQCNO-SSLCONFIGPTR     POINTER.
** Offset of MQSCO structure for client connection
15 MQCNO-SSLCONFIGOFFSET  PIC S9(9) BINARY.
** Unique connection identifier
15 MQCNO-CONNECTIONID     PIC X(24).
** Offset of MQCSP structure for security parameters
15 MQCNO-SECURITYPARMSOFFSET PIC S9(9) BINARY.
** Address of MQCSP structure for security parameters
15 MQCNO-SECURITYPARMSPTR POINTER.
** Length of string identified by CCDTURLOFFSET or CCDTURLPTR
15 MQCNO-CCDTURLLENGTH
** Pointer to a string which contains a URL, to identify the location of the client
connection channel
15 MQCNO-CCDTURLPTR
** Address of string which contains a URL that identifies the location of the client
connection channel table
15 MQCNO-CCDTURLOFFSET
** Reserved field to pad to 64 bit boundary
15 MQCNO-RESERVED
** Name set by the application to identify the connection to the queue manager
15 MQCNO-APPLNAME
** Reserved field to pad to 64 bit boundary
15 MQCNO-RESERVED2
** Address of the MQBMO structure
15 MQCNO-BALANCEPARMSOFFSET
** Pointer to the MQBMO structure
15 MQCNO-BALANCEPARMSPTR

```

MQCNO için PL/I bildirimi

#### V 9.2.0

```

dcl
1 MQCNO based,
3 StrucId          char(4),          /* Structure identifier */

```

```

3 Version          fixed bin(31), /* Structure version number */
3 Options          fixed bin(31), /* Options that control the action
of MQCONN * */
3 ClientConnOffset fixed bin(31), /* Offset of MQCD structure for
client connection * */
3 ClientConnPtr   pointer, /* Address of MQCD structure for
client connection * */
3 ConnTag         char(128), /* Queue manager connection tag * */
3 SSLConfigPtr    pointer, /* Address of MQSCO structure for
client connection * */
3 SSLConfigOffset fixed bin(31), /* Offset of MQSCO structure for
client connection * */
3 ConnectionId    char(24), /* Unique connection identifier
3 SecurityParmsOffset fixed bin(31); /* Offset of MQCSP structure for
security parameters * */
3 SecurityParmsPtr pointer, /* Address of MQCSP structure for
security parameters * */
3 CCDTUrlLength   fixed bin(31) /* Length of string identified by CCDTUrlPtr
or CCDTUrlOffset * */
3 CCDTUrlOffset   fixed bin(31) /* Offset in bytes to URL of client connection channel * */
3 CCDTUrlPtr      pointer /* Pointer to string containing URL * */
3 Reserved        char(4) /* Reserved field to pad out to 64 bit boundary * */
3 ApplName        char(28) /* Name set by the application to identify the
connection to
the queue manager * */
3 Reserved2       char(4) /* Reserved field to pad out to 64 bit boundary
*/

```

#### V 9.2.4

```

dcl
1 MQCNO based,
3 StrucId          char(4), /* Structure identifier * */
3 Version          fixed bin(31), /* Structure version number * */
3 Options          fixed bin(31), /* Options that control the action
of MQCONN * */
3 ClientConnOffset fixed bin(31), /* Offset of MQCD structure for
client connection * */
3 ClientConnPtr   pointer, /* Address of MQCD structure for
client connection * */
3 ConnTag         char(128), /* Queue manager connection tag * */
3 SSLConfigPtr    pointer, /* Address of MQSCO structure for
client connection * */
3 SSLConfigOffset fixed bin(31), /* Offset of MQSCO structure for
client connection * */
3 ConnectionId    char(24), /* Unique connection identifier
3 SecurityParmsOffset fixed bin(31) /* Offset of MQCSP structure for
security parameters * */
3 SecurityParmsPtr pointer, /* Address of MQCSP structure for
security parameters * */
3 CCDTUrlLength   fixed bin(31) /* Length of string identified by CCDTUrlPtr
or CCDTUrlOffset * */
3 CCDTUrlOffset   fixed bin(31) /* Offset in bytes to URL of client connection channel * */
3 CCDTUrlPtr      pointer /* Pointer to string containing URL * */
3 Reserved        char(4) /* Reserved field to pad out to 64 bit boundary * */
3 ApplName        char(28) /* Name set by the application to identify the
connection to
the queue manager * */
3 Reserved2       char(4) /* Reserved field to pad out to 64 bit boundary * */
3 BalanceParmsOffset fixed bin(31) /* Offset of the MQBMO structure * */
3 BalanceParmsPtr pointer /* Address of the MQBMO structure * */

```

### MQCNO için High Level Assembler (Üst Düzey Çevirici) bildirimi

#### V 9.2.0

```

MQCNO          DSECT
MQCNO_STRUCID DS CL4  Structure identifier
MQCNO_VERSION DS F    Structure version number
MQCNO_OPTIONS DS F    Options that control the action of
*                MQCONN
MQCNO_CLIENTCONNOFFSET DS F    Offset of MQCD structure for client
*                connection
MQCNO_CLIENTCONNPTR DS F    Address of MQCD structure for client
*                connection
MQCNO_CONNTAG   DS XL128 Queue manager connection tag
*
MQCNO_CONNECTIONID DS XL24 Unique connection identifier

```

* MQCNO_SSLCONFIGOFFSET	DS	F	Offset of MQCSP structure for security parameters
* MQCNO_SSLCONFIGPTR	DS	F	Address of MQCSP structure for security parameters
* MQCNO_LENGTH	EQU	*-MQCNO	
	ORG	MQCNO	
MQCNO_AREA	DS	CL(MQCNO_LENGTH)	
MQCNO_CCDTURLLENGTH	DS	F	Length of string identified by CCDURLPTR or CCDTURLOFFSET
* MQCNO_CCDTURLOFFSET	DS	F	Offset in bytes to URL of client connection channel
MQCNO_CCDTURLPTR	DS	F	Pointer to string containing URL
RESERVED	DS	XL4	Reserved field to pad out to 64 bit boundary
APPLNAME	DS	CL28	Name set by the application to identify the connection to the queue manager
* RESERVED2	DS	XL4	Reserved field to pad out to 64 bit boundary

### V 9.2.4

MQCNO	DSECT		
MQCNO_STRUCID	DS	CL4	Structure identifier
MQCNO_VERSION	DS	F	Structure version number
MQCNO_OPTIONS	DS	F	Options that control the action of MQCONN
* MQCNO_CLIENTCONNOFFSET	DS	F	Offset of MQCD structure for client connection
* MQCNO_CLIENTCONNPTR	DS	F	Address of MQCD structure for client connection
* MQCNO_CONNTAG	DS	XL128	Queue manager connection tag
MQCNO_CONNECTIONID	DS	XL24	Unique connection identifier
MQCNO_SSLCONFIGOFFSET	DS	F	Offset of MQCSP structure for security parameters
* MQCNO_SSLCONFIGPTR	DS	F	Address of MQCSP structure for security parameters
* MQCNO_LENGTH	EQU	*-MQCNO	
	ORG	MQCNO	
MQCNO_AREA	DS	CL(MQCNO_LENGTH)	
MQCNO_CCDTURLLENGTH	DS	F	Length of string identified by CCDURLPTR or CCDTURLOFFSET
* MQCNO_CCDTURLOFFSET	DS	F	Offset in bytes to URL of client connection channel
MQCNO_CCDTURLPTR	DS	F	Pointer to string containing URL
RESERVED	DS	XL4	Reserved field to pad out to 64 bit boundary
APPLNAME	DS	CL28	Name set by the application to identify the connection to the queue manager
* RESERVED2	DS	XL4	Reserved field to pad out to 64 bit boundary
MQCNO_BALANCEPARMSOFFSET	DS	F	Offset of the MQBMO structure
MQCNO_BALANCEPARMSPTR	DS	F	Address of the MQBMO structure

### MQCNO için Visual Basic bildirimi

### V 9.2.0

Type MQCNO			
StrucId	As String*4		'Structure identifier'
Version	As Long		'Structure version number'
Options	As Long		'Options that control the action of 'MQCONNX'
ClientConnOffset	As Long		'Offset of MQCD structure for client 'connection'
ClientConnPtr	As MQPTR		'Address of MQCD structure for client 'connection'
ConnTag	As MQBYTE128		'Queue manager connection tag'
SSLConfigPtr	As MQPTR		'Address of MQSCO structure for client 'connection'
SSLConfigOffset	As Long		'Offset of MQSCO structure for client 'connection'
ConnectionId	As MQBYTE24		'Unique connection identifier'
SecurityParmsOffset	As Long		'Offset of MQCSP structure for security 'parameters'
SecurityParmsPtr	As MQPTR		'Address of MQCSP structure for security 'parameters'
CCDTUrlLength	As Long		'Length of string identified by CCDUrlPtr 'or CCDTurloffset'
CCDTUrlOffset	As Long		'Offset in bytes to URL of client connection channel'
CCDTUrlPtr	As MQPTR		'Pointer to string containing URL'
Reserved	As MQBYTE4		'Reserved field to pad out to 64 bit boundary'
ApplName	As String*28		'Name set by the application to identify the connection to 'the queue manager'

```
Reserved2      As MQBYTE4 'Reserved field to pad out to 64 bit boundary'
End Type
```

## V9.2.4

```
Type MQCNO
StrucId        As String*4 'Structure identifier'
Version        As Long     'Structure version number'
Options        As Long     'Options that control the action of'
                'MQCONN'
ClientConnOffset As Long   'Offset of MQCD structure for client'
                'connection'
ClientConnPtr   As MQPTR   'Address of MQCD structure for client'
                'connection'
ConnTag         As MQBYTE128 'Queue manager connection tag'
SSLConfigPtr    As MQPTR   'Address of MQSCO structure for client'
                'connection'
SSLConfigOffset As Long   'Offset of MQSCO structure for client'
                'connection'
ConnectionId    As MQBYTE24 'Unique connection identifier'
SecurityParmsOffset As Long 'Offset of MQCSP structure for security'
                'parameters'
SecurityParmsPtr As MQPTR   'Address of MQCSP structure for security'
                'parameters'
CCDTUrlLength   As Long   'Length of string identified by CCDTUrlPtr'
                'or CCDTUrlOffset'
CCDTUrlOffset   As Long   'Offset in bytes to URL of client connection channel'
CCDTUrlPtr      As MQPTR   'Pointer to string containing URL'
Reserved        As MQBYTE4 'Reserved field to pad out to 64 bit boundary'
ApplName        As String*28 'Name set by the application to identify the connection to'
                'the queue manager'
Reserved2       As MQBYTE4 'Reserved field to pad out to 64 bit boundary'
BalanceParmsOffset As Long 'Offset in bytes to MQBNO structure'
BalanceParmsPtr  As MQPTR   'Address of MQBNO structure'
End Type
```

### İlgili görevler

[MQCONN kullanılıyor](#)

### StrucId (MQCHAR4)

StrucId her zaman bir giriş alanıdır. Başlangıç değeri MQCNO\_STRUC\_ID 'dir.

Değer şu olmalıdır:

#### MQCNO\_STRUC\_ID

Bağlantı seçenekleri yapısına ilişkin tanıttıcı.

C programlama dili için, sabit MQCNO\_STRUC\_ID\_ARRAY değişmezi de tanımlanır; bu değişmez, MQCNO\_STRUC\_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisidir.

### Sürüm (MQLONG)

Sürüm her zaman bir giriş alanıdır. İlk değeri MQCNO\_VERSION\_1 'dir.

Değer aşağıdakilerden biri olmalıdır:

#### MQCNO\_VERSION\_1

Version-1 bağlantı seçenekleri yapısı.

#### MQCNO\_VERSION\_2

Version-2 bağlantı seçenekleri yapısı.

#### MQCNO\_VERSION\_3

Version-3 bağlantı seçenekleri yapısı.

#### MQCNO\_VERSION\_4

Version-4 bağlantı seçenekleri yapısı.

#### MQCNO\_VERSION\_5

Version-5 bağlantı seçenekleri yapısı.

#### MQCNO\_VERSION\_6

Version-6 bağlantı seçenekleri yapısı.

## **MQCNO\_VERSION\_7**

Version-7 bağlantı seçenekleri yapısı.

V 9.2.4

## **MQCNO\_VERSION\_8**

Version-8 bağlantı seçenekleri yapısı.

Yalnızca yapının daha yeni sürümlerinde bulunan alanlar, alanların tanımlarında bu şekilde tanımlanır. Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

## **MQCNO\_CURRENT\_VERSION**

Bağlantı seçenekleri yapısının yürürlükteki sürümü.

## **Seçenekler (MQUZE)**

MQCONN 'in işlemini denetleyen seçenekler.

## **Hesap seçenekleri**

Aşağıdaki seçenekler, **AccountingConnOverride** kuyruk yöneticisi özniteliği MQMON\_ENABLED değerine ayarlıysa, muhasebe tipini denetler:

### **MQCNO\_ACCOUNTING\_MQI\_ENABLED**

When monitoring data collection is disabled in the queue manager definition by setting the **MQIAccounting** attribute to MQMON\_OFF, setting this flag enables MQI accounting data collection.

### **MQCNO\_ACCOUNTING\_MQI\_DISABLLANT**

When monitoring data collection is disabled in the queue manager definition by setting the **MQIAccounting** attribute to MQMON\_OFF, setting this flag stops MQI accounting data collection.

### **MQCNO\_ACCOUNTING\_Q\_ENABLED**

When queue-accounting data collection is disabled in the queue manager definition by setting the **MQIAccounting** attribute to MQMON\_OFF, setting this flag enables accounting data collection for those queues that specify a queue manager in the *MQIAccounting* field of their queue definition.

### **MQCNO\_ACCOUNTING\_Q\_DISABET**

When queue-accounting data collection is disabled in the queue manager definition by setting the **MQIAccounting** attribute to MQMON\_OFF, setting this flag switches off accounting data collection for those queues that specify a queue manager in the *MQIAccounting* field of their queue definition.

Bu işaretlerin hiçbiri tanımlanmazsa, bağlantının muhasebesi, kuyruk yöneticisi özniteliklerinde tanımlandığı gibidir.

## **Bağlama seçenekleri**

Aşağıdaki seçenekler, kullanılacak IBM MQ bağ tanımı tipini denetler. Bu seçeneklerden yalnızca birini belirtin:

### **MQCNO\_STANDARD\_BINDING**

Uygulama ve yerel kuyruk yöneticisi aracı (kuyruğa alma işlemlerini yöneten bileşen) ayrı yürütme birimlerinde çalıştırılıyor (genellikle, ayrı süreçlerde). Bu düzenleme kuyruk yöneticisinin bütünlüğünü korur; yani, kuyruk yöneticisini errant programlarından korur.

Kuyruk yöneticisi birden çok bağ tanımlama tipini destekliyse ve MQCNO\_STANDARD\_BINBding değerini belirlerseniz, kuyruk yöneticisi gerçek bağ tanımı tipini seçmek için qm.ini dosyasındaki Connection ' ta **DefaultBindType** özniteliğini kullanır. Bu stanza tanımlanmadıysa ya da değer kullanılamıyorsa ya da uygulama için uygun değilse, kuyruk yöneticisi uygun bir bağ tanımı tipi seçer. Kuyruk yöneticisi, bağlanma seçeneklerinde kullanılan gerçek bağ tanımlama tipini ayarlar.

Uygulamanın tam sınanmamış olabileceği ya da güvenilir olmayan ya da güvenilmez olduğu durumlarda MQCNO\_STANDARD\_BINDING seçeneğini kullanın. MQCNO\_STANDARD\_BINDING varsayılan değerdir.

Bu seçenek tüm ortamlarda desteklenir.

mqm kitaplığına bağlantı oluşturuyorsanız, önce varsayılan bağ tanımlama tipini kullanan bir standart sunucu bağlantısı girişiminde bulunulması gerekir. Temeldeki sunucu kitaplığı yüklenemediyse, bunun yerine bir istemci bağlantısı girişiminde bulunulması gerekir.

- MQCONN (ya da MQCONNX 'in MQCNO\_STANDARD\_BINDING belirtildiyse MQCONNX) davranışını değiştirmek için, MQ\_CONNECT\_TYPE ortam değişkenini aşağıdaki seçeneklerden birine ayarlayın. Bunun için bir kural dışı durum olduğunu unutmayın: MQ\_CONNECT\_TYPE değeri YEREL ya da STANDART olarak belirlenirse, fastpath bağlantıları, uygulama için ilgili bir değişiklik yapılmaksızın yönetici tarafından düşürülebilir.

Çizelge 481. MQCONN ya da MQCONNX 'in işleyişini değiştiren MQ_CONNECT_TYPE değerleri	
Değer	Anlamı
CLIENT	Yalnızca istemci bağlantısı deniyor.
FastPath	Bu değer önceki yayınlarda desteklendi, ancak belirtilirse yoksayılır.
LOCAL	Yalnızca sunucu bağlantısı denendi. Fastpath bağlantıları, standart bir sunucu bağlantısına düşürülebilir.
Standart	Önceki yayın düzeyleriyle uyumluluk için desteklenir. Bu değer şimdi LOCAL olarak ele alınır.

- MQCONNX çağrıldığında MQ\_CONNECT\_TYPE ortam değişkeni ayarlanmazsa, varsayılan bağ tanımlama tipini kullanan bir standart sunucu bağlantısı girişiminde bulunulması denir. Sunucu kitaplığı yüklenemezse, bir istemci bağlantısı girişiminde bulunmaya çalışılır.




### MQCNO\_FASTPATH\_BINDING

Uygulama ve yerel kuyruk yöneticisi aracı aynı yürütme biriminin bir parçasıdır. Bu, uygulamanın ve yerel kuyruk yöneticisi aracısının ayrı yürütme birimlerinde çalıştırıldığı tipik bağ tanımlama yöntemine karşılık olarak gelir.

Kuyruk yöneticisi bu bağ tanımlama tipini desteklemiyorsa, MQCNO\_FASTPATH\_BINDING yoksayılır; işlem, seçenek belirlenmemiş gibi devam eder.

MQCNO\_FASTPATH\_BINDING, birden çok işlemin uygulama tarafından kullanılan genel kaynaktan daha fazla kaynak tükettiği durumlarda avantaj olabilir. Fastpath bağ tanımını kullanan bir uygulama, *güvenilir uygulama* olarak bilinir.

Fastpath bağ tanımını kullanıp kullanmayacağına karar verirken aşağıdaki önemli noktaları göz önünde bulundurun:

- MQCNO\_FASTPATH\_BINDING seçeneğinin kullanılması, bir uygulamanın kuyruk yöneticisine ait iletileri ve diğer veri alanlarını değiştirmesini ya da değiştirmesini engellemektedir. Bu seçeneği yalnızca, bu sorunları tam olarak değerlendirdiğiniz durumlarda kullanın.
- Uygulama, MQCNO\_FASTPATH\_BINDING ile zamanuyumsuz sinyaller ya da süreölçer kesintileri ( sigkill gibi) kullanmamalıdır. Paylaşılan bellek kesimlerinin kullanımına ilişkin kısıtlamalar da vardır.
- Uygulama, kuyruk yöneticisinden bağlantıyı kesmek için MQDISC çağrısını kullanmalıdır.
- Kuyruk yöneticisini endmqm komutuyla sona erdirmeden önce uygulamanın bitmesi gerekir.
-  On IBM i, the job must run under a user profile that belongs to the QMQMADM group. Ayrıca, programın olağan dışı bir şekilde durmaması gerekir; tersi durumda, beklenmedik sonuçlar ortaya çıkabilir.
-   AIX and Linux üzerinde, mqm kullanıcı kimliği etkin kullanıcı kimliği olmalı ve mqm grup tanıtıcısı etkin grup tanıtıcısı olmalıdır. To make the application run in this way, configure the program so that it is owned by the mqm user identifier and mqm group identifier, and then set the setuid and setgid permission bits on the program.



The IBM MQ Object Authority Manager (OAM) still uses the real user ID for authority checking.

- **Windows** Windows üzerinde program, mqm grubunun bir üyesi olmalıdır. Fastpath bağ tanımı 64 bit uygulamaları için desteklenmez.

Aşağıdaki ortamlarda MQCNO\_FASTPATH\_BINDING seçeneği desteklenmektedir:

- **AIX** AIX
- **IBM i** IBM i
- **Linux** Linux
- **Windows** Windows

• **z/OS** z/OS üzerinde, seçenek kabul edilir, ancak yoksayılr.

Güvenilir uygulamaların kullanılmasıyla ilgili daha fazla bilgi için bkz. [Güvenilen uygulamalara ilişkin kısıtlamalar](#).

### MQCNO\_SHARED\_BINDING

MQCNO\_SHARED\_BINDING ile, uygulama ve yerel kuyruk yöneticisi aracısı bazı kaynakları paylaşır. Kuyruk yöneticisi bu bağ tanımı tipini desteklemiyorsa, MQCNO\_SHARED\_BINDING yoksayılr. Bu seçenek belirlenmemiş gibi işleme devam eder.

### MQCNO\_ISOLATED\_BINDING

Bu durumda, uygulama işlemi ve yerel kuyruk yöneticisi aracısı, kaynakları paylaşmadıkları için birbirlerinden yalıtılır. Kuyruk yöneticisi bu bağ tanımlama tipini desteklemiyorsa, MQCNO\_ISOLATED\_BINDING yoksayılr. Bu seçenek belirlenmemiş gibi işleme devam eder.

### MQCNO\_CLIENT\_BINDING

Uygulamanın yalnızca istemci bağlantısı denemesini sağlamak için bu seçeneği belirleyin. Bu seçenek aşağıdaki sınırlamalara sahiptir:

- **z/OS** MQCNO\_CLIENT\_BINDING, z/OS üzerinde yoksayıldı.
- MQCNO\_CLIENT\_BINDING, MQCNO\_STANDARD\_BINDING dışında herhangi bir MQCNO bağ tanımı seçeneğiyle belirtildiye, MQRC\_OPTIONS\_ERROR ile reddedildi.
- MQCNO\_CLIENT\_BINDING, bağ tanımlama tipini seçmek için kendi yöntemlerine sahip oldukları için Java ya da .NET için kullanılamaz.

### MQCNO\_LOCAL\_BINDING

Uygulamanın bir sunucu bağlantısını denemesini sağlamak için bu seçeneği belirleyin. MQCNO\_FASTPATH\_BINDING, MQCNO\_ISOLATED\_BINDING ya da MQCNO\_SHARED\_BINDING değeri de belirtilirse, bağlantı bu tipte olur ve bu bölümde belgelenir. Ters durumda, varsayılan bağ tanımlama tipi kullanılarak standart bir sunucu bağlantısı girişiminde bulunmaya çalışılır. MQCNO\_LOCAL\_BINDING, aşağıdaki sınırlamalara sahiptir:

- **z/OS** MQCNO\_LOCAL\_BINDING, z/OS üzerinde yoksayıldı.
- MQCNO\_RECONNECT\_AS\_DEF dışında herhangi bir MQCNO yeniden bağlanma seçeneğiyle belirtilirse, MQCNO\_LOCAL\_BINDING, MQRC\_OPTIONS\_ERROR ile reddedilir.
- MQCNO\_LOCAL\_BINDING, bağ tanımlama tipini seçmek için kendi yöntemlerine sahip oldukları için Java ya da .NET için kullanılamaz.

On the following platforms, you can use the environment variable MQ\_CONNECT\_TYPE with the bind type specified by the Options field, to control the type of binding used.

- **AIX** AIX



- **Linux** Linux
- **Windows** Windows

If you specify this environment variable, it must have the value FASTPATH or STANDARD ; if it has a different value, it is ignored. Ortam değişkeninin değeri büyük ve küçük harfe duyarlıdır; ek bilgi için [MQCONN environment variable](#) konusuna bakın.

Ortam değişkeni ve *Options* alanı aşağıdaki şekilde etkileşimde bulunur:

- Ortam değişkenini atlarsanız ya da desteklenmeyen bir değer verdiyseniz, fastpath bağ tanımının kullanımı yalnızca *Options* alanı tarafından belirlenir.
- Ortam değişkenine desteklenen bir değer verirdeniz, fastpath bağ tanımı yalnızca hem ortam değişkeni, hem de *Options* alanı, fastpath bağ tanımını belirtirse kullanılır.

## Bağlantı biçim imi seçenekleri

**LTS** These options are supported only when connecting to a z/OS queue manager and they control the use of the connection tag ConnTag. Bu seçeneklerden yalnızca birini belirleyebilirsiniz.

**V 9.2.0** Bağlantı etiketlerinin kesin olarak somutlaması, IBM MQ for z/OS ile IBM MQ for Multiplatforms arasında farklılık gösterir:

- **z/OS** Aşağıdaki seçenekler, *MQCNO\_GENERATE\_CONN\_TAG* dışında, yalnızca bir z/OS kuyruk yöneticisine bağlanırken desteklenir ve bağlantı etiketinin kullanımını denetlemektedir. Desteklenen seçeneklerden yalnızca birini belirleyebilirsiniz.
- **ALW** *MQCNO\_GENERATE\_CONN\_TAG* , yalnızca z/OS dışındaki altyapılarda desteklenir.

**V 9.2.0** **ALW** **MQCNO\_GENERATE\_CONN\_TAG**

Kuyruk yöneticisinin bu bağlantıyla ilişkilendirdiği bağlantı etiketini, çıkış MQCNO yapısıyla döndürür. Döndürülen bağlantı etiketi, kuyruk yöneticisinin tek bir Uygulama Eşgörünümü olarak kabul ettiği tüm bağlantılar için aynı olur.

**z/OS** **MQCNO\_SERIALIZE\_CONN\_TAG\_Q\_MGR**

Bu seçenek, yerel kuyruk yöneticisinden bağlantı etiketinin dışlayıcı kullanımını ister. Bağlantı etiketi yerel kuyruk yöneticisinde zaten kullanıldıysa, MQCONNX çağrısı başarısız olur; neden kodu: MQRC\_CONN\_TAG\_IN\_USE. Çağrımın sonucu, yerel kuyruk yöneticisinin ait olduğu kuyruk paylaşım grubunun başka bir yerindeki bağlantı etiketi kullanılarak etkilenmez.

**z/OS** **MQCNO\_SERIALIZE\_CONN\_TAG\_QSG**

Bu seçenek, yerel kuyruk yöneticisinin ait olduğu kuyruk paylaşım grubu içindeki bağlantı etiketinin dışlayıcı kullanımını ister. Bağlantı etiketi kuyruk paylaşım grubunda zaten kullanılmışsa, MQCONNX çağrısı, MQRC\_CONN\_TAG\_IN\_USE neden kodlarıyla başarısız olur.

**z/OS** **MQCNO\_RESTRICT\_CONN\_TAG\_Q\_MGR**

Bu seçenek, yerel kuyruk yöneticisinden bağlantı etiketinin paylaşımlı kullanımını ister. Bağlantı etiketi yerel kuyruk yöneticisinde zaten kullanıldıysa, istekte bulunan uygulama, etiketin var olan kullanıcıyla aynı işlem kapsamında çalışıyorsa, MQCONNX çağrısı başarılı olabilir. Bu koşul karşılanmazsa, MQCONNX çağrısı başarısız olur; neden kodu MQRC\_CONN\_TAG\_IN\_USE. Çağrımın sonucu, yerel kuyruk yöneticisinin ait olduğu kuyruk paylaşım grubunun başka bir yerindeki bağlantı etiketinin kullanılmasından etkilenmez.

- Uygulamalar, bağlantı etiketini paylaşmak için aynı MVS adres alanı içinde çalışmalıdır. Bağlantı etiketini kullanan uygulama bir istemci uygulamasıysa, MQCNO\_RESTRICT\_CONN\_TAG\_Q\_MGR ' ye izin verilmez.

Bu seçenek, yerel kuyruk yöneticisinin ait olduğu kuyruk paylaşım grubu içindeki bağlantı etiketinin paylaşımli olarak kullanılmasını ister. Bağlantı etiketi kuyruk paylaşım grubunda zaten kullanılmışsa, istekte bulunan uygulamanın aynı işleme kapsamında çalıştığı ve etiketin var olan kullanıcıyla aynı kuyruk yöneticisine bağlı olması koşuluyla, MQCONNX çağrısı başarılı olabilir.

Bu koşullar karşılanmazsa, MQCONNX çağrısı başarısız olur; neden kodu MQRC\_CONN\_TAG\_IN\_USE.

- Uygulamalar, bağlantı etiketini paylaşmak için aynı MVS adres alanı içinde çalışmalıdır. Bağlantı etiketini kullanan uygulama bir istemci uygulamasıysa, MQCNO\_RESTRICT\_CONN\_TAG\_QSG izin verilmez.

Bu seçeneklerden hiçbiri belirtilmediyse, ConnTag kullanılmaz. Version , MQCNO\_VERSION\_3 değerinden küçükse, bu seçenekler geçerli değildir.

## Tanıtıcı paylaşım seçenekleri

### Multi

Bu seçenekler aşağıdaki ortamlarda desteklenir:

- **AIX** AIX
- **IBM i** IBM i
- **Linux** Linux
- **Windows** Windows

Bunlar aynı süreç içinde farklı iş parçacıkları (koşut işleme birimleri) arasındaki tutamaçların paylaşımını denetler. Bu seçeneklerden yalnızca birini belirleyebilirsiniz:

### MQCNO\_HANDLE\_SHARE\_NONE

Bu seçenek, bağlantı ve nesne tutamaçlarının yalnızca, ayrılmasına neden olan iş parçacığı tarafından (yani, MQCONN, MQCONNX ya da MQCONNX çağrısını veren iş parçacığıysa) kullanılabilceğini gösterir. Tutamaçlar, aynı işleme ait diğer iş parçacıkları tarafından kullanılamaz.

### MQCNO\_HANDLE\_SHARE\_BLOCK

Bu seçenek, bir işlemin bir iş parçacığı tarafından ayrılan bağlantı ve nesne tutamaçlarının, aynı işleme ait diğer iş parçacıkları tarafından kullanılabilceği anlamına gelir. Ancak, her defasında yalnızca bir iş parçacığı belirli bir tanıtıcıyı kullanabilir; yani, yalnızca bir tutamaçla dizisel olarak kullanılmasına izin verilir. Bir iş parçacığı, zaten başka bir iş parçacığı tarafından kullanılan bir tutamacı kullanmaya çalışırsa, tutamaç kullanılabilir duruma gelinceye kadar çağrı öbekleri (bekler).

### MQCNO\_HANDLE\_SHARE\_NO\_BLOCK

Bu, MQCNO\_HANDLE\_SHARE\_BLOCK ile aynıdır; ancak, tanıtıcı başka bir iş parçacığı tarafından kullanılsa da, çağrı hemen tamamlanmaya kadar MQCC\_FAILED ve MQRC\_CALL\_IN\_PROGRESS ile tamamlanmaya kadar arama tamamlanır.

Bir iş parçacığıda sıfır ya da bir paylaşılmayan tutamaçlar olabilir:

- MQCNO\_HANDLE\_SHARE\_NONE belirten her MQCONN ya da MQCONNX çağrısı, ilk çağrıda yeni paylaşılmayan bir tanıtıcı döndürür ve ikinci ve daha sonraki çağrılarda da aynı paylaşılmayan tanıtıcı (MQDISC çağrısını araya girmez varsayılarak) döndürür. Neden kodu, ikinci ve sonraki çağrılar için MQRC\_ALREADY\_CONNECTED olur.
- MQCNO\_HANDLE\_SHARE\_BLOCK ya da MQCNO\_HANDLE\_SHARE\_NO\_BLOCK belirten her MQCONNX çağrısı, her çağrıda yeni bir paylaşılan tanıtıcı döndürür.

Nesne tutamaçları, aynı paylaşım özelliklerini, nesne tanıtıcısını yaratan MQOPEN çağrısında belirlenen bağlantı tanıtıcısı ile edinir. Ayrıca, iş birimleri, iş birimini başlatmak için kullanılan bağlantı tanıtıcısı ile aynı paylaşım özelliklerini devralır; iş birimi paylaşılan bir tanıtıcı kullanılarak tek bir iş parçacıkta başlatıldıysa, iş birimi aynı tanıtıcıyı kullanarak başka bir iş parçacığıda güncellenebilir.

Bir tutamaç paylaşımı seçeneği belirtmezseniz, varsayılan değer ortam tarafından belirlenir:

- **Windows** Microsoft Transaction Server (MTS) ortamında varsayılan değer, MQCNO\_HANDLE\_SHARE\_BLOCK ile aynıdır.
- Diğer ortamlarda, varsayılan değer MQCNO\_HANDLE\_SHARE\_NONE ile aynıdır.

## Yeniden bağlanma seçenekleri

Yeniden bağlantı seçenekleri, bir bağlantının yeniden bağlanabilir olup olmadığını belirler. Yalnızca istemci bağlantıları yeniden bağlanabilir.

### MQCNO\_RECONNECT\_AS\_DEF

Yeniden bağlanma seçeneği, varsayılan değerine çözümlenir. Varsayılan değer ayarlanmazsa, bu seçeneğin değeri DISABLE olarak çözülüyor. Bu seçeneğin değeri sunucuya geçirilir ve PCF ve MQSC tarafından sorgulanabilir.

### MQCNO\_RECONNECT

Uygulama, MQCONNX parametresinin **QmgrName** parametresinin değeriyle tutarlı herhangi bir kuyruk yöneticisiyle yeniden bağlanabilir. MQCNO\_RECONNECT seçeneğini, istemci uygulaması ile ilk olarak bir bağlantı kurduğu kuyruk yöneticisi arasında bir benzeşme yoksa kullanın. Bu seçeneğin değeri sunucuya geçirilir ve PCF ve MQSC tarafından sorgulanabilir.

### MQCNO\_RECONNECT\_DISABLE

Uygulama yeniden bağlanamaz. Seçeneğin değeri sunucuya iletilmedi.

### MQCNO\_RECONNECT\_Q\_MGR

Uygulama, yalnızca başlangıçta bağlı olduğu kuyruk yöneticisiyle yeniden bağlanabilir. Bir istemci yeniden bağlanabiliyorsa bu değeri kullanın; ancak, istemci uygulaması ile bir bağlantı kurulduğu kuyruk yöneticisi arasında bir benzerlik vardır. Bir istemcinin yüksek kullanılabilirlikli bir kuyruk yöneticisinin yedek yönetim ortamına otomatik olarak yeniden bağlanmasını istiyorsanız bu değeri seçin. Bu seçeneğin değeri sunucuya geçirilir ve PCF ve MQSC tarafından sorgulanabilir.

Yalnızca istemci bağlantıları için MQCNO\_RECONNECT, MQCNO\_RECONNECT\_DISABLE ve MQCNO\_RECONNECT\_Q\_MGR seçeneklerini kullanın. Bir bağlama bağlantısı için seçenekler kullanılırsa, MQCONNX tamamlanma kodu MQCC\_FAILED ve neden kodu MQRC\_OPTIONS\_ERROR ile başarısız olur. Otomatik istemci yeniden bağlanması IBM MQ classes for Javatafından desteklenmiyor

## Etkileşim paylaşma seçenekleri

Aşağıdaki seçenekler yalnızca TCP/IP istemci bağlantıları için geçerlidir. SNA, SPX ve NetBios kanalları için bu değerler yoksayılr ve kanal, ürünün önceki sürümlerinde olduğu gibi çalışır.

### MQCNO\_NO\_CONV\_SHARING

Bu seçenek, etkileşim paylaşımına izin vermez.

Sohbetlerin yoğun olarak yüklendiği durumlarda, MQCNO\_NO\_CONV\_SHARING kullanabilirsiniz; bu nedenle, çekişmenin, paylaşım etkileşimlerinin var olduğu kanal yönetim ortamının sunucu bağlantısı ucunda yer alan bir olasılık olduğu durumlarda kullanılır. MQCNO\_NO\_CONV\_SHARING, etkileşim paylaşımını desteklemeyen bir kanala bağlandığında, sharecnv (1) ve etkileşim paylaşımını desteklemeyen bir kanala bağlandığında, sharecnv (0) gibi davranır.

## **MQCNO\_ALL\_CONVS\_SHARE**

Bu seçenek, etkileşim paylaşımına izin verir; uygulama, kanal eşgörünümündeki bağlantı sayısına herhangi bir sınır yerleştirmez. Bu seçenek, varsayılan değerdir.

Uygulama, kanal örneğinin paylaşılabilirliğini gösteriyorsa, ancak kanalın sunucu bağlantısı ucundaki *SharingConversations* (SHARECNV) tanımlaması bir paylaşım ayarlıdır, paylaşım oluşmaz ve uygulamaya hiçbir uyarı verilmez.

Benzer şekilde, uygulama paylaşımına izin verildiğini gösteriyorsa, ancak sunucu bağlantısı *SharingConversations* tanımlaması sıfıra ayarlıysa, uyarı verilmez ve uygulama, IBM WebSphere MQ 7.0' dan önceki sürümlerde bir istemciyle aynı davranışı sergiler; paylaşım sohbetleriyle ilgili uygulama ayarı dikkate alınmaz.

MQCNO\_NO\_CONV\_SHARING ve MQCNO\_ALL\_CONVS\_SHARE karşılıklı olarak birbirini dışlar. Belirli bir bağlantıda her iki seçenek de belirtilirse, bağlantı, MQRC\_OPTIONS\_ERROR neden koduyla reddedilir.

## **Kanal tanımlama seçenekleri**

Aşağıdaki seçenekler, MQCNO içinde geçirilen kanal tanımlama yapısının kullanımını denetler:

### **MQCNO\_CD\_FOR\_OUTPUT\_ONLY**

Bu seçenek, MQCNO 'daki kanal tanımlama yapısının yalnızca başarılı bir MQCONNX çağrısında kullanılan kanal adını döndürmek için kullanılmasına izin verir.

Geçerli bir kanal tanımlama yapısı sağlanamazsa, çağrı neden kodu MQRC\_CD\_ERROR ile başarısız olur.

Uygulama istemci olarak çalışmazsa, bu seçenek yoksayılr.

Döndürülen kanal adı, aynı kanal tanımlamasını kullanarak yeniden bağlantı kurmak için MQCNO\_USA\_CD\_SELECTION seçeneğini kullanarak sonraki bir MQCONNX çağrısında kullanılabilir. Bu, istemci kanal çizelgesinde birden çok geçerli kanal tanımlaması olduğunda yararlı olabilir.

### **MQCNO\_USE\_CD\_SEÇİMİ**

Bu seçenek, MQCONNX çağrısının, MQCNO 'da geçirilen kanal tanımlaması yapısında bulunan kanal adını kullanarak bağlanmasına izin verir.

MQSERVER ortam değişkeni ayarlandıysa, bu değişken tarafından tanımlanan kanal tanımlaması kullanılır. MQSERVER belirlenmezse, istemci kanal çizelgesi kullanılır.

Eşleşen kanal adı ve kuyruk yöneticisi adı bulunan bir kanal tanımlaması bulunamazsa, çağrı neden kodu MQRC\_Q\_MGR\_NAME\_ERROR neden ile başarısız olur.

Geçerli bir kanal tanımlama yapısı sağlanamazsa, çağrı neden kodu MQRC\_CD\_ERROR ile başarısız olur.

Uygulama istemci olarak çalışmazsa, bu seçenek yoksayılr.

## **Varsayılan seçenek**

Yukarıda açıklanan seçeneklerden hiçbirine gerek duymuyorsanız, aşağıdaki seçeneği kullanabilirsiniz:

### **MQCNO\_NONE**

Seçenek belirlenmez.

Program belgelerine yardımcı olmak için MQCNO\_NONE değerini kullanın. Bu seçeneğin diğer herhangi bir MQCNO\_\* seçeneği ile kullanılması amaçlanmaz, ancak değeri sıfır olduğu için, bu tür bir kullanım algılanamaz.

### **ClientConnGörelı Konumu (MQUZE)**

ClientConnGörelı Konumu, MQCNO yapısının başlangıcındaki bir MQCD kanal tanımlama yapısının bayt cinsinden görelı konudur. Görelı konum pozitif ya da negatif olabilir. Bu alan, başlangıçtaki 0 değeri olan bir giriş alanıdır.

*ClientConnOffset* yalnızca, MQCONNX çağrısını yayınlayan uygulama bir IBM MQ MQI clientolarak çalışıyorsa kullanın. Bu alanın nasıl kullanılacağı hakkında bilgi için *ClientConnPtr* alanının açıklamasına bakın.

*Version* , MQCNO\_VERSION\_2değerinden küçükse bu alan yok sayılır.

### **ClientConnPtr (MQPTR)**

ClientConnPtr, giriş alanıdır. Başlangıç değeri, işaretçileri destekleyen programlama dillerindeki boş değeri gösterge dir ve tersi durumda, boş değeri bir byte dizisidir.

Yalnızca, MQCONNX çağrısını yayınlayan uygulama IBM MQ MQI clientolarak çalışırken *ClientConnOffset* ve *ClientConnPtr* ' yi kullanın. Uygulama, bu alanlardan birini ya da bir diğeri belirterek, gereken değerleri içeren bir MQCD kanal tanımlaması yapısı sağlayarak istemci bağlantı kanalının tanımını denetleyebilir.

Uygulama bir IBM MQ MQI clientolarak çalışıyorsa, ancak bir MQCD yapısı sağlamıyorsa, kanal tanımlamasını seçmek için MQSERVER ortam değişkeni kullanılır. MQSERVER belirlenmezse, istemci kanal çözeliğesi kullanılır.

Uygulama IBM MQ MQI client, *ClientConnOffset* ve *ClientConnPtr* olarak çalışmazsa, yoksayılr.

Uygulama bir MQCD yapısı sağlıyorsa, alanları gereken değerlere ayarlayın; MQCD ' deki diğ er alanlar yoksayılr. Karakter dizimlerini, alanın uzunluğuna göre boşluklarla doldurur ya da boş değeri bir karakterle sonlandırabilirsiniz. MQCD yapısındaki alanlarla ilgili ek bilgi için [“Alanlar” sayfa 1457 belgesine](#) bakın.

Çizelge 482. MQCD ' deki alanlar

<b>MQCD ' deki alan</b>	<b>Değer</b>
<i>ChannelName</i>	Kanal adı.
<i>Version</i>	Yapı sürüm numarası. MQCD_VERSION_7değerinden küçük olmamalıdır.
<i>TransportType</i>	Desteklenen herhangi bir iletim tipi.
<i>ModeName</i>	LU 6.2 kip adı.
<i>TpName</i>	LU 6.2 hareket programı adı.
<i>SecurityExit</i>	Kanal güvenliği çıkışının adı.
<i>SendExit</i>	Çıkış kanalının çıkış adı.
<i>ReceiveExit</i>	Kanaldan çıkışa ilişkin çıkış adı.
<i>MaxMsgLength</i>	İstemci bağlantı kanalı üzerinden gönderilebilecek ileti bayt cinsinden uzunluk üst sınırı.
<i>SecurityUserData</i>	Güvenlik çıkışı için kullanıcı verileri.
<i>SendUserData</i>	Çıkış göndermek için kullanıcı verileri.
<i>ReceiveUserData</i>	Alma çıkışa ilişkin kullanıcı verileri.
<i>UserIdentifier</i>	LU 6.2 oturumu oluşturmak için kullanılacak kullanıcı kimliği.
<i>Password</i>	LU 6.2 oturumu oluşturmak için kullanılacak parola.
<i>ConnectionName</i>	Bağlantı adı.
<i>HeartbeatInterval</i>	Sağlıklı işletim bildirim akışları arasındaki süre (saniye).

Çizelge 482. MQCD 'deki alanlar (devamı var)

MQCD 'deki alan	Değer
<i>StrucLength</i>	MQCD yapısının uzunluğu.
<i>ExitNameLength</i>	<i>SendExitPtr</i> ve <i>ReceiveExitPtr</i> tarafından adreslenen çıkış adlarının uzunluğu. <i>SendExitPtr</i> ya da <i>ReceiveExitPtr</i> boş değerli gösterge olmayan bir değere ayarlıysa, sıfırdan büyük olmalıdır.
<i>ExitDataLength</i>	<i>SendUserDataPtr</i> ve <i>ReceiveUserDataPtr</i> tarafından adreslenen çıkış verileri uzunluğu. <i>SendUserDataPtr</i> ya da <i>ReceiveUserDataPtr</i> boş değerli gösterge olmayan bir değere ayarlıysa, sıfırdan büyük olmalıdır.
<i>SendExitsDefined</i>	<i>SendExitPtr</i> tarafından adreslenen gönderme çıkış sayısı. Sıfır, <i>SendExit</i> ve <i>SendUserData</i> ise çıkış adını ve verileri sağlar. If greater than zero, <i>SendExitPtr</i> and <i>SendUserDataPtr</i> provide the exit names and data, and <i>SendExit</i> and <i>SendUserData</i> must be blank.
<i>ReceiveExitsDefined</i>	<i>ReceiveExitPtr</i> tarafından adreslenen alma çıkış sayısı. Sıfır, <i>ReceiveExit</i> ve <i>ReceiveUserData</i> ise çıkış adını ve verileri sağlar. If greater than zero, <i>ReceiveExitPtr</i> and <i>ReceiveUserDataPtr</i> provide the exit names and data, and <i>ReceiveExit</i> and <i>ReceiveUserData</i> must be blank.
<i>SendExitPtr</i>	İlk gönderme çıkışının adı.
<i>SendUserDataPtr</i>	İlk gönderme çıkışa ilişkin verilerin adresi.
<i>ReceiveExitPtr</i>	İlk alma çıkışının adının adresi.
<i>ReceiveUserDataPtr</i>	İlk alma çıkışa ilişkin verilerin adresi.
<i>LongRemoteUserIdLength</i>	Uzun uzak kullanıcı kimliğinin uzunluğu.
<i>LongRemoteUserIdPtr</i>	Uzun uzak kullanıcı kimliğinin adresi.
<i>RemoteSecurityId</i>	Uzak güvenlik tanıtıcısı.
<i>SSLCipherSpec</i>	TLS CipherSpec.
<i>SSLPeerNamePtr</i>	TLS eşdüzey adının adresi.
<i>SSLPeerNameLength</i>	TLS eşdüzey adının uzunluğu.
<i>KeepAliveInterval</i>	Kanala ilişkin canlı tutma zamanlaması için iletişim yığına geçirilen değer
<i>LocalAddress</i>	Kullanılacak yerel ağ bağdaştırıcısının IP adresi ve giden bağlantılar için kullanılacak bağlantı noktaları da içinde olmak üzere, yerel iletişim adresi.

Kanal tanımlama yapısını aşağıdaki iki yoldan biriyle sağlayın:

- By using the offset field *ClientConnOffset*

Bu durumda, uygulamanın bir MQCNO içeren bir bileşik yapıyı, ardından MQCD 'nin izlediği bir MQCNO içeren bir bileşik yapı bildirmesi ve *ClientConnOffset* 'un, MQCNO ' nun başlangıcındaki kanal tanımlama yapısının görel konumuna ayarlaması gerekir. Bu görel konumun doğru olduğundan emin olun. *ClientConnPtr* boş (null) işaretçi ya da boş değerli (null) byte olarak ayarlanmalıdır.

İşaretçi veri tipini desteklemeyen programlama dilleri için *ClientConnOffset* kullanın ya da gösterge veri tipini, farklı ortamlara (örneğin, COBOL programlama dili) portable olmayan bir biçimde uygulayın.

Visual Basic programlama dili için, adı verilen bir bileşik yapı MQCNOCD, CMQXB.BAS; bu yapı bir MQCNO yapısını ve bunu izleyen bir MQCD yapısını içerir. MQCNOCD\_DEFAULTS alt yordamı çağrılarak MQCNOCD ' yi başlatın. MQCNOCD şu şekilde kullanıldı: MQCONNX çağrısının MQCONNXAny çeşitlemesi; ek ayrıntılar için MQCONNX çağrısının açıklamasına bakın.

- *ClientConnPtr* işaretçi alanını kullanarak

Bu durumda, uygulama kanal tanımlama yapısını MQCNO yapısından ayrı olarak bildirebilir ve *ClientConnPtr* 'ı kanal tanımlama yapısının adresi olarak ayarlayabilir. *ClientConnOffset* 'ı sıfır olarak ayarlayın.

İşaretçi veri tipini, farklı ortamlara (örneğin, C programlama dili) taşınabilir bir şekilde destekleyen programlama dilleri için *ClientConnPtr* kullanın.

C programlama dilinde, MQCD\_CLIENT\_CONN\_XX\_ENCODE\_CASE\_ONE default makro değişkenini kullanarak, MQCONNX çağrısında kullanılmak üzere, MQCD\_XX\_ENCODE\_CASE\_ONE default tarafından sağlanan başlangıç değerlerinden daha uygun olan yapıya ilişkin başlangıç değerlerini sağlayabilirsiniz.

Seçtiğiniz teknik, *ClientConnOffset* ve *ClientConnPtr* 'den yalnızca birini kullanabilirsiniz; Her ikisi de sıfır dışında bir durumda, çağrı neden kodu MQRC\_CLIENT\_CONN\_ERROR ile başarısız olur.

MQCONNX çağrısı tamamlandıktan sonra, MQCD yapısına yeniden gönderme yapılmaz.

*Version* , MQCNO\_VERSION\_2değerinden küçükse bu alan yok sayılır.

**Not:** Programlama dilinin gösterge veri tipini desteklemediği altyapılarda, bu alan uygun uzunluğun byte dizgisi olarak bildirilir ve ilk değer, tüm boş değerli byte dizgisi olarak kabul edilir.

V 9.2.0

### Çoklu Platformlar üzerinde ConnTag (MQBYTE128)

Bağlantı etiketi kavramsal olarak bir bağlantı tanıtıcısıyla benzerdir, ancak birden çok ilgili bağlantıya yayılabilir ve bunları tek bir uygulama eşgörünümü olarak tanımlayabilir. Çoklu platformlar üzerinde, bağlantı etiketi bağlantı sırasında kuyruk yöneticisi tarafından oluşturulur.

Daha fazla bilgi için bkz. [bağlantı tanıtıcısı](#) ve [uygulama örneği](#).

Oluşturulan bağlantı etiketleri semî insan tarafından okunabilir. Yani, bunlar, yerel karakter kümesindeki dizgiler olarak, MQSC 'de görüntülenebilir ve süzülebilir. İlgili olmak üzere IBM MQ tarafından bilinen bağlantılara otomatik olarak aynı bağlantı etiketi atanır. Bu atama özellikle [uygulama dengeleme](#) açısından önemlidir.

Oluşturulan bağlantı etiketi üç şekilde görünür:

- MQCONNX çağrısına ilişkin MQCNO yapısında, [MQCNO\\_GENERATE\\_CONN\\_TAG](#) belirtildiğinde.
- [GÖRÜNEN EKRAM](#) (ya da programlı eşdeğerleri) çıkışından elde edilen çıkışta.
- [API 'YI GÖRÜNTÜLE](#) (ya da eşdeğerleri) çıkışından elde edilen çıkışta.

Etiket, uygulama sona erdirildiğinde ya da MQDISC çağrısını yayınlarken geçerli olacak şekilde durur.

### İlgili başvurular

“ConnTag (MQBYTE128)- IBM MQ for z/OS” sayfa 331

Bir bağlantı etiketi kavramsal olarak bir bağlantı tanıtıcısına benzer, ancak bunları tek bir uygulama örneği olarak tanımlayan birden çok ilgili bağlantıya yayılabilir. IBM MQ for z/OS üzerinde bağlantı etiketi, uygulama tarafından sağlanan ve o uygulama eşgörünümünden bağlantıları diziselleştirmek için MQCNO\_\*\_CONN\_TAG seçenekleriyle birlikte kullanılan bir giriş alanıdır.

z/OS

### ConnTag (MQBYTE128)- IBM MQ for z/OS

Bir bağlantı etiketi kavramsal olarak bir bağlantı tanıtıcısına benzer, ancak bunları tek bir uygulama örneği olarak tanımlayan birden çok ilgili bağlantıya yayılabilir. IBM MQ for z/OS üzerinde bağlantı etiketi, uygulama tarafından sağlanan ve o uygulama eşgörünümünden bağlantıları diziselleştirmek için MQCNO\_\*\_CONN\_TAG seçenekleriyle birlikte kullanılan bir giriş alanıdır.

Bir uygulamanın eşzamanlı olarak bağlanması amaçlanan birden çok eşgörünümü varsa, her birinin bu alan için benzersiz bir değer sağlaması gerekir. Daha fazla ayrıntı için bu [bağlantı etiketi seçeneklerinin](#) açıklamalarına bakın.

### Notlar:

- IBM MQ for z/OS üzerinde, yürütme sırasında bir uygulamayla ilişkili bağlantı etiketini denetimci olarak belirlemenin bir yolu yoktur.

- ASCII ya da EBCDIC ' de MQ ile başlayan bağlantı etiketi değerleri, IBM ürünleri tarafından kullanılmak üzere ayrılmıştır. Bu harflerle başlayan bağlantı etiketi değerlerini kullanmayın.

Etiket gerekmiyorsa aşağıdaki özel değeri kullanın:

### **MQCT\_NONE**

Değer, alanın uzunluğu için ikili sıfırdır.

C programlama dili için, MQCT\_NONE\_ARRAY değişmezi de tanımlanır; bu değişmezin değeri MQCT\_NONE ile aynı, ancak dizgi yerine bir karakter dizisi.

ConnTag alanı, bir z/OS kuyruk yöneticisine bağlanırken kullanılır.

Bu alanın uzunluğu MQ\_CONN\_TAG\_LENGTH tarafından verilir. *Version* , MQCNO\_VERSION\_3' den küçükse bu alan yok sayılır.

**Multi** IBM MQ for Multiplatforms üzerindeki bağlantı etiketini kullanmaya ilişkin bilgi için bkz. [“Çoklu Platformlar üzerinde ConnTag \(MQBYTE128\)” sayfa 331](#) .

### **SSLConfigPtr (PMQSCO)**

SSLConfigPtr , bir giriş alanıdır. Başlangıç değeri, işaretçileri destekleyen programlama dillerindeki boş değerli göstergedir ve tersi durumda, boş değerli bir byte dizisidir.

Use *SSLConfigPtr* and *SSLConfigOffset* only when the application issuing the MQCONN call is running as an IBM MQ MQI client and the channel protocol is TCP/IP. Uygulama bir IBM MQ istemcisi olarak çalışmazsa ya da kanal iletişim kuralı TCP/IP değilse, *SSLConfigPtr* ve *SSLConfigOffset* yok sayılır.

By specifying *SSLConfigPtr* or *SSLConfigOffset*, plus either *ClientConnPtr* or *ClientConnOffset*, the application can control the use of TLS for the client connection. TLS bilgileri bu şekilde belirtildiğinde, MQSSLKEYR ve MQSSLCRYP ortam değişkenleri yok sayılır; istemci kanal tanımlama çizelgesindeki (CCDT) TLS ile ilgili bilgiler de dikkate alınmaz.

TLS bilgileri yalnızca şu şekilde belirtilebilir:

- İstemci işlemine ilişkin ilk MQCONN çağrısı ya da
- Kuyruk yöneticisine önceki tüm TLS bağlantıları MQDISC kullanılarak sonuçlandırıldığında, sonraki bir MQCONN çağrısı sonlandırıldı.

Bunlar, süreç genelindeki TLS ortamının kullanıma hazırlanabileceği tek durumlardır. TLS ortamı zaten var olduğunda TLS bilgileri belirten bir MQCONN çağrısı yayınlanırsa, çağrıdaki TLS bilgileri yok sayılır ve bağlantı, var olan TLS ortamı kullanılarak yapılır; çağrı, bu durumda MQCC\_UYARI tamamlanma kodunu döndürür ve MQRC\_SSL\_ALREADY\_INITIALIZED neden kodunu döndürür.

MQSCO yapısını, *SSLConfigPtr* içinde bir adres belirterek ya da *SSLConfigOffset* içinde bir görelî konum belirterek, MQCD yapısıyla aynı şekilde sağlayabilirsiniz. Bunun nasıl yapacağını ilişkin ayrıntılar için *ClientConnPtr* ' in açıklamasını görmektedir. Ancak, *SSLConfigPtr* ve *SSLConfigOffset* ' den en fazla birini kullanabilirsiniz. Çağrı, MQRC\_SSL\_CONFIG\_ERROR neden koduyla başarısız olur. Her ikisi de sıfırsa.

MQCONN çağrısı tamamlandıktan sonra, MQSCO yapısına yeniden gönderme yapılmamıştır.

*Version* , MQCNO\_VERSION\_4 değerinden küçükse bu alan yok sayılır.

**Not:** Programlama dilinin işaretçi veri tipini desteklemediği altyapılarda, bu alan uygun uzunluğun byte dizisi olarak bildirilir.

### **SSLConfigOffset (MQUZE)**

SSLConfigOffset , MQCNO yapısının başlangıcındaki bir MQSCO yapısının bayt cinsinden görelî konudur. Görelî konum pozitif ya da negatif olabilir. Bu alan bir giriş alanıdır ve ilk değeri 0 olur.



*SSLConfigOffset* yalnızca, MQCONNX çağrısını yayınlayan uygulama bir IBM MQ MQI clientolarak çalışıyorsa kullanın. Bu alanın nasıl kullanılacağı hakkında bilgi için *SSLConfigPtr* alanının açıklamasına bakın.

*Version* , MQCNO\_VERSION\_4değerinden küçükse bu alan yok sayılır.

### **ConnectionId (MQBYTE24)**

ConnectionId , IBM MQ ' un bir uygulamayı güvenilir bir şekilde tanımasını sağlayan, benzersiz 24 baytlık bir tanıtıcıdır. Uygulama, PUT ve GET çağrılarında ilinti için bu tanıtıcıyı kullanabilir. Bu çıkış parametresi, tüm programlama dillerinde 24 boş baytın ilk değerine sahiptir.

Kuyruk yöneticisi, tüm bağlantılara benzersiz bir tanıtıcı atar, ancak bunlar kurulur. Bir MQCONNX, sürüm 5 MQCNO ile bağlantı kurarsa, uygulama döndürülen MQCNO 'dan ConnectionId ' yi saptayabilir. Atanan tanıtıcı, IBM MQ tarafından oluşturulan, CorrelId, MsgIDve GroupIdgibi diğer tüm tanıtıcılar arasında benzersiz olacak şekilde garantilidir.

PCF komut Sorgula Bağlantı ya da MQSC komutu DISPLAY CONN komutunu kullanarak uzun çalışma birimlerini tanımlamak için ConnectionId komutunu kullanın. MQSC komutları (CONN) tarafından kullanılan ConnectionId , burada döndürülen ConnectionId ögesinden türetilir. The PCF Inquire and Stop Connection commands can use the ConnectionId returned here without modification.

You can use the ConnectionId to force the end of a long running unit of work, by specifying the ConnectionId using the PCF command Stop Connection or the MQSC command STOP CONN. Bu komutların kullanılmasıyla ilgili ek bilgi için Bağlantıyı Durdur ve DURDUR CONN başlıklı konuya bakın.

Sürüm MQCNO\_VERSION\_5değerinden küçükse bu alan döndürülemez.

Bu alanın uzunluğu MQ\_CONNECTION\_ID\_LENGTH tarafından verilir.

### **SecurityParmsGoreli Konumu (MQUZE)**

SecurityParmsGoreli Konumu, MQCNO yapısının başlangıcındaki MQCSP yapısının bayt cinsinden görelî konutdur. Görelî konum pozitif ya da negatif olabilir. Bu alan bir giriş alanıdır ve ilk değeri 0 olur.

*Sürüm* , MQCNO\_VERSION\_5değerinden küçükse bu alan yok sayılır.

MQCSP yapısı "MQCSP-Güvenlik deęiřtirgeleri" sayfa 335içinde tanımlıdır.

### **SecurityParmsPtr (PMQCSP)**

SecurityParmsPtr, yetkilendirme hizmeti tarafından kimlik doęrulaması için bir kullanıcı kimlięi ve parola belirlemek üzere kullanılan MQCSP yapısının adresidir. Bu alan bir giriş alanıdır ve başlangıç değeri boş deęerli ya da boş deęerli bayttır.

*Sürüm* , MQCNO\_VERSION\_5değerinden küçükse bu alan yok sayılır.

MQCSP yapısı "MQCSP-Güvenlik deęiřtirgeleri" sayfa 335içinde tanımlıdır.

### **Reserved (Ayrılmıř) (MQBYTE4)**

Yapıyı 64 bitlik bir sınırla doldurmak için ayrılmıř bir alandır. Alanın ilk değeri, alanın uzunluęuna iliřkin ikili sıfır deęeridir.

*Version* , MQCNO\_VERSION\_6değerinden küçükse bu alan yok sayılır.

### **CCDTUrlLength (MQLONG)**

CCDTUrlLength , baęlantı için kullanılacak istemci baęlantı kanalı çizelgesinin yerini tanıtan bir URL içeren CCDTUrlPtr ya da CCDTUrlOffset ile tanıtılan dizginin uzunluęudur. Alanın ilk değeri sıfırdır.

CCDTUrlLength komutunu, MQCONNX çağrılarını yayınlayan uygulama IBM MQ MQI clientolarak çalıştıęında kullanın.

Bu, MQCHLLIB ve MQCHLTAB ortam deęiřkenlerini ayarlamaya yönelik programlı bir alternatiftir.

Uygulama istemci olarak çalışmıyorsa, CCDTUrlLength yoksayılır.

Version , MQCNO\_VERSION\_6' dan küçükse bu alan yoksayılır.

### **CCDTUrlPtr (PMQCHAR)**

CCDTUrlPtr , bağlantı için kullanılacak istemci bağlantı kanalı çizelgesinin yerini saptamak için, URL içeren bir dizginin isteğe bağlı göstergesidir. Bu alan, göstergeleri destekleyen programlama dillerinde boş değerli bir işaretçinin ilk değeri ve tersi durumda tümü boş değerli bir byte dizgisi olan bir giriş alanıdır.

CCDTUrlPtr komutunu, MQCONNX çağrılarını yayınlayan uygulama IBM MQ MQI clientolarak çalıştığında kullanın.

**Önemli:** CCDTUrlPtr ve CCDTUrlOffset türlerinden yalnızca birini kullanabilirsiniz. Her iki alan da sıfır değilse, çağrı MQRC\_CCDT\_URL\_ERROR neden koduyla başarısız olur.

Bu, MQCHLLIB ve MQCHLTAB ortam değişkenlerini ayarlamaya yönelik programlı bir alternatiftir.

Uygulama istemci olarak çalışmıyorsa, CCDTUrlPtr yoksayılır.

Version , MQCNO\_VERSION\_6' dan küçükse bu alan yoksayılır.

### **CCDTUrlOffset (MQLONG)**

CCDTUrlOffset , MQCNO yapısının başlangıcından bağlantı için kullanılacak istemci bağlantı kanalı çizelgesinin konumunu tanımlayan bir URL ' yi içeren bir dizgiye kadar bayt cinsinden görelî konumdur. Görelî konum pozitif ya da negatif olabilir ve alanın ilk değeri sıfırdır.

CCDTUrlOffset komutunu, MQCONNX çağrılarını yayınlayan uygulama IBM MQ MQI clientolarak çalıştığında kullanın.

**Önemli:** CCDTUrlPtr ve CCDTUrlOffset türlerinden yalnızca birini kullanabilirsiniz. Her iki alan da sıfır değilse, çağrı MQRC\_CCDT\_URL\_ERROR neden koduyla başarısız olur.

Bu, MQCHLLIB ve MQCHLTAB ortam değişkenlerini ayarlamaya yönelik programlı bir alternatiftir.

Uygulama istemci olarak çalışmıyorsa, CCDTUrlOffset yoksayılır.

Version , MQCNO\_VERSION\_6' dan küçükse bu alan yoksayılır.

### **V 9.2.0 ApplName (MQCHAR28)**

Kuyruk yöneticiyle olan bağlantıyı tanımlamak için uygulama tarafından belirlenen ad. Alanın ilk değeri MQAN\_NONE\_DIZISI (boşluk karakterleri) olur.

Version değeri MQCNO\_VERSION\_7 değerinden küçükse ya da değer boşluk olarak ayarlandıysa bu alan yoksayılır.

**z/OS** Bu alanı z/OS üzerinde ayarlayamazsınız. Bu işlemi gerçekleştirmeye çalışırsanız, MQRC\_CNO\_ERROR neden kodunu geri alırsınız.

### **V 9.2.0 Reserved2 (MQBYTE4)**

Yapıyı 64 bitlik bir sınırla doldurmak için ayrılmış bir alandır. Alanın ilk değeri, alanın uzunluğuna ilişkin ikili sıfır değeridir.

Version , MQCNO\_VERSION\_7 değerinden küçükse bu alan yok sayılır.

### **V 9.2.4 BalanceParmsGörelî Konumu (MQUZE)**

MQBNO tipinde bir yapı için, uygulamanın dengeleme davranışına ilişkin bilgileri içeren bellek konumu. Uygulama, bir istemci kanalının üzerinden bağlanmadıkça, tamamen yoksayılır.

Version , MQCNO\_VERSION\_8 değerinden küçükse bu alan yok sayılır.

Ek bilgi için MQBNO başlıklı konuya bakın.

Bu alanı sağlıyorsa, "BalanceParmsPtr (MQPTR)" sayfa 335 alanını sağlayamazsınız. Her iki alanı da sağlamaya çalışırsanız, bir MQRC\_CNO\_ERROR alırsınız. Bu alan yalnızca istemci bağlantılarıyla ilgiliyse, bu alanı diğer bağlantı tiplerinde sağlamak da MQRC\_CNO\_ERROR sonucunu içerir.

## V 9.2.4 BalanceParmsPtr (MQPTR)

MQBNO tipinde bir yapı için, uygulamanın dengeleme davranışına ilişkin bilgileri içeren bir yapı için bellek yeri göstergesi. Uygulama, bir istemci kanalının üzerinden bağlanmadıkça, tamamen yoksayılr.

Version , MQCNO\_VERSION\_8değerinden küçükse bu alan yok sayılır.

Ek bilgi için MQBNO başlıklı konuya bakın.

Bu alanı sağlıyorsa, “BalanceParmsGörelili Konumu (MQUZE)” sayfa 334 alanını sağlayamazsınız. Her iki alanı da sağlamaya çalışırsanız, bir MQRC\_CNO\_ERROR alırsınız. Bu alan yalnızca istemci bağlantılarıyla ilgiliyse, bu alanı diğer bağlantı tiplerinde sağlamak da MQRC\_CNO\_ERROR sonucunu içerir.

## MQCSP-Güvenlik deęiřtirgeleri

MQCSP yapısı, yetkilendirme hizmetinin bir kullanıcı kimliğini ve parolayı doğrulamasını sağlar. MQCONNX çağırısında MQCSP bağlantı güvenliği deęiřtirgeleri yapısını belirtirsiniz.

**Uyarı:** Bazı durumlarda, bir istemci uygulamasına ilişkin MQCSP yapısındaki parola aę üzerinden düz metin olarak gönderilir. İstemci uygulaması parolalarının uygun şekilde korunduęundan emin olmak için bkz. MQCSP parola koruması.

## Kullanılabilirlik

MQCSP yapısı desteklenen tüm IBM MQ altyapılarında kullanılabilir.

## Karakter kümesi ve kodlama

MQCSP ' deki veriler, yerel kuyruk yöneticisinin karakter kümesi ve kodlamasında olmalıdır; bunlar sırasıyla **CodedCharSetId** kuyruk yöneticisi öznitelięi ve MQENC\_NATIVE tarafından verilir.

## Alanlar

**Not:** Ařaęıdaki tabloda, alanlar alfabetik olarak deęil, kullanıma göre gruplanır. Alt konular aynı sırayı izler.

Çizelge 483. MQCSP ' deki Alanlar		
Alan adı ve açıklaması	Deęiřmezin adı	Deęiřmezin ilk deęeri (varsa)
StrucId (yapı tanıtıcısı)	MQCSP_STRUC_ID	' CSP- '
Sürüm (yapı sürümü numarası)	MQCSP_VERSION_1	1
AuthenticationType (kimlik doğrulama tipi)	Yok	MQCSP_AUTH_NONE
Reserved1 ( IBM üzerinde işaretiçli hizalaması için gereklidir)	Yok	Boş deęerli dizilim ya da boşluklar
CSPUserIdPtr (kullanıcı kimliğinin adresi)	Yok	Boş deęerli gösterge ya da boş deęerli byte
CSPUserIdGörelili Konum (kullanıcı kimliğinin görelili konumu)	Yok	0
CSPUserIdLength (kullanıcı kimliği uzunluęu)	Yok	0
Reserved2 ( IBM üzerinde işaretiçli hizalaması için gereklidir)	Yok	Boş deęerli dizilim ya da boşluklar
CSPPasswordPtr (parola adresi)	Yok	Boş deęerli gösterge ya da boş deęerli byte
CSPPasswordOffset (parolanın görelili konumu)	Yok	0

Çizelge 483. MQCSP ' deki Alanlar (devamı var)

Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
CSPPasswordLength (parola uzunluğu)	Yok	0

**Notlar:**

- ↪ simgesi, tek bir boş karakteri temsil eder.
- C programlama dilinde, makro değişkeniMQCSP\_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlar için başlangıç değerleri sağlamak üzere aşağıdaki şekilde kullanılabilir:

```
MQCSP MyCSP = {MQCSP_DEFAULT};
```

## Dil bildirimleri

### MQCSP için C bildirimi

```
typedef struct tagMQCSP MQCSP;
struct tagMQCSP {
    MQCHAR4    StructId;           /* Structure identifier */
    MQLONG     Version;           /* Structure version number */
    MQLONG     AuthenticationType; /* Type of authentication */
    MQBYTE4    Reserved1;        /* Required for IBM i pointer
    alignment */
    MQPTR      CSPUserIdPtr;      /* Address of user ID */
    MQLONG     CSPUserIdOffset;   /* Offset of user ID */
    MQLONG     CSPUserIdLength;   /* Length of user ID */
    MQBYTE8    Reserved2;        /* Required for IBM i pointer
    alignment */
    MQPTR      CSPPasswordPtr;    /* Address of password */
    MQLONG     CSPPasswordOffset; /* Offset of password */
    MQLONG     CSPPasswordLength; /* Length of password */
};
```

### MQCSP için COBOL bildirimi

```
** MQCSP structure
10 MQCSP.
** Structure identifier
15 MQCSP-STRUCID PIC X(4).
** Structure version number
15 MQCSP-VERSION PIC S9(9) BINARY.
** Type of authentication
15 MQCSP-AUTHENTICATIONTYPE PIC S9(9) BINARY.
** Required for IBM i pointer alignment
15 MQCSP-RESERVED1 PIC X(4).
** Address of user ID
15 MQCSP-CSPUSERIDPTR POINTER.
** Offset of user ID
15 MQCSP-CSPUSERIDOFFSET PIC S9(9) BINARY.
** Length of user ID
15 MQCSP-CSPUSERIDLENGTH PIC S9(9) BINARY.
** Required for IBM i pointer alignment
15 MQCSP-RESERVED2 PIC X(4).
** Address of password
15 MQCSP-CSPPASSWORDPTR POINTER.
** Offset of password
15 MQCSP-CSPPASSWORDOFFSET PIC S9(9) BINARY.
** Length of password
15 MQCSP-CSPPASSWORDLENGTH PIC S9(9) BINARY.
```

### MQCSP için PL/I bildirimi

```
dcl
1 MQCSP based,
```

```

3 StrucId          char(4),          /* Structure identifier */
3 Version          fixed bin(31), /* Structure version number */
3 AuthenticationType fixed bin(31), /* Type of authentication */
3 Reserved1       char(4),          /* Required for IBM i pointer
                                   alignment */
3 CSPUserIdPtr    pointer,          /* Address of user ID */
3 CSPUserIdOffset fixed bin(31), /* Offset of user ID */
3 CSPUserIdLength fixed bin(31), /* Length of user ID */
3 Reserved2       char(8),          /* Required for IBM i pointer
                                   alignment */
3 CSPPasswordPtr  pointer,          /* Address of password */
3 CSPPasswordOffset fixed bin(31), /* Offset of user ID */
3 CSPPasswordLength fixed bin(31); /* Length of user ID */

```

MQCSP için Visual Basic bildirimi

```

Type MQCSP
  StrucId          As String*4      'Structure identifier'
  Version          As Long          'Structure version number'
  AuthenticationType As Long        'Type of authentication'
  Reserved1       As MQBYTE4       'Required for IBM i pointer'
                                   'alignment'
  CSPUserIdPtr    As MQPTR         'Address of user ID'
  CSPUserIdOffset As Long          'Offset of user ID'
  CSPUserIdLength As Long          'Length of user ID'
  Reserved2       As MQBYTE8       'Required for IBM i pointer'
                                   'alignment'
  CSPPasswordPtr  As MQPTR         'Address of password'
  CSPPasswordOffset As Long        'Offset of password'
  CSPPasswordLength As Long        'Length of password'
End Type

```

### **StrucId (MQCHAR4)**

Yapı tanıtıcısı.

Değer şu olmalıdır:

#### **MQCSP\_STRUC\_ID**

Güvenlik parametreleri yapısına ilişkin tanıtıcı.

C programlama dili için, MQCSP\_STRUC\_ID\_ARRAY değişmezi de tanımlanır; bu, MQCSP\_STRUC\_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisidir.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQCSPSTRUC\_ID ' dir.

### **Sürüm (MQUZE)**

Yapı sürüm numarası.

Değer şu olmalıdır:

#### **MQCSP\_VERSION\_1**

Version-1 güvenlik parametreleri yapısı.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

#### **MQCSP\_XX\_ENCODE\_CASE\_ONE current\_version**

Güvenlik değiştirgeleri yapısının yürürlükteki sürümü.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQCSP\_VERSION\_1' dir.

### **AuthenticationType (MQUZE)**

AuthenticationType , bir giriş alanıdır. İlk değeri MQCSP\_AUTH\_NONE olur.

Bu, gerçekleştirilecek kimlik doğrulamasının tipidir. Geçerli değerler şunlardır:

#### **MQCSP\_AUTH\_NONE**

Kullanıcı kimliği ve parola alanlarını kullanmayın.

#### **MQCSP\_AUTH\_USER\_ID\_AND\_PWD**

Kullanıcı kimliği ve parola alanlarını doğrulayın.

Varsayılan değer MQCSP\_AUTH\_NONE değeridir. Varsayılan ayar ile parola koruması gerçekleştirilmez.

Kimlik doğrulamaya gereksinim duyarsanız, **MQCSP**'i ayarlamanız gerekir. **AuthenticationType** -MQCSP\_AUTH\_USER\_ID\_AND\_PWD.

Ek bilgi için [MQCSP password protection](#) başlıklı konuya bakın.

### **Reserved1 (MQBYTE4)**

A reserved field, required for pointer alignment on IBM i.

Bu bir giriş alanıdır. Bu alanın ilk değeri boş değerli.

### **CSPUserIdPtr (MQPTR)**

Bu, kimlik doğrulamasında kullanılacak kullanıcı kimliğinin bayt cinsinden adresidir.

Bu bir giriş alanıdır. Bu alanın ilk değeri, göstergeleri destekleyen programlama dillerindeki boş değerli göstergedir ve tersi durumda, boş değerli byte dizilimidir. *Version*, MQCNO\_VERSION\_5 değerinden küçükse bu alan yok sayılır.

This field can contain an operating system user ID when an **AUTHTYPE** of *İŞLEMLER* is named in the **CONNAUTH** field of the queue manager.

Windows üzerinde, tam olarak nitelenmiş bir etki alanı kullanıcı kimliği olabilir.

Kuyruk yöneticisinin **CONNAUTH** alanında bir *IDPWLDAP* adı verilen bir **AUTHTYPE** değeri, bu alanda bir LDAP Kullanıcı Kimliği içerebilir.

### **CSPUserIdGörelili Konumu (MQUZE)**

Bu, kimlik doğrulamasında kullanılacak kullanıcı kimliğinin bayt cinsinden görelili konudur. Görelili konum pozitif ya da negatif olabilir.

Bu bir giriş alanıdır. Bu alanın ilk değeri 0 'tır.

### **CSPUserIdUzunluğu (MQUZE)**

Bu alan, kimlik doğrulamasında kullanılacak kullanıcı kimliğinin uzunluğudur.

Kullanıcı kimliğinin uzunluk üst sınırı altyapıya bağlıdır ve Kullanıcı Kimliklerikonusuna bakın. Kullanıcı kimliğinin uzunluğu izin verilen üst sınırdan büyükse, kimlik denetimi isteği MQRC\_NOT\_AUTHENTICATION ile başarısız olur.

Bu alan bir giriş alanıdır. Bu alanın ilk değeri 0 'tır.

### **Reserved2 (MQBYTE8)**

A reserved field, required for pointer alignment on IBM i.

Bu bir giriş alanıdır. Bu alanın ilk değeri boş değerli.

### **CSPPasswordPtr (MQPTR)**

Bu, kimlik doğrulamasında kullanılacak parolanın bayt cinsinden adresidir.

Bu bir giriş alanıdır. Bu alanın ilk değeri, göstergeleri destekleyen programlama dillerindeki boş değerli göstergedir ve tersi durumda, boş değerli byte dizilimidir. *Version*, MQCNO\_VERSION\_5 değerinden küçükse bu alan yok sayılır.

Bu alan, kuruluşa bağlı olarak işletim sistemi ya da LDAP parolası denetimi tarafından reddedilen, ancak kimlik doğrulama yöntemini geçmeden önce IBM MQ tarafından reddedilmemiş boş bir parola içerebilir.

### **CSPPasswordOffset (MQUZE)**

Bu, kimlik doğrulamasında kullanılacak parolanın bayt cinsinden görelili konudur. Görelili konum pozitif ya da negatif olabilir.

Bu bir giriş alanıdır. Bu alanın ilk değeri 0 'tır.

### **CSPPasswordLength (MQUZE)**

Bu alan, kimlik doğrulamasında kullanılacak parolanın uzunluğudur.

Parola uzunluğu üst sınırı, 256 karakter olan MQ\_CSP\_PASSWORD\_LENGTH ' dir. Parolanın uzunluğu izin verilen üst sınırdan büyükse, kimlik doğrulama isteği MQRC\_NOT\_AUTHENTICAL ile başarısız olur.

Bu alan bir giriş alanıdır. Bu alanın ilk değeri 0 'tır.

### **MQCTLO-Denetim geri çağırma seçenekleri yapısı**

MQCTLO yapısı, bir denetim geri çağırma işleviyle ilgili seçenekleri belirtmek için kullanılır. Yapı, MQCTL çağırısındaki bir giriş ve çıkış deęiştirgesidir.

#### **Kullanılabilirlik**

MQCTLO yapısı ařaęıdaki altyapılarda bulunur:

-  AIX
-  IBM i
-  Linux
-  Windows
-  z/OS

ve bu sistemlere baęlı IBM MQ MQI clients için.

#### **Sürüm**

MQCTLO ' nun yürürlükteki sürümü MQCTLO\_VERSION\_1.

#### **Karakter kümesi ve kodlama**

MQCTLO içindeki veriler, **CodedCharSetId** kuyruk yöneticisi öznitelięi tarafından belirtilen karakter kümesinde ve MQENC\_NATIVE tarafından verilen yerel kuyruk yöneticisinin kodlamasında olmalıdır. Ancak, uygulama bir MQ MQI istemcisi olarak çalışıyorsa, yapı istemcinin karakter kümesinde ve kodlamasında olmalıdır.

#### **Alanlar**

**Not:** Ařaęıdaki tabloda, alanlar alfabetik olarak deęil, kullanıma göre gruplanır. Alt konular aynı sırayı izler.

Alan adı ve açıklaması	Deęişmezin adı	Deęişmezin ilk değeri (varsa)
StrucID (yapı tanıtıcısı)	MQCTLO_STRUC_ID	' CTLO '
Sürüm (yapı sürümü numarası)	MQCTLO_VERSION_1	1
Seçenekler (seçenekler)	MQCTLO_NONE	Boş Deęerler
Seçenekler (ayrılmış alan)	Ayrılmış alan	
ConnectionArea (geri çağırma işlevinin kullanılacağı alan)	Yok	Boş değeri gösterge ya da boş değeri byte

Çizelge 484. MQCTLO 'daki Alanlar (devamı var)

Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
<b>Notlar:</b> 1. C programlama dilinde, makro değişkeniMQCTLO_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlar için başlangıç değerleri sağlamak üzere bunu aşağıdaki şekilde kullanın: <pre>MQCTLO MyCTLO = {MQCTLO_DEFAULT};</pre>		

## Dil bildirimleri

MQCTLO için C bildirimini

```
typedef struct tagMQCTLO MQCTLO;
struct tagMQCTLO {
    MQCHAR4   StrucId;           /* Structure identifier */
    MQLONG    Version;          /* Structure version number */
    MQLONG    Options;          /* Options that control the action of MQCTL */
    MQLONG    Reserved;         /* Reserved field */

    MQPTR     ConnectionArea; /* Connection work area passed to the function */
};
```

MQCTLO için COBOL bildirimini

```
** MQCTLO structure
10  MQCTLO.
** Structure Identifier
15  MQCTLO-STRUCID                PIC X(4).
** Structure Version
15  MQCTLO-VERSION                PIC S9(9) BINARY.
** Options
15  MQCTLO-OPTIONS                PIC S9(9) BINARY.
** Reserved
15  MQCTLO-RESERVED                PIC S9(9) BINARY.
** ConnectionArea
15  MQCTLO-CONNECTIONAREA          POINTER
```

MQCTLO için PL/I bildirimini

```
dcl
1  MQCTLO based,
3  StrucId          char(4),          /* Structure identifier */
3  Version          fixed bin(31),    /* Structure version */
3  Options          fixed bin(31),    /* Options */
3  Reserved         fixed bin(31),
3  ConnectionArea  pointer;          /* Connection work area */
```

### **StrucId (MQCHAR4)**

Denetim seçenekleri yapısı- StrucId alanı

Bu, yapı tanıtıcısıdır; değer şöyle olmalıdır:

#### **MQCTLO\_STRUC\_ID**

Denetim Seçenekleri yapısına ilişkin tanıtıcı.

C programlama dili için, MQCTLO\_STRUC\_ID\_ARRAY değişmezi de tanımlanır; bu, MQCTLO\_STRUC\_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisidir.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQCTLO\_STRUC\_ID 'dir.



## **Sürüm (MQUZE)**

Denetim seçenekleri yapısı-Sürüm alanı

Bu, yapı sürüm numarasıdır; değer şu olmalıdır:

### **MQCTLO\_VERSION\_1**

Version-1 Denetim seçenekleri yapısı.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

### **MQCTLO\_CURRENT\_VERSION**

Denetim seçenekleri yapısının yürürlükteki sürümü.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQCTLO\_VERSION\_1' dir.

## **Seçenekler (MQUZE)**

Denetim seçenekleri yapısı-Seçenekler alanı

MQCTL ' nin işlemini denetleyen seçenekler.

### **MQCTLO\_FAIL\_IF QUIESCING**

Kuyruk yöneticisi ya da bağlantı susturulmuş durumdaysa, MQCTL çağrısını başarısız olacak şekilde zorlayın.

MQCB çağrısında geçirilen MQGMO seçeneklerinde MQGMO\_FAIL\_IF QUIESCING değerini belirtin; bu seçenek, susturma sırasında ileti tüketicilerine bildirilmesine neden olur.

### **MQCTLO\_THREAD\_AFFINITY**

Bu seçenek, uygulamanın, aynı iş parçacığının üzerinde aynı bağlantı için tüm ileti tüketicilerinin çağrılmasını gerektirdiğini bildirir. Bu iş parçacığı, bağlantı durduruluncaya kadar tüketicilere yönelik tüm çağrılar için kullanılır.

**Varsayılan seçenek:** Açıklanan seçeneklerden herhangi birine gereksiniminiz yoksa, aşağıdaki seçeneği kullanın:

### **MQCTLO\_NONE**

Başka bir seçenek belirtilmemeyi belirtmek için bu değeri kullanın; tüm seçenekler varsayılan değerlerini kabul eder. MQCTLO\_NONE, program belgelerine yardımcı olmak için tanımlıdır; bu seçeneğin diğeriyle birlikte kullanılması amaçlanmaz, ancak değeri sıfır olduğu için, bu tür bir kullanım saptanamaz.

Bu bir giriş alanıdır. *Options* alanının ilk değeri MQCTLO\_NONE olur.

## **Ayrılmış (MQUZE)**

Bu ayrılmış bir alandır. Değer sıfır olmalıdır.

## **ConnectionArea (MQPTR)**

Denetim seçenekleri yapısı- ConnectionArea alanı

Bu alan, geri bildirme işlevi için kullanılabilir bir alandır.

Kuyruk yöneticisi, bu alanın içeriğine dayalı olarak karar vermez ve geri bildirme için bir giriş parametresi olan MQCBC yapısındaki ConnectionArea alanına değiştirilmeden geçirilir.

Bu alan, MQOP\_START ve MQOP\_START\_WAN dışındaki tüm işlemler için dikkate alınmaz.

Bu, geri bildirme işlevi için bir giriş ve çıkış alanıdır. Bu alanın başlangıç değeri, boş değerli ya da boş değerli bayttır.

## **MQDH-Dağıtım üstbilgisi**

MQDH yapısı, ileti bir iletim kuyruğunda saklanan bir dağıtım listesi iletilisi olduğunda iletide bulunan ek verileri tanımlar. Dağıtım listesi iletilisi, birden çok hedef kuyruğuna gönderilen bir iletidir. Ek veriler, MQDH yapısının ardından bir MQOR kayıtları dizisi ve bir MQPMR kayıtları dizisinden oluşur. Bu yapı, iletileri doğrudan iletim kuyruklarına koyan ya da iletileri iletim kuyruklarından (örneğin, ileti kanalı araçları)

kaldiren özel uygulamalar tarafından kullanılır. Dağıtım listelerine ileti koymak isteyen uygulamalar bu yapıyı kullanmamalıdır. Bunun yerine, dağıtım listesindeki hedefleri tanımlamak için MQOD yapısını ve ileti özelliklerini belirtmek için MQPMO yapısını kullanmalı ya da tek tek hedeflere gönderilen iletilere ilişkin bilgileri almalıdır.

## Kullanılabilirlik

MQDH yapısı aşağıdaki altyapılarda kullanılabilir:

-  AIX
-  IBM i
-  Linux
-  Windows

ve bu sistemlere bağlı IBM MQ MQI clients için.

## Biçim Adı

MQFMT\_DIST\_HEADER

## Karakter kümesi ve kodlama

MQDH 'deki veriler, **CodedCharSetId** kuyruk yöneticisi özniteliği tarafından belirtilen karakter kümesinde ve MQENC\_NATIVE tarafından verilen yerel kuyruk yöneticisinin kodlamasında yer almalıdır.

MQDH 'nin karakter kümesini ve kodlamasını aşağıdaki alanlardaki *CodedCharSetId* ve *Encoding* alanlarına ayarlayın:

- MQMD (MQDH yapısı ileti verilerinin başındadır) ya da
- MQDH yapısından önce gelen üstbilgi yapısı (diğer tüm durumlar).

## Kullanım

Bir uygulama dağıtım listesine bir ileti koyduğunda ve hedeflerin bazıları ya da tümü uzak olduğunda, kuyruk yöneticisi uygulama iletisi verilerini MQXQH ve MQDH yapılarıyla örnekler ve iletiyi ilgili iletim kuyruğuna yerleştirir. Bu nedenle, ileti bir iletim kuyruğunda olduğunda veriler aşağıdaki sırada gerçekleşir:

- MQXQH yapısı
- MQDH yapısı ve MQOR ve MQPMR kayıtlarının dizileri
- Uygulama iletisi verileri

Hedeflere bağlı olarak, kuyruk yöneticisi bu tür birden çok ileti oluşturabilir ve bunu farklı iletim kuyruklarına yerleştirebilir. Bu durumda, bu iletilerdeki MQDH yapıları, uygulama tarafından açılan dağıtım listesi tarafından tanımlanan hedeflerin farklı alt kümelerini tanımlar.

Dağıtım listesi iletisini doğrudan bir iletim kuyruğuna koyan bir uygulama, daha önce açıklanan sıraya uymalı ve MQDH yapısının doğru olduğundan emin olmalıdır. MQDH yapısı geçerli değilse, kuyruk yöneticisi MQPUT ya da MQPUT1 çağrısından MQRC\_DH\_ERROR neden koduyla başarısız olabilir.

İletileri bir kuyrukta dağıtım listesi biçiminde saklayabilirsiniz; ancak, kuyruğu dağıtım listesi iletilerini destekleyebilecek biçimde tanımladıysanız. “Kuyruklara ilişkin öznitelikler” sayfa 821’inde açıklanan **DistLists** kuyruk özniteliğine bakın. Bir uygulama dağıtım listesi iletisini dağıtım listelerini desteklemeyen bir kuyruğa doğrudan koyarsa, kuyruk yöneticisi dağıtım listesi iletisini tek tek iletilere böler ve bunun yerine kuyruktaki iletileri yerleştirir.

## Alanlar

**Not:** Aşağıdaki tabloda, alanlar alfabetik olarak değil, kullanıma göre gruplanır. Alt konular aynı sırayı izler.

Çizelge 485. MQDH için MQDH 'deki Alanlar		
Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
StrucId (yapı tanıtıcısı)	MQDH_STRUC_ID	'DH↯↯'
Sürüm (yapı sürümü numarası)	MQDH_VERSION_1	1
StrucLength (MQDH yapısının uzunluğu ve aşağıdaki kayıtlar)	Yok	0
Kodlama (MQPMR kayıtları dizisini izleyen verilerin sayısal kodlaması)	Yok	0
CodedCharSetId (MQPMR kayıtları dizisini izleyen verilerin karakter kümesi tanıtıcısı)	MQCCSI_AYRILMAMIŞ	0
Biçim (MQPMR kayıtları dizisini izleyen verilerin biçim adı)	MQFMT_NONE	Boşluklar
İşaretler (genel işaretler)	MQDHF_NONE	0
PutMsgRecFields (hangi MQPMR alanlarının var olduğunu gösteren işaretler)	MQPMRF_NONE	0
RecsPresent (var olan nesne kaydı sayısı)	Yok	0
ObjectRecGörelili Konum (MQDH 'nin başlangıcındaki ilk nesne kaydının görelili konumu)	Yok	0
PutMsgRecOffset (MQDH 'nin başlangıcından itibaren ilk koyma iletisi kaydının görelili konumu)	Yok	0
<b>Notlar:</b>		
1. ↯ simgesi, tek bir boş karakteri temsil eder.		
2. C programlama dilinde, makro değişkeniMQDH_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlar için başlangıç değerleri sağlamak üzere bunu aşağıdaki şekilde kullanın:		
<pre>MQDH MyDH = {MQDH_DEFAULT};</pre>		

## Dil bildirimleri

MQDH için C bildirimi

```
typedef struct tagMQDH MQDH;
struct tagMQDH {
    MQCHAR4   StrucId;           /* Structure identifier */
    MQLONG    Version;          /* Structure version number */
    MQLONG    StrucLength;      /* Length of MQDH structure plus following
                                MQOR and MQPMR records */
    MQLONG    Encoding;         /* Numeric encoding of data that follows
                                the MQOR and MQPMR records */
    MQLONG    CodedCharSetId;   /* Character set identifier of data that
                                follows the MQOR and MQPMR records */
    MQCHAR8   Format;           /* Format name of data that follows the
                                MQOR and MQPMR records */
    MQLONG    Flags;            /* General flags */
    MQLONG    PutMsgRecFields;  /* Flags indicating which MQPMR fields are
                                present */
    MQLONG    RecsPresent;      /* Number of MQOR records present */
};
```

```

MQLONG   ObjectRecOffset; /* Offset of first MQOR record from start
                          of MQDH */
MQLONG   PutMsgRecOffset; /* Offset of first MQPMR record from start
                          of MQDH */
};

```

### MQDH için COBOL bildirimi

```

**   MQDH structure
10  MQDH.
**   Structure identifier
15  MQDH-STRUCID      PIC X(4).
**   Structure version number
15  MQDH-VERSION     PIC S9(9) BINARY.
**   Length of MQDH structure plus following MQOR and MQPMR records
15  MQDH-STRUCLNGTH PIC S9(9) BINARY.
**   Numeric encoding of data that follows the MQOR and MQPMR records
15  MQDH-ENCODING    PIC S9(9) BINARY.
**   Character set identifier of data that follows the MQOR and MQPMR
**   records
15  MQDH-CODEDCHARSETID PIC S9(9) BINARY.
**   Format name of data that follows the MQOR and MQPMR records
15  MQDH-FORMAT      PIC X(8).
**   General flags
15  MQDH-FLAGS       PIC S9(9) BINARY.
**   Flags indicating which MQPMR fields are present
15  MQDH-PUTMSGRECFIELDS PIC S9(9) BINARY.
**   Number of MQOR records present
15  MQDH-RECSPRESENT PIC S9(9) BINARY.
**   Offset of first MQOR record from start of MQDH
15  MQDH-OBJECTRECOFFSET PIC S9(9) BINARY.
**   Offset of first MQPMR record from start of MQDH
15  MQDH-PUTMSGRECOFFSET PIC S9(9) BINARY.

```

### MQDH için PL/I bildirimi

```

dcl
1  MQDH based,
3  StrucId      char(4),          /* Structure identifier */
3  Version     fixed bin(31),    /* Structure version number */
3  StrucLength fixed bin(31),    /* Length of MQDH structure plus
                               following MQOR and MQPMR
                               records */
3  Encoding     fixed bin(31),   /* Numeric encoding of data that
                               follows the MQOR and MQPMR
                               records */
3  CodedCharSetId fixed bin(31), /* Character set identifier of data
                               that follows the MQOR and MQPMR
                               records */
3  Format       char(8),          /* Format name of data that follows
                               the MQOR and MQPMR records */
3  Flags       fixed bin(31),    /* General flags */
3  PutMsgRecFields fixed bin(31), /* Flags indicating which MQPMR
                               fields are present */
3  RecsPresent fixed bin(31),    /* Number of MQOR records present */
3  ObjectRecOffset fixed bin(31), /* Offset of first MQOR record from
                               start of MQDH */
3  PutMsgRecOffset fixed bin(31); /* Offset of first MQPMR record from
                               start of MQDH */

```

### MQDH için Visual Basic bildirimi

```

Type MQDH
StrucId      As String*4 'Structure identifier'
Version      As Long     'Structure version number'
StrucLength  As Long     'Length of MQDH structure plus following'
              'MQOR and MQPMR records'
Encoding     As Long     'Numeric encoding of data that follows'
              'the MQOR and MQPMR records'
CodedCharSetId As Long   'Character set identifier of data that'
              'follows the MQOR and MQPMR records'
Format       As String*8 'Format name of data that follows the'
              'MQOR and MQPMR records'
Flags        As Long     'General flags'
PutMsgRecFields As Long  'Flags indicating which MQPMR fields are'

```

RecsPresent	As Long	'present'
ObjectRecOffset	As Long	'Number of MQOR records present'
		'Offset of first MQOR record from start'
		'of MQDH'
PutMsgRecOffset	As Long	'Offset of first MQPMR record from start'
		'of MQDH'
End Type		

### **StrucId (MQCHAR4)**

Değer şu olmalıdır:

#### **MQDH\_STRUC\_ID**

Dağıtım üstbilgisi yapısına ilişkin tanıtıcı.

C programlama dili için, MQDH\_STRUC\_ID\_ARRAY değişmezi de tanımlanır; bu, MQDH\_STRUC\_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisidir.

Bu alanın ilk değeri MQDH\_STRUC\_ID 'dir.

### **Sürüm (MQUZE)**

Değer şu olmalıdır:

#### **MQDH\_VERSION\_1**

Dağıtım üstbilgisi yapısına ilişkin sürüm numarası.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

#### **MQDH\_XX\_ENCODE\_CASE\_ONE current\_version**

Dağıtım üstbilgisi yapısının yürürlükteki sürümü.

Bu alanın ilk değeri MQDH\_VERSION\_1' dir.

### **StrucLength (MQUZE)**

Bu, MQDH yapısının başlangıcından, MQOR ve MQPMR kayıtlarının dizilerini izleyen ileti verilerinin başlangıcından byte sayıdır. Veriler aşağıdaki sırayla gerçekleşir:

- MQDH yapısı
- MQOR kayıtları dizisi
- MQPMR kayıtları dizisi
- İleti Verileri

MQOR ve MQPMR kayıtlarının dizileri, MQDH yapısının içerdiği görelî kümelerle adreslenir. Bu görelî konum, bir ya da daha çok MQDH yapısı, kayıt dizileri ve ileti verileri arasında kullanılmayan baytlarla sonuçlanırsa, kullanılmayan byte 'lar *StrucLength* değerine eklenmelidir; ancak bu baytların içeriği kuyruk yöneticisi tarafından korunmaz. MQPMR kayıtları dizisi için, MQOR kayıtları diziden önce gelen dizi için geçerlidir.

Bu alanın ilk değeri 0 'tır.

### **Kodlama (MQUZE)**

Bu, MQOR ve MQPMR kayıtları dizilerini izleyen verilerin sayısal kodlamasıdır; MQDH yapısındaki sayısal veriler için geçerli değildir.

MQPUT ya da MQPUT1 çağrısında, uygulama bu alanı veri için uygun değere ayarlamalıdır.

Bu alanın ilk değeri 0 'tır.

### **CodedCharSetId (MQHOMER)**

Bu, MQOR ve MQPMR kayıtları dizilerini izleyen verilerin karakter takımı tanıtıcısıdır; MQDH yapısındaki karakter verileri için geçerli değildir.

MQPUT ya da MQPUT1 çağrısında, uygulama bu alanı veri için uygun değere ayarlamalıdır. Aşağıdaki özel değeri kullanabilirsiniz:

### **MQCCSI\_INHERIT**

Bu yapının karakter kümesi tanıtıcısını devralır.

Character data in the data *takip edilen* this structure is in the same character set as this structure.

Kuyruk yöneticisi, bu değeri, iletide belirtilen yapının gerçek karakter kümesi tanıtıcısına göre değiştirir. Herhangi bir hata oluşmazsa, MQGET çağrısı MQCCSI\_INHERIT değerini döndürmez.

MQMD 'deki *PutAppLType* alanının değeri MQAT\_BROKER ise, MQCCSI\_INHERIT kullanamazsınız.

Bu değer aşağıdaki ortamlarda desteklenir:

-  AIX
-  IBM i
-  Linux
-  Windows

ve bu sistemlere bağlı IBM MQ istemcileri için geçerli olur.

Bu alanın ilk değeri MQCCSI\_UNDEFED değerine sahip.

### **Biçim (MQCHAR8)**

Bu, MQOD ve MQPMR kayıtları dizilerini izleyen verilerin biçimidir (en son hangisi oluşur).

MQPUT ya da MQPUT1 çağrısında, uygulama bu alanı veri için uygun değere ayarlamalıdır. Bu alanı kodlamaya ilişkin kurallar, MQMD 'deki *Format* alanı için yapılan kurullarla aynıdır.

Bu alanın ilk değeri MQFMT\_NONE olur.

### **İşaretler (MQUZE)**

Aşağıdaki işareti belirtebilirsiniz:

#### **MQDHF\_NEW\_MSG\_IDS**

Dağıtım listesinde her hedef için yeni bir ileti tanıtıcısı oluşturun. Bu değeri yalnızca, herhangi bir put-message kaydı yoksa ya da kayıtlar varsa, ancak *MsgId* alanını içermiyorsa ayarlayın.

Bu işaret, dağıtım listesi iletisinin son olarak tek tek iletilere bölünmesine kadar ileti tanıtıcılarının oluşturulmalarını korumaktadır. Bu, dağıtım listesi iletilisiyle akışı gereken denetim bilgileri miktarını en aza indirir.

Bir uygulama bir dağıtım listesine ileti koyduğunda, kuyruk yöneticisi, aşağıdaki deyimlerin her ikisi de doğru olduğunda oluşturduğu MQDH 'ye MQDHF\_NEW\_MSG\_IDS' yi ayarlar:

- Uygulama tarafından sağlanan herhangi bir put-message kaydı yok ya da sağlanan kayıtlar *MsgId* alanını içermiyor.
- MQMD 'deki *MsgId* alanı MQMI\_NONE olur ya da MQPMO' daki *Options* alanı MQPMO\_NEW\_MSG\_ID ' yi içerir.

Herhangi bir işaret gerekli değilse, aşağıdakileri belirtin:

#### **MQDHF\_NONE**

Herhangi bir işaret belirtilmedi. MQDHF\_NONE, program belgelerine yardımcı olmak için tanımlıdır. Bu değişimin başka biriyle birlikte kullanılması amaçlanmaz, ancak değeri sıfır olduğu için bu tür bir kullanım saptanamaz.

Bu alanın ilk değeri MQDHF\_NONE olur.

### **PutMsgRecFields (MQlong)**

Aşağıdaki işaretlerin hiçbirini ya da birkaçını belirtebilirsiniz:

**MQPMRF\_MSG\_ID**

İleti tanıtıcısı alanı var.

**MQPMRF\_CORREL\_ID**

İlinti tanıtıcısı alanı var.

**MQPMRF\_GROUP\_ID**

Grup tanıtıcısı alanı var.

**MQPMRF\_FEEDBACK**

Geribildirim alanı mevcut.

**MQPMRF\_ACCOUNTING\_TOKEN**

Muhasebe belirteci alanı var.

Hiçbir MQPMR alanı yoksa, aşağıdaki bilgileri belirtin:

**MQPMRF\_NONE**

Hiçbir put-message kayıt alanı mevcut değil. MQPMRF\_NONE, program belgelerine yardımcı olmak için tanımlıdır. Bu değişmezin başka biriyle birlikte kullanılması amaçlanmaz, ancak değeri sıfır olduğu için bu tür bir kullanım saptanamaz.

Bu alanın ilk değeri MQPMRF\_NONE olur.

**RecsPresent (MQUZE)**

Bu, varış noktalarının sayısıdır. Bir dağıtım listesi her zaman en az bir hedef içermeli, bu nedenle *RecsPresent* her zaman sıfırdan büyük olmalıdır.

Bu alanın ilk değeri 0 'tır.

**ObjectRecGörelili Konumu (MQUZE)**

Bu, hedef kuyrukların adlarını içeren MQOR nesne kayıtları dizideki ilk kaydın bayt cinsinden görelili konumu verir. Bu dizideki *RecsPresent* kayıtları var. Bu kayıtlar (artı ilk nesne kaydı ile önceki alan arasında atlanan tüm baytlar), *StrucLength* alanı tarafından verilen uzunluğa eklenir.

Bir dağıtım listesi her zaman en az bir hedef içermeli, bu nedenle *ObjectRecOffset* her zaman sıfırdan büyük olmalıdır.

Bu alanın ilk değeri 0 'tır.

**PutMsgRecOffset (MQlong)**

Bu, MQPMR dizesindeki ilk kaydın byte cinsinden görelili konumu verir, ileti özelliklerini içeren ileti kayıtlarını içerir. Varsa, bu dizideki *RecsPresent* kayıtları vardır. Bu kayıtlar (artı, ilk put ileti kaydı ile önceki alan arasında atlanan tüm baytlar), *StrucLength* alanı tarafından verilen uzunluğa eklenir.

Put ileti kayıtları isteğe bağlıdır; kayıt sağlanmaz, *PutMsgRecOffset* sıfır ve *PutMsgRecFields* MQPMRF\_NONE değerini içerir.

Bu alanın ilk değeri 0 'tır.

**MQDLH-Harf üstbilgisi**

MQDLH yapısı, teslim edilmeyen iletiler (teslim edilmeyen ileti) kuyruğundaki iletilerin uygulama iletileri verilerinin önekini oluşturan bilgileri tanımlar. Kuyruk yöneticisi ya da ileti kanalı aracısı iletiyi kuyruğa yeniden yönlendirdiği için ya da bir uygulama iletiyi doğrudan kuyruğa koyduğundan, ileti gönderilmeyen ileti kuyruğuna gelebilir.

**Biçim Adı**

MQFMT\_DEAD\_LETTER\_HEADER

## Karakter kümesi ve kodlama

MQDLH yapısındaki alanlar, *CodedCharSetId* ve *Encoding* alanları tarafından verilen karakter takımındaki ve kodlamasındaki alanlardır. Bunlar, MQDLH ' den önce gelen üstbilgi yapısında ya da MQDLH uygulama ileti verilerinin başlangıcında ise MQMD yapısında belirtilir.

Karakter takımı, kuyruk adlarında geçerli olan karakterler için tek baytlık karakterler içeren bir karakter olmalıdır.

Java/JMS için IBM MQ sınıflarını kullanıyorsanız ve MQMD ' de tanımlanan kod sayfası Java sanal makinesi tarafından desteklenmiyorsa, MQDLH UTF-8 karakter kümesine yazılır.

## Kullanım

İletileri doğrudan ileti kuyruğuna koyan uygulamalar, ileti verilerine bir MQDLH yapısıyla örnek eklemeli ve alanları uygun değerlerle kullanıma hazırlamalıdır. Ancak, kuyruk yöneticisi bir MQDLH yapısının var olmasını ya da alanlar için geçerli değerlerin belirtilmesini gerektirmez.

Bir ileti gelmeyen iletiler kuyruğuna konamayacak kadar uzunsa, uygulamanın aşağıdakilerden birini yapması gerekir:

- İleti verilerini, gelmeyen iletiler kuyruğuna sığacak şekilde kısaltın.
- İletiyi yardımcı belleğe kaydedin ve gitmeyen iletiler kuyruğuna bunu gösteren bir kural dışı durum raporu iletileri yerleştirin.
- İletiyi atın ve oluşturan bir hata döndürün. İleti kritik bir iletiyse (ya da önemli bir ileti olabiliyorsa), bunu yalnızca oluşturanın iletinin bir kopyasına sahip olduğu biliniyorsa yapın; örneğin, bir iletişim kanalından bir ileti kanalı aracısı tarafından alınan bir ileti.

Önceki işlemlerden hangisi uygunsuzsa (varsa), uygulamanın tasarımına bağlıdır.

Kuyruk yöneticisi, kesim olan bir ileti öndeki MQDLH yapısıyla yerleştirildiğinde özel işlem gerçekleştirir; daha fazla ayrıntı için MQMDE yapısının tanımına bakın.

## İletileri gönderilmeyen iletiler kuyruğuna koyma

Teslim edilmeyen ileti kuyruğuna bir ileti bulunduğunda, MQPUT ya da MQPUT1 çağrısı için kullanılan MQMD yapısı, aşağıdaki kural dışı durumlar dışında, iletiyle ilişkilendirilmiş MQMD ile (genellikle MQGET çağrısı tarafından döndürülen MQMD) aynı olmalıdır:

- *CodedCharSetId* ve *Encoding* alanlarını, MQDLH yapısındaki alanlar için kullanılan karakter kümesi ve kodlamaya ayarlayın.
- Verilerin bir MQDLH yapısıyla başladığını belirtmek için *Format* alanını MQFMT\_DEAD\_LETTER\_HEADER olarak ayarlayın.
- Koşullara uygun bir bağlam seçeneği kullanarak bağlam alanlarını (*AccountingToken*, *AppIdentityData*, *AppOriginData*, *PutAppName*, *PutAppType*, *PutDate*, *PutTime*, *UserIdentifier*) ayarlayın:
  - Gitmeyen ileti kuyruğuna, önceki iletilerle ilgili olmayan bir iletiyi koyan bir uygulamanın MQPMO\_DEFAULT\_CONTEXT seçeneğini kullanması gerekir; bu, kuyruk yöneticisinin ileti tanımlayıcısındaki tüm bağlam alanlarını varsayılan değerlerine ayarlamasına neden olur.
  - Az önce aldığı bir ileti kuyruğuna konan bir sunucu uygulaması, özgün bağlam bilgilerini korumak için MQPMO\_PASS\_ALL\_CONTEXT seçeneğini kullanmalıdır.
  - Yeni aldığı bir iletiye *yanıt* gönderen bir sunucu uygulaması, MQPMO\_PASS\_IDENTITY\_CONTEXT seçeneğini kullanmalıdır; bu, kimlik bilgilerini korur, ancak kaynak bilgilerini sunucu uygulamasının kaynağı olacak şekilde ayarlar.
  - Bir ileti kanalı aracısı, iletişim kanalından aldığı bir ileti kuyruğuna, özgün bağlam bilgilerini korumak için MQPMO\_SET\_ALL\_CONTEXT seçeneğini kullanmalıdır.

MQDLH yapısının kendisinde, alanları aşağıdaki gibi ayarlayın:



- *CodedCharSetId*, *Encoding* ve *Format* alanlarını MQDLH yapısını izleyen verileri (genellikle özgün ileti tanımlayıcısındaki değerleri) tanımlayan değerlere ayarlayın.
- *PutApplType*, *PutApplName*, *PutDate* ve *PutTime* bağlam alanlarını, iletiyi gitmeyen ileti kuyruğuna koyan uygulamaya uygun değerlere ayarlayın; bu değerler özgün iletiyle ilgili değildir.
- Diğer alanları uygun şekilde ayarlayın.

Tüm alanların geçerli değerlere sahip olduğunu ve karakter alanlarının alanın tanımlı uzunluğuna kadar boşluklarla doldurulması gerektiğini doğrulayın; kuyruk yöneticisi, boş ve sonraki karakterleri MQDLH yapısında boşluklara dönüştürmediği için, karakter verilerini boş bir karakter kullanarak zamanından önce sona erdirmeyin.

## Gitmeyen iletiler kuyruğundan ileti alınıyor

Gitmeyen iletiler kuyruğundan ileti alan uygulamalar, iletilerin bir MQDLH yapısıyla başladığını doğrulamalıdır. Uygulama, ileti tanımlayıcısı MQMD 'deki *Format* alanını inceleyerek bir MQDLH yapısının olup olmadığını saptayabilir; alan MQFMT\_DEAD\_LETTER\_HEADER değerine sahipse, ileti verileri bir MQDLH yapısıyla başlar. Ayrıca, uygulamaların gitmeyen iletiler kuyruğundan alacakları iletilerin kuyruk için çok uzun olması durumunda kesilebileceğini de unutmayın.

## Alanlar

**Not:** Aşağıdaki tabloda, alanlar alfabetik olarak değil, kullanıma göre gruplanır. Alt konular aynı sırayı izler.

Çizelge 486. MQDLH için MQDLH 'deki Alanlar		
Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
<u>StrucId</u> (yapı tanıtcısı)	MQDLH_STRUC_ID	'DLH→'
<u>Sürüm</u> (yapı sürümü numarası)	MQDLH_VERSION_1	1
<u>Neden</u> (ileti gitmeyen iletiler kuyruğuna neden geldi)	MQRC_NONE	0
<u>DestQName</u> (özgün hedef kuyruğun adı)	Yok	Boş değerli dizilim ya da boşluklar
<u>DestQMgrAd</u> (özgün hedef kuyruk yöneticisinin adı)	Yok	Boş değerli dizilim ya da boşluklar
<u>Kodlama</u> (MQDLH 'yi izleyen verilerin sayısal kodlaması)	Yok	0
<u>CodedCharSetId</u> (MQDLH 'yi izleyen verilerin karakter kümesi tanıtcısı)	MQCCSI_AYRILMAMIŞ	0
<u>Biçim</u> (MQDLH 'yi izleyen verilerin biçim adı)	MQFMT_NONE	Boşluklar
<u>PutApplTip</u> (iletiyi teslim almayan ileti kuyruğuna koyan uygulamanın tipi)	Yok	0
<u>PutApplAdı</u> (iletiyi gönderilmeyen iletiler kuyruğuna koyan uygulamanın adı)	Yok	Boş değerli dizilim ya da boşluklar
<u>PutDate</u> (iletinin ölü mektup kuyruğuna konma tarihi)	Yok	Boş değerli dizilim ya da boşluklar
<u>PutTime</u> (iletinin ileti gönderilmeyen iletiler kuyruğuna konma süresi)	Yok	Boş değerli dizilim ya da boşluklar

Çizelge 486. MQLDH için MQLDH 'deki Alanlar (devamı var)

Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
<p><b>Notlar:</b></p> <ol style="list-style-type: none"> <li>1. ~ simgesi, tek bir boş karakteri temsil eder.</li> <li>2. Boş değerli dizilim ya da boşluk değeri, C dizgisindeki boş dizgiyi ve diğer programlama dillerindeki boş karakterleri gösterir.</li> <li>3. C programlama dilinde, makro değişkeniMQLDH_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlar için başlangıç değerleri sağlamak üzere bunu aşağıdaki şekilde kullanın:</li> </ol> <pre>MQLDH MyDLH = {MQLDH_DEFAULT};</pre>		

## Dil bildirimleri

### MQLDH için C bildirimi

```
typedef struct tagMQLDH MQLDH;
struct tagMQLDH {
    MQLCHAR4   StrucId;           /* Structure identifier */
    MQLONG     Version;          /* Structure version number */
    MQLONG     Reason;           /* Reason message arrived on dead-letter
    (undelivered-message) queue */
    MQLCHAR48  DestQName;        /* Name of original destination queue */
    MQLCHAR48  DestQMgrName;     /* Name of original destination queue
    manager */
    MQLONG     Encoding;         /* Numeric encoding of data that follows
    MQLDH */
    MQLONG     CodedCharSetId;   /* Character set identifier of data that
    follows MQLDH */
    MQLCHAR8   Format;           /* Format name of data that follows
    MQLDH */
    MQLONG     PutAppType;       /* Type of application that put message on
    dead-letter (undelivered-message)
    queue */
    MQLCHAR28  PutAppName;      /* Name of application that put message on
    dead-letter (undelivered-message)
    queue */
    MQLCHAR8   PutDate;         /* Date when message was put on dead-letter
    (undelivered-message) queue */
    MQLCHAR8   PutTime;         /* Time when message was put on the
    dead-letter (undelivered-message)
    queue */
};
```

### MQLDH için COBOL bildirimi

```
** MQLDH structure
10 MQLDH.
** Structure identifier
15 MQLDH-STRUCID PIC X(4).
** Structure version number
15 MQLDH-VERSION PIC S9(9) BINARY.
** Reason message arrived on dead-letter (undelivered-message) queue
15 MQLDH-REASON PIC S9(9) BINARY.
** Name of original destination queue
15 MQLDH-DESTQNAME PIC X(48).
** Name of original destination queue manager
15 MQLDH-DESTQMGRNAME PIC X(48).
** Numeric encoding of data that follows MQLDH
15 MQLDH-ENCODING PIC S9(9) BINARY.
** Character set identifier of data that follows MQLDH
15 MQLDH-CODEDCHARSETID PIC S9(9) BINARY.
** Format name of data that follows MQLDH
15 MQLDH-FORMAT PIC X(8).
** Type of application that put message on dead-letter
(undelivered-message) queue
```

```

15 MQDLH-PUTAPPLTYPE PIC S9(9) BINARY.
** Name of application that put message on dead-letter
** (undelivered-message) queue
15 MQDLH-PUTAPPLNAME PIC X(28).
** Date when message was put on dead-letter (undelivered-message)
** queue
15 MQDLH-PUTDATE PIC X(8).
** Time when message was put on the dead-letter (undelivered-message)
** queue
15 MQDLH-PUTTIME PIC X(8).

```

### MQDLH için PL/I bildirimi

```

dcl
1 MQDLH based,
3 StrucId char(4), /* Structure identifier */
3 Version fixed bin(31), /* Structure version number */
3 Reason fixed bin(31), /* Reason message arrived on
dead-letter (undelivered-message)
queue */
3 DestQName char(48), /* Name of original destination
queue */
3 DestQMgrName char(48), /* Name of original destination queue
manager */
3 Encoding fixed bin(31), /* Numeric encoding of data that
follows MQDLH */
3 CodedCharSetId fixed bin(31), /* Character set identifier of data
that follows MQDLH */
3 Format char(8), /* Format name of data that follows
MQDLH */
3 PutApplType fixed bin(31), /* Type of application that put
message on dead-letter
(undelivered-message) queue */
3 PutApplName char(28), /* Name of application that put
message on dead-letter
(undelivered-message) queue */
3 PutDate char(8), /* Date when message was put on
dead-letter (undelivered-message)
queue */
3 PutTime char(8); /* Time when message was put on the
dead-letter (undelivered-message)
queue */

```

### MQDLH için High Level Assembler (Üst Düzey Çevirici) bildirimi

```

MQDLH          DSECT
MQDLH_STRUCID  DS CL4  Structure identifier
MQDLH_VERSION  DS F    Structure version number
MQDLH_REASON   DS F    Reason message arrived on dead-letter
*              (undelivered-message) queue
MQDLH_DESTQNAME DS CL48 Name of original destination queue
MQDLH_DESTQMGRNAME DS CL48 Name of original destination queue
*              manager
MQDLH_ENCODING DS F    Numeric encoding of data that follows
*              MQDLH
MQDLH_CODEDCHARSETID DS F Character set identifier of data that
*              follows MQDLH
MQDLH_FORMAT   DS CL8  Format name of data that follows MQDLH
MQDLH_PUTAPPLTYPE DS F  Type of application that put message on
*              dead-letter (undelivered-message) queue
MQDLH_PUTAPPLNAME DS CL28 Name of application that put message on
*              dead-letter (undelivered-message) queue
MQDLH_PUTDATE  DS CL8  Date when message was put on
*              dead-letter (undelivered-message) queue
MQDLH_PUTTIME  DS CL8  Time when message was put on the
*              dead-letter (undelivered-message) queue
*
MQDLH_LENGTH   EQU *-MQDLH
                ORG MQDLH
MQDLH_AREA     DS CL(MQDLH_LENGTH)

```

### MQDLH için Visual Basic bildirimi

```

Type MQDLH
StrucId As String*4 'Structure identifier'

```

Version	As Long	'Structure version number'
Reason	As Long	'Reason message arrived on dead-letter' '(undelivered-message) queue'
DestQName	As String*48	'Name of original destination queue'
DestQMgrName	As String*48	'Name of original destination queue' 'manager'
Encoding	As Long	'Numeric encoding of data that follows' 'MQDLH'
CodedCharSetId	As Long	'Character set identifier of data that' 'follows MQDLH'
Format	As String*8	'Format name of data that follows MQDLH'
PutApplType	As Long	'Type of application that put message on' 'dead-letter (undelivered-message) queue'
PutApplName	As String*28	'Name of application that put message on' 'dead-letter (undelivered-message) queue'
PutDate	As String*8	'Date when message was put on dead-letter' '(undelivered-message) queue'
PutTime	As String*8	'Time when message was put on the' 'dead-letter (undelivered-message) queue'
End Type		

### **StrucId (MQCHAR4)**

StrucId , yapı tanıtıcısıdır.

Değer şu olmalıdır:

#### **MQDLH\_STRUC\_ID**

Ölü-harf üstbilgisi yapısına ilişkin tanıtıcı.

C programlama dili için, MQDLH\_STRUC\_ID\_ARRAY değişmezi de tanımlanır; bu değer, MQDLH\_STRUC\_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisi olur.

Bu alanın ilk değeri MQDLH\_STRUC\_ID ' dir.

### **Sürüm (MQUZE)**

Sürüm, yapı sürüm numarasıdır.

Değer şu olmalıdır:

#### **MQDLH\_VERSION\_1**

Ölü-mektup üstbilgisi yapısına ilişkin sürüm numarası.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

#### **MQDLH\_CURRENT\_VERSION**

Yürürlükteki ölü harf üstbilgisi yapısının yürürlükteki sürümü.

Bu alanın ilk değeri MQDLH\_VERSION\_1' dir.

### **Neden (MQUZE)**

Neden alanı, iletinin, özgün hedef kuyruğu yerine, ölü-mektup kuyruğuna yerleştirilmesinin nedenini belirtir.

Bu durum, iletinin, özgün hedef kuyruğu yerine, ölüme neden olan kuyruğun üzerine yerleştirilmesinin nedenini belirtir. Bu, MQFB\_ \* ya da MQRC\_ \* değerlerinden biri olmalıdır (örneğin, MQRC\_Q\_FULLL). Oluşabilecek ortak MQFB\_ \* değerlerine ilişkin ayrıntılar için "[MQMD-İleti tanımlayıcı](#)" sayfa 416 içindeki *Feedback* alanının açıklamasına bakın.

Değer MQFB\_IMS\_SON yoluyla MQFB\_IMS\_LAST aralığında olursa, gerçek IMS hata kodu, *Reason* alanının değerinden MQFB\_IMS\_ERROR çıkarılarak saptlanabilir.

Bazı MQFB\_ \* değerleri yalnızca bu alanda oluşur. Bunlar, hedef kuyruk kuyruğuna aktarılan havuz iletilerine, tetikleme iletilerine ya da iletim kuyruğu iletilerine ilişkin ilişki. Bu bilgiler şunlardır:

#### **MQFB\_APPL\_CANNOT\_BI\_STARTED ( X'00000109')**

Tetikleme iletisini işleyen bir uygulama, tetikleme iletisinin *ApplId* alanında adı belirtilen uygulamayı başlatamaz (bkz. "[MQTM-Tetikleyici iletisi](#)" sayfa 589).

z/OSüzerinde, CKTI CICS işlemi, tetikleme iletilerini işleyen bir uygulama örneğidir.

**MQFB\_APPL\_TYPE\_ERROR ( X'0000010B')**

Tetikleyici iletisinin *ApplType* alanı geçerli olmadığından, bir tetikleme iletisini işleyen bir uygulama, uygulamayı başlatamıyor (bkz. “MQTM-Tetikleyici iletisi” sayfa 589).

z/OSüzerinde, CKTI CICS işlemi, tetikleme iletilerini işleyen bir uygulama örneğidir.

**MQFB\_BIND\_OPEN\_CLUSRCVR\_DEL ( X'00000119')**

İleti, SYSTEM.CLUSTER.TRANSMIT.QUEUE , MQOO\_BIND\_ON\_OPEN seçeneğiyle açılan bir küme kuyruğu için tasarlanmış, ancak iletiyi hedef kuyruğa iletmek için kullanılacak uzak küme alıcı kanalı, ileti gönderilmeden önce silindi. MQOO\_BIND\_ON\_OPEN belirtildiğinden, yalnızca kuyruk açıldığında seçilen kanal iletiyi iletmek için kullanılabilir. Bu kanal artık kullanılamadığı için, ileti, ölü-mektup kuyruğuna yerleştirilir.

**MQFB\_NOT\_A\_REPOSITORY\_MSG ( X'00000118')**

İleti bir havuz iletisi değil.

**MQFB\_STOPPED\_BY\_CHAD\_EXIT ( X'00000115')**

İleti, kanal otomatik tanımlama çıkışı tarafından durduruldu.

**MQFB\_STOPPED\_BY\_MSG\_EXIT ( X'0000010D')**

İleti, kanal iletisi çıkışı tarafından durduruldu.

**MQFB\_TM\_ERROR ( X'0000010A')**

MQMD' deki *Format* alanı MQFMT\_TRIGGER değerini belirtiyor, ancak ileti geçerli bir MQTM yapısıyla başlamıyor. Örneğin, *StrucId* animsaticısı göz yakalama programı geçerli olmayabilir, *Version* tanınmamış olabilir ya da tetikleme iletisinin uzunluğu MQTM yapısını içermemek için yetersiz olabilir.

z/OSüzerinde, CKTI CICS işlemi, tetikleme iletilerini işleyen ve bu geribildirim kodunu oluşturabilecek bir uygulama örneğidir.

**MQFB\_XMIT\_Q\_MSG\_ERROR ( X'0000010F')**

İleti kanalı aracısı, iletim kuyruğunda bir iletinin doğru biçimde olmadığını saptadı. İleti kanalı aracısı, bu geribildirim kodunu kullanarak iletiyi ölü harf kuyruğuna yerleştiriyor.

Bir iletinin doğrudan iletim kuyruğuna konması yaygın bir nedendir, bu nedenle iletinin beklenen XQH üstbilgisi yok. Uygulama, MQXQH üstbilgisini oluşturmadığı sürece, iletiler uzak bir kuyruğun üzerinden iletim kuyruğuna konmalıdır.

Bu alanın ilk değeri MQRC\_NONE olur.

**DestQName (MQCHAR48)**

DestQName , ileti için özgün hedef olan ileti kuyruğunun adıdır.

Bu alanın uzunluğu MQ\_Q\_NAME\_LENGTH değeri tarafından verilir. Bu alanın başlangıç değeri, C ' de boş değer dizgisidir ve diğer programlama dillerindeki 48 boş karakter dizilimidir.

**DestQMgrAdı (MQCHAR48)**

DestQMgrAdı, ileti için özgün hedef olan kuyruk yöneticisinin adıdır.

Bu alanın uzunluğu MQ\_Q\_MGR\_NAME\_LENGTH değeri tarafından verilir. Bu alanın başlangıç değeri, C ' de boş değer dizgisidir ve diğer programlama dillerindeki 48 boş karakter dizilimidir.

**Kodlama (MQUZE)**

Kodlama, MQDLH yapısını izleyen verilerin sayısal kodlamasıdır (genellikle özgün iletiden gelen veriler); MQDLH yapısındaki sayısal veriler için geçerli değildir.

MQPUT ya da MQPUT1 çağrısında, uygulama bu alanı veri için uygun değere ayarlamalıdır.

Bu alanın başlangıç değeri 0' dir.

## **CodedCharSetId (MQHOMER)**

CodedCharSetId , MQDLH yapısıyla (genellikle özgün iletiden gelen veriler) akan verilerin karakter takımı tanıtıcısıdır; MQDLH yapısındaki karakter verileri için geçerli değildir.

MQPUT ya da MQPUT1 çağrısında, uygulama bu alanı veri için uygun değere ayarlamalıdır. Aşağıdaki özel değer kullanılabilir:

### **MQCCSI\_INHERIT**

Bu yapıyı izleyen verilerdeki karakter verileri, bu yapıyla aynı karakter kümesinde yer alıyor.

Kuyruk yöneticisi, bu değeri, iletide belirtilen yapının gerçek karakter kümesi tanıtıcısına göre değiştirir. Herhangi bir hata oluşmaması koşuluyla, MQCCSI\_INHERIT değeri MQGET çağrısıyla döndürülmez.

MQMD ' deki *PutApplType* alanının değeri MQAT\_BROKER ise, MQCCSI\_INHERIT kullanamazsınız.

Bu değer aşağıdaki ortamlarda desteklenir:

-  AIX
-  IBM i
-  Linux
-  Windows

ve bu sistemlere bağlı IBM MQ istemcileri için geçerli olur.

Bu alanın ilk değeri MQCCSI\_UNDEFED değerine sahip.

## **Biçim (MQCHAR8)**

Biçim, MQDLH yapısından sonra gelen verilerin biçim adıdır (genellikle özgün iletiden gelen veriler).

MQPUT ya da MQPUT1 çağrısında, uygulama bu alanı veri için uygun değere ayarlamalıdır. Bu alanın kodlanması için kurallar, MQMD ' de *Format* alanının kodlanması için bu kurallarla aynıdır.

Bu alanın uzunluğu MQ\_FORMAT\_LENGTH tarafından verilir. Bu alanın ilk değeri MQFMT\_NONE olur.

## **PutApplTipi (MQUZE)**

PutApplTip, iletiyi ölü harf (teslim edilemeyen ileti) kuyruğuna koyan uygulama tipidir.

Bu alan, ileti tanımlayıcısı MQMD ' deki *PutApplType* alanıyla aynı anlama sahiptir (ayrıntılar için [“MQMD-İleti tanımlayıcı” sayfa 416](#) konusuna bakın).

Kuyruk yöneticisi iletiyi ölü harf kuyruğuna yeniden yönlendirirse, *PutApplType* MQAT\_QMGR değerine sahiptir.

Bu alanın başlangıç değeri 0' dir.

## **PutApplAd (MQCHAR28)**

PutApplAd, iletiyi ölü harf (teslim edilemeyen ileti) kuyruğuna yerleştiren uygulamanın adıdır.

Adın biçimi *PutApplType* alanına bağlıdır. Biçim, serbest bırakılacak yayın düzeyini gösterebilir. [“MQMD-İleti tanımlayıcı” sayfa 416](#) içindeki *PutApplName* alanının açıklamasına bakın.

Kuyruk yöneticisi iletiyi ölü harf kuyruğuna yeniden yönlendirirse, *PutApplName* kuyruk yöneticisi adının ilk 28 karakterini içerir, gerekirse boşluklarla doldurulur.

Bu alanın uzunluğu MQ\_PUT\_APPL\_NAME\_LENGTH tarafından verilir. Bu alanın ilk değeri, C içindeki boş değerli dizedir ve diğer programlama dillerindeki 28 boşluk karakteridir.

### ***PutDate (MQCHAR8)***

PutDate , iletinin ölü harf (teslim edilemeyen ileti) kuyruğuna konduğu tarihtir.

Bu alanın kuyruk yöneticisi tarafından oluşturulduğu tarih için kullanılan biçim şöyledir:

- YYYYAAAGG

karakterlerin gösterdiği yer:

#### **YYYY**

yıl (dört sayısal rakam)

#### **mm**

yıl ay (01-12)

#### **DD**

ay günü (01-31 arası)

Greenwich Mean Time (GMT) is used for the *PutDate* and *PutTime* fields, subject to the system clock being set accurately to GMT.

Bu alanın uzunluğu MQ\_PUT\_DATE\_LENGTH tarafından verilir. Bu alanın ilk değeri, C içindeki boş değerli dizedir ve diğer programlama dillerinde sekiz boş karakter olur.

### ***PutTime (MQCHAR8)***

PutTime (PutTime), iletinin ölü harf (teslim edilemeyen ileti) kuyruğuna konduğu süredir.

Bu alanın kuyruk yöneticisi tarafından oluşturulduğu saat için kullanılan biçim şöyledir:

- HHMMSSSTH

karakterlerin gösterdiği yer:

#### **HH**

saat (00-23 arası)

#### **mm**

dakika (00-59)

#### **SS**

saniye (00-59; bkz. not)

#### **T**

saniyenin onda biri (0-9 arası)

#### **H**

saniyenin kamarası (0-9 arası)

**Not:** If the system clock is synchronized to an very accurate time standard, it is possible on rare occasions for 60 or 61 to be returned for the seconds in *PutTime*. Bu, küresel zaman standardına sıçrama saniyeleri eklenince oluşur.

Greenwich Mean Time (GMT) is used for the *PutDate* and *PutTime* fields, subject to the system clock being set accurately to GMT.

Bu alanın uzunluğu MQ\_PUT\_TIME\_LENGTH tarafından verilir. Bu alanın ilk değeri, C içindeki boş değerli dizedir ve diğer programlama dillerinde sekiz boş karakter olur.

## **MQDMHO-İleti işleme seçeneklerini sil**

**MQDMHO** yapısı, uygulamaların ileti tanıtıcılarının nasıl silineceğini denetleyen seçenekleri belirtmesine olanak sağlar. Yapı, **MQDLTMH** çağrısındaki bir giriş parametresidir.

## **Karakter kümesi ve kodlama**

**MQDMHO** içindeki veriler, uygulamanın karakter kümesinde ve uygulamanın kodlamasında olmalıdır (**MQENC\_NATIVE**).

## Alanlar

**Not:** Aşağıdaki tabloda, alanlar alfabetik olarak değil, kullanıma göre gruplanır. Alt konular aynı sırayı izler.

Çizelge 487. MQDMHO içindeki alanlar		
Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
StrucId (yapı tanıtıcısı)	MQDMHO_STRUC_ID	'DMHO'
Sürüm (yapı sürümü numarası)	MQDMHO_VERSION_1	1
Seçenekler (seçenekler)	MQDMHO_NONE	0

**Notlar:**

1. C programlama dilinde, makro değişkeni MQDMHO\_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlar için başlangıç değerleri sağlamak üzere aşağıdaki şekilde kullanılabilir:

```
MQDMHO MyDMHO = {MQDMHO_DEFAULT};
```

## Dil bildirimleri

MQDMHO için C bildirimini

```
typedef struct tagMQDMHO;
struct tagMQDMHO {
    MQCHAR4   StrucId;           /* Structure identifier */
    MQLONG    Version;          /* Structure version number */
    MQLONG    Options;          /* Options that control the action of MQDLTMH */
};
```

MQDMHO için COBOL bildirimini

```
** MQDMHO structure
10 MQDMHO.
** Structure identifier
15 MQDMHO-STRUCID PIC X(4).
** Structure version number
15 MQDMHO-VERSION PIC S9(9) BINARY.
** Options that control the action of MQDLTMH
15 MQDMHO-OPTIONS PIC S9(9) BINARY.
```

MQDMHO için PL/I bildirimini

```
dcl
1 MQDMHO based,
3 StrucId char(4), /* Structure identifier */
3 Version fixed bin(31), /* Structure version number */
3 Options fixed bin(31), /* Options that control the action of MQDLTMH */
```

MQDMHO için High Level Assembler (Üst Düzey Çevirici) bildirimini

```
MQDMHO DSECT
MQDMHO_STRUCID DS CL4 Structure identifier
MQDMHO_VERSION DS F Structure version number
MQDMHO_OPTIONS DS F Options that control the action of
* MQDLTMH
MQDMHO_LENGTH EQU *-MQDMHO
MQDMHO_AREA DS CL(MQDMHO_LENGTH)
```

## StrucId (MQCHAR4)



Bu, yapı tanıtıcısıdır; değer şöyle olmalıdır:

### **MQDMHO\_STRUC\_ID**

İleti tanıtıcısı seçenekleri yapısının silinmesine ilişkin tanıtıcı.

For the C programming language, the constant **MQDMHO\_STRUC\_ID\_ARRAY** is also defined; this has the same value as **MQDMHO\_STRUC\_ID**, but is an array of characters instead of a string.

Bu her zaman bir giriş alanıdır. Bu alanın başlangıç değeri **MQDMHO\_STRUC\_ID**' dir.

### **Sürüm (MQUZE)**

Bu, yapı sürüm numarasıdır; değer şu olmalıdır:

### **MQDMHO\_VERSION\_1**

Version-1 , ileti tanıtıcısı seçenekleri yapısını siler.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

### **MQDMHO\_CURRENT\_VERSION**

İleti silme işlemi seçenekleri yapısının yürürlükteki sürümü.

Bu her zaman bir giriş alanıdır. Bu alanın başlangıç değeri **MQDMHO\_VERSION\_1**' dir.

### **Seçenekler (MQUZE)**

Değer şu olmalıdır:

### **MQDMHO\_NONE**

Seçenek belirtilmedi.

Bu her zaman bir giriş alanıdır. Bu alanın başlangıç değeri **MQDMHO\_NONE**' dir.

## **MQDMPO-İleti silme özelliği seçenekleri**

MQDMPO yapısı, uygulamaların iletilerin özelliklerinin nasıl silineceğini denetleyen seçenekleri belirtmesine olanak sağlar. Yapı, MQDLTMP çağrısına ilişkin bir giriş değiştirgesidir.

### **Karakter kümesi ve kodlama**

MQDMPO içindeki veriler, uygulamanın karakter kümesinde ve uygulamanın kodlamasında (MQENC\_NATIVE) olmalıdır.

### **Alanlar**

**Not:** Aşağıdaki tabloda, alanlar alfabetik olarak değil, kullanıma göre gruplanır. Alt konular aynı sırayı izler.

<i>Çizelge 488. MQDMPO içindeki alanlar</i>		
<b>Alan adı ve açıklaması</b>	<b>Değişmezin adı</b>	<b>Değişmezin ilk değeri (varsa)</b>
<u>StrucId</u> (yapı tanıtıcısı)	MQDMPO_STRUC_ID	' DMPO '
<u>Sürüm</u> (yapı sürümü numarası)	MQDMPO_VERSION_1	1
<u>Seçenekler</u> (MQDMPO işlemi denetleyen seçenekler)	MQDLTMP işlemi denetleyen seçenekler	MQDMPO_NONE

Çizelge 488. MQDPMO içindeki alanlar (devamı var)

Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
<b>Notlar:</b>		
1. C programlama dilinde, makro değişkeniMQDPMO_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlar için başlangıç değerleri sağlamak üzere bunu aşağıdaki şekilde kullanın:		
<pre>MQDPMO MyDPMO = {MQDPMO_DEFAULT};</pre>		

## Dil bildirimleri

### MQDPMO için C bildirim

```
typedef struct tagMQDPMO MQDPMO;
struct tagMQDPMO {
    MQCHAR4  StrucId;          /* Structure identifier */
    MQLONG   Version;        /* Structure version number */
    MQLONG   Options;        /* Options that control the action of
                             MQDLTMP */
};
```

### MQDPMO için COBOL bildirim

```
** MQDPMO structure
   10 MQDPMO.
**   Structure identifier
   15 MQDPMO-STRUCID          PIC X(4).
**   Structure version number
   15 MQDPMO-VERSION        PIC S9(9) BINARY.
**   Options that control the action of MQDLTMP
   15 MQDPMO-OPTIONS        PIC S9(9) BINARY.
```

### MQDPMO için PL/I bildirim

```
Dcl
  1 MQDPMO based,
  3 StrucId      char(4),          /* Structure identifier */
  3 Version      fixed bin(31), /* Structure version number */
  3 Options      fixed bin(31), /* Options that control the action
                               of MQDLTMP */
```

### MQDPMO için High Level Assembler bildirim

```
MQDPMO          DSECT
MQDPMO_STRUCID  DS   CL4  Structure identifier
MQDPMO_VERSION  DS   F    Structure version number
MQDPMO_OPTIONS  DS   F    Options that control the
*                action of MQDLTMP
MQDPMO_LENGTH   EQU   *-MQDPMO
MQDPMO_AREA     DS   CL(MQDPMO_LENGTH)
```

## StrucId (MQCHAR4)

İleti özelliği seçenekleri yapısını sil- StrucId alanı

Bu, yapı tanıtıcısıdır. Değer şu olmalıdır:

### MQDPMO\_STRUCT\_ID

İleti özelliği seçenekleri yapısının silinmesine ilişkin tanıtıcı.

C programlama dili için, sabit MQDMPO\_STRUC\_ID\_ARRAY değişmezi de tanımlanır; bu değer MQDMPO\_STRUCT\_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisidir.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQDMPO\_STRUCT\_ID 'dir.

### **Sürüm (MQUZE)**

İleti özelliği seçenekleri yapısını sil-Sürüm alanı

Bu, yapı sürüm numarasıdır. Değer şu olmalıdır:

#### **MQDMPO\_VERSION\_1**

İleti özelliği seçenekleri yapısının silinmesine ilişkin sürüm numarası.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

#### **MQDMPO\_CURRENT\_VERSION**

İleti özelliği seçenekleri yapısının yürürlükteki sürümü.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQDMPO\_VERSION\_1' dir.

### **Seçenekler (MQUZE)**

İleti özelliği seçenekleri yapısını sil-Seçenekler alanı

**Konum seçenekleri:** Aşağıdaki seçenekler, özellik imlecine göre özelliğin görelisiyle ilgilidir.

#### **MQDMPO\_DEL\_FIRST**

Belirtilen adla eşleşen ilk özelliğin silinmesine neden olur.

#### **MQDMPO\_DEL\_PROP\_UNDER\_CURSOR**

Özellik imlecinin gösterdiği özelliği siler; bu, MQIMPO\_INQ\_FIRST ya da MQIMPO\_INQ\_Next seçeneğini kullanarak en son sorgulayan özeldir.

İleti tanıtıcısı yeniden kullanıldığında özellik imleci sıfırlanır. Ayrıca, bir MQGET çağrısındaki MQGMO yapısının *MsgHandle* alanında ya da MQPUT çağrısında MQPMO yapısıyla ilgili ileti tanıtıcısı belirtildiğinde de ilk durumuna getirilir.

Özellik imleci henüz oluşturulmadığında bu seçenek kullanılırsa, çağrı tamamlanma kodu MQCC\_FAILED ile başarısız olur ve MQRC\_PROPERTY\_NOT\_AVAISLE neden olur. Özellik imlecinin işaret ettiği özellik zaten silindiye, arama da MQCC\_FAILED tamamlanma kodu ile başarısız olur ve MQRC\_PROPERTY\_NOT\_AVAISSE neden olur.

İki seçenektan hiçbiri seçeneği gerekmiyorsa, aşağıdaki seçenek kullanılabilir:

#### **MQDMPO\_NONE**

Seçenek belirtilmedi.

Bu alan her zaman bir giriş alanıdır. Bu alanın ilk değeri MQDMPO\_DEL\_FIRST olarak değerlendirilmektedir.

### **MQEPH-Gömülü PCF üstbilgisi**

MQEPH yapısı, ileti programlanabilir bir komut biçimi (PCF) iletisi olduğunda iletide bulunan ek verileri tanımlar. *PCFHeader* alanı, bu yapıyı izleyen PCF değiştirgelerini tanımlar ve bu, PCF ileti verilerini diğer üstbilgilerle izlemenizi sağlar.

### **Biçim Adı**

MQFMT\_EMBEDDED\_PCF

### **Karakter kümesi ve kodlama**

MQEPH içindeki veriler, **CodedCharSetId** kuyruk yöneticisi özniteliği tarafından belirtilen karakter kümesinde ve MQENC\_NATIVE tarafından verilen yerel kuyruk yöneticisinin kodlamasında yer almalıdır.

MQEPH 'nin karakter kümesini ve kodlamasını MQMD' deki *CodedCharSetId* ve *Encoding* alanlarına (MQEPH yapısı ileti verilerinin başındadır) ya da MQEPH yapısından önce gelen üstbilgi yapısına (diğer tüm durumlar) ayarlayın.

## Kullanım

Komut sunucusuna ya da başka bir kuyruk yöneticisi PCF kabul eden sunucuya komut göndermek için MQEPH yapılarını kullanamazsınız.

Benzer şekilde, komut sunucusu ya da başka bir kuyruk yöneticisi PCF kabul eden sunucu, MQEPH yapıları içeren yanıtlar ya da olaylar oluşturmaz.

## Alanlar

**Not:** Aşağıdaki tabloda, alanlar alfabetik olarak değil, kullanıma göre gruplanır. Alt konular aynı sırayı izler.

Çizelge 489. MQEPH için MQEPH 'deki Alanlar		
Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
<u>StrucId</u> (yapı tanıtıcısı)	MQEPH_STRUC_ID	' EPH↵ '
<u>Sürüm</u> (yapı sürümü numarası)	MQEPH_VERSION_1	1
<u>StrucLength</u> (MQEPH yapısının uzunluğu ve bunu izleyen MQCFH ve değiştirge yapıları)	MQEPH_STRUC_LENGT H_FIXED	68
Kodlama (son PCF parametre yapısını izleyen verilerin sayısal kodlaması)	Yok	0
<u>CodedCharSetId</u> (son PCF parametre yapısını izleyen verilerin karakter kümesi tanıtıcısı)	MQCCSI_AYRILMAMIŞ	0
<u>Biçim</u> (son PCF parametre yapısını izleyen verilerin biçim adı)	MQFMT_NONE	Boşluklar
<u>İşaretler</u> (işaretler)	MQEPH_NONE	0
<u>PCFHeader</u> (programlanabilir komut biçimi (PCF) üstbilgisi)	Çizelge 490 sayfa 364 içinde tanımlandığı şekilde adlar ve değerler	0
<p><b>Notlar:</b></p> <ol style="list-style-type: none"> <li>↵ simgesi, tek bir boş karakteri temsil eder.</li> <li>C programlama dilinde, makro değişkeniMQEPH_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlar için başlangıç değerleri sağlamak üzere bunu aşağıdaki şekilde kullanın:</li> </ol> <pre>MQEPH MyEPH = {MQEPH_DEFAULT};</pre>		

## Dil bildirimleri

MQEPH için C bildirimi

```
typedef struct tagMQEPH MQEPH;
struct tagMQDH {
    MQCHAR4 StrucId;          /* Structure identifier */
    MQLONG  Version;         /* Structure version number */
    MQLONG  StrucLength;     /* Total length of MQEPH including the MQCFH
                             and parameter structures that follow it */
    MQLONG  Encoding;        /* Numeric encoding of data that follows last
                             PCF parameter structure */
};
```

```

MQLONG   CodedCharSetId; /* Character set identifier of data that
                          follows last PCF parameter structure */
MQCHAR8  Format;          /* Format name of data that follows last PCF
                          parameter structure */
MQLONG   Flags;          /* Flags */
MQCFH    PCFHeader;     /* Programmable command format header */
};

```

## MQEPH için COBOL bildirimi

```

** MQEPH structure
10 MQEPH.
**   Structure identifier
15 MQEPH-STRUCID      PIC X(4).
**   Structure version number
15 MQEPH-VERSION     PIC S9(9) BINARY.
**   Total length of MQEPH structure including the MQCFH
**   and parameter structures that follow it
15 MQEPH-STRUCLNGTH  PIC S9(9) BINARY.
**   Numeric encoding of data that follows last
**   PCF structure
15 MQEPH-ENCODING    PIC S9(9) BINARY.
**   Character set identifier of data that
**   follows last PCF parameter structure
15 MQEPH-CODEDCHARSETID PIC S9(9) BINARY.
**   Format name of data that follows last PCF
**   parameter structure
15 MQEPH-FORMAT      PIC X(8).
**   Flags
15 MQEPH-FLAGS       PIC S9(9) BINARY.
**   Programmable command format header
15 MQEPH-PCFHEADER.
**   Structure type
20 MQEPH-PCFHEADER-TYPE      PIC S9(9) BINARY.
**   Structure length
20 MQEPH-PCFHEADER-STRUCLNGTH PIC S9(9) BINARY.
**   Structure version number
20 MQEPH-PCFHEADER-VERSION   PIC S9(9) BINARY.
**   Command identifier
20 MQEPH-PCFHEADER-COMMAND   PIC S9(9) BINARY.
**   Message sequence number
20 MQEPH-PCFHEADER-MSGSEQNUMBER PIC S9(9) BINARY.
**   Control options
20 MQEPH-PCFHEADER-CONTROL   PIC S9(9) BINARY.
**   Completion code
20 MQEPH-PCFHEADER-COMPCODE   PIC S9(9) BINARY.
**   Reason code qualifying completion code
20 MQEPH-PCFHEADER-REASON     PIC S9(9) BINARY.
**   Count of parameter structures
20 MQEPH-PCFHEADER-PARAMETERCOUNT PIC S9(9) BINARY.

```

## MQEPH için PL/I bildirimi

```

dcl
1 MQEPH based,
3 StrucId      char(4), /* Structure identifier */
3 Version      fixed bin(31), /* Structure version number */
3 StrucLength  fixed bin(31), /* Total Length of MQEPH including the
                              MQCFH and parameter structures that
                              follow it
3 Encoding     fixed bin(31), /* Numeric encoding of data that follows
                              last PCF parameter structure
3 CodedCharSetId fixed bin(31), /* Character set identifier of data that
                              follows last PCF parameter structure
3 Format        char(8), /* Format name of data that follows last
                              PCF parameter structure */
3 Flags        fixed bin(31), /* Flags */
3 PCFHeader, /* Programmable command format header
5 Type         fixed bin(31), /* Structure type */
5 StrucLength  fixed bin(31), /* Structure length */
5 Version      fixed bin(31), /* Structure version number */
5 Command      fixed bin(31), /* Command identifier */
5 MsgseqNumber fixed bin(31), /* Message sequence number */
5 Control      fixed bin(31), /* Control options */
5 CompCode     fixed bin(31), /* Completion code */

```

```

5 Reason          fixed bin(31), /* Reason code qualifying completion code */
5 ParameterCount  fixed bin(31); /* Count of parameter structures */

```

## MQEPH için High Level Assembler (Üst Düzey Çevirici) bildirimi

```

MQEPH                                DSECT
MQEPH_STRUCID                        DS    CL4  Structure identifier
MQEPH_VERSION                        DS    F    Structure version number
MQEPH_STRUCLNGTH                     DS    F    Total length of MQEPH including the
*                                     MQCFH and parameter structures that
*                                     follow it
MQEPH_ENCODING                        DS    F    Numeric encoding of data that follows
*                                     last PCF parameter structure
MQEPH_CODEDCHARSETID                 DS    F    Character set identifier of data that
*                                     follows last PCF parameter structure
MQEPH_FORMAT                          DS    CL8  Format name of data that follows last
*                                     PCF parameter structure
MQEPH_FLAGS                           DS    F    Flags
MQEPH_PCFHEADER                       DS    0F    Force fullword alignment
MQEPH_PCFHEADER_TYPE                  DS    F    Structure type
MQEPH_PCFHEADER_STRUCLNGTH            DS    F    Structure length
MQEPH_PCFHEADER_VERSION               DS    F    Structure version number
MQEPH_PCFHEADER_COMMAND               DS    F    Command identifier
MQEPH_PCFHEADER_MSGSEQNUMBER          DS    F    Structure length
MQEPH_PCFHEADER_CONTROL               DS    F    Control options
MQEPH_PCFHEADER_COMPCODE               DS    F    Completion code
MQEPH_PCFHEADER_REASON                 DS    F    Reason code qualifying completion code
MQEPH_PCFHEADER_PARAMETER COUNT       DS    F    Count of parameter structures
MQEPH_PCFHEADER_LENGTH                 EQU   *-MQEPH_PCFHEADER
*                                     ORG   MQEPH_PCFHEADER
MQEPH_PCFHEADER_AREA                  DS    CL(MQEPH_PCFHEADER_LENGTH)
*
MQEPH_LENGTH                          EQU   *-MQEPH
*                                     ORG   MQEPH
MQEPH_AREA                             DS    CL(MQEPH_LENGTH)

```

## MQEPH için Visual Basic bildirimi

```

Type MQEPH
  StrucId      As String*4 'Structure identifier'
  Version     As Long      'Structure version number'
  StrucLength  As Long      'Total length of MQEPH structure including the MQCFH'
  Encoding     As Long      'and parameter structures that follow it'
  CodedCharSetId As Long    'Numeric encoding of data that follows last'
  Format       As String*8  'PCF parameter structure'
  Flags       As Long      'Character set identifier of data that'
  PCFHeader   As MQCFH     'follows last PCF parameter structure'
End Type
Global MQEPH_DEFAULT As MQEPH

```

### **StrucId (MQCHAR4)**

Değer şu olmalıdır:

#### **MQEPH\_STRUCT\_ID**

Dağıtım üstbilgisi yapısına ilişkin tanıtıcı.

C programlama dili için, sabit MQPH\_STRUCT\_ID\_ARRAY değişmezi de tanımlanır; bu, MQDH\_STRUCT\_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisidir.

Bu alanın ilk değeri MQEF\_STRUCT\_ID 'dir.

### **Sürüm (MQUZE)**

Değer şu olmalıdır:

#### **MQEPH\_VERSION\_1**

Gömülü PCF üstbilgi yapısına ilişkin sürüm numarası.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

### **MQCFH\_VERSION\_3**

Yerleşik PCF üstbilgi yapısının yürürlükteki sürümü.

Bu alanın ilk değeri MQEPH\_VERSION\_1' dir.

### **StrucLength (MQUZE)**

Bu, bir sonraki üstbilgi yapısından önceki verilerin miktarıdır. Bu ürün aşağıdakileri içerir:

- MQEPH üstbilgisinin uzunluğu
- Üstbilgiyi izleyen tüm PCF parametrelerinin uzunluğu
- Bu değiştirgeleri izleyen boş doldurma işlemi

StrucLength (Yapısal Uzunluğu), 4 'un katı olmalıdır.

Yapının değişmez uzunluktaki kısmı MQPH\_STRUCT\_LENGTH\_FIXED ile tanımlanır.

Bu alanın ilk değeri 68 'dir.

### **Kodlama (MQUZE)**

Bu, MQEPH yapısını ve ilişkili PCF parametrelerini izleyen verilerin sayısal kodlamasıdır; MQEPH yapısındaki karakter verileri için geçerli değildir.

Bu alanın ilk değeri 0 'tır.

### **CodedCharSetId (MQHOMER)**

Bu, MQEPH yapısını ve ilişkili PCF parametrelerini izleyen verilerin karakter takımı tanıtcısıdır; MQEPH yapısındaki karakter verileri için geçerli değildir.

Bu alanın ilk değeri MQCCSI\_UNDEFED değerine sahip.

### **Biçim (MQCHAR8)**

Bu ad, MQEPH yapısını ve ilişkili PCF değiştirgelerini izleyen verilerin biçim adıdır.

Bu alanın ilk değeri MQFMT\_NONE olur.

### **İşaretler (MQUZE)**

Aşağıdaki değerler kullanılabilir:

#### **MQEPH\_NONE**

Herhangi bir işaret belirtilmedi. MQEPH\_NONE, program belgelerine yardımcı olmak için tanımlıdır. Bu değişmezin başka biriyle birlikte kullanılması amaçlanmaz, ancak değeri sıfır olduğu için bu tür bir kullanım saptanamaz.

#### **MQEF\_CCSID\_EMBEDDED**

Karakter verilerini içeren parametrelerin karakter kümesi, her yapıdaki CodedCharSetId alanında tek tek belirtilir. StrucId ve Biçim alanlarının karakter kümesi, MQPH yapısından önce gelen üstbilgi yapısındaki CodedCharSetId alanı tarafından ya da MQEPH, iletinin başlangıcındaki MQMD ' deki CodedCharSetId alanı tarafından tanımlanır.

Bu alanın ilk değeri MQEPH\_NONE olur.

### **PCFÜstbilgisi (MQCFH)**

Bu, PCF (programlanabilir komut biçimi) üstbilgisidir ve MQEPH yapısını izleyen PCF parametrelerini tanımlıyor. Bu, PCF ileti verilerini diğer üstbilgilerle izlemenize olanak sağlar.

PCF üstbilgisi başlangıçta aşağıdaki değerlerle tanımlanır:

Çizelge 490. MQCFH 'deki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
Type	MQCFT_NONE	0
StrucLength	MQCFH_STRUC_LENGTH	36
Version	MQCFH_VERSION_3	3
StrucLength	Yok	0
Command	MQCMD_NONE	0
MsgSeqNumber	Yok	1
Control	MQCFC_SON	1
CompCode	MQCC_OK	0
Reason	MQRC_NONE	0
ParameterCount	Yok	0

The application must change the Type from MQCFT\_NONE to a valid structure type for the use it is making of the embedded PCF header.

## MQGMO-Get-message seçenekleri

MQGMO yapısı, uygulamanın iletilerin kuyruklardan nasıl kaldırılacağını denetlemesini sağlar. Yapı, MQGET çağırısına ilişkin bir giriş/çıkış değiştirgesidir.

## Sürüm

MQGMO ' nun yürürlükteki sürümü: MQGMO\_VERSION\_4. Belirli alanlar yalnızca MQGMO ' nun belirli sürümlerinde kullanılabilir. Birden çok ortam arasında uygulamaları bağlamanız gerekiyorsa, MQGMO sürümünün tüm ortamlarda tutarlı olduğundan emin olmanız gerekir. Yalnızca yapının belirli sürümlerinde var olan alanlar, "[MQGMO-Get-message seçenekleri](#)" sayfa 364 içinde ve alan açıklamalarında olduğu gibi tanımlanır.

Desteklenen programlama dilleri için sağlanan üstbilgi, COPY ve INCLUDE dosyaları, ortam tarafından desteklenen ancak *Version* alanının ilk değeri MQGMO\_VERSION\_1olarak ayarlanmış MQGMO ' nun en son sürümünü içerir. version-1 yapısında var olmayan alanları kullanmak için, *Version* alanını gerekli sürümün sürüm numarasına ayarlayın.

## Karakter kümesi ve kodlama

MQGMO ' daki veriler, **CodedCharSetId** kuyruk yöneticisi özneliği tarafından belirtilen karakter kümesinde ve MQENC\_NATIVE tarafından verilen yerel kuyruk yöneticisinin kodlamasında yer almalıdır. Ancak, uygulama bir MQ MQI istemcisi olarak çalışıyorsa, yapı istemcinin karakter kümesinde ve kodlamasında olmalıdır.

## Alanlar

**Not:** Aşağıdaki tabloda, alanlar alfabetik olarak değil, kullanıma göre gruplanır. Alt konular aynı sırayı izler.

Çizelge 491. MQGMO için MQGMO içindeki alanlar		
Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
StrucId (yapı tanıtıcısı)	MQGMO_STRUC_ID	'GMO-'
Sürüm (yapı sürümü numarası)	MQGMO_VERSION_1	1



Çizelge 491. MQGMO için MQGMO içindeki alanlar (devamı var)

Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
MQGMO-Options alanı (MQGET işlemini denetleyen seçenekler)	MQGMO_NO_WAIT	0
WaitInterval (bekleme aralığı)	Yok	0
Signal1 (sinyal)	Yok	z/OS üzerinde boş değerli gösterge; 0 tersi durumda
Signal2 (sinyal tanıtıcısı)	Yok	0
ResolvedQName (hedef kuyruğun çözülmüş adı)	Yok	Boş değerli dizilim ya da boşluklar
<b>Not:</b> <i>Version</i> , MQGMO_VERSION_2' den küçükse kalan alanlar yoksayılr.		
MatchOptions (MQGET için kullanılan seçim ölçütlerini denetleyen seçenekler)	MQMO_MATCH_MSG_ID + MQMO_MATCH_CORREL_ID	3
GroupStatus (alınan iletinin bir grupta olup olmadığını gösteren işaret)	MQGS_NOT_IN_GROUP	' - '
SegmentStatus (alınan iletinin bir mantıksal iletinin parçası olup olmadığını gösteren işaret)	MQSS_NOT_A_SEGMENTS	' - '
Bölümleme (alınan ileti için daha fazla bölümlenmeye izin verilip verilmediğini gösteren işaret)	MQSEG_ENGELLEME	' - '
Reserved1 (ayrılmış)	Yok	' - '
<b>Not:</b> <i>Version</i> , MQGMO_VERSION_3' den küçükse kalan alanlar yoksayılr.		
MsgToken (ileti simgesi)	MQMTOK_NONE	Boş Değerler
ReturnedLength (döndürülen ileti verilerinin bayt cinsinden uzunluğu)	MQRL_UNDEFINED	-1
<b>Not:</b> <i>Version</i> , MQGMO_VERSION_4' den küçükse kalan alanlar yoksayılr.		
Reserved2 (ayrılmış)	Yok	' - '
MsgHandle (kuyruktan alınmakta olan iletinin özellikleriyle doldurulacak bir ileti tanıtıcısı)	MQHM_NONE	0
<p><b>Notlar:</b></p> <ol style="list-style-type: none"> <li>1. - simgesi, tek bir boş karakteri temsil eder.</li> <li>2. Boş değerli dizilim ya da boşluk değeri, C dizgisindeki boş dizgiyi ve diğer programlama dillerindeki boş karakterleri gösterir.</li> <li>3. C programlama dilinde, makro değişkeniMQGMO_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlar için başlangıç değerleri sağlamak üzere aşağıdaki şekilde kullanılabilir:</li> </ol> <pre>MQGMO MyGMO = {MQGMO_DEFAULT};</pre>		

## Dil bildirimleri

### MQGMO için C bildirimi

```
typedef struct tagMQGMO MQGMO;
struct tagMQGMO {
    MQCHAR4   StrucId;           /* Structure identifier */
    MQLONG    Version;          /* Structure version number */
    MQLONG    Options;          /* Options that control the action of */
                                /* MQGET */
    MQLONG    WaitInterval;     /* Wait interval */
    MQLONG    Signal1;          /* Signal */
    MQLONG    Signal2;          /* Signal identifier */
    MQCHAR48  ResolvedQName;    /* Resolved name of destination queue */
    /* Ver:1 */
    MQLONG    MatchOptions;     /* Options controlling selection */
                                /* criteria used for MQGET */
    MQCHAR    GroupStatus;      /* Flag indicating whether message */
                                /* retrieved is in a group */
    MQCHAR    SegmentStatus;    /* Flag indicating whether message */
                                /* retrieved is a segment of a logical */
                                /* message */
    MQCHAR    Segmentation;     /* Flag indicating whether further */
                                /* segmentation is allowed for the */
                                /* message retrieved */
    MQCHAR    Reserved1;       /* Reserved */
    /* Ver:2 */
    MQBYTE16  MsgToken;         /* Message token */
    MQLONG    ReturnedLength;   /* Length of message data returned */
                                /* (bytes) */
    /* Ver:3 */
    MQLONG    Reserved2;       /* Reserved */
    MQHMSG    MsgHandle;        /* Message handle */
    /* Ver:4 */
};
```

**Not:** z/OS sistemlerinde *Signal1* alanı *PMQLONG* olarak bildirilir.

### MQGMO için COBOL bildirimi

```
** MQGMO structure
10 MQGMO.
** Structure identifier
15 MQGMO-STRUCID PIC X(4).
** Structure version number
15 MQGMO-VERSION PIC S9(9) BINARY.
** Options that control the action of MQGET
15 MQGMO-OPTIONS PIC S9(9) BINARY.
** Wait interval
15 MQGMO-WAITINTERVAL PIC S9(9) BINARY.
** Signal
15 MQGMO-SIGNAL1 PIC S9(9) BINARY.
** Signal identifier
15 MQGMO-SIGNAL2 PIC S9(9) BINARY.
** Resolved name of destination queue
15 MQGMO-RESOLVEDQNAME PIC X(48).
** Options controlling selection criteria used for MQGET
15 MQGMO-MATCHOPTIONS PIC S9(9) BINARY.
** Flag indicating whether message retrieved is in a group
15 MQGMO-GROUPSTATUS PIC X.
** Flag indicating whether message retrieved is a segment of a
logical message
15 MQGMO-SEGMENTSTATUS PIC X.
** Flag indicating whether further segmentation is allowed for the
message retrieved
15 MQGMO-SEGMENTATION PIC X.
** Reserved
15 MQGMO-RESERVED1 PIC X.
** Message token
15 MQGMO-MSGTOKEN PIC X(16).
** Length of message data returned (bytes)
15 MQGMO-RETURNEDLENGTH PIC S9(9) BINARY.
** Reserved
15 MQGMO-RESERVED2 PIC S9(9) BINARY.
** Message handle
15 MQGMO-MSGHANDLE PIC S9(18) BINARY.
```

**Not:** z/OSsistemlerinde *Signal1* alanı POINTERolarak bildirilir.

MQGMO için PL/I bildirimi

```
dcl
  1 MQGMO based,
  3 StrucId      char(4),          /* Structure identifier */
  3 Version      fixed bin(31),   /* Structure version number */
  3 Options      fixed bin(31),   /* Options that control the action of
                                MQGET */
  3 WaitInterval fixed bin(31),   /* Wait interval */
  3 Signal1      fixed bin(31),   /* Signal */
  3 Signal2      fixed bin(31),   /* Signal identifier */
  3 ResolvedQName char(48),       /* Resolved name of destination
                                queue */
  3 MatchOptions fixed bin(31),   /* Options controlling selection
                                criteria used for MQGET */
  3 GroupStatus  char(1),         /* Flag indicating whether message
                                retrieved is in a group */
  3 SegmentStatus char(1),       /* Flag indicating whether message
                                retrieved is a segment of a logical
                                message */
  3 Segmentation char(1),       /* Flag indicating whether further
                                segmentation is allowed for the
                                message retrieved */
  3 Reserved1    char(1),         /* Reserved */
  3 MsgToken     char(16),        /* Message token */
  3 ReturnedLength fixed bin(31); /* Length of message data returned
                                (bytes) */
  3 Reserved2    fixed bin(31); /* Reserved */
  3 MsgHandle     fixed bin(63); /* Message handle */
```

**Not:** z/OSsistemlerinde *Signal1* alanı pointerolarak bildirilir.

MQGMO için High Level Assembler bildirimi

```
MQGMO          DSECT
MQGMO_STRUCID  DS    CL4  Structure identifier
MQGMO_VERSION  DS    F    Structure version number
MQGMO_OPTIONS  DS    F    Options that control the action of
*              MQGET
MQGMO_WAITINTERVAL DS  F    Wait interval
MQGMO_SIGNAL1  DS    F    Signal
MQGMO_SIGNAL2  DS    F    Signal identifier
MQGMO_RESOLVEDQNAME DS CL48 Resolved name of destination queue
MQGMO_MATCHOPTIONS DS  F    Options controlling selection criteria
*              used for MQGET
MQGMO_GROUPSTATUS DS  CL1  Flag indicating whether message
*              retrieved is in a group
MQGMO_SEGMENTSTATUS DS CL1  Flag indicating whether message
*              retrieved is a segment of a logical
*              message
MQGMO_SEGMENTATION DS  CL1  Flag indicating whether further
*              segmentation is allowed for the message
*              retrieved
MQGMO_RESERVED1 DS    CL1  Reserved
MQGMO_MSGTOKEN  DS    XL16 Message token
MQGMO_RETURNEDLENGTH DS  F    Length of message data returned (bytes)
MQGMO_RESERVED2 DS    F    Reserved
MQGMO_MSGHANDLE DS    D    Message handle
MQGMO_LENGTH    EQU    *-MQGMO
                ORG    MQGMO
MQGMO_AREA      DS    CL(MQGMO_LENGTH)
```

MQGMO için High Level Assembler bildirimi

```
Type MQGMO
  StrucId      As String*4  'Structure identifier'
  Version      As Long      'Structure version number'
  Options      As Long      'Options that control the action of MQGET'
  WaitInterval As Long      'Wait interval'
  Signal1      As Long      'Signal'
  Signal2      As Long      'Signal identifier'
  ResolvedQName As String*48 'Resolved name of destination queue'
  MatchOptions As Long      'Options controlling selection criteria'
```

GroupStatus	As String*1	'used for MQGET' 'Flag indicating whether message' 'retrieved is in a group'
SegmentStatus	As String*1	'Flag indicating whether message' 'retrieved is a segment of a logical' 'message'
Segmentation	As String*1	'Flag indicating whether further' 'segmentation is allowed for the message' 'retrieved'
Reserved1	As String*1	'Reserved'
MsgToken	As MQBYTE16	'Message token'
ReturnedLength	As Long	'Length of message data returned (bytes)'
End Type		

## PROPCTL kanal seçenekleri MQGMO

Daha önceki bir IBM MQ sürümünden bir ortak kuyruk yöneticisine IBM MQ 9.2 kuyruk yöneticisinden gönderilen bir iletiye hangi ileti özelliklerinin dahil edildiğini denetlemek için **PROPCTL** kanal özneliğini kullanın.

Çizelge 492. Kanal iletişi özellik özneliği ayarları

PROPCTL	Açıklama
TÜMÜ	<p>Daha önceki bir sürümden iş ortağı kuyruk yöneticisine bağlı uygulamalar, bir IBM MQ 9.2 uygulaması tarafından iletiye yerleştirilen özellikleri işleyebiliyorsa bu seçeneği kullanın.</p> <p>Tüm özellikler, MQRFH2 içine yerleştirilen ad-değer çiftlerine ek olarak ortak kuyruk yöneticisine gönderilir.</p> <p>İki uygulama tasarımı sorununu göz önünde bulundurmanız gerekir:</p> <ol style="list-style-type: none"> <li>1. Ortak kuyruk yöneticisine bağlı bir uygulama, IBM MQ 9.2 kuyruk yöneticisinde oluşturulan MQRFH2 üstbilgilerini içeren iletileri işleyebilmelidir.</li> <li>2. Ortak kuyruk yöneticisine bağlı uygulamanın, MQPD_SUPPORT_REQUIRED ile doğru şekilde işaretlenmiş yeni ileti özelliklerini işleme gerekir.</li> </ol> <p>ALL kanal seçeneği ayarlandığında, JMS uygulamaları kanalı kullanarak IBM MQ 9.2 ile daha önceki bir sürüm arasında birlikte çalışabilir. İleti özelliklerini kullanan yeni IBM MQ 9.2 uygulamaları, önceki sürüm uygulamasının MQRFH2 üstbilgilerini nasıl işlediğine bağlı olarak daha önceki bir sürümdeki uygulamalarla birlikte çalışabilir.</p>

Çizelge 492. Kanal iletisi özellik özneteliği ayarları (devamı var)

PROPCTL	Açıklama
DERLEME	<p>Bazı durumlarda daha önceki bir sürüm iş ortağı kuyruk yöneticisine bağlı uygulamalara ileti özellikleri göndermek için bu seçeneği kullanın, ancak tümünü göndermeyin. İleti özellikleri yalnızca iki koşul karşılandığında gönderilir:</p> <ol style="list-style-type: none"><li>Hiçbir özellik, ileti özelliği işlenmesi gerektiren olarak işaretlenmemelidir.</li><li>İleti özelliklerinden en az biri "ayrılmış" bir klasörde olmalıdır; bkz. <a href="#">Not</a>.</li></ol> <p>COMPAT kanal seçeneği ayarlandığında, JMS uygulamaları kanalı kullanarak IBM MQ 9.2 ile önceki bir sürüm arasında birlikte çalışabilir.</p> <p>Kanal, ileti özelliklerini kullanan her uygulama tarafından kullanılamaz, yalnızca ayrılmış klasörleri kullanan uygulamalar tarafından kullanılabilir. İletinin ya da özelliğin gönderilip gönderilmediğine ilişkin kurallar şunlardır:</p> <ol style="list-style-type: none"><li>İletinin özellikleri varsa, ancak özelliklerin hiçbiri "ayrılmış" bir klasörle ilişkilendirilmediyse, ileti özellikleri gönderilmez.</li><li>"Ayrılmış" bir özellik klasöründe herhangi bir ileti özelliği yaratıldıysa, iletiyle ilişkili tüm ileti özellikleri gönderilir. Ancak:<ol style="list-style-type: none"><li>İleti özelliklerinden herhangi biri gerekli destek olarak işaretliyse (MQPD_SUPPORT_REQUIRED ya da MQPD_SUPPORT_REQUIRED_IF_LOCAL), iletinin tamamı reddedilir. Rapor seçeneklerinin değerine göre geri döndürülür, atılır ya da teslim edilmeyen iletiler kuyruğuna gönderilir.</li><li>Herhangi bir ileti özelliği gerekli destek olarak işaretlenmezse, tek bir özellik gönderilmeyebilir. İleti özelliği tanımlayıcı alanlarından herhangi biri varsayılan olmayan değerlere ayarlanırsa, tek tek özellik gönderilmez. İleti gönderilmeye devam ediyor. Varsayılan olmayan bir özellik tanımlayıcı alan değeri örneği: MQPD_USER_CONTEXT.</li></ol></li></ol> <p><b>Not:</b> "Ayrılmış" klasör adları mcd . , jms . , us1 . ya da mqext . ile başlar. Bu klasörler, JMS arabirimini kullanan uygulamalar için oluşturulur. IBM MQ 9.2 ' de bu klasörlere yerleştirilen ad-değer çiftleri ileti özellikleri olarak kabul edilir.</p> <p>İleti özellikleri, MQRFH2 üstbilgisine yerleştirilen ad-değer çiftlerine ek olarak bir MQRFH2 üstbilgisinde gönderilir. MQRFH2 üstbilgisine yerleştirilen ad-değer çiftleri, ileti reddedilmediği sürece gönderilir.</p>
YOK	<p>Daha önceki bir sürüm iş ortağı kuyruk yöneticisine bağlı uygulamalara ileti özelliklerinin gönderilmesini önlemek için bu seçeneği kullanın. Ad-değer çiftlerini ve ileti özelliklerini içeren bir MQRFH2 yine de gönderilir, ancak yalnızca ad-değer çiftleriyle birlikte gönderilir.</p> <p>NONE kanal seçeneği ayarlandığında, bir JMS iletisi, herhangi bir JMS ileti özelliği olmayan bir JMSTextMessage ya da JMSByteMessage olarak gönderilir. Önceki bir sürüm uygulamasının IBM MQ 9.2 uygulamasında ayarlanan tüm özellikleri yoksaması mümkünse, bu uygulamayla birlikte çalışabilir.</p>

### PROPCTL MQGMO için kuyruk seçenekleri

İleti özelliklerinin, herhangi bir **MQGMO** ileti özelliği seçeneği ayarlanmadan **MQGET** çağıran bir uygulamaya nasıl döndürüleceğini denetlemek için **PROPCTL** kuyruk özneteliğini kullanın.

Çizelge 493. Kuyruk iletisi özellik özneliği ayarları

PROPCTL	Açıklama
TÜMÜ	<p>Aynı kuyruktan ileti okuyan farklı uygulamaların iletiyi farklı şekillerde işleyebilmesi için ALL seçeneğini kullanın.</p> <ul style="list-style-type: none"><li>• Daha önceki bir sürümden değiştirilmeden geçirilen bir uygulama, MQRFH2 ögesini doğrudan okumaya devam edebilir. MQRFH2 üstbilgisinde özelliklere doğrudan erişilebilir.</li></ul> <p>Yeni özellikleri ve yeni özellik özneliklerini işlemek için uygulamayı değiştirmeniz gerekir. Uygulama, MQRFH2 üstbilgilerinin yerleşim düzenindeki ve sayısındaki değişikliklerden etkilenebilir. Bazı klasör öznelikleri kaldırılabilir ya da IBM MQ , önceki bir sürümde yoksaydığı MQRFH2 üstbilgisinin yerleşim düzeninde bir hata bildiriyor olabilir.</p> <ul style="list-style-type: none"><li>• Yeni ya da değiştirilen bir uygulama, ileti özelliklerini sorgulamak ve MQRFH2 üstbilgisinde ad-değer çiftlerini doğrudan okumak için MQI ileti özelliğini kullanabilir.</li></ul> <p>İletideki tüm özellikler uygulamaya döndürülür.</p> <ul style="list-style-type: none"><li>• Uygulama bir ileti tanıtıcısı oluşturmak için MQCRTMH ' i çağırırsa, ileti özelliklerini MQINQMPkullanarak sorgulamalıdır. İleti özellikleri olmayan ad-değer çiftleri, ileti özelliklerinin soyulduğu MQRFH2içinde kalır.</li><li>• Uygulama bir ileti tanıtıcısı oluşturmazsa, tüm ileti özellikleri ve ad-değer çiftleri MQRFH2içinde kalır.</li></ul> <p>ALL bu etkiyi yalnızca, alan uygulama bir MQGMO_PROPERTIES seçeneği ayarlı değilse ya da MQGMO_PROPERTIES_AS_Q_DEFolarak ayarlıysa gösterir.</p>

Çizelge 493. Kuyruk iletisi özellik özneliği ayarları (devamı var)

PROPCTL	Açıklama
COMPAT (varsayılan)	<p>COMPAT varsayılan seçenektir. GMO_PROPERTIES_* ayarlanmazsa, önceki bir sürümdeki değiştirilmemiş bir uygulamada olduğu gibi, COMPAT varsayılr. Belirtik olarak MQRFH2 yaratmayan daha önceki bir sürüm uygulaması olan COMPAT seçeneği varsayılan olarak IBM MQ 9.2 üzerinde değişiklik yapılmadan çalışır.</p> <p>JMS iletilerini okumak için daha önceki bir sürüm uygulaması MQI uygulaması yazdıysanız bu seçeneği kullanın.</p> <ul style="list-style-type: none"><li>MQRFH2 üstbilgisinde saklanan JMS özellikleri, mcd. , jms. , us1. ya da mqextile başlayan adlara sahip klasörlerdeki MQRFH2 üstbilgisinde uygulamaya döndürülür.</li><li>İletide JMS klasörleri varsa ve bir IBM MQ 9.2 uygulaması iletiye yeni özellik klasörleri eklerse, bu özellikler MQRFH2 içinde de döndürülür. Sonuç olarak, yeni özellikleri ve yeni özellik özneliklerini işlemek için uygulamayı değiştirmeniz gerekir. Değiştirilmemiş bir uygulama, MQRFH2 üstbilgilerinin yerleşim düzenindeki ve sayısındaki değişikliklerden etkilenebilir. Bazı klasör öznelikleri kaldırılmış olabilir ya da IBM MQ , önceki bir sürümde yoksaydığı MQRFH2 üstbilgisinin yerleşim düzeninde hatalar bulabilir.</li></ul> <p><b>Not:</b> Bu senaryoda, uygulamanın önceki bir sürümüne ya da IBM MQ 9.2 kuyruk yöneticisine bağlı olması fark etmez, uygulamanın davranışı aynıdır. Kanal <b>PROPCTL</b> özneliği COMPAT ya da ALL olarak ayarlanırsa, iletide önceki sürüm ortak kuyruk yöneticisine yeni ileti özellikleri gönderilir.</p> <ul style="list-style-type: none"><li>İleti bir JMS iletisi değilse, ancak diğer özellikleri içeriyorsa, bu özellikler bir MQRFH2 üstbilgisinde uygulamaya döndürülmez.<sup>1</sup></li><li>Bu seçenek, birçok durumda, belirtik olarak bir MQRFH2 yaratan önceki sürüm uygulamalarının doğru bir şekilde çalışmasını da sağlar. Örneğin, MQRFH2 İçeren JMS ileti özelliklerini yaratan bir MQI programı doğru çalışmaya devam eder. JMS ileti özellikleri olmadan, ancak diğer bazı MQRFH2 klasörleriyle bir ileti oluşturulursa, klasörler uygulamaya döndürülür. Yalnızca klasörler ileti özelliği klasörleriyse, bu belirli klasörler MQRFH2' den kaldırılır. İleti özelliği klasörleri content= 'properties' adlı yeni klasör özneliğine sahip olarak tanıtılır ya da Tanımlı özellik klasörü adı ya da Gruplanmamış özellik klasörü adlistesinde yer alan klasörlerdir.</li><li>Uygulama bir ileti tanıtıcısı oluşturmak için MQCRTMH ' i çağırırsa, ileti özelliklerini MQINQMP kullanarak sorgulamalıdır. İleti özellikleri MQRFH2 üstbilgilerinden kaldırılır. İleti özellikleri olmayan ad-değer çiftleri MQRFH2 içinde kalır.</li><li>Uygulama bir ileti tanıtıcısı oluşturmak için MQCRTMH ögesini çağırırsa, iletinin JMS klasörleri olup olmadığına bakılmaksızın tüm ileti özelliklerini sorgulayabilir.</li><li>Uygulama bir ileti tanıtıcısı oluşturmazsa, tüm ileti özellikleri ve ad-değer çiftleri MQRFH2 içinde kalır.</li></ul> <p>Bir ileti yeni kullanıcı özelliği klasörleri içeriyorsa, iletinin yeni ya da değiştirilmiş bir IBM MQ 9.2 uygulaması tarafından oluşturulduğunu çıkarabilirsiniz. Alan uygulama bu yeni özellikleri doğrudan bir MQRFH2 içinde işlemek için kullanıyorsa, uygulamayı ALL seçeneğini kullanacak şekilde değiştirmeniz gerekir. Varsayılan COMPAT seçeneği ayarlandığında, değiştirilmemiş bir uygulama, IBM MQ 9.2 özellikleri olmadan MQRFH2' in geri kalanını işlemeye devam eder.</p> <p>PROPCTL arabiriminin amacı, MQRFH2 klasörlerini okuyan eski uygulamaları ve ileti özelliği arabirimini kullanan yeni ve değiştirilmiş uygulamaları desteklemektir. Yeni uygulamaların tüm kullanıcı iletisi özellikleri için ileti özelliği arabirimini kullanmasını ve MQRFH2 üstbilgilerini doğrudan okumaktan ve yazmaktan kaçınmasını hedefleyin.</p> <p>COMPAT yalnızca, alan uygulama bir MQGMO_PROPERTIES seçeneği ayarlı değilse ya da MQGMO_PROPERTIES_AS_Q_DEF olarak ayarladıysa bu etkiyi gösterir.</p>

Çizelge 493. Kuyruk iletisi özellik özneteliği ayarları (devamı var)

PROPCTL	Açıklama
ZORLA	<p>FORCE seçeneği, tüm ileti özelliklerini MQRFH2 üstbilgilerine yerleştirir. MQRFH2 üstbilgilerindeki tüm ileti özellikleri ve ad-değer çiftleri iletide kalır. İleti özellikleri MQRFH2' den kaldırılmaz ve bir ileti tanıtıcısı aracılığıyla kullanılabilir kıılır. FORCE seçeneğinin belirlenmesinin etkisi, yeni geçirilen bir uygulamanın MQRFH2 üstbilgilerinden ileti özelliklerini okumasını sağlamaktır.</p> <p>Bir uygulamayı IBM MQ 9.2 ileti özelliklerini işleyecek şekilde değiştirdiğinizi, ancak daha önce olduğu gibi MQRFH2 üstbilgileriyle doğrudan çalışma yeteneğini koruduğunu varsayın. Başlangıçta PROPCTL kuyruk özneteliğini FORCEolarak ayarlayarak, uygulamanın ileti özelliklerini kullanmaya ne zaman geçeceğine karar verebilirsiniz. İleti özelliklerini kullanmaya başlamaya hazır olduğunuzda <b>PROPCTL</b> kuyruk özneteliğini başka bir değere ayarlayın. Uygulamadaki yeni işlev beklediğiniz gibi davranmazsa, <b>PROPCTL</b> seçeneğini FORCEolarak ayarlayın.</p> <p>FORCE yalnızca, alan uygulama bir MQGMO_PROPERTIES seçeneği belirlemediyse ya da bunu MQGMO_PROPERTIES_AS_Q_DEFolarak ayarladıysa bu etkiyi gösterir.</p>
YOK	<p>Var olan bir uygulamanın bir iletiyi işleyebilmesi, tüm ileti özelliklerini yoksayması ve yeni ya da değiştirilen bir uygulamanın ileti özelliklerini sorgulaması için NONE seçeneğini kullanın.</p> <ul style="list-style-type: none"><li>• Uygulama bir ileti tanıtıcısı oluşturmak için MQCRTMH ' i çağırırsa, ileti özelliklerini MQINQMPkullanarak sorgulamalıdır. İleti özellikleri olmayan ad-değer çiftleri, ileti özelliklerinin soyulduğu MQRFH2içinde kalır.</li><li>• Uygulama bir ileti tanıtıcısı yaratmazsa, tüm ileti özellikleri MQRFH2' den kaldırılır. MQRFH2 üstbilgilerindeki ad-değer çiftleri iletide kalır.</li></ul> <p>NONE değeri yalnızca, alan uygulama bir MQGMO_PROPERTIES seçeneği ayarlı değilse ya da MQGMO_PROPERTIES_AS_Q_DEFdeğerine ayarlıysa bu etkiyi gösterir.</p>
V6COMPAT	<p>Gönderildiği biçimde bir MQRFH2 almak için bu seçeneği kullanın. Gönderen uygulama ya da kuyruk yöneticisi ek ileti özellikleri yaratırsa, bunlar ileti tanıtıcısında döndürülür.</p> <p>Bu seçenek, hem gönderen hem de alan kuyruklarda ve araya giren iletim kuyruklarında belirlenmeli. Kuyruk adı çözüm yolundaki kuyruk tanımlamalarında ayarlanan diğer PROPCTL seçeneklerini geçersiz kılar.</p> <p>V6COMPAT seçeneğini yalnızca istisnai durumlarda kullanın. Örneğin, uygulamaları önceki bir sürümden IBM MQ 9.2sürümüne geçiriyorsanız, önceki sürümün davranışını koruduğu için bu seçenek değerlidir. Bu seçeneğin ileti verimi üzerinde bir etkisi olabilir. Ayrıca, yönetmek daha zordur; seçeneğin gönderen, alıcı ve müdahale eden iletim kuyruklarında ayarlandığından emin olmanız gerekir.</p> <p>V6COMPAT yalnızca, alan uygulama bir MQGMO_PROPERTIES seçeneği ayarlı değilse ya da MQGMO_PROPERTIES_AS_Q_DEFolarak ayarladıysa bu etkiyi yaratır.</p>

İleti özellikleri ve ad-değer çiftleri hakkında daha fazla bilgi için bkz. [“NameValueVerileri \(MQCHARn\)” sayfa 525.](#)

## MQGMO için ileti özelliği seçenekleri

İleti özelliklerinin bir uygulamaya nasıl döndürüleceğini denetlemek için **MQGMO** ileti özelliği seçeneklerini kullanın.

<sup>1</sup> IBM MQ classes for JMS tarafından oluşturulan belirli özellik klasörlerinin varlığı bir JMS iletisini gösterir. Özellik klasörleri: mcd., jms., usr. ya da mqext.



Çizelge 494. MQGMO ileti özelliği seçeneği ayarları

MQGMO Seçenek	Açıklama
MQGMO_PROPERTIES_AS_Q_DEF	<p>Aynı kuyruktan okuyan ve GMO_PROPERTIES_* ayarını tanımlamayan IBM MQ uygulamaları, ileti özelliklerini farklı şekilde alır. İleti tanıtıcısı yaratmayan IBM MQ uygulamaları, kuyruk <b>PROPCTL</b> özniteliği tarafından denetlenir. IBM MQ uygulaması, MQRFH2 içindeki ileti özelliklerini almayı seçebilir ya da bir ileti tanıtıcısı yaratıp ileti özelliklerini sorgulayabilir. Uygulama bir ileti tanıtıcısı oluşturursa, özellikler MQRFH2' den kaldırılır.</p> <ul style="list-style-type: none"><li>• GMO_PROPERTIES_* ayarını tanımlamayan ya da MQGMO_PROPERTIES_AS_Q_DEF olarak ayarlayan yeni ya da değiştirilmiş bir IBM MQ uygulaması, ileti özelliklerini sorgulamayı seçebilir. MQINQMP MQI çağrısını kullanarak ileti tanıtıcısı ve sorgu iletisi özellikleri yaratmak için MQCRTMH ayarlanmalıdır.</li><li>• Yeni ya da değiştirilen bir uygulama ileti tanıtıcısı yaratmazsa, doğrudan MQRFH2 üstbilgilerinden aldığı ileti özelliklerini okuması gerekir.</li><li>• <b>PROPCTL</b> kuyruk özniteliği FORCE olarak ayarlanırsa, ileti tanıtıcısında hiçbir özellik döndürülmez. Tüm özellikler MQRFH2 üstbilgilerinde döndürülür.</li><li>• <b>PROPCTL</b> kuyruk özniteliği NONE ya da COMPAT olarak ayarlanırsa, ileti tanıtıcısı yaratan bir IBM MQ uygulaması tüm ileti özelliklerini alır.</li></ul>
MQGMO_PROPERTIES_IN_HANDLE	<p>Bir uygulamayı ileti özelliklerini kullanmaya zorlayın. Değiştirilen bir uygulamanın ileti tanıtıcısı yaratmakta başarısız olup olmadığını saptamak için bu seçeneği kullanın. Uygulama, ileti özelliklerini MQINQMPyerine doğrudan bir MQRFH2 içinden okumaya çalışıyor olabilir.</p>
MQGMO_NO_PROPERTIES	<ul style="list-style-type: none"><li>• Tüm özellikler kaldırılır. JMS özellikleri gibi kuyruk yöneticisi tarafından oluşturulan özellikler kaldırılır.</li><li>• Bir ileti tanıtıcısı yaratıldığında bile özellikler kaldırılır. Diğer MQRFH2 klasörlerindeki ad-değer çiftleri, ileti verilerinde bulunur.</li></ul>
MQGMO_PROPERTIES_FORCE_MQRFH2	<p>Bir ileti tanıtıcısı oluşturulsa da, özellikler MQRFH2 üstbilgilerinde döndürülür.</p> <ul style="list-style-type: none"><li>• MQINQMP , bir ileti tanıtıcısı yaratıldığında bile ileti özellikleri döndürmez. Bir özellik sorulursa MQRC_PROPERTY_NOT_AVAILABLE döndürülür.</li></ul>

Çizelge 494. MQGMO ileti özelliği seçeneği ayarları (devamı var)

MQGMO Seçenek	Açıklama
MQGMO_PROPERTIES_COMPATIBILITY	<p>İleti bir JMS istemcisiyse, MQRFH2 üstbilgilerinde JMS özellikleri döndürülür. İleti tanıtıcısı yaratan yeni ya da değiştirilmiş IBM MQ uygulamaları farklı davranır.</p> <ul style="list-style-type: none"><li>• İleti bir mcd . , jms . , usx . ya da mqext klasörü içeriyorsa, herhangi bir ileti özelliği klasöründeki tüm özellikler döndürülür.</li><li>• İleti bir mcd . , jms . , usx . ya da mqext klasörü değil, özellik klasörleri içeriyorsa, bir MQRFH2 içinde ileti özellikleri döndürülmez.</li><li>• Yeni ya da değiştirilmiş bir IBM MQ uygulaması bir ileti tanıtıcısı yaratırsa, ileti özelliklerini MQINQMP MQI çağrısını kullanarak sorgulayın. Tüm ileti özellikleri MQRFH2'iden kaldırılır.</li><li>• Yeni ya da değiştirilmiş bir IBM MQ uygulaması bir ileti tanıtıcısı yaratırsa, iletideki tüm özellikler sorgulanabilir. İleti bir mcd . , jms . , usx . ya da mqext klasörü içermese bile, tüm ileti özellikleri sorgulanabilir.</li></ul>

### İlgili başvurular

PROPCTL

2471 (09A7) (RC2471): MQRC\_PROPERTY\_NOT\_ALLOWED

### StrucId (MQCHAR4)

Bu, yapı tanıtıcısıdır. Değer şu olmalıdır:

#### MQGMO\_STRUC\_ID

Get-message options yapısına ilişkin tanıtıcı.

C programlama dili için, sabit MQGMO\_STRUC\_ID\_ARRAY değişmezi de tanımlanır; bu, MQGMO\_STRUC\_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisi olur.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQGMO\_STRUC\_ID ' dir.

### Sürüm (MQUZE)

Sürüm, yapı sürüm numarasıdır.

Değer aşağıdakilerden biri olmalıdır:

#### MQGMO\_VERSION\_1

Version-1 alma iletisi seçenekleri yapısı.

Bu sürüm, tüm ortamlarda desteklenir.

#### MQGMO\_VERSION\_2

Version-2 get-message options yapısı.

Bu sürüm, tüm ortamlarda desteklenir.

#### MQGMO\_VERSION\_3

Version-3 alma iletisi seçenekleri yapısı.

Bu sürüm, tüm ortamlarda desteklenir.

#### MQGMO\_VERSION\_4

Version-4 alma iletisi seçenekleri yapısı.

Bu sürüm, tüm ortamlarda desteklenir.

Yalnızca yapının daha yeni sürümlerinde var olan alanlar, alanların açıklamalarında olduğu gibi tanımlanır. Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

### **MQGMO\_CURRENT\_VERSION**

İleti alma seçenekleri yapısının yürürlükteki sürümü.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQGMO\_VERSION\_1' dir.

### **MQGMO için seçenekler (MQlong)**

**MQGMO** seçenekleri, MQGETişlemini denetler. Seçeneklerden sıfır ya da daha fazlasını belirleyebilirsiniz. Birden fazla isteğe bağlı değer gerekliyse:

- Değerleri ekleyin (aynı değişmezi bir kereden fazla eklemeyin) ya da
- Değerleri bit bilgiyle OR işlemini kullanarak (programlama dili bit işlemleri destekliyse) birleştirin.

Geçerli olmayan seçenek birleşimleri dikkat edilir; diğer tüm birleşimler geçerlidir.

### **Bekleme seçenekleri**

Aşağıdaki seçenekler, kuyruğa varması için bekleyen iletilerin bekleme ile ilgilidir:

#### **MQGMO\_BEKLE**

Uygulama, uygun bir ileti gelene kadar bekler. Uygulamanın bekleyeceği süre üst sınırı *WaitInterval* içinde belirtilir.

**Önemli:** Hemen uygun bir ileti varsa, bekleme ya da gecikme yoktur.

MQGET istekleri engellenirse ya da MQGET istekleri bekleme sırasında engellenirse, bekleme işlemi iptal edilir. The call completes with MQCC\_FAILED and reason code MQRC\_GET\_INHIBITED, regardless of whether there are suitable messages on the queue.

You can use MQGMO\_WAIT with the MQGMO\_BROWSE\_FIRST or MQGMO\_BROWSE\_NEXT options.

Birden çok uygulama aynı paylaşılan kuyruğun üzerinde bekliyorsa, uygun bir ileti geldiğinde hangi uygulamanın etkinleştirildiğini aşağıdaki kurallar seçmektedir:

<i>Çizelge 495. Rules for activating MQGET calls on a shared queue.</i>		
<b>Etkinleştirilmeyi bekleyen MQGET çağrılarının sayısı</b>		<b>Sonuç</b>
<b>GÖZ AT seçeneği ile</b>	<b>BROWSE seçeneği olmadan<sup>2</sup></b>	
Yok	bir ya da daha fazla	GÖZ seçeneği olmayan bir MQGET çağrısı etkinleştirilmektedir.
bir ya da daha fazla	Yok	Bir BROWSE seçeneğiyle tüm MQGET çağrıları etkinleştirilir.
bir ya da daha fazla	bir ya da daha fazla	GÖZ seçeneği olmayan bir MQGET çağrısı etkinleştirilmektedir. Etkinleştirilen BROWSE seçeneği olan MQGET çağrılarının sayısı tahmin edilemez.

Aynı kuyruğun üzerinde BROWSE seçeneği olmayan birden çok MQGET çağrısı varsa, yalnızca biri etkinleştirilmektedir. Kuyruk yöneticisi, aşağıdaki sırada bekleyen çağrılara öncelik vermeyi dener:

1. Yalnızca belirli iletiler (örneğin, belirli bir MsgId ya da CorrelId ) ile (ya da her ikisi) karşılanabilen belirli bekleme istekleri.
2. Herhangi bir iletiyle karşılanabilen genel alma isteği istekleri.

#### **Not:**

- İlk kategori içinde, daha belirli alma bekleme isteklerine ek öncelik verilmez. Örneğin, hem MsgId hem de CorrelId belirtilmesini sağlar.

<sup>2</sup> MQGMO\_LOCK seçeneğini belirten bir MQGET çağrısı, göz atmayan bir çağrı olarak işlem görür.

- Her iki kategori içinde, hangi uygulamanın seçildiği tahmin edilemez. Özellikle, en uzun süre bekleyen uygulama, seçilen en uzun süre için gerekli değildir.
- İşletim sisteminin yol uzunluğu ve öncelik zamanlaması konuları, düşük işletim sistemi önceliğinin beklenenden beklenen bir uygulamanın iletiyi almasının beklenenden daha düşük bir şekilde uygulanması anlamına gelebilir.
- Beklemedeki bir uygulamanın, tercihteki iletiyi olan bir iletiyi almasını da gerçekleştirmesi olabilir.

**z/OS** z/OS' ta aşağıdaki noktalar geçerlidir:

- Uygulamanın, iletinin gelmesini beklerken diğer çalışmalarla devam etmesi için, bunun yerine sinyal seçeneğini (MQGMO\_SET\_SIGNAL) kullanmayı düşünün. Ancak, sinyal seçeneği ortalamadır; farklı ortamlar arasında bağlantı noktalamak için kullanabileceğiniz uygulamalar bunu kullanmamalıdır.
- Aynı iletiyi bekleyen birden fazla MQGET çağrısı varsa, bekleme ve işaret seçenekleri karışımıyla, bekleyen her çağrı eşit olarak kabul edilir. It is an error to specify MQGMO\_SET\_SIGNAL with MQGMO\_WAIT. Ayrıca, işaretin ödenmemiş olduğu bir kuyruk tanıtıcısı ile bu seçeneği belirtme hatasıdır.
- If you specify MQGMO\_WAIT or MQGMO\_SET\_SIGNAL for a queue that has an IndexType of MQIT\_MSG\_TOKEN, no selection criteria are permitted. Bu da şu anlama gelir:
  - Bir version-1 MQGMO kullanıyorsanız, MQGET çağrısında belirtilen MQMD 'deki MsgId ve CorrelId alanlarını MQMI\_NONE ve MQCI\_NONE olarak ayarlayın.
  - If you are using a version-2 or later MQGMO, set the MatchOptions field to MQMO\_NONE.
- Bir MQGET çağrısı paylaşılan bir kuyruksa ve çağrı bir göz atma isteği ya da bir grup iletinin yıkıcı alma isteği olduğundan, MsgId ya da CorrelId eşleşmediği için, ECB işaretiniz 200 milisaniyeden sonra MQEC\_MSG\_VARDI ileti gönderilir.

Bu durum, kuyruğa uygun bir ileti gelmemiş olsa da, kuyruğun MQEC\_WAIT\_INTERVAL\_EXERDID ile birlikte gönderilmesiyle ilgili bekleme süresi sona erinceye kadar ortaya çıkar. MQEC\_MSG\_VARDI gönderildiğinde iletiyi almak için ikinci bir MQGET çağrısı yeniden yayınlamanız gerekir; varsa, iletiyi almak için.

Bu teknik, bir ileti gelişinin zamanında size bildirilmesini sağlamak için kullanılır; ancak, paylaşılmayan bir kuyrukta benzer bir çağrı dizisiyle karşılaştırıldığında, beklenmeyen bir işleme ek yükü olarak görünebilirsiniz.

MQGMO\_BROWSE\_MSG\_UNDER\_CURSOR ya da MQGMO\_MSG\_UNDER\_CURSOR ile belirtilirse, MQGMO\_WAIT yoksayılr; herhangi bir hata ortaya çıktı.

### **MQGMO\_NO\_BEKLEME**

Uygun bir ileti bulunmazsa, uygulama beklemeyi. MQGMO\_NO\_WAIT , MQGMO\_WAIT' in tersidir. MQGMO\_NO\_WAIT , program belgelerine yardımcı olmak için tanımlıdır. İkisi de belirtilmediyse, bu varsayılan değerdir.

### **MQGMO\_SET\_SIGNAL**

Bu seçeneği, Signal1 ve Signal2 alanlarıyla birlikte kullanın. Bir iletinin gelmesini beklerken uygulamaların diğer çalışmalarla devam etmesini sağlar. Ayrıca, (uygun işletim sistemi olanakları varsa), birden çok kuyruğa gelen iletilerin gelmesini beklemek için uygulamalar da kullanılabilir.

**Not:** MQGMO\_SET\_SIGNAL seçeneği, ortama özgü bir seçenektir; bu seçeneği, bağlantı noktası olarak kullanmak istediğiniz uygulamalar için kullanmayın.

İki durumda, arama işlemi, bu seçeneğin belirlenmemiş olduğu şekilde tamamlanır:

1. Şu anda kullanılabilir bir ileti, ileti tanımlayıcısında belirtilen ölçütleri karşıladığında.
2. Bir parametre hatası ya da başka bir zamanuyumlu hata saptanırsa.

İleti tanımlayıcısında belirtilen ölçütlere uygun bir ileti görüntülenmiyorsa, denetim, iletinin gelmesini beklemeden uygulamaya döner. **CompCode** ve **Reason** parametreleri, MQCC\_WARNING ve MQRC\_SIGNAL\_REQUEST\_ACCEPTED olarak ayarlanır. İleti tanımlayıcısındaki diğer çıkış alanları ve MQGET çağrısının çıkış değişirgeleri tanımlı değil. Daha sonra uygun bir ileti geldiğinde, ECB ' ye posta gönderilerek sinyal teslim edilir.

Bu durumda, iletiyi almak için MQGET çağrısını yeniden yayınlamalıdır. Uygulama, işletim sistemi tarafından sağlanan işlevleri kullanarak bu sinyali bekleyebilir.

İşletim sistemi birden çok bekleme mekanizması sağlıyorsa, bu işlemi birkaç kuyruktan herhangi birine gelen bir iletinin gelmesini beklemek için kullanabilirsiniz.

Sıfır olmayan bir WaitInterval belirtilirse, bekleme süresi sona erdikten sonra sinyal teslim edilir. Kuyruk yöneticisi beklemeyi de iptal edebilir, bu durumda sinyal teslim edilir.

Birden çok MQGET çağrısı, aynı ileti için bir işaret ayarlayabilir. Uygulamaların etkinleştirildiği sıra, MQGMO\_WAIT için açıklananlarla aynıdır.

Aynı iletiyi birden fazla MQGET çağrısı bekliyorsa, bekleyen her çağrı eşit olarak kabul edilir. Aramalar, bekleme ve sinyal seçeneklerinin bir karışımının içermesini sağlar.

Belirli koşullar altında, MQGET çağrısı bir iletiyi alabilir ve aynı iletinin gelmesinden kaynaklanan bir sinyal teslim edilebilir. Bir sinyal teslim edildiğinde, hiçbir ileti gönderilmeyecek şekilde bir uygulama hazırlanmalıdır.

Bir kuyruk tanıtıcısı, bekleyen birden fazla sinyal isteği olamaz.

Bu seçenek, aşağıdaki seçeneklerin hiçbiriyle geçerli değildir:

- MQGMO\_UNLOCK
- MQGMO\_WAIT

Bir MQGET çağrısı paylaşılan bir kuyruksa ve çağrı bir göz atma isteği ya da bir grup iletisinin yıkıcı alma isteği olduğundan, MsgId ya da CorrelId eşleştirilmezse, kullanıcının sinyal ECB değeri 200 milisaniyeden sonra MQEC\_MSG\_ARRIVED ' a gönderilir.

Bu durum, kuyruğa uygun bir ileti gönderilmemiş olsa da, kuyruk MQEC\_WAIT\_INTERVAL\_EXPIRED ile birlikte gönderildiğinde bekleme süresi doluncaya kadar kuyruğa girmemiş olabilir. MQEC\_MSG\_ARRIVED yayınlandığında, varsa iletiyi almak için ikinci bir MQGET çağrısı yeniden yayınlamanız gerekir.

Bu teknik, bir ileti gelişinin zamanında size bildirilmesini sağlamak için kullanılır; ancak, paylaşılmayan bir kuyrukta benzer bir çağrı dizisiyle karşılaştırıldığında, beklenmeyen bir işleme ek yükü olarak görünebilirsiniz.

Bu ileti, iletiler sık olarak eklendiğinde, verimli bir ileti alma yöntemi değildir. Göz atma durumunda bu ek yükü önlemek için, MQGET çağrısıyla eşleşen MsgId değerini (dizinlenmemiş ya da MsgId ile dizinlenmediyse) ya da CorrelId ( CorrelId ile dizinlendiyse) belirtin.

**z/OS** Bu seçenek yalnızca z/OS üzerinde desteklenir.

## MQGMO\_FAIL\_IF QUIESCING

Kuyruk yöneticisi susturulmuş durumdaysa, MQGET çağrısını başarısız olmak için zorlayın.

**z/OS** z/OS' ta bu seçenek, bağlantı ( CICS ya da IMS uygulaması için) susturulmuş durumdaysa, MQGET çağrısını başarısız olarak da zorlar.

Bu seçenek MQGMO\_WAIT ya da MQGMO\_SET\_SIGNAL ile belirtilirse ve bekleme ya da işaret, kuyruk yöneticisinin susturulmuş duruma girdiği sırada olağanüstüyorsa:

- The wait is canceled and the call returns completion code MQCC\_FAILED with reason code MQRC\_Q\_MGR QUIESCING or MQRC\_CONNECTION QUIESCING.
- Sinyal, ortama özgü bir sinyal tamamlanma kodu ile iptal edilir.

**z/OS** z/OS' ta sinyal, olay tamamlanma kodu MQEC\_Q\_MGR QUIESCING ya da MQEC\_CONNECTION QUIESCING ile tamamlanır.

MQGMO\_FAIL\_IF QUIESCING belirtilmemişse ve kuyruk yöneticisi ya da bağlantı susturma durumuna girerse, bekleme ya da işaret iptal edilmez.

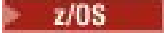
## Eşitleme noktası seçenekleri

The following options relate to the participation of the MQGET call within a unit of work:

### MQGMO\_SYNCPOINT

Bu istek, olağan çalışma birimi protokolleri içinde işlem yapmak. İleti, diğer uygulamalar tarafından kullanılamaz olarak işaretlenir, ancak iş birimi kesinleştirildiğinde kuyruktan silinir. İş birimi geriletilirse, ileti yeniden kullanılabilir kılınır.

MQGMO\_SYNCPOINT ve MQGMO\_NO\_SYNCPOINT kümeden ayrılabilirsiniz. Bu durumda, alma isteğinin iş birimi protokollerine dahil edilmesi, kuyruk yöneticisini çalıştıran ortam tarafından belirlenir. Bu, uygulamayı çalıştıran ortam tarafından belirlenmez.

-  On z/OS, the get request is within a unit of work.
- z/OS'dışındaki tüm ortamlarda, alma isteği bir iş birimi içinde değildir.

Because of these differences, an application that you want to port must not allow this option to default; specify MQGMO\_SYNCPOINT or MQGMO\_NO\_SYNCPOINT explicitly.

Bu seçenek, aşağıdaki seçeneklerin hiçbiriyle geçerli değildir:

- MQGMO\_BROWSE\_FIRST
- MQGMO\_BROWSE\_MSG\_UNDER\_CURSOR
- MQGMO\_BROWSE\_NEXT
- MQGMO\_LOCK
- MQGMO\_NO\_SYNCPOINT
- MQGMO\_SYNCPOINT\_IF\_PERSISTENT
- MQGMO\_UNLOCK

### MQGMO\_SYNCPOINT\_IF\_PERSISTENT

Bu istek, yalnızca alınan iletinin kalıcı olması durumunda, olağan çalışma birimi protokollerinde işlem görmektedir. Kalıcı ileti, MQMD' deki Persistence alanında MQPER\_PERSISTENT değerine sahiptir.

- If the message is persistent, the queue manager processes the call as though the application had specified MQGMO\_SYNCPOINT.
- If the message is not persistent, the queue manager processes the call as though the application had specified MQGMO\_NO\_SYNCPOINT.

Bu seçenek, aşağıdaki seçeneklerin hiçbiriyle geçerli değildir:

- MQGMO\_BROWSE\_FIRST
- MQGMO\_BROWSE\_MSG\_UNDER\_CURSOR
- MQGMO\_BROWSE\_NEXT
- MQGMO\_COMPLETE\_MSG
- MQGMO\_MARK\_SKIP\_BACKOUT
- MQGMO\_NO\_SYNCPOINT
- MQGMO\_SYNCPOINT
- MQGMO\_UNLOCK

Bu seçenek aşağıdaki ortamlarda desteklenir:

-  AIX
-  IBM i
-  Linux
-  z/OS

ve bu sistemlere baęlı IBM MQ MQI clients için.

### **MQGMO\_NO\_SYNCPOINT**

Bu istek, olaęan alıřma birimi protokollerinin dıřında iřlem yapmak. Gz atma seeneęi olmayan bir ileti aldıysanız, ileti hemen kuyruktan silinir. İleti, iř biriminin yedeklenmesiyle yeniden kullanılabilir kılınamaz.

MQGMO\_BROWSE\_FIRST ya da MQGMO\_BROWSE\_NEXT belirtilirse bu seenek varsayılan olarak kabul edilir.

MQGMO\_SYNCPOINT ve MQGMO\_NO\_SYNCPOINT kmeden ayrılabilirsiniz. Bu durumda, alma isteęinin iř birimi protokollerine dahil edilmesi, kuyruk yneticisini alıřtıran ortam tarafından belirlenir. Bu, uygulamayı alıřtıran ortam tarafından belirlenmez.

- **z/OS** On z/OS, the get request is within a unit of work.
- z/OS dıřındaki tm ortamlarda, alma isteęi bir iř birimi iinde deęildir.

Because of these differences, an application that you want to port must not allow this option to default; specify either MQGMO\_SYNCPOINT or MQGMO\_NO\_SYNCPOINT explicitly.

Bu seenek, ařaęıdaki seeneklerin hibiriyle geerli deęildir:

- MQGMO\_MARK\_SKIP\_BACKOUT
- MQGMO\_SYNCPOINT
- MQGMO\_SYNCPOINT\_IF\_PERSISTENT

### **z/OS MQGMO\_MARK\_SKIP\_BACKOUT**

Bu seenekle iřaretlenen iletiyi kuyruęa geri vermeden, bir iř birimini geri alma.

Bu seenek yalnızca z/OS zerinde desteklenir.

Bu seenek belirtilirse, MQGMO\_SYNCPOINT de belirtilmelidir. MQGMO\_MARK\_SKIP\_BACKOUT ařaęıdaki seeneklerden herhangi biriyle geerli deęildir:

- MQGMO\_BROWSE\_FIRST
- MQGMO\_BROWSE\_MSG\_UNDER\_CURSOR
- MQGMO\_BROWSE\_NEXT
- MQGMO\_LOCK
- MQGMO\_NO\_SYNCPOINT
- MQGMO\_SYNCPOINT\_IF\_PERSISTENT
- MQGMO\_UNLOCK

**Not:** On IMS and CICS, you might have to issue an extran IBM MQ call after backing out a unit of work containing a message marked with MQGMO\_MARK\_SKIP\_BACKOUT. İřaretli iletiyi ieren yeni iř birimini kesinleřtirmeden nce bir IBM MQ aęrısı yayınlamanız gerekir. Arama, istedięiniz herhangi bir IBM MQ aęrısı olabilir.

1. IMS zerinde, IMS APAR PN60855 ' i uygulamadıysanız ve bir IMS MPP ya da BMP uygulaması alıřtırıyorsanız.
2. Herhangi bir uygulama alıřtırıyorsanız, CICS zerinde.

Her iki durumda da, yedeklenmiř iletiyi ieren yeni iř birimini kesinleřtirmeden nce herhangi bir IBM MQ aęrısı yayınlayın.

**Not:** Bir iř birimi iinde, geri alma iřlemi atlanıyor olarak iřaretlenen tek bir alma isteęi yanı sıra, hibir ya da birden ok iřaretlenmemiř alma isteęi olabilir.

Bir uygulama, bir iř biriminden dıřarı ıkmıřsa, MQGMO\_MARK\_SKIP\_BACKOUT kullanılarak alınan bir ileti nceki durumuna geri yklenmez. Dięer kaynak gncellemeleri yedeklenir. İleti, geri alma isteęi tarafından bařlatılan yeni bir iř biriminde alınmıř gibi iřlem grr. İleti, MQGMO\_MARK\_SKIP\_BACKOUT seeneęi olmadan alınır.

Bazı kaynaklar değiştirildikten sonra, iş biriminin başarıyla tamamlanamadığı anlaşılırsa, MQGMO\_MARK\_SKIP\_BACKOUT yararlı olur. Bu seçeneği çıkarırsanız, iş birimini yedeklemek, iletiyi kuyruktan geri almak için geri getirmenizi sağlar. İleti bir sonraki alındığında aynı olay dizisi yine ortaya çıkar.

Ancak, özgün MQGET çağrısında MQGMO\_MARK\_SKIP\_BACKOUT değerini belirtirseniz, iş birimi yedeklenirken diğer kaynaklardaki güncellemeleri de geri çekmektedir. İleti, yeni bir iş birimi altında alınmış gibi işlem görür. Uygulama uygun hata işlemeyi gerçekleştirebilir. Özgün iletiyi gönderene bir rapor ileti gönderebilir ya da özgün iletiyi ölüme mektup kuyruğuna yerleştirebilir. Daha sonra yeni iş birimini kesinleştirebilir. Yeni iş birimi kesinleştirilirken, ileti özgün kuyruktan kalıcı olarak kaldırılır.

MQGMO\_MARK\_SKIP\_BACKOUT , tek bir fiziksel ileti işaretler. İleti bir ileti grubuna aitse, gruptaki diğer iletiler imlenmez. Benzer bir şekilde, işaretli ileti bir mantıksal iletinin bir kesimiye, mantıksal iletteki diğer kesimler imlenmez.

Bir gruptaki herhangi bir ileti işaretlenir, ancak iletiler MQGMO\_LOGICAL\_ORDER kullanılarak alınır, gruptaki ilk iletiyi işaretlemek avantajlıdır. İş birimi geriletilirse, ilk (işaretli) ileti yeni iş birimine taşınır. Gruptaki ikinci ve sonraki iletiler kuyruğa yeniden yürürlüğe girilir. Kuyruğunda kalan iletiler, MQGMO\_LOGICAL\_ORDER kullanılarak başka bir uygulama tarafından alınamıyor. Gruptaki ilk ileti artık kuyruğa değil. However, the application that backed the unit of work out can retrieve the second and later messages into the new unit of work using the MQGMO\_LOGICAL\_ORDER option. İlk ileti zaten alındı.

Zaman zaman yeni iş birimini yedeklemeniz gerekebilir. Örneğin, dead-letter kuyruğu dolu olduğu için ve ileti atılmamalı. Yeni iş birimi yedeklenirken, iletinin kaybolmasına engel olan ileti özgün kuyruğa yeniden yürürlüğe konmasını sağlar. Ancak, bu durumda işleme devam edilemiyor. Yeni iş birimini yedekledikten sonra, uygulamanın, işletmene ya da yöneticiye kurtarılamaz bir hata olduğunu bildirmesi ve sonra bitirilmesinin gerektiği konusunda bilgilendirmesi gerekir.

MQGMO\_MARK\_SKIP\_BACKOUT yalnızca, alma isteğini içeren iş birimi uygulamanın geri vermesiyle kesintiye uğratılırsa çalışır. Alma isteğini içeren iş birimi, işlem ya da sistem başarısız olduğu için yedeklenirse, MQGMO\_MARK\_SKIP\_BACKOUT yoksayıdır. Bu seçenek kullanılarak alınan tüm iletiler, bu seçenek olmadan alınan iletiyle aynı şekilde kuyruğa yeniden başlatılır.

## Göz Atma Seçenekleri

Aşağıdaki seçenekler kuyruklardaki iletilere göz atmak için ilişkilendir:

### MQGMO\_BROWSE\_FIRST

Bir kuyruk MQOO\_BROWSE seçeneğiyle açıldığında, kuyruktaki ilk iletiden önce mantıksal olarak konumlandırılmış bir göz atma imleci oluşturulur. You can then use MQGET calls specifying the MQGMO\_BROWSE\_FIRST, MQGMO\_BROWSE\_NEXT, or MQGMO\_BROWSE\_MSG\_UNDER\_CURSOR option to retrieve messages from the queue nondestructively. The browse cursor marks the position, within the messages on the queue, from which the next MQGET call with MQGMO\_BROWSE\_NEXT searches for a suitable message.

MQGMO\_BROWSE\_FIRST aşağıdaki seçeneklerden herhangi biriyle geçerli değildir:

- MQGMO\_BROWSE\_MSG\_UNDER\_CURSOR
- MQGMO\_BROWSE\_NEXT
- MQGMO\_MARK\_SKIP\_BACKOUT
- MQGMO\_MSG\_UNDER\_CURSOR
- MQGMO\_SYNCPOINT
- MQGMO\_SYNCPOINT\_IF\_PERSISTENT
- MQGMO\_UNLOCK

Ayrıca, kuyruk göz atma için açılmamışsa da bu bir hatadır.

MQGMO\_BROWSE\_FIRST ile bir MQGET çağrısı, imleci göz atma imlecinin önceki konumunu yoksayar. Kuyruğa belirtilen ilk ileti, ileti tanımlayıcısında belirtilen koşulları karşıladığını gösterir. İleti kuyruğun üzerinde kalır ve göz atma imleci bu iletiye konumlanır.



Bu çağrıdan sonra, göz atma imleci döndürülen iletiye konumlanır. The message might be removed from the queue before the next MQGET call with MQGMO\_BROWSE\_NEXT is issued. Bu durumda, imleç, bu konum artık boş olsa bile, iletinin meşgul olduğu kuyrukta konumdaki konumda kalır.

Use the MQGMO\_MSG\_UNDER\_CURSOR option with a non-browse MQGET call, to remove the message from the queue.

Göz atma imleci, aynı *Hobj* tutamacını kullansa da, göz atma dışı bir MQGET çağrısı tarafından taşınmaz. Nor is it moved by a browse MQGET call that returns a completion code of MQCC\_FAILED, or a reason code of MQRC\_TRUNCATED\_MSG\_FAILED.

Göz atılan iletiyi kilitlemek için, bu seçenikle MQGMO\_LOCK seçeneğini belirleyin.

Mantıksal iletilerin gruplarındaki ve bölümlerindeki iletilerin işlenmesini denetleyen MQGMO\_\* ve MQMO\_\* seçeneklerinin herhangi bir geçerli birleşimi ile MQGMO\_BROWSE\_FIRST değerini belirtebilirsiniz.

MQGMO\_LOGICAL\_ORDER belirtilirse, iletiler mantıksal sırayla göz atılır. Bu seçeneği çıkarırsanız, iletiler fiziksel sırayla göz atılır. MQGMO\_BROWSE\_FIRST belirtirseniz, mantıksal sipariş ile fiziksel sipariş arasında geçiş yapabilirsiniz. Subsequent MQGET calls using MQGMO\_BROWSE\_NEXT browse the queue in the same order as the most recent call that specified MQGMO\_BROWSE\_FIRST for the queue handle.

Kuyruk yöneticisi, MQGET çağrılarında ilişkin iki grup ve bölüm bilgilerini saklar. Göz atma çağrılarında ilişkin grup ve bölüm bilgileri, kuyruktan iletileri alan çağrılara ilişkin bilgilerden ayrı olarak saklanır. MQGMO\_BROWSE\_FIRST belirtirseniz, kuyruk yöneticisi göz atılmasına ilişkin grup ve bölüm bilgilerini yoksayar. Yürürlükteki grup ve yürürlükteki mantıksal ileti yok gibi kuyrukları tarar. MQGET çağrısı başarılı, tamamlanma kodu MQCC\_OK ya da MQCC\_WARNING ise, göz atma için grup ve bölüm bilgileri, döndürülen iletinin o değerine ayarlanır. Arama başarısız olursa, grup ve bölüm bilgileri aramadan önce olduğu gibi kalır.

## MQGMO\_BROWSE\_NEXT

Advance the browse cursor to the next message on the queue that satisfies the selection criteria specified on the MQGET call. İleti uygulamaya geri döndürüldü, ancak kuyruğun üzerinde kalır.

MQGMO\_BROWSE\_NEXT aşağıdaki seçeneklerden herhangi biriyle geçerli değildir:

- MQGMO\_BROWSE\_FIRST
- MQGMO\_BROWSE\_MSG\_UNDER\_CURSOR
- MQGMO\_MARK\_SKIP\_BACKOUT
- MQGMO\_MSG\_UNDER\_CURSOR
- MQGMO\_SYNCPOINT
- MQGMO\_SYNCPOINT\_IF\_PERSISTENT
- MQGMO\_UNLOCK

Ayrıca, kuyruk göz atma için açılmamışsa da bu bir hatadır.

MQGMO\_BROWSE\_NEXT , kuyruğa göz atma için açıldıktan sonra, kuyruğa göz atmak için ilk çağrıysa, MQGMO\_BROWSE\_FIRST ile aynı şekilde davranır.

The message under cursor might be removed from the queue before the next MQGET call with MQGMO\_BROWSE\_NEXT is issued. Göz atma imleci, bu konum artık boş olsa bile, mantıksal olarak iletinin meşgul olduğu kuyrukta konumdaki konumda kalır.

İletiler kuyrukta iki yoldan biriyle saklanır:

- (MQMDS\_PRIORITY) önceliğindeki FIFO ya da
- Öncelik ne olursa olsun FIFO (MQMDS\_FIFO)

**MsgDeliverySequence** kuyruk özneliği, hangi yöntemin geçerli olduğunu belirtir (ayrıntılar için [“Kuyruklara ilişkin öznelikler” sayfa 821](#) konusuna bakın).

Bir kuyruk `MsgDeliverySequence / MQMDS_PRIORITY` olabilir. Kuyruğa, göz atma imlecine işaret eden daha yüksek önceliğe sahip bir ileti gönderilir. In which case, the higher priority message is not found during the current sweep of the queue using `MQGMO_BROWSE_NEXT`. Yalnızca, göz atma imlecinin `MQGMO_BROWSE_FIRST` ile ilk durumuna getirilmesinden ya da kuyruğun yeniden açılmasından sonra bulunabilir.

The `MQGMO_MSG_UNDER_CURSOR` option can be used with a non-browse `MQGET` call if required, to remove the message from the queue.

Göz atma imleci, aynı `Hobj` tanıtıcısı kullanılarak `MQGET` ' a göz atmayan çağrılar tarafından taşınmaz.

Göz atılacak iletiyi kilitlemek için bu seçenikle `MQGMO_LOCK` seçeneğini belirtin.

Mantıksal iletilerin gruplarındaki ve bölümlerindeki iletilerin işlenmesini denetleyen `MQGMO_*` ve `MQMO_*` seçeneklerinin herhangi bir geçerli birleşimi ile `MQGMO_BROWSE_NEXT` değerini belirtebilirsiniz.

`MQGMO_LOGICAL_ORDER` belirtilirse, iletiler mantıksal sırayla göz atılır. Bu seçeneği çıkarırsanız, iletiler fiziksel sırayla göz atılır. `MQGMO_BROWSE_FIRST` belirtirseniz, mantıksal sipariş ile fiziksel sipariş arasında geçiş yapabilirsiniz. Subsequent `MQGET` calls using `MQGMO_BROWSE_NEXT` browse the queue in the same order as the most recent call that specified `MQGMO_BROWSE_FIRST` for the queue handle. The call fails with reason code `MQRC_INCONSISTENT_BROWSE` if this condition is not satisfied.

**Not:** `MQGMO_LOGICAL_ORDER` belirtilmemişse, ileti grubunun sonuna göz atmak için `MQGET` çağrısını kullanırken özel özen gösteriniz. Örneğin, gruptaki son iletinin, kuyruқта gruptaki ilk iletisinden önce geldiğini varsayın. Using `MQGMO_BROWSE_NEXT` to browse beyond the end of the group, specifying `MQMO_MATCH_MSG_SEQ_NUMBER` with `MsgSeqNumber` set to 1 returns the first message in the group already browsed. Bu sonuç hemen gerçekleşebilir ya da araya müdahale eden gruplar varsa, `MQGET` ' un bir sayısı daha sonra olabilir. Aynı dikkat edilmesi, bir grupta olmayan mantıksal bir ileti için geçerlidir.

Göz atma çağrılarına ilişkin grup ve bölüm bilgileri, kuyruktan iletileri alan çağrılara ilişkin bilgilerden ayrı olarak saklanır.

### **MQGMO\_BROWSE\_MSG\_UNDER\_CURSOR**

Retrieve the message pointed to by the browse cursor nondestructively, regardless of the `MQMO_*` options specified in the `MatchOptions` field in `MQGMO`.

`MQGMO_BROWSE_MSG_UNDER_CURSOR` aşağıdaki seçeneklerden herhangi biriyle geçerli değildir:

- `MQGMO_BROWSE_FIRST`
- `MQGMO_BROWSE_NEXT`
- `MQGMO_MARK_SKIP_BACKOUT`
- `MQGMO_MSG_UNDER_CURSOR`
- `MQGMO_SYNCPOINT`
- `MQGMO_SYNCPOINT_IF_PERSISTENT`
- `MQGMO_UNLOCK`

Ayrıca, kuyruk göz atma için açılmamışsa da bu bir hatadır.

Göz atma imlecinin işaret ettiği ileti, `MQGMO_BROWSE_FIRST` ya da `MQGMO_BROWSE_NEXT` seçeneği kullanılarak en son alınan ileti. Bu kuyruk açıldığından bu kuyruk için bu çağrılar hiçbir gönderilmezse, çağrı başarısız olur. Ayrıca, göz atma imlecinin altındaki ileti yok edici olarak geri alındıysa, çağrı da başarısız olur.

Göz atma imlecinin konumu, bu çağrıya göre değiştirilmez.

The `MQGMO_MSG_UNDER_CURSOR` option can be used with a non-browse `MQGET` call, to remove the message from the queue.

Göz atma imleci, aynı Hobj tutamacını kullansa da, göz atma dışı bir MQGET çağrısı tarafından taşınmaz. Nor is it moved by a browse MQGET call that returns a completion code of MQCC\_FAILED, or a reason code of MQRC\_TRUNCATED\_MSG\_FAILED.

MQGMO\_BROWSE\_MSG\_UNDER\_CURSOR MQGMO\_LOCK ile belirtilirse:

- Önceden kilitlenmiş bir ileti varsa, bu ileti, imlecin altındaki tek bir ileti olmalıdır; bu nedenle, kilidi açmadan ve kitlemeden geri döndürülür. İleti kilitli kalır.
- Kilitli bir ileti yoksa ve göz at imlecinin altında bir ileti varsa, bu ileti kilitlenir ve uygulamaya geri döndürülür. Göz atma imlecinin altında herhangi bir ileti yoksa, arama başarısız olur.

If MQGMO\_BROWSE\_MSG\_UNDER\_CURSOR is specified without MQGMO\_LOCK:

- Önceden kilitlenmiş bir ileti varsa, imlecin altındaki bu ileti olmalıdır. İleti uygulamaya geri döndürüldü ve daha sonra kilidi açıldı. İletinin kilidi açıldığı için, iletinin yeniden göz atılabileceğinin garantisi yoktur ya da aynı uygulama tarafından yok edici bir şekilde alınır. Kuyruktan ileti almak başka bir uygulama tarafından yok edici olarak alınmış olabilir.
- Kilitli bir ileti yoksa ve göz at imlecinin altında bir ileti varsa, bu ileti uygulamaya geri döndürülür. Göz atma imlecinin altında bir ileti yoksa, arama başarısız olur.

If MQGMO\_COMPLETE\_MSG is specified with MQGMO\_BROWSE\_MSG\_UNDER\_CURSOR, the browse cursor must identify a message whose Offset field in MQMD is zero. Bu koşul karşılanmazsa, çağrı neden kodu MQRC\_INVALID\_MSG\_UNDER\_CURSOR ile başarısız olur.

Göz atma çağrılarında ilişkin grup ve bölüm bilgileri, kuyruktan iletileri alan çağrılara ilişkin bilgilerden ayrı olarak saklanır.

### **MQGMO\_MSG\_UNDER\_CURSOR**

Retrieve the message pointed to by the browse cursor, regardless of the MQMO\_\* options specified in the MatchOptions field in MQGMO. İleti kuyruktan kaldırılır.

Göz atma imlecinin işaret ettiği ileti, MQGMO\_BROWSE\_FIRST ya da MQGMO\_BROWSE\_NEXT seçeneği kullanılarak en son alınan ileti.

If MQGMO\_COMPLETE\_MSG is specified with MQGMO\_MSG\_UNDER\_CURSOR, the browse cursor must identify a message whose Offset field in MQMD is zero. Bu koşul karşılanmazsa, çağrı neden kodu MQRC\_INVALID\_MSG\_UNDER\_CURSOR ile başarısız olur.

Bu seçenek, aşağıdaki seçeneklerin hiçbiriyle geçerli değildir:

- MQGMO\_BROWSE\_FIRST
- MQGMO\_BROWSE\_MSG\_UNDER\_CURSOR
- MQGMO\_BROWSE\_NEXT
- MQGMO\_UNLOCK

Ayrıca, hem göz atma hem de giriş için kuyruk açılmamışsa, bu bir hatadır. Göz atma imleci yeniden alınabilir bir iletiyi göstermiyorsa, MQGET çağrısı tarafından bir hata döndürülür.

### **MQGMO\_MARK\_BROWSE\_HANDLE**

Başarılı bir MQGET tarafından döndürülen ya da döndürülen MsgToken ile tanıtilen ileti işaretlendi. Bu işaret, çağrıda kullanılan nesne tanıtıcısı için özgüdür.

İleti kuyruktan kaldırılmaz.

MQGMO\_MARK\_BROWSE\_HANDLE , yalnızca aşağıdaki seçeneklerden biri de belirtilirse geçerlidir:

- MQGMO\_BROWSE\_FIRST
- MQGMO\_BROWSE\_MSG\_UNDER\_CURSOR
- MQGMO\_BROWSE\_NEXT

MQGMO\_MARK\_BROWSE\_HANDLE aşağıdaki seçeneklerden herhangi biriyle geçerli değildir:

- MQGMO\_ALL\_MSGS\_AVAILABLE
- MQGMO\_ALL\_SEGMENTS\_AVAILABLE

- MQGMO\_COMPLETE\_MSG
- MQGMO\_LOCK
- MQGMO\_LOGICAL\_ORDER
- MQGMO\_UNLOCK

Aşağıdaki olaylardan biri gerçekleşinceye kadar ileti bu durumda kalır:

- İlgili nesne tanıtıcısı olağan biçimde ya da başka bir şekilde kapatılır.
- İleti, MQGMO\_UNMARK\_BROWSE\_HANDLE seçeneğiyle MQGET arasında bir çağrı yaparak bu tanıtıcı için işaretlenmemiş.
- İleti, MQCC\_OK ya da MQCC\_WARNING ile tamamlanan, yıkıcı MQGET çağrısından geri döndürülür. MQGET daha sonra geri döndürülese bile ileti durumu değişmeye devam eder.
- İletinin süresi doluyor.

### **MQGMO\_MARK\_BROWSE\_CO\_OP**

Başarılı bir MQGET tarafından döndürülen ya da döndürülen *MsgToken* ile tanıtılan ileti, işbirliği kümesindeki tüm tutamaçlar için işaretlenir.

İşbirliği düzeyi işareti, ayarlanmış olabilecek herhangi bir tanıtıcı düzeyi işaretine ek olarak.

İleti kuyruktan kaldırılmaz.

MQGMO\_MARK\_BROWSE\_CO\_OP is valid only if the object handle used was returned by a call to MQOPEN that specified MQOO\_CO\_OP. Ayrıca, aşağıdaki MQGMO seçeneklerinden birini de belirtmeniz gerekir:

- MQGMO\_BROWSE\_FIRST
- MQGMO\_BROWSE\_MSG\_UNDER\_CURSOR
- MQGMO\_BROWSE\_NEXT

Bu seçenek, aşağıdaki seçeneklerin hiçbiriyle geçerli değildir:

- MQGMO\_ALL\_MSGS\_AVAILABLE
- MQGMO\_ALL\_SEGMENTS\_AVAILABLE
- MQGMO\_COMPLETE\_MSG
- MQGMO\_LOCK
- MQGMO\_LOGICAL\_ORDER
- MQGMO\_UNLOCK

İleti önceden işaretlenmişse ve MQGMO\_UNMARKED\_BROWSE\_MSG seçeneği belirtilmemişse, çağrı MQCC\_FAILED ile başarısız olur ve neden kodu MQRC\_MSG\_MARKED\_BROWSE\_CO\_OP.

Aşağıdaki olaylardan biri gerçekleşinceye kadar ileti bu durumda kalır:

- İşbirliği kümesindeki tüm nesne tutamaçları kapatılır.
- İleti, MQGMO\_UNMARK\_BROWSE\_CO\_OP seçeneği ile MQGET çağrısı ile işbirliği yapan tarayıcılar için işaretlenmemiş.
- İleti, kuyruk yöneticisi tarafından otomatik olarak işaretlenmez.
- İleti, göz atma olmayan MQGET çağrısından geri döndürülür. MQGET daha sonra geri döndürülese bile ileti durumu değişmeye devam eder.
- İletinin süresi doluyor.

### **MQGMO\_UNMARKED\_BROWSE\_MSG**

A call to MQGET that specifies MQGMO\_UNMARKED\_BROWSE\_MSG returns a message that is considered to be unmarked for its handle. İleti, tanıtıcısı için işaretlenmişse bir ileti döndürmez. It also does not return the message if the queue was opened by a call to MQOPEN, with the option MQOO\_CO\_OP, and the message has been marked by a member of the cooperating set.

Bu seçenek, aşağıdaki seçeneklerin hiçbiriyle geçerli değildir:

- MQGMO\_ALL\_MSGS\_AVAILABLE
- MQGMO\_ALL\_SEGMENTS\_AVAILABLE
- MQGMO\_COMPLETE\_MSG
- MQGMO\_LOCK
- MQGMO\_LOGICAL\_ORDER
- MQGMO\_UNLOCK

#### **MQGMO\_UNMARK\_BROWSE\_CO\_OP**

Bu seçeneği belirten bir MQGET çağrısının ardından, iş birliği işlemi için işaretlenecek işbirliği noktaları kümesindeki herhangi bir açık tanıtıcı tarafından ileti artık dikkate alınmamaktadır. İleti, bu çağrıdan önce tanıtıcı düzeyinde işaretlenmişse, bu ileti hala tanıtıcı düzeyinde işaretlenir.

Using MQGMO\_UNMARK\_BROWSE\_CO\_OP is valid only with a handle returned by a successful call to MQOPEN with the option MQOO\_CO\_OP. MQGET , iş birliği çekme noktaları kümesi tarafından işaretlenmemiş olarak kabul edilmese de başarılı olur.

MQGMO\_UNMARK\_BROWSE\_CO\_OP is not valid on a non-browse MQGET call, or with any of the following options:

- MQGMO\_ALL\_MSGS\_AVAILABLE
- MQGMO\_ALL\_SEGMENTS\_AVAILABLE
- MQGMO\_COMPLETE\_MSG
- MQGMO\_LOCK
- MQGMO\_LOGICAL\_ORDER
- MQGMO\_MARK\_BROWSE\_CO\_OP
- MQGMO\_UNLOCK
- MQGMO\_UNMARKED\_BROWSE\_MSG

#### **MQGMO\_UNMARK\_BROWSE\_HANDLE**

Bu seçeneği belirten bir MQGET çağrısının ardından, bulunan iletinin artık bu tanıtıcı tarafından işaretlendiği kabul edilmiyor.

İleti, bu tanıtıcı için işaretlenmemiş olsa bile çağrı başarılı olur.

Bu seçenek, göz atma dışı MQGET çağrılarında ya da aşağıdaki seçeneklerden herhangi biriyle geçerli değildir:

- MQGMO\_ALL\_MSGS\_AVAILABLE
- MQGMO\_ALL\_SEGMENTS\_AVAILABLE
- MQGMO\_COMPLETE\_MSG
- MQGMO\_LOCK
- MQGMO\_LOGICAL\_ORDER
- MQGMO\_MARK\_BROWSE\_CO\_OP
- MQGMO\_UNLOCK
- MQGMO\_UNMARKED\_BROWSE\_MSG

### **Kilitleme seçenekleri**

Aşağıdaki seçenekler kuyruklardaki iletileri kilitlemek için ilişkilendirir:

#### **MQGMO\_LOCK**

Göz atılmasına neden olan iletiyi kilitleyin; böylece, ileti kuyruk için açık olan diğer herhangi bir tanıtıcı için görünmez olur. Bu seçenek, yalnızca aşağıdaki seçeneklerden biri de belirtilirse belirlenebilir:

- MQGMO\_BROWSE\_FIRST
- MQGMO\_BROWSE\_NEXT

- MQGMO\_BROWSE\_MSG\_UNDER\_CURSOR

Her kuyruk tanıtıcısı için yalnızca bir ileti kilitlenmiş olabilir. İleti, mantıksal bir ileti ya da fiziksel bir ileti olabilir:

- MQGMO\_COMPLETE\_MSG belirtilirse, mantıksal iletiyi oluşturan tüm ileti bölümleri kuyruk tanıtıcısı için kilitlenir. İletiler, kuyruğun üzerinde olmalı ve alma için kullanılabilir olmalıdır.
- MQGMO\_COMPLETE\_MSG'ı atlarsanız, kuyruk sapına yalnızca tek bir fiziksel ileti kilitlenir. If this message happens to be a segment of a logical message, the locked segment prevents other applications using MQGMO\_COMPLETE\_MSG to retrieve or browse the logical message.

Kilitlenen ileti, her zaman göz atma imlecinin altındaki iletidir. İleti, MQGMO\_MSG\_UNDER\_CURSOR seçeneğini belirten daha sonraki bir MQGET çağrısı tarafından kuyruktan kaldırılabilir. Kuyruk tanıtıcısını kullanan diğer MQGET çağrıları da iletiyi kaldırabilir (örneğin, kilitli iletinin ileti tanıtıcısını belirten bir çağrıdır).

If the call returns completion code MQCC\_FAILED, or MQCC\_WARNING with reason code MQRC\_TRUNCATED\_MSG\_FAILED, no message is locked.

Uygulama iletiyi kuyruktan kaldırmazsa, kilit aşağıdaki eylemlerden biri tarafından serbest bırakılır:

- Bu tanıtıcı için MQGMO\_BROWSE\_FIRST ya da MQGMO\_BROWSE\_NEXT belirtildiğinde başka bir MQGET çağrısı yayınlayın. Çağrı MQCC\_OK ya da MQCC\_WARNING ile tamamlanırsa, kilit serbest bırakılır. Çağrı MQCC\_FAILED ile tamamlanırsa, ileti kilitlenir. Ancak, aşağıdaki istisnalar geçerlidir:
  - MQCC\_WARNING MQRC\_TRUNCATED\_MSG\_FAILED ile birlikte döndürülürse, iletinin kilidi açılmaz.
  - MQCC\_FAILED MQRC\_NO\_MSG\_AVAILABLE ile birlikte döndürülürse, iletinin kilidi açılır.

MQGMO\_LOCK belirtilirse, döndürülen ileti kilitlenir. MQGMO\_LOCK'ı atlarsanız, aramadan sonra kilitli bir ileti yoktur.

MQGMO\_WAIT belirtilirse ve hemen kullanılabilir bir ileti yoksa, bekleme başlamadan önce özgün iletinin kilidi açılır.

- Issuing another MQGET call for this handle, with MQGMO\_BROWSE\_MSG\_UNDER\_CURSOR, without MQGMO\_LOCK. Çağrı MQCC\_OK ya da MQCC\_WARNING ile tamamlanırsa, kilit serbest bırakılır. Çağrı MQCC\_FAILED ile tamamlanırsa, ileti kilitlenir. Ancak, aşağıdaki kural dışı durum geçerlidir:
  - MQCC\_WARNING MQRC\_TRUNCATED\_MSG\_FAILED ile birlikte döndürülürse, iletinin kilidi açılmaz.
- Issuing another MQGET call for this handle with MQGMO\_UNLOCK.
- Tanıtıcıyı kullanarak bir MQCLOSE çağrısı yayınlayın. The MQCLOSE might be implicit, caused by the application ending.

Eşlik eden bir göz atma seçeneğini belirlemek için gereken MQ00\_BROWSE dışındaki MQGMO\_LOCK değerini belirtmek için özel bir MQOPEN seçeneği gerekli değildir.

MQGMO\_LOCK aşağıdaki seçeneklerden herhangi biriyle geçerli değildir:

- MQGMO\_MARK\_SKIP\_BACKOUT
- MQGMO\_SYNCPOINT
- MQGMO\_SYNCPOINT\_IF\_PERSISTENT
- MQGMO\_UNLOCK

## MQGMO\_UNLOCK

The message to be unlocked must have been previously locked by an MQGET call with the MQGMO\_LOCK option. Bu tanıtıcı için kilitlenmiş bir ileti yoksa, çağrı MQCC\_WARNING ve MQRC\_NO\_MSG\_LOCKED ile tamamlanır.

MQGMO\_UNLOCK belirtilirse, **MsgDesc**, **BufferLength**, **Buffer** ve **DataLength** parametreleri denetlenmez ya da değiştirilmez. *Buffer*'da ileti döndürülmez.

No special open option is required to specify MQGMO\_UNLOCK (although MQ00\_BROWSE is needed to issue the lock request in the first place).

Bu seçenek, aşağıdakiler dışında herhangi bir seçenekle geçerli değildir:

- MQGMO\_NO\_WAIT
- MQGMO\_NO\_SYNCPOINT

Bu seçeneklerin her ikisi de, belirtilip belirlenmemekte olup olmadığı varsayılır.

## İleti-veri seçenekleri

Aşağıdaki seçenekler, ileti kuyruktan okunduğunda ileti verilerinin işlenmesine ilişkin bilgi içerir:

### MQGMO\_ACCEPT\_TRUNCATED\_MSG

If the message buffer is too small to hold the complete message, allow the MQGET call to fill the buffer. MQGET , arabelleği, iletinin alabileceği kadar büyük bir iletiyle doldurur. Bu, bir uyarı tamamlama kodu yayınlar ve işleme işlemi tamamlar. Bu da şu anlama gelir:

- İletilere göz atılırken, göz at imleci döndürülen iletiye kadar ilerletilir.
- İletileri kaldırırken, döndürülen ileti kuyruktan kaldırılır.
- Başka bir hata oluşmazsa neden kodu MQRC\_TRUNCATED\_MSG\_ACCEPTED döndürülür.

Bu seçenek olmadan, arabellek tutulabildiği kadar iletinin doldurulsa da doldurulmaz. Bir uyarı tamamlama kodu verilir, ancak işleme tamamlanmaz. Bu da şu anlama gelir:

- İletilere göz atılırken göz at imleci gelişmiş değildir.
- İletileri kaldırırken, ileti kuyruktan kaldırılmaz.
- Başka bir hata oluşmazsa neden kodu MQRC\_TRUNCATED\_MSG\_FAILED döndürülür.

### MQGMO\_CONVERT

Bu seçenek, iletteki uygulama verilerini, MQGET çağrısındaki **MsgDesc** parametresinde belirtilen CodedCharSetId ve Encoding değerlerine uygun şekilde dönüştürür. Veriler, **Buffer** parametresine kopyalanmadan dönüştürüldü.

İleti konduğunda belirtilen Format alanı, iletteki verilerin niteini tanımlamak için dönüştürme işlemi tarafından kabul edilir. İleti verileri, yerleşik biçimler için kuyruk yöneticisi tarafından ve diğer biçimler için kullanıcı tarafından yazılmış bir çıkışa dönüştürülür. Veri dönüştürme çıkışıyla ilgili ayrıntılar için bkz. [“Veri dönüştürme çıkışı” sayfa 891](#) .

- Dönüştürme başarılı olursa, **MsgDesc** parametresinde belirtilen CodedCharSetId ve Encoding alanları, MQGET çağrısından geri dönmeye değiştirilmez.
- Yalnızca dönüştürme başarısız olursa, ileti verileri döndürülmezse, MsgDesc içindeki CodedCharSetId ve Encoding alanları dönüştürülmemiş ileti için dönüştürülen değerlere ilişkin değerler olarak ayarlanır. Tamamlama kodu, bu durumda MQCC\_WARNING olur.

Her iki durumda da bu alanlar, **Buffer** parametresine döndürülen ileti verilerinin karakter kümesi tanıtıcısını ve kodlamasını tanımlar.

Kuyruk yöneticisinin dönüştürmeyi gerçekleştirdiği biçim adlarının listesi için [“MQMD-İleti tanımlayıcı” sayfa 416](#) içinde açıklanan *Format* alanına bakın.

## Grup ve bölüm seçenekleri

Aşağıdaki seçenekler, mantıksal iletilerin gruplarındaki ve bölümlerindeki iletilerin işlenmesine ilişkin bilgi içerir. Seçenek açıklamalarından önce, aşağıdaki önemli koşullara ilişkin bazı tanımlamalar yer alıyor:

### Fiziksel ileti

Fiziksel ileti, bir kuyruktan yerleştirilebilecek ya da kuyruktan kaldırılabilir en küçük bilgi birimidir. Bu, genellikle tek bir MQPUT, MQPUT1 ya da MQGET çağrısında belirtilen ya da alınan bilgilere karşılık gelir. Her fiziksel ileti kendi ileti tanımlayıcısına ( MQMD) sahiptir. Genellikle, fiziksel iletiler ileti tanıtıcısı için farklı değerlere göre ayırt edilir, MQMD' deki MsgId alanı. Kuyruk yöneticisi farklı değerleri zorunlu kılmaz.

## Mantıksal ileti

Mantıksal ileti, uygulama bilgilerinin tek bir birimidir. Sistem kısıtlamalarının yokluğunda, mantıksal ileti fiziksel iletiyle aynıdır. Mantıksal iletiler büyükse, sistem kısıtlamaları, mantıksal bir iletiyi bölümler olarak adlandırılan iki ya da daha çok fiziksel iletiye bölmeyi önerebilir ya da gerekli kılabilir.

Kesimlere ayrılmış bir mantıksal ileti, MQMD içinde aynı boş değerli olmayan grup tanıtıcısına sahip iki ya da daha çok fiziksel iletten oluşur. GroupId . Bunlar, MQMD' de aynı ileti sıra numarasına, MsgSeqNumber alanına sahiptir. The segments are distinguished by differing values for the segment offset, Offset field in MQMD. Bölüm görelî konumu, mantıksal iletiden verilerin başlangıcındaki fiziksel iletteki verilerin görelî konutdur. Her bir bölüm fiziksel bir ileti olduđu için, mantıksal bir iletteki kesimlerin genellikle farklı ileti tanıtıcıları vardır.

Kesimlere ayrılmış olmayan, ancak gönderme uygulaması tarafından bölümlemeye izin verilen bir mantıksal ileti, boş değerli olmayan bir grup tanıtıcısı da içeriyor. Bu durumda, mantıksal ileti bir ileti grubuna ait değilse, o grup tanıtıcısına sahip tek bir fiziksel ileti vardır. Logical messages, for which segmentation has been inhibited by the sending application, have a null group identifier, MQGI\_NONE, unless the logical message belongs to a message group.

## İleti grubu

İleti grubu, aynı boş değerli grup tanıtıcısına sahip bir ya da daha fazla mantıksal ileti kümesidir. Gruptaki mantıksal iletiler, ileti sıra numarası için farklı değerlere göre ayırt edilir. Sıra numarası, 1-n aralığında bir tamsayıdır; burada n, gruptaki mantıksal iletilerin sayısıdır. Mantıksal iletilerden biri ya da daha fazlası bölümlendiye, grupta n fiziksel ileti sayısı daha fazla olur.

## MQGMO\_LOGICAL\_ORDER

MQGMO\_LOGICAL\_ORDER , kuyruk tanıtıcısı için ardışık MQGET çağruları tarafından iletilerin döndürüldüğü sırayı denetler. Seçeneğin her bir çağrıda belirtilmesi gerekir.

If MQGMO\_LOGICAL\_ORDER is specified for successive MQGET calls for the same queue handle, messages in groups are returned in the order of their message sequence numbers. Mantıksal iletilerin bölümleri, bölüm görelî konumlarından belirtilen sırayla döndürülür. Bu sipariş, kuyrukta ileti ve kesimlerin oluştuđu sıradan farklı olabilir.

**Not:** MQGMO\_LOGICAL\_ORDER belirtildiğinde, gruplara ait olmayan ve kesimler olmayan iletiler üzerinde herhangi bir olumsuz sonuç yoktur. Sonuç olarak, bu tür iletiler her biri tek bir iletiyle oluşan bir ileti grubuna aitmiş gibi işlem görür. Gruplardaki iletilerin, ileti kesimlerinin ve bölümlenmemiş iletilerin gruplar halinde olmadığı bir ileti karışımının bulunduğu kuyruklardan ileti alınırken MQGMO\_LOGICAL\_ORDER ' in belirtilmesi güvenlidir.

İletileri gereken sırayla geri döndürmek için, kuyruk yöneticisi, ardışık MQGET çağruları arasında grup ve bölüm bilgilerinin korur. Grup ve bölüm bilgileri, kuyruk tanıtıcısı için yürürlükteki ileti grubunu ve yürürlükteki mantıksal iletiyi tanıtır. Ayrıca, grup ve mantıksal ileti içindeki geçerli konumu ve iletilerin bir iş birimi içinde alınıp alınmadığını belirtir. Kuyruk yöneticisi bu bilgileri koruduğu için, uygulamanın her MQGET çağrısından önce grup ve bölüm bilgilerinin ayarlamaya gerek yoktur. Özellikle, uygulamanın MQMD' de GroupId, MsgSeqNumber ve Offset alanlarını ayarlamaya gerek olmadığı anlamına gelir. Ancak, uygulamanın her çağrıda MQGMO\_SYNCPOINT ya da MQGMO\_NO\_SYNCPOINT seçeneğini doğru bir şekilde ayarlaması gerekir.

Kuyruk açıldığında, yürürlükteki ileti grubu yok ve yürürlükteki mantıksal ileti yok. MQGET çağrısı içeren bir ileti MQMF\_MSG\_IN\_GROUP çağrısı tarafından döndürüldüğünde, ileti grubu yürürlükteki ileti grubu olur. Art arda çağrılarda belirtilen MQGMO\_LOGICAL\_ORDER ile, aşağıdaki gibi bir ileti döndürülünceye kadar bu grup yürürlükteki grup olarak kalır:

- MQMF\_SEGMENT olmadan MQMF\_LAST\_MSG\_IN\_GROUP (yani, gruptaki son mantıksal ileti kesimlere ayrılmış değildir) ya da
- MQMF\_LAST\_SEGMENT ile MQMF\_LAST\_MSG\_IN\_GROUP (yani, döndürülen ileti, gruptaki son mantıksal iletinin son bölüğüdür).

Böyle bir ileti döndürüldüğünde, ileti grubu sonlandırılır ve MQGET çağrısı başarıyla tamamlandığında artık geçerli bir grup olmaz. Benzer bir şekilde, MQGET çağrısı içeren bir ileti MQMF\_SEGMENT çağrısı tarafından döndürüldüğünde, mantıksal ileti yürürlükteki mantıksal ileti olur. MQMF\_LAST\_SEGMENT işaretine sahip ileti döndürüldüğünde mantıksal ileti sonlandırılır.



Seçim ölçütü belirtilmezse, ardışık MQGET çağruları, doğru sırayla, kuyrukta ilk ileti grubuna ilişkin iletileri döndürür. Daha sonra, daha fazla ileti kalmayınca kadar ikinci ileti grubunun iletilerini vb. için geri getirirler. It is possible to select the particular message groups returned by specifying one or more of the following options in the MatchOptions field:

- MQMO\_MATCH\_MSG\_ID
- MQMO\_MATCH\_CORREL\_ID
- MQMO\_MATCH\_GROUP\_ID

Ancak, bu seçenekler yalnızca, yürürlükteki ileti grubu ya da mantıksal ileti olmadığında etkindir. Ek ayrıntılar için “MQGMO-Get-message seçenekleri” sayfa 364 içinde açıklanan MatchOptions alanına bakın.

Çizelge 496 sayfa 389 , kuyruk yöneticisinin MQGET çağrısına geri dönmek üzere bir ileti bulmayı denerken arayacağı MsgId, CorrelId, GroupId, MsgSeqNumber ve Offset alanlarının değerlerini gösterir. Kurallar hem kuyruktan ileti kaldırmak, hem de kuyruklardaki iletilere göz atmak için geçerlidir. Tabloda ya Evet ya da Hayır anlamına gelir:

#### LOG ORD

Çağrıda MQGMO\_LOGICAL\_ORDER seçeneğinin belirtilip belirtilmediğini belirtir.

#### Cur grp

Çağrıdan önce geçerli bir ileti grubunun var olup olmadığını belirtir.

#### Cur log msg

Çağrıdan önce geçerli bir mantıksal iletinin var olup olmadığını belirtir.

#### Diğer kolonlar

Kuyruk yöneticisinin görünüşü için gereken değerleri gösterir. Önceki ileti, kuyruk tanıtıcısı için önceki iletide alan için döndürülen değeri belirtir.

Çizelge 496. Mantıksal iletilerin gruplarındaki ve kısımlarındaki iletilere ilişkin MQGET seçenekleri							
Belirttiğiniz seçenekler	Aramadan önce grup ve günlük-msg durumu		Kuyruk yöneticisinin bakışlarına ilişkin değerler				
	LOG ORD	Cur grp	Cur log msg	MsgId	CorrelId	GroupId	MsgSeqNumber
Evet	Hayır	Hayır	MatchOption tarafından denetlenen	MatchOption tarafından denetlenen	MatchOption tarafından denetlenen	1	0
Evet	Hayır	Evet	Herhangi bir ileti tanıtıcısı	Herhangi bir ilinti tanıtıcısı	Önceki grup tanıtıcısı	1	Önceki görel konum + önceki bölüm uzunluğu
Evet	Evet	Hayır	Herhangi bir ileti tanıtıcısı	Herhangi bir ilinti tanıtıcısı	Önceki grup tanıtıcısı	Önceki sıra numarası + 1	0
Evet	Evet	Evet	Herhangi bir ileti tanıtıcısı	Herhangi bir ilinti tanıtıcısı	Önceki grup tanıtıcısı	Önceki sıra numarası	Önceki görel konum + önceki bölüm uzunluğu
Hayır	Herhangi biri	Herhangi biri	MatchOption tarafından denetlenen	MatchOption tarafından denetlenen	MatchOption tarafından denetlenen	MatchOption tarafından denetlenen	MatchOption tarafından denetlenen

Kuyrukta birden çok ileti grubu varsa ve geri dönmeye hak kazanırsa, gruplar, her gruptaki ilk mantıksal iletinin ilk bölümünün kuyruğunda belirlenen sırayla döndürülür. Yani, ileti sıra numaraları 1 ve görelî konum 0 olan fiziksel iletiler, uygun grupların döndürülebiyecek sırayı belirler.

MQGMO\_LOGICAL\_ORDER seçeneđi, iş birimlerini aşıđıdaki gibi etkiler:

- Bir iş birimi içinde bir gruptaki ilk mantıksal ileti ya da bölüm alınır, aynı kuyruk tanıtıcısı kullanır, gruptaki diđer tüm mantıksal iletiler ve kesimler bir iş birimi içinde alınmalıdır. Ancak, bunlar aynı iş birimi içinde alınmaları gerekir. Bu, birçok fiziksel iletilerden oluşan bir ileti grubunun kuyruk tanıtıcısı için iki ya da daha fazla sayıda iş birimi bölünmesine olanak sağlar.
- Bir gruptaki ilk mantıksal ileti ya da bölüm, bir iş birimi içinde alınmaz ve aynı kuyruk tanıtıcısı kullanılır, gruptaki diđer mantıksal ileti ve kesimlerin hiçbirini bir iş birimi içinde alamaz.

Bu koşullar karşılanmazsa, MQGET çağırısı neden kodu MQRC\_INCONSISTENT\_UOW ile başarısız olur.

MQGMO\_LOGICAL\_ORDER belirtildiğinde, MQGET çağırısında belirtilen MQGMO , MQGMO\_VERSION\_2 değerinden küçük olamaz ve MQMD MQMD\_VERSION\_2 değerinden küçük olmamalıdır. Bu koşul karşılanmazsa, uygun olduđu şekilde, çağrı neden kodu MQRC\_WRONG\_GMO\_VERSION ya da MQRC\_WRONG\_MD\_VERSION ile başarısız olur.

If MQGMO\_LOGICAL\_ORDER is not specified for successive MQGET calls for the queue handle, messages are returned without regard for whether they belong to message groups, or whether they are segments of logical messages. Bu, belirli bir gruptan ya da mantıksal ileten gelen iletilerin ya da bölümlerin, diđer gruplardan ya da mantıksal iletilerden gelen iletilerle ya da bölümlerle ya da gruplarda olmayan ve kesimler olmayan iletilerle birlikte gönderilebileceđi anlamına gelir. Bu durumda, art arda MQGET çağrıları tarafından döndürülen iletiler, bu çağrılarda belirtilen MQMO\_\* seçenekleri tarafından denetlenir (bu seçeneklerin ayrıntıları için "[MQGMO-Get-message seçenekleri](#)" sayfa 364 ' de açıklanan *MatchOptions* alanına bakın).

Sistem başarısızlığı oluştuktan sonra, ortadaki bir ileti grubunu ya da mantıksal iletiyi yeniden başlatmak için kullanılabilir teknik budur. When the system restarts, the application can set the GroupId, MsgSeqNumber, Offset, and MatchOptions fields to the appropriate values, and then issue the MQGET call with MQGMO\_SYNCPOINT or MQGMO\_NO\_SYNCPOINT set, but without specifying MQGMO\_LOGICAL\_ORDER. If this call is successful, the queue manager retains the group and segment information, and subsequent MQGET calls using that queue handle can specify MQGMO\_LOGICAL\_ORDER as normal.

Kuyruk yöneticisinin MQGET çağırısı için sakladığı grup ve kesim bilgileri, MQPUT çağırısına ilişkin sakladığı grup ve bölüm bilgilerinden ayrıdır. Ayrıca, kuyruk yöneticisi aşıđıdaki ile ilişkin ayrı bilgileri saklar:

- MQGET , iletileri kuyruktan kaldırma çağrıları sağlar.
- MQGET , kuyruklardaki iletilere göz atmalarını sağlar.

For any given queue handle, the application can mix MQGET calls that specify MQGMO\_LOGICAL\_ORDER with MQGET calls that do not. Ancak, aşıđıdaki noktalara dikkat edin:

- MQGMO\_LOGICAL\_ORDER' ı atlıyorsanız, her başarılı MQGET çağırısı, kuyruk yöneticisinin kayıtlı grup ve bölüm bilgilerini döndürülen iletiye karşılık gelen değerlere ayarlamasına neden olur; bu durum, kuyruk tanıtıcısı için kuyruk yöneticisi tarafından saklanan varolan grup ve kesim bilgilerinin yerini alır. Yalnızca arama işlemine uygun olan bilgiler (göz atma ya da kaldırma) deđiştirilir.
- If you omit MQGMO\_LOGICAL\_ORDER, the call does not fail if there is a current message group or logical message; the call might succeed with an MQCC\_WARNING completion code. [Çizelge 497 sayfa 391](#) , oluşabilecek çeşitli vakaları gösterir. Bu durumlarda, tamamlanma kodu MQCC\_OK deđilse, neden kodu aşıđıdakilerden biridir (uygun olduđu gibi):
  - MQRC\_INCOMPLETE\_GROUP
  - MQRC\_INCOMPLETE\_MSG
  - MQRC\_INCONSISTENT\_UOW

**Not:** Kuyruk yöneticisi, bir kuyruğa göz atılırken grup ve bölüm bilgilerini denetmez ya da göz atma için açılan ancak giriş olmayan bir kuyruğu kapatırken, bu durumda tamamlama kodu her zaman MQCC\_OK olur (başka hata olmadığını varsayarlar).

Çizelge 497. MQGET ya da MQCLOSE çağrısının grup ve bölüm bilgileriyle tutarlı olmadığı durumlarda sonuç		
Yürürlükteki çağrı	Önceki arama MQGMO_LOGICAL_ORDER ile MQGET idi	Previous call was MQGET without MQGMO_LOGICAL_ORDER
MQGET şu ürünü geçir MQGMO_LOGICAL_ORDER	MQCC_FAILED	MQCC_FAILED
MQGMO_LOGICAL_ORDER olmadan MQGET	MQCC_WARNING	MQCC_OK
Sonlandırılmamış bir grupla ya da mantıksal iletiyle MQCLOSE	MQCC_WARNING	MQCC_OK

Mantıksal sırayla iletileri ve bölümleri almak isteyen uygulamaların, kullanılması gereken en basit seçenek olduğu için MQGMO\_LOGICAL\_ORDER' u belirtmeleri önerilir. Bu seçenek, kuyruk yöneticisi bu bilgileri yönettiği için, grup ve bölüm bilgilerini yönetme gereksiniminin uygulanını giderir. Ancak, özelleştirilmiş uygulamaların MQGMO\_LOGICAL\_ORDER seçeneği tarafından sağlanandan daha fazla denetime gereksinimi olabilir ve bu, bu seçeneği belirtmeyerek elde edilebilir. The application must then ensure that the MsgId, CorrelId, GroupId, MsgSeqNumber, and Offset fields in MQMD, and the MQMO\_\* options in MatchOptions in MQGMO, are set correctly, before each MQGET call.

Örneğin, bu iletilerin gruplar halinde mi, yoksa mantıksal ileti bölümleri mi olduğunu dikkate almadan, aldığı fiziksel iletileri iletmek isteyen bir uygulama, MQGMO\_LOGICAL\_ORDER belirtmemelidir. Kuyruk yöneticileri gönderme ve alma arasında birden çok yol içeren karmaşık bir ağda, fiziksel iletiler sıradan çıkabilirler. MQPUT çağrısında ne MQGMO\_LOGICAL\_ORDER, ne de karşılık gelen MQPMO\_LOGICAL\_ORDER belirtilirse, iletmeye uygulaması her fiziksel iletiyi alır almaz, gelecek mantıksal sırada beklemek zorunda kalmadan, her fiziksel iletiyi alabilir ve iletebilirler.

You can specify MQGMO\_LOGICAL\_ORDER with any of the other MQGMO\_\* options, and with various of the MQMO\_\* options in appropriate circumstances (see preceding section).

- **z/OS** On z/OS, this option is supported for private and shared queues, but the queue must have an index type of MQIT\_GROUP\_ID. Paylaşılan kuyruklar için, kuyruk eşlemelerinin CFLEVEL (3) ya da sonraki bir yerde olması gereken CFSTRUCT nesnesi.
- Bu seçenek, aşağıdaki altyapılara ilişkin tüm yerel kuyruklar için desteklenir:

- **AIX** AIX
- **Linux** Linux
- **IBM i** IBM i
- **Windows** Windows

ve bu sistemlere bağlı IBM MQ MQI clients için.

### MQGMO\_COMPLETE\_MSG

MQGET çağrısı, yalnızca tam bir mantıksal ileti döndürülebilir. Mantıksal ileti bölümlenmiş bir iletiyse, kuyruk yöneticisi kesimleri yeniden oluşturur ve uygulamaya ilişkin tam mantıksal iletiyi döndürür; mantıksal iletinin bölümlenmiş olması, uygulamanın alınırken uygulamanın belirgin olmamasını sağlar.

**Not:** Kuyruk yöneticisinin ileti bölümlerini yeniden bir araya getirmesine neden olan tek seçenek budur. Belirlenmezse, kuyruklarda varsa, bölümler uygulamaya tek tek döndürülür (ve MQGET çağrısında belirlenen diğer seçim ölçütlerine uymuyorsa). Tek tek parça almak istemeyen uygulamaların her zaman MQGMO\_COMPLETE\_MSG belirtmesi gerekir.

Bu seçeneği kullanmak için uygulama, iletiyi tamamlamak için yeterli büyüklükte bir arabellek sağlamalıdır ya da MQGMO\_ACCEPT\_TRUNCATED\_MSG seçeneğini belirtmelidir.

Kuyruk bazı kesimlere ayrılmış iletiler içeriyorsa (ağda geciktikleri için ve henüz gelmedikleri için), MQGMO\_COMPLETE\_MSG belirtildiğinde eksik mantıksal iletilere ait bölümlerin alınmasını önler. Ancak, bu ileti bölümleri yine de **CurrentQDepth** kuyruk özniteliğinin değerine katkıda bulunur; bu, *CurrentQDepth* sıfırdan büyük olmasına rağmen yeniden alınabilecek mantıksal iletiler olabileceği anlamına gelir.

Kalıcı iletiler için, kuyruk yöneticisi kesimleri yalnızca bir iş birimi içinde yeniden birleştirebilir:

- MQGET çağrısı, kullanıcı tanımlı bir iş birimi içinde çalışıyorsa, bu iş birimi kullanılır. Arama, yeniden çevirme işlemi sırasında başarısız olursa, kuyruk yöneticisi yeniden birleştirme sırasında kaldırılan kesimleri yeniden kuyruğa almak için geri yükler. Ancak, hata, iş biriminin başarıyla kesinleştirilmesini engellememektedir.
- Arama, kullanıcı tanımlı bir iş birimi dışında çalışıyorsa ve var olan bir kullanıcı tanımlı iş birimi yoksa, kuyruk yöneticisi arama süresi için bir iş birimi yaratır. Arama başarılı olursa, kuyruk yöneticisi iş birimini otomatik olarak kesinleştirir (uygulamanın bu işlemi yapması gerekmez). Arama başarısız olursa, kuyruk yöneticisi iş birimini geri arar.
- Arama, kullanıcı tanımlı bir iş birimi dışında çalışıyorsa, ancak kullanıcı tanımlı bir iş birimi varsa, kuyruk yöneticisi yeniden bir araya getiremez. İleti yeniden yapılanmasını gerektirmiyorsa, arama hala başarılı olabilir. Ancak ileti yeniden yapılanmasını gerektiriyorsa, arama başarısız olur ve neden kodu MQRC\_UOW\_NOT\_AVAILABLE olur.

Kalıcı olmayan iletiler için, kuyruk yöneticisi yeniden birleştirme gerçekleştirmek için bir iş birimi kullanılabilir durumda olmasını gerektirmez.

Bir kesim olan her fiziksel ileti, kendi ileti tanımlayıcısına sahiptir. Tek bir mantıksal ileti oluşturan kesimler için, ileti tanımlayıcısındaki alanların çoğu, mantıksal iletteki tüm kesimler için aynıdır; tipik olarak, mantıksal iletteki kesimler arasında farklılık gösteren yalnızca *MsgId*, *Offset* ve *MsgFlags* alanlarıdır. Ancak, bir bölüm, ara kuyruk yöneticisinde bir kesilen iletiler kuyruğuna yerleştirilirse, DLQ işleyicisi MQGMO\_CONVERT seçeneğini belirten iletiyi alır ve bu, karakter kümesi ya da kesimin kodlamasıyla sonuçlanabilir. DLQ işleyicisi, kesimi kendi yolunda başarıyla gönderirse, kesim hedef kuyruk yöneticisine geldiğinde mantıksal iletiden farklı kesimlerden farklı bir karakter takımı ya da kodlama olabilir.

*CodedCharSetId* ve *Encoding* alanlarının farklı olduğu bölümlerden oluşan mantıksal bir ileti, kuyruk yöneticisi tarafından tek bir mantıksal iletiyle yeniden birleştirilemez. Instead, the queue manager reassembles and returns the first few consecutive segments at the start of the logical message that have the same character-set identifiers and encodings, and the MQGET call completes with completion code MQCC\_WARNING and reason code MQRC\_INCONSISTENT\_CCIDS or MQRC\_INCONSISTENT\_ENCODINGS, as appropriate. Bu, MQGMO\_CONVERT ' un belirtilip belirtilmediğine bakılmaksızın gerçekleşir. To retrieve the remaining segments, the application must reissue the MQGET call without the MQGMO\_COMPLETE\_MSG option, retrieving the segments one by one. MQGMO\_LOGICAL\_ORDER , siparişteki kalan bölümleri almak için kullanılabilir.

Bölümleri koyan bir uygulama, ileti tanımlayıcısındaki diğer alanları, bölümler arasında farklılık gösteren değerlere de ayarlayabilir. Ancak, alma uygulaması mantıksal iletiyi almak için MQGMO\_COMPLETE\_MSG kullanıyorsa bu işlemi gerçekleştirmenin bir avantajı yoktur. Kuyruk yöneticisi bir mantıksal iletiyi yeniden derlerken, ileti tanımlayıcısında ilk kesime ilişkin ileti tanımlayıcısındaki değerleri döndürür; tek kural dışı durum *MsgFlags* alanıdır; kuyruk yöneticisi, yeniden birleştirilen iletinin tek kesim olduğunu göstermek için ayarlar.

Bir rapor iletisi için MQGMO\_COMPLETE\_MSG belirtilmişse, kuyruk yöneticisi özel işleme gerçekleştirir. Kuyruk yöneticisi, kuyrukta mantıksal iletiyle ilgili farklı bölümlerle ilgili rapor tipine ilişkin tüm rapor iletilerinin olup olmadığını görmek için kuyruğu denetler. Bunlar varsa, MQGMO\_COMPLETE\_MSG belirtilerek tek bir ileti olarak alınabilirler. Bunun mümkün olması için, rapor iletileri bir kuyruk yöneticisi tarafından ya da bölümlendirmeyi destekleyen bir MCA tarafından oluşturulmalıdır; ya da kaynak uygulama en az 100 bayt ileti verisi istemelidir (yani, uygun MQRO\_\*\_WITH\_DATA ya da MQRO\_\*\_WITH\_FULL\_DATA seçenekleri belirtilmelidir). Bir kesime ilişkin

uygulama verilerinin tam miktarından azsa, eksik byte 'lar, döndürülen rapor iletisinde boş değerler ile değiştirilir.

MQGMO\_MSG\_UNDER\_CURSOR ya da MQGMO\_BROWSE\_MSG\_UNDER\_CURSOR ile MQGMO\_COMPLETE\_MSG belirtilirse, göz atma imlecinin MQMD içindeki *Offset* alanı 0 değerine sahip bir ileti üzerine konumlandırılması gerekir. Bu koşul karşılanmazsa, çağrı neden kodu MQRC\_INVALID\_MSG\_UNDER\_CURSOR ile başarısız olur.

MQGMO\_COMPLETE\_MSG implies MQGMO\_ALL\_SEGMENTS\_AVAILABLE, which need not therefore be specified.

MQGMO\_COMPLETE\_MSG , diğer MQGMO\_\* seçeneklerinin herhangi biriyle MQGMO\_SYNCPOINT\_IF\_PERSISTENT dışında ve MQMO\_MATCH\_OFFSET ögesinden ayrı olan MQMO\_\* seçenekleriyle belirtilebilir.

- **z/OS** z/OS' ta bu seçenek özel ve paylaşılan kuyruklar için desteklenir, ancak kuyruğun bir MQIT\_GROUP\_ID izin tipine sahip olması gerekir. Paylaşılan kuyruklar için, kuyruk eşleminin CFLEVEL (3) ya da daha sonraki bir yerde olması gereken CFSTRUCT nesnesinden biri.

• Aşağıdaki altyapılarda:

- **AIX** AIX
- **IBM i** IBM i
- **Linux** Linux
- **Windows** Windows

ve bu sistemlere bağlı IBM MQ MQI clients için, bu seçenek tüm yerel kuyruklar için desteklenir.

### MQGMO\_ALL\_MSGS\_AVAILABLE

Bir gruptaki iletiler, yalnızca gruptaki tüm iletiler kullanılabilir olduğunda alınabilmeye hazır olur. Kuyrukta bazı iletiler eksik (ağda gecikmiş olduğundan ve henüz gelmedikleri için) içeren ileti grupları varsa, MQGMO\_ALL\_MSGS\_AVAILABLE belirtildiğinde, eksik gruplara ait iletilerin alınmasını engeller. Ancak, bu iletiler yine de **CurrentQDepth** kuyruk özniteliğinin değerine katkıda bulunur; bu, CurrentQDepth sıfırdan büyük olmasına rağmen yeniden alınabilecek ileti grupları olabileceğinden anlamına gelir. Yeniden alınabilir başka bir ileti yoksa, belirtilen bekleme aralığı (varsa) sona erdikten sonra neden kodu MQRC\_NO\_MSG\_AVAILABLE döndürülmez.

MQGMO\_ALL\_MSGS\_AVAILABLE işleminin işlenmesi MQGMO\_LOGICAL\_ORDER ' un da belirtilip belirtilmediğine bağlıdır:

- Her iki seçenek de belirtilirse, MQGMO\_ALL\_MSGS\_AVAILABLE yalnızca geçerli grup ya da mantıksal ileti olmadığında bir etkiye sahip olur. Geçerli bir grup ya da mantıksal ileti varsa, MQGMO\_ALL\_MSGS\_AVAILABLE yoksayılar. Bu, MQGMO\_ALL\_MSGS\_AVAILABLE ' un iletileri mantıksal sırada işlerken üzerinde kalabileceği anlamına gelir.
- If MQGMO\_ALL\_MSGS\_AVAILABLE is specified without MQGMO\_LOGICAL\_ORDER, MQGMO\_ALL\_MSGS\_AVAILABLE always has an effect. Bu, gruptaki geri kalan iletileri kaldırabilmek için, gruptaki ilk iletinin kuyruktan kaldırılmasından sonra, seçeneğin kapatılması gerektiği anlamına gelir.

Successful completion of an MQGET call specifying MQGMO\_ALL\_MSGS\_AVAILABLE means that at the time that the MQGET call was issued, all the messages in the group were on the queue. Ancak, diğer uygulamaların yine de gruptaki iletileri kaldırabileceğinin farkında olun (grup, gruptaki ilk iletiyi alan uygulamaya kilitlemez).

Bu seçeneği çıkarırsanız, grup tamamlanmamış olsa bile gruplara ait iletiler alınabilir.

MQGMO\_ALL\_MSGS\_AVAILABLE implies MQGMO\_ALL\_SEGMENTS\_AVAILABLE, which need not therefore be specified.

MQGMO\_ALL\_MSGS\_AVAILABLE , diğer MQGMO\_\* seçeneklerinden herhangi biriyle ve herhangi bir MQMO\_\* seçeneği ile belirtilebilir.

- **z/OS** z/OS' ta bu seçenek özel ve paylaşılan kuyruklar için desteklenir, ancak kuyruğun bir MQIT\_GROUP\_ID izin tipine sahip olması gerekir. Paylaşılan kuyruklar için, kuyruk eşleminin CFLEVEL (3) ya da daha sonraki bir yerde olması gereken CFSTRUCT nesnesinden biri.

- Aşağıdaki altyapılarda:

- **AIX** AIX
- **IBM i** IBM i
- **Linux** Linux
- **Windows** Windows

ve bu sistemlere bağlı IBM MQ MQI clients için, bu seçenek tüm yerel kuyruklar için desteklenir.

### MQGMO\_ALL\_SEGMENTS\_AVALABILIR

Mantıksal bir iletteki kesimler, mantıksal iletiyle ilgili tüm kesimler kullanılabilir olduğunda alınabilmeye hazır olur. Kuyruk bazı kesimlere ayrılmış iletiler içeriyorsa (ağda geciktikleri için ve henüz gelmedikleri için), MQGMO\_ALL\_SEGMENTS\_AVAILABLE belirtildiğinde eksik mantıksal iletilere ait bölümlerin alınmasını önler. Ancak, bu kesimler yine de **CurrentQDepth** kuyruk özniteliğinin değerine katkıda bulunur; bu, CurrentQDepth sıfırdan büyük olsa bile, yeniden alınabilir mantıksal ileti olabileceğinden söz edilir. Yeniden alınabilir başka bir ileti yoksa, belirtilen bekleme aralığı (varsa) sona erdikten sonra neden kodu MQRC\_NO\_MSG\_AVAILABLE döndürülmez.

MQGMO\_ALL\_SEGMENTS\_AVAILABLE işleminin işlenmesi MQGMO\_LOGICAL\_ORDER ' un da belirtilip belirtilmediğine bağlıdır:

- Her iki seçenek de belirtilirse, MQGMO\_ALL\_SEGMENTS\_AVAILABLE yalnızca geçerli mantıksal ileti olmadığında bir etkiye sahiptir. Geçerli bir mantıksal ileti varsa, MQGMO\_ALL\_SEGMENTS\_AVAILABLE yoksayılr. Bu, MQGMO\_ALL\_SEGMENTS\_AVAILABLE ' un iletileri mantıksal sırada işlerken üzerinde kalabileceği anlamına gelir.
- If MQGMO\_ALL\_SEGMENTS\_AVAILABLE is specified without MQGMO\_LOGICAL\_ORDER, MQGMO\_ALL\_SEGMENTS\_AVAILABLE always has an effect. Bu, mantıksal iletiden kalan bölümleri kaldırabilmek için, mantıksal iletiden gelen ilk kesimin kuyruktan kaldırılmasından sonra, seçeneğin kapatılması gerektiği anlamına gelir.

Bu seçenek belirlenmezse, mantıksal ileti tamamlanmamış olsa bile ileti bölümleri alınabilir.

Hem MQGMO\_COMPLETE\_MSG hem de MQGMO\_ALL\_SEGMENTS\_AVAILABLE , herhangi birinin alınabilmesi için tüm kesimlerin kullanılabilir olmasını gerektirirken, eski ileti tam iletiyi döndürür, ancak ikincisi kesimlerin birer birer alınmasını sağlar.

Bir rapor iletisi için MQGMO\_ALL\_SEGMENTS\_AVAILABLE belirtilmişse, kuyruk yöneticisi, tam mantıksal iletiyi oluşturan her bir kesim için en az bir rapor iletisi olup olmadığını görmek için kuyruğu denetler. Varsa, MQGMO\_ALL\_SEGMENTS\_AVAILABLE koşulu karşılanır. Ancak, kuyruk yöneticisi, rapor iletilerinin *tipini* denetmez ve bu nedenle, rapor iletilerinde mantıksal iletinin bölümleriyle ilgili rapor tiplerinin bir karışımı olabilir. Sonuç olarak, MQGMO\_ALL\_SEGMENTS\_AVAILABLE 'in başarılı MQGMO\_COMPLETE\_MSG ' in başarılı olacağı anlamına gelmez. Belirli bir mantıksal iletinin kesimleri için bir rapor tipi karışımı varsa, bu rapor iletileri birer birer alınmalıdır.

Diğer MQGMO\_\* seçeneklerinden herhangi biriyle ve herhangi bir MQMO\_\* seçeneği ile MQGMO\_ALL\_SEGMENTS\_AVAILABLE ögesini belirtebilirsiniz.

- z/OS' ta bu seçenek özel ve paylaşılan kuyruklar için desteklenir, ancak kuyruğun bir MQIT\_GROUP\_ID izin tipine sahip olması gerekir. Paylaşılan kuyruklar için, kuyruk eşleminin CFLEVEL (3) ya da daha sonraki bir yerde olması gereken CFSTRUCT nesnesinden biri.

- Aşağıdaki altyapılarda:

- **AIX** AIX
- **IBM i** IBM i

-  Linux
-  Windows

ve bu sistemlere bağı IBM MQ MQI clients için, bu seçenek tüm yerel kuyruklar için desteklenir.

## Özellik seçenekleri

Aşağıdaki seçenekler iletinin özellikleriyle ilgilidir:

### MQGMO\_PROPERTIES\_AS\_Q\_DEF

İletinin özellikleri (ya da uzantısı), ileti tanımlayıcısında (ya da uzantısında) bulunanlar dışında, **PropertyControl** kuyruk özneliği tarafından tanımlandığı şekilde gösterilmelidir. Bir `MsgHandle` sağlandıysa, bu seçenek yoksayılr ve iletinin özellikleri, **PropertyControl** kuyruk özneliğinin değeri `MQPROP_FORCE_MQRFH2`değilse, `MsgHandle` aracılığıyla kullanılabilir.

Özellik seçeneği belirlenmezse, bu varsayılan işlemdir.

### MQGMO\_PROPERTIES\_IN\_HANDLE

İletinin özellikleri, `MsgHandle` aracılığıyla kullanılabilir kılınmalıdır. Bir ileti tanıtıcısı sağlanmazsa, çağrı başarısız olur; neden `MQRC_HMSG_ERROR` ile başarısız olur.

**Not:** İleti daha sonra, ileti tanıtıcısı oluşturmayan bir uygulama tarafından okunduysa, kuyruk yöneticisi ileti özelliklerini bir `MQRFH2` yapısına yerleştirir. Beklenmeyen bir `MQRFH2` üstbilgisi varlığının, var olan bir uygulamanın davranışını kesintiye uğradığını da bulabilirsiniz.

### MQGMO\_NO\_PROPERTIES

İleti tanımlayıcısının (ya da uzantısında) içerilenler dışında, iletinin hiçbir özelliği alınacak. Bir `MsgHandle` belirtilirse, bu değer yoksayılacak.

### MQGMO\_PROPERTIES\_FORCE\_MQRFH2

İletinin özellikleri, ileti tanımlayıcısında (ya da uzantısında) içerilenler dışında, `MQRFH2` üstbilgileri kullanılarak gösterilmelidir. Bu, özellikleri almayı bekleyen, ancak ileti tanıtıcılarını kullanacak şekilde değiştirilememiş olan uygulamaların önceki sürümleriyle uyumluluk sağlar. Bir `MsgHandle` belirtilirse, bu değer yoksayılr.

### MQGMO\_PROPERTIES\_COMPATIBILITY

If the message contains a property with a prefix of "**mcd.**", "**jms.**", "**usr.**", or "**mqext.**", all message properties are delivered to the application in an `MQRFH2` header. Ters durumda, ileti tanımlayıcısında (ya da uzantısında) yer alan durumlar dışında, iletinin tüm özellikleri atılır ve artık bu uygulamanın erişilemeyecek şekilde bulunur.

## Varsayılan seçenek

Tanımlanan seçeneklerden hiçbiri gerekli değilse, aşağıdaki seçenek kullanılabilir:

### MQGMO\_NONE

Başka bir seçenek belirtilmemeyi belirtmek için bu değeri kullanın; tüm seçenekler varsayılan değerlerini kabul eder. `MQGMO_NONE`, program belgelerine yardımcı olur; bu seçeneğin diğeriyle birlikte kullanılması amaçlanmaz, ancak değeri sıfır olduğu için, bu tür bir kullanım saptanamaz.

Options alanının başlangıç değeri, `MQGMO_NO_WAIT` artı `MQGMO_PROPERTIES_AS_Q_DEF`' dir.

### **WaitInterval (MQUZE)**

`MQGET` çağrısının gelmesi için uygun bir ileti beklemesi (yani, `MQGET` çağrısının **MsgDesc** parametresinde belirtilen seçim ölçütlerini karşılayan bir iletidir), milisaniye olarak ifade edilen yaklaşık süredir.

**Önemli:** Hemen uygun bir ileti varsa, bekleme ya da gecikme yoktur.



Daha ayrıntılı bilgi için “MQMD-İleti tanımlayıcı” sayfa 416 içinde açıklanan *MsgId* alanına bakın). Bu süre geçtikten sonra uygun bir ileti gelmezse, çağrı MQCC\_FAILED ve neden kodu MQRC\_NO\_MSG\_AVAILABLE ile tamamlanır.

z/OS işletim sisteminde, MQGET çağrısının gerçekte bekleyeceği süre sistem yükleme ve iş zamanlama konularından etkilenir ve *WaitInterval* için belirtilen değer ile *WaitInterval*' tan yaklaşık 100 milisaniye daha büyük arasında değişiklik gösterebilir.

*WaitInterval* , MQGMO\_WAIT ya da MQGMO\_SET\_SIGNAL seçeneği ile birlikte kullanılır. Bunların hiçbiri belirtilmediyse bu değer dikkate alınmaz. Bunlardan biri belirtilirse, *WaitInterval* sıfırdan büyük ya da sıfıra eşit olmalı ya da şu özel değere sahip olmalıdır:

#### **MQWI\_UNSNMA**

Sınırsız bekleme aralığı.

Bu alanın başlangıç değeri 0' dir.

#### **Signal1 (MQUZA)**

Bu, yalnızca MQGMO\_SET\_SIGNAL seçeneğiyle birlikte kullanılan bir giriş alanıdır; bir ileti kullanılabilir olduğunda teslim edilecek sinyali tanımlar.

**Not:** Bu alanın veri tipi ve kullanımı ortam tarafından belirlenir; bu nedenle, farklı ortamlar arasında bağlantı noktası yapmak istediğiniz uygulamalar sinyal kullanmamalıdır.

- z/OS' ta bu alan, bir Olay Denetim Bloğu (ECB) adresini içermelidir. MQGET çağrısı yayınlanmadan önce, ECB ' nin uygulama tarafından temizlenmesi gerekir. ECB ' yi içeren saklama alanı, kuyruk kapatılıncaya kadar serbest bırakılmamalı. ECB, tanımlanan sinyal tamamlama kodlarından biriyle kuyruk yöneticisi tarafından gönderilir. These completion codes are set in bits 2 through 31 of the ECB, the area defined in the z/OS mapping macro IHAECB as being for a user completion code.
- Diğer tüm ortamlarda, bu ayrılmış bir alandır; değeri önemli değildir.

Sinyal tamamlama kodları şunlardır:

#### **MQEC\_MSG\_GELDI**

Kuyruğa uygun bir ileti geldi. Bu ileti çağırıcı için ayrılmadı; ikinci bir MQGET isteği yayınlanmalıdır, ancak ikinci istek yapılmadan önce başka bir uygulama iletiyi alabilir.

#### **MQEC\_WAIT\_INTERVAL\_EXERD**

Belirtilen *WaitInterval* belirtilen uygun bir ileti olmadan süresi doldu.

#### **MQEC\_WAIT\_IPTAL edildi**

Belirsiz bir nedenle (kuyruk yöneticisi sonlandırılması ya da kuyruk geçersiz kılınan kuyruk gibi) bekleme işlemi iptal edildi. Daha ayrıntılı tanılama istiyorsanız, isteği yeniden yayınlayın.

#### **MQEC\_Q\_MGR QUIESCING**

Kuyruk yöneticisi susturma durumuna girdiği için bekleme iptal edildi (MQGET çağrısında MQGMO\_FAIL\_IF QUIESCING belirtildi).

#### **MQEC\_CONNECTION QUIESCING**

Bağlantı, susturma durumuna girdiği için iptal edildi (MQGET çağrısında MQGMO\_FAIL\_IF QUIESCING belirtildi).

Bu alanın ilk değeri ortam tarafından belirlenir:

- z/OS üzerinde, ilk değer boş değerli göstergedir.
- Diğer tüm ortamlarda, ilk değer 0 olur.

#### **Signal2 (MQUZA)**

Bu giriş alanı, yalnızca MQGMO\_SET\_SIGNAL seçeneğiyle birlikte kullanılan bir giriş alanıdır. Ayrılmış bir alandır; değeri anlamlı değildir.

Bu alanın başlangıç değeri 0' dir.

#### **ResolvedQName (MQCHAR48)**



Bu, kuyruk yöneticisinin, yerel kuyruk yöneticisine tanımlandığı şekilde, iletinin alındığı kuyruğun yerel adına ayarlandığı bir çıkış alanıdır. Bu, aşağıdaki durumlarda kuyruğu açmak için kullanılan addan farklıdır:

- Bir diğer ad kuyruğu açıldı (bu durumda, diğer adın çözüldüğü yerel kuyruğun adı döndürülür) ya da
- Bir model kuyruğu açıldı (bu durumda, dinamik yerel kuyruğun adı döndürülür).

Bu alanın uzunluğu MQ\_Q\_NAME\_LENGTH değeri tarafından verilir. Bu alanın başlangıç değeri, C ' de boş değer dizisidir ve diğer programlama dillerindeki 48 boş karakter dizilimidir.

### **MatchOptions (MQUZE)**

Bu seçenekler, uygulamanın, MQGET çağrısının döndürdüğü iletiyi seçmek için kullanılacak **MsgDesc** parametresindeki alanları seçmesine olanak tanır. Uygulama, bu alanda gerekli seçenekleri belirler ve **MsgDesc** parametresindeki ilgili alanları, bu alanlar için gereken değerlere ayarlar. Yalnızca, ileti için MQMD ' de bu değerleri içeren iletiler, MQGET çağrısındaki **MsgDesc** değiştirgesini kullanarak alma için adaydır. Döndürülemeyecek ileti seçilirken, karşılık gelen eşleşme seçeneğinin belirlenmediği alanlar yoksayılar. MQGET çağrısında seçim ölçütü belirtmediyseniz (yani, *herhangi biri* iletisi kabul edilebilir), *MatchOptions* olarak MQMO\_NONE olarak ayarlayın.

- z/OS' ta, kullanılacak seçim ölçütleri, kuyruk için kullanılan dizin tipi ile sınırlanmış olabilir. Ek ayrıntılar için **IndexType** kuyruk özniteliğine bakın.

MQGMO\_LOGICAL\_ORDER belirtilirse, sonraki MQGET çağrısıyla döndürülecek yalnızca bazı iletiler kullanılabilir:

- If there is no current group or logical message, only messages that have *MsgSeqNumber* equal to 1 and *Offset* equal to 0 are eligible for return. Bu durumda, uygun iletilerden hangilerinin döndürülmesini seçmek için aşağıdaki eşleşme seçeneklerinden birini ya da birkaçını kullanabilirsiniz:

- MQMO\_MATCH\_MSG\_ID
- MQMO\_MATCH\_COREL\_ID
- MQMO\_MATCH\_GROUP\_ID

- Geçerli bir grup ya da mantıksal ileti varsa, mantıksal iletide yalnızca gruptaki ya da sonraki bölümdeki sonraki ileti geri dönmek için uygun olur ve MQMO\_\* seçenekleri belirlenerek bu ileti değiştirilemez.

Önceki durumların her ikisinde de, geçerli olmayan eşleşme seçeneklerini belirtebilirsiniz; ancak, **MsgDesc** parametresindeki ilgili alanın değeri, döndürülecek iletiyle ilgili alanın değeriyle eşleşmelidir; çağrılan hata, MQRC\_MATCH\_OPTIONS\_ERROR neden koduyla başarısız olur; bu koşul karşılanmaz.

*MatchOptions* is ignored if you specify either MQGMO\_MSG\_UNDER\_CURSOR or MQGMO\_BROWSE\_MSG\_UNDER\_CURSOR.

İleti özelliğine dayalı olarak ileti alınması eşleştirme seçenekleri kullanılarak gerçekleştirilmez; daha fazla bilgi için bkz. "SelectionString (MQCHARV)" sayfa 486.

Aşağıdaki eşleşme seçeneklerinden birini ya da birkaçını belirtebilirsiniz:

#### **MQMO\_MATCH\_MSG\_ID**

Alınacak iletinin, MQGET çağrısının **MsgDesc** parametresindeki *MsgId* alanının değeriyle eşleşen bir ileti tanıtıcısı olmalıdır. Bu eşleşme, geçerli olabilecek diğer eşleşmelere ek olarak (örneğin, ilinti tanıtıcısıdır).

Bu seçeneği çıkarırsanız, **MsgDesc** parametresindeki *MsgId* alanı yoksayılar ve herhangi bir ileti tanıtıcısı eşleşir.

**Not:** MQMI\_NONE ileti tanıtıcısı, iletiyle ilgili olarak MQMD ' deki herhangi bir ileti tanıtıcısıyla eşleşen özel bir değerdir. Bu nedenle, MQMI\_MATCH\_MSG\_ID ile MQMI\_MATCH\_MSG\_ID belirtildiğinde, MQMO\_MATCH\_MSG\_ID belirtilmeyen aynı durum söz edilir.

#### **MQMO\_MATCH\_COREL\_ID**

Alınacak iletinin, MQGET çağrısının **MsgDesc** parametresindeki *CorrelId* alanının değeriyle eşleşen bir ilinti tanıtıcısı olmalıdır. Bu eşleşme, geçerli olabilecek diğer eşleşmelere ek olarak (örneğin, ileti tanıtıcısı).

Bu seçeneği çıkarırsanız, **MsgDesc** parametresindeki *CorrelId* alanı yoksayılır ve herhangi bir ilinti tanımlayıcısı eşleşir.

**Not:** MQCI\_NONE ilinti tanıtcısı, ileti için MQMD ' deki *any* ilinti tanıtcısıyla eşleşen özel bir değerdir. Bu nedenle, MQCI\_NONE ile MQMO\_MATCH\_CORIL\_ID değerinin belirlenmesi, MQMO\_MATCH\_COREL\_ID belirtmemesiyle aynı.

#### **MQMO\_MATCH\_GROUP\_ID**

Alınacak iletinin, MQGET çağrısının **MsgDesc** parametresindeki *GroupId* alanının değeriyle eşleşen bir grup tanıtcısı olmalıdır. Bu eşleşme, geçerli olabilecek diğer eşleşmelere ek olarak (örneğin, ilinti tanıtcısıdır).

Bu seçeneği çıkarırsanız, **MsgDesc** parametresindeki *GroupId* alanı yoksayılır ve herhangi bir grup tanıtcısı eşleşir.

**Not:** MQGI\_NONE grup tanıtcısı, ileti için MQMD ' deki *herhangi bir* grup tanıtcısıyla eşleşen özel bir değerdir. Bu nedenle, MQGI\_NONE ile MQMO\_MATCH\_GROUP\_ID değerinin belirtilmesi, MQMO\_MATCH\_GROUP\_ID belirtmemesiyle aynı.

#### **MQMO\_MATCH\_MSG\_SEQ\_NUMBER**

Alınacak iletinin sıra numarası, MQGET çağrısının **MsgDesc** parametresindeki *MsgSeqNumber* alanının değeriyle eşleşen bir ileti sıra numarasına sahip olmalıdır. Bu eşleşme, geçerli olabilecek diğer eşleşmelere ek olarak (örneğin, grup tanıtcısı).

Bu seçeneği çıkarırsanız, **MsgDesc** parametresindeki *MsgSeqNumber* alanı yoksayılır ve herhangi bir ileti dizisi numarası eşleşir.

#### **MQMO\_MATCH\_OFFSET**

Alınacak ileti, MQGET çağrısının **MsgDesc** parametresindeki *Offset* alanının değeriyle eşleşen bir görel konumuna sahip olmalıdır. Bu eşleşme, geçerli olabilecek diğer eşleşmelere ek olarak (örneğin, ileti sıra numarası).

Bu seçeneği çıkarırsanız, **MsgDesc** parametresindeki *Offset* alanı yoksayılır ve herhangi bir görel konum eşleşir.

- Bu seçenek z/OSüzerinde desteklenmez.

#### **MQMO\_MATCH\_MSG\_TOKEN**

Alınacak iletinin, MQGET çağrısında belirtilen MQGMO yapısındaki *MsgToken* alanının değeriyle eşleşen bir ileti simgesi olmalıdır.

Bu seçeneği tüm yerel kuyruklar için belirleyebilirsiniz. If you specify it for a queue that has an *IndexType* of MQIT\_MSG\_TOKEN (a WLM-managed queue), you can specify no other match options with MQMO\_MATCH\_MSG\_TOKEN.

MQGMO\_WAIT ya da MQGMO\_SET\_SIGNAL ile MQMO\_MATCH\_MSG\_TOKEN değerini belirtebilirsiniz. If the application wants to wait for a message to arrive on a queue that has an *IndexType* of MQIT\_MSG\_TOKEN, specify MQMO\_NONE.

Bu seçeneği çıkarırsanız, MQGMO ' daki *MsgToken* alanı yoksayılır ve herhangi bir ileti simgesi eşleşir.

Tanımlanan seçeneklerin hiçbirini belirtmediyseniz, aşağıdaki seçeneği kullanabilirsiniz:

#### **MQMO\_NONE**

Döndürülebilecek iletiyi seçerken eşleşme yok; kuyrukta bulunan tüm iletiler alma için uygundur (ancak, MQGMO\_ALL\_MSGS\_AVALABILIR, MQGMO\_ALL\_SEGMENTS\_AVALABILIR ve MQGMO\_COMPLE\_MSG seçenekleri tarafından denetime tabi olur).

MQMO\_NONE yardımları programı belgeleri. Bu seçeneğin diğer MQMO\_ \* seçeneği ile kullanılması amaçlanmaz, ancak değeri sıfır olduğu için bu tür bir kullanım algılanamaz.

Bu bir giriş alanıdır. Bu alanın ilk değeri MQMO\_MATCH\_MSG\_ID, MQMO\_MATCH\_COREL\_ID ile olur. *Version* , MQGMO\_VERSION\_2değerinden küçükse bu alan yok sayılır.

**Not:** *MatchOptions* alanının ilk değeri, önceki MQSeries kuyruk yöneticileriyle uyumluluk için tanımlıdır. However, when reading a series of messages from a queue without using selection criteria, this initial

value requires the application to reset the *MsgId* and *CorrelId* fields to MQMI\_NONE and MQCI\_NONE before each MQGET call. Avoid the need to reset *MsgId* and *CorrelId* by setting *Version* to MQGMO\_VERSION\_2, and *MatchOptions* to MQMO\_NONE.

### İlgili kavramlar

JMS ' de ileti seçicileri

#### **GroupStatus (MQCHAR)**

Bu işaret, alınan iletinin bir grup içinde olup olmadığını belirtir.

Aşağıdaki değerlerden birine sahiptir:

##### **MQGS\_NOT\_IN\_GROUP**

İleti bir grupta yer almıyor.

##### **MQGS\_MSG\_IN\_GROUP**

İleti bir grup içinde, ancak gruptaki son kişi değil.

##### **MQGS\_LAST\_MSG\_IN\_GROUP**

Bu, gruptaki son iletidir.

Bu değer, grup yalnızca bir iletiyle oluşuyorsa, döndürülen değer de budur.

Bu bir çıkış alanıdır. Bu alanın ilk değeri MQGS\_NOT\_IN\_GROUP olur. *Version* , MQGMO\_VERSION\_2değerinden küçükse bu alan yok sayılır.

#### **SegmentStatus (MQCHAR)**

Bu işaret, alınan iletinin mantıksal bir iletinin bir parçası olup olmadığını gösterir. Aşağıdaki değerlerden birine sahiptir:

##### **MQSS\_NOT\_A\_KESIMI**

İleti bir bölüm değil.

##### **MQSS\_SEGMENT**

İleti bir bölümdür, ancak mantıksal iletinin son bölümü değildir.

##### **MQSS\_LAST\_SEGMENT**

İleti, mantıksal iletinin son bölümüdür.

Mantıksal ileti yalnızca bir kesimden oluşuyorsa, bu değer de döndürülen değerdir.

z/OSüzerinde, kuyruk yöneticisi bu alanı her zaman MQSS\_NOT\_A\_SEGMENT olarak ayarlar.

Bu bir çıkış alanıdır. Bu alanın ilk değeri MQSS\_NOT\_A\_SEGMENT'IDIR. *Version* , MQGMO\_VERSION\_2değerinden küçükse bu alan yok sayılır.

#### **Bölümleme (MQCHAR)**

Bu işaret, alınan ileti için ek bölümlemeye izin verilip verilmediğini gösteren bir işarettir. Aşağıdaki değerlerden birine sahiptir:

##### **MQSEG\_INHIBITED**

Bölümlemeye izin verilmiyor.

##### **MQSEG\_ALLOWD**

Bölümlemeye izin verilir.

z/OSüzerinde, kuyruk yöneticisi bu alanı her zaman MQSEG\_INHIBITED olarak ayarlar.

Bu bir çıkış alanıdır. Bu alanın ilk değeri MQSEG\_INHIBITED olur. *Version* , MQGMO\_VERSION\_2değerinden küçükse bu alan yok sayılır.

#### **Reserved1 (MQCHAR)**

Bu ayrılmış bir alandır. Bu alanın ilk değeri boşluk karakteridir. *Version* , MQGMO\_VERSION\_2değerinden küçükse bu alan yok sayılır.

## **MsgToken (MQBYTE16)**

MsgToken alanı-MQGM0 yapısı. Bu alan, kuyruk yöneticisi tarafından bir iletiyi benzersiz bir şekilde tanımlamak için kullanılır.

Bu, kuyruk yöneticisi tarafından bir kuyruğun benzersiz bir şekilde tanınması için oluşturulan bayt dizilimidir. İleti simgesi, ileti kuyruk yöneticisine ilk kez yerleştirildiğinde oluşturulur ve kuyruk yöneticisi yeniden başlatılmadıkça, ileti kuyruk yöneticisinden kalıcı olarak kaldırılincaya kadar iletiyle kalır.

İleti kuyruktan kaldırıldığında, iletinin o örneğini tanımlayan *MsgToken* artık geçerli değildir ve hiçbir zaman yeniden kullanılmaz. Kuyruk yöneticisi yeniden başlatılırsa, yeniden başlatmadan önce kuyruğun üzerinde bir ileti tanımlayan *MsgToken* değeri, yeniden başlatma işleminden sonra geçerli olmayabilir. Ancak, *MsgToken* , farklı bir ileti eşgörünümünü tanımlamak için hiçbir zaman yeniden kullanılmaz. *MsgToken* , kuyruk yöneticisi tarafından oluşturulur ve herhangi bir dış uygulama tarafından görülemez.

Bir MQGET çağrısı tarafından bir Sürüm 3 ya da üstü bir MQGM0 belirtildiğinde bir ileti döndürüldüğünde, kuyruk yöneticisi tarafından MQGM0 ' ya kuyruksa iletiyi tanıtan *MsgToken* iletisi döndürülür. Bunun bir kural dışı durumu vardır: İleti, eşitleme noktası dışında kuyruksa kaldırıldığında, kuyruk yöneticisi sonraki bir MQGET çağrısında döndürülen iletiyi tanıtmakta yararlı olmadığı için bir *MsgToken* döndürmeyebilir. Uygulamaların, sonraki MQGET çağrılarında iletiye başvurması için yalnızca *MsgToken* ' u kullanması gerekir.

Bir *MsgToken* belirtilirse ve *MatchOption* MQMO\_MATCH\_MSG\_TOKENspecifiedbelirtilirse ve MQGM0\_MSG\_UNDER\_CURSOR ya da MQGM0\_BROWSE\_MSG\_UNDER\_CURSOR belirtilirse, yalnızca bu *MsgToken* ile tanıtılan ileti döndürüleabilmektedir. Bu seçenek, INDXTYPE ile bağımsız olarak tüm yerel kuyruksa geçerlidir ve z/OS üzerinde, yalnızca İş Yükü Yöneticisi (WLM) kuyruksa INDXTYPE (MSGTOKEN) olanağını kullanmanız gerekir.

Belirtilen diğer *MatchOptions* sayısı imlenir ve eşleşmezse, MQRC\_NO\_MSG\_AVAILEABLE döndürülür. MQGM0\_BROWSE\_NEXT, MQMO\_MATCH\_MSG\_TOKEN ile kodlandysa, *MsgToken* ile tanıtılan ileti yalnızca, çağırana tanıtıcı için göz atma imlecinin ötesinde olduğunda döndürülür.

MQGM0\_MSG\_UNDER\_CURSOR ya da MQGM0\_BROWSE\_MSG\_UNDER\_CURSOR belirtilirse, MQMO\_MATCH\_MSG\_TOKEN yoksayılır.

MQMO\_MATCH\_MSG\_TOKEN, aşağıdaki alma iletisi seçenekleriyle geçerli değil:

- MQGM0\_BEKLE
- MQGM0\_SET\_SIGNAL

MQMO\_MATCH\_MSG\_TOKEN MQGET çağrısına ilişkin çağrıya bir MQGM0 sürüm 3 ya da üstü sağlanmalıdır; tersi durumda, MQRC\_WRONG\_GMO\_VERSION döndürülür.

*MsgToken* bu sırada geçerli değilse, başka bir hata olmadığı sürece, MQRC\_NO\_MSG\_AVAILEABLE ile MQCCT\_BARINIZ geri döndü.

## **ReturnedLength (MQUZE)**

Bu, kuyruk yöneticisinin **Buffer** parametresindeki MQGET çağrısının döndürdüğü ileti verilerinin bayt cinsinden uzunluğuna ayarlayan bir çıkış alanıdır. Kuyruk yöneticisi bu yeteneği desteklemiyorsa, *ReturnedLength* MQRL\_UNDEFINED değeri olarak ayarlanır.

İletiler kodlamalar ya da karakter kümeleri arasında dönüştürüldüğünde, ileti verileri bazen boyut değiştirebilir. MQGET çağrısından geri dönüş:

- *ReturnedLength* MQRL\_UNDEFINED değilse, döndürülen ileti verilerinin bayt sayısı *ReturnedLength* tarafından verilir.
- *ReturnedLength* , MQRL\_UNDEFINED değerine sahipse, döndürülen ileti verilerinin bayt sayısı genellikle *BufferLength* ve *DataLength* ' nin daha küçü tarafından verilir; ancak, MQGET çağrısı tamamlanmışsa, bu durum *küçüktür* . MQRC\_TRUNCATED\_MSG\_ACCEPTED neden kodunu tamamlar. Bu gerçekleşirse, **Buffer** parametresindeki önemsiz byte ' lar boş değere ayarlanır.

Aşağıdaki özel değer tanımlıdır:

## **MQRL\_UNDEFED**

Döndürülen verilerin uzunluğu tanımlı değil.

z/OS' ta, *ReturnedLength* alanı için döndürülen değer her zaman MQRL\_UNDEFINED değerine sahip olur.

Bu alanın ilk değeri MQRL\_UNDEFED olur. *Version* , MQGMO\_VERSION\_3değerinden küçükse bu alan yok sayılır.

## **Reserved2 (MQUZE)**

Bu ayrılmış bir alandır. Bu alanın ilk değeri boşluk karakteridir. *Version* , MQGMO\_VERSION\_4değerinden küçükse bu alan yok sayılır.

## **MsgHandle (MQHMSG)**

MQGMO\_PROPERTIES\_AS\_Q\_DEF seçeneği belirtilirse ve *PropertyControl* kuyruk özniteliği MQPROP\_FORCE\_MQRFH2 olarak ayarlanmazsa, bu, kuyruktan alınan iletinin özellikleriyle doldurulacak bir iletiyle, tanıtıcıdır. Tanıtıcı, bir MQCRTMH çağrısı tarafından yaratılır. Bir ileti alınmadan önce, bu tanıtıcı ile önceden ilişkilendirilmiş özellikler temizlenir.

Aşağıdaki değer de belirtilebilir:

MQHM\_NONE

İleti tanıtıcısı sağlanmadı.

Geçerli bir ileti tanıtıcısı sağlandıysa ve ileti özelliklerini içermek üzere çıkışta kullanılırsa, MQGET çağrısında ileti tanımlayıcısı gerekmez; giriş alanları için ileti tanıtıcısı ile ilişkili ileti tanımlayıcısı kullanılır.

MQGET çağrısında bir ileti tanımlayıcısı belirtilirse, ileti tanıtıcısı ile ilişkili ileti tanımlayıcısına her zaman öncelik kazanır.

MQGMO\_PROPERTIES\_FORCE\_MQRFH2 belirtildiyse ya da MQGMO\_PROPERTIES\_AS\_Q\_DEF belirtildiyse ve *PropertyControl* kuyruk özniteliği MQPROP\_FORCE\_MQRFH2 ise, ileti tanımlayıcı değıştirgesi belirtilmediğinde çağrı neden kodu MQRC\_MD\_ERROR ile başarısız olur.

MQGET çağrısından dönüşte, bu ileti tanıtıcısı ile ilişkilendirilmiş özellikler ve ileti tanımlayıcısı, alınan iletinin durumunu (MQGET çağrısında belirtildiyse, ileti tanımlayıcısının yanı sıra ileti tanımlayıcısını) yansıtacak şekilde güncellenir. Bundan sonra, iletinin özellikleri MQINQMP çağrısını kullanarak sorgulanabilir.

İleti tanımlayıcı uzantıları dışında, varsa, MQINQMP çağrısıyla sorgulanabilen bir özellik ileti verilerinde bulunmuyorsa; kuyrukta bulunan ileti ileti verilerinde yer alan özellikler, veriler uygulamaya döndürülmeden önce ileti verilerinden kaldırılmışsa.

If no message handle is provided or Version is less than MQGMO\_VERSION\_4 then you must supply a valid message descriptor on the MQGET call. İleti verileri, MQGMO yapısındaki ve *PropertyControl* kuyruk öznitelikteki özellik seçeneklerinin değerine konu olan ileti verileri (ileti tanımlayıcısında yer alan durumlar dışında) herhangi bir ileti özelliğini döndürmektedir.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQHM\_NONE olur. *Version* , MQGMO\_VERSION\_4değerinden küçükse bu alan yok sayılır.

## **MQIIH- IMS bilgi üstbilgisi**

MQIIH yapısı, IMS köprüsünde IMS ' e gönderilen bir iletiye ilişkin üstbilgi bilgilerini tanımlar. Desteklenen herhangi bir IBM MQ altyapısı için MQIIH yapısını içeren bir ileti yaratabilir ve iletebilir, ancak IMS köprüsünü yalnızca bir IBM MQ for z/OS kuyruk yöneticisi kullanabilir. Bu nedenle, iletinin z/OS dışı bir kuyruk yöneticisinden IMS ' e ulaşabilmesi için kuyruk yöneticisi ağızının, iletinin yönetilebileceği en az bir z/OS kuyruk yöneticisi içermesi gerekir.

## Kullanılabilirlik

Tüm IBM MQ sistemleri ve IBM MQ istemcileri.

## Biçim Adı

MQFMT\_IMS

## Karakter kümesi ve kodlama

MQIIH yapısı ve uygulama iletisi verileri için kullanılan karakter kümesi ve kodlama için özel koşullar geçerlidir:

- IMS köprü kuyruğuna sahip kuyruk yöneticisine bağlanan uygulamalar, kuyruk yöneticisinin karakter kümesinde ve kodlamasında bulunan bir MQIIH yapısı sağlamalıdır. Bunun nedeni, bu durumda MQIIH yapısına ilişkin veri dönüştürmenin gerçekleştirilmemiş olmasıdır.
- Diğer kuyruk yöneticilerine bağlanan uygulamalar, desteklenen karakter kümelerinde ve kodlamasında bulunan bir MQIIH yapısı sağlayabilir; IMS köprü kuyruğunun sahibi olan kuyruk yöneticisine bağlı alıcı ileti kanalı aracısı MQIIH ' yi dönüştürür.
- MQIIH yapısını izleyen uygulama iletisi verileri, MQIIH yapısıyla aynı karakter kümesinde ve kodlamada olmalıdır. MQIIH yapısındaki *CodedCharSetId* ve *Encoding* alanlarını, uygulama ileti verilerinin karakter kümesini ve kodlamasını belirtmek için kullanmayın.

Veriler, kuyruk yöneticisi tarafından desteklenen yerleşik biçimlerden biri değilse, uygulama ileti verilerini dönüştürmek için bir veri dönüştürme çıkışı sağlamanız gerekir.

## Alanlar

**Not:** Aşağıdaki tabloda, alanlar alfabetik olarak değil, kullanıma göre gruplanır. Alt konular aynı sırayı izler.

Çizelge 498. MQIIH için MQIIH ' deki Alanlar		
Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
<u>StrucId</u> (yapı tanıtıcısı)	MQIIH_STRUC_ID	' I IH - '
<u>Sürüm</u> (yapı sürümü numarası)	MQIIH_VERSION_1	1
<u>StrucLength</u> (MQIIH yapısının uzunluğu)	MQIIH_LENGTH_1	84
<u>Kodlama</u> (ayrılmış-bkz. "Karakter kümesi ve kodlama" sayfa 402)	Yok	0
<u>CodedCharSetId</u> (ayrılmış-bkz. "Karakter kümesi ve kodlama" sayfa 402)	Yok	0
<u>Biçim</u> (MQIIH ' yi izleyen verilerinMQ biçimi adı)	MQFMT_NONE	Boşluklar
<u>İşaretler</u> (işaretler)	MQIIH_NONE	0
<u>LTermOverride</u> (mantıksal uçbirim geçersiz kılma)	Yok	Boşluklar
<u>MFSMapName</u> (ileti biçimi hizmetleri eşlemi adı)	Yok	Boşluklar
<u>ReplyToBiçim</u> (MQ yanıt iletisinin biçim adı)	MQFMT_NONE	Boşluklar
<u>Authenticator</u> (RACF parolası ya da geçiş bileti)	MQIAUT_NONE	Boşluklar
<u>TranInstanceId</u> (hareket eşgörünümü tanıtıcısı)	MQITII_NONE	Boş Değerler
<u>TranState</u> (hareket durumu)	MQITS_NOT_IN_ETKILE ŞİM	' - '

Çizelge 498. MQIIH için MQIIH 'deki Alanlar (devamı var)

Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
<u>CommitMode</u> (kesinleştirme kipi)	MQICM_COMMIT_THEN_SEND	'0'
<u>SecurityScope</u> (güvenlik kapsamı)	MQISS_CHECK	'C'
<u>Ayrılmış</u> (ayrılmış)	Yok	'-'

**Notlar:**

1. - simgesi, tek bir boş karakteri temsil eder.
2. C programlama dilinde, makro değişkeniMQIIH\_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlar için başlangıç değerleri sağlamak üzere aşağıdaki şekilde kullanılabilir:

```
MQIIH MyIIH = {MQIIH_DEFAULT};
```

## Dil bildirimleri

### MQIIH için C bildirimini

```
typedef struct tagMQIIH MQIIH;
struct tagMQIIH {
    MQCHAR4    StrucId;          /* Structure identifier */
    MQLONG     Version;         /* Structure version number */
    MQLONG     StrucLength;     /* Length of MQIIH structure */
    MQLONG     Encoding;       /* Reserved */
    MQLONG     CodedCharSetId; /* Reserved */
    MQCHAR8    Format;         /* MQ format name of data that follows
                               MQIIH */
    MQLONG     Flags;          /* Flags */
    MQCHAR8    LTermOverride; /* Logical terminal override */
    MQCHAR8    MFMapName;     /* Message format services map name */
    MQCHAR8    ReplyToFormat; /* MQ format name of reply message */
    MQCHAR8    Authenticator; /* RACF password or passticket */
    MQBYTE16   TranInstanceId; /* Transaction instance identifier */
    MQCHAR     TranState;     /* Transaction state */
    MQCHAR     CommitMode;    /* Commit mode */
    MQCHAR     SecurityScope; /* Security scope */
    MQCHAR     Reserved;     /* Reserved */
};
```

### MQIIH için COBOL bildirimini

```
** MQIIH structure
10 MQIIH.
** Structure identifier
15 MQIIH-STRUCID PIC X(4).
** Structure version number
15 MQIIH-VERSION PIC S9(9) BINARY.
** Length of MQIIH structure
15 MQIIH-STRUCLNGTH PIC S9(9) BINARY.
** Reserved
15 MQIIH-ENCODING PIC S9(9) BINARY.
** Reserved
15 MQIIH-CODEDCHARSETID PIC S9(9) BINARY.
** MQ format name of data that follows MQIIH
15 MQIIH-FORMAT PIC X(8).
** Flags
15 MQIIH-FLAGS PIC S9(9) BINARY.
** Logical terminal override
15 MQIIH-LTERMOVERRIDE PIC X(8).
** Message format services map name
15 MQIIH-MFSMAPNAME PIC X(8).
** MQ format name of reply message
15 MQIIH-REPLYTOFORMAT PIC X(8).
```

```

**      RACF password or passticket
15 MQIIH-AUTHENTICATOR PIC X(8).
**      Transaction instance identifier
15 MQIIH-TRANINSTANCEID PIC X(16).
**      Transaction state
15 MQIIH-TRANSTATE      PIC X.
**      Commit mode
15 MQIIH-COMMITMODE     PIC X.
**      Security scope
15 MQIIH-SECURITYSCOPE  PIC X.
**      Reserved
15 MQIIH-RESERVED       PIC X.

```

### MQIIH için PL/I bildirimi

```

dcl
  1 MQIIH based,
  3 StrucId      char(4),          /* Structure identifier */
  3 Version      fixed bin(31),   /* Structure version number */
  3 StrucLength  fixed bin(31),   /* Length of MQIIH structure */
  3 Encoding     fixed bin(31),   /* Reserved */
  3 CodedCharSetId fixed bin(31), /* Reserved */
  3 Format        char(8),         /* MQ format name of data that follows
                                MQIIH */
  3 Flags        fixed bin(31),   /* Flags */
  3 LTermOverride char(8),       /* Logical terminal override */
  3 MFSMapName   char(8),       /* Message format services map name */
  3 ReplyToFormat char(8),      /* MQ format name of reply message */
  3 Authenticator char(8),      /* RACF password or passticket */
  3 TranInstanceId char(16),    /* Transaction instance identifier */
  3 TranState    char(1),       /* Transaction state */
  3 CommitMode   char(1),       /* Commit mode */
  3 SecurityScope char(1),     /* Security scope */
  3 Reserved     char(1);       /* Reserved */

```

### MQIIH için High Level Assembler (Üst Düzey Çevirici) bildirimi

```

MQIIH          DSECT
MQIIH_STRUCID  DS   CL4  Structure identifier
MQIIH_VERSION  DS   F    Structure version number
MQIIH_STRUCLNGTH DS  F    Length of MQIIH structure
MQIIH_ENCODING DS   F    Reserved
MQIIH_CODEDCHARSETID DS  F    Reserved
MQIIH_FORMAT   DS   CL8  MQ format name of data that follows
*              MQIIH
MQIIH_FLAGS    DS   F    Flags
MQIIH_LTERM_OVERRIDE DS  CL8 Logical terminal override
MQIIH_MFSMAPNAME DS   CL8 Message format services map name
MQIIH_REPLYTOFORMAT DS  CL8 MQ format name of reply message
MQIIH_AUTHENTICATOR DS  CL8 RACF password or passticket
MQIIH_TRANINSTANCEID DS  XL16 Transaction instance identifier
MQIIH_TRANSTATE DS   CL1 Transaction state
MQIIH_COMMITMODE DS   CL1 Commit mode
MQIIH_SECURITYSCOPE DS  CL1 Security scope
MQIIH_RESERVED DS   CL1 Reserved
*
MQIIH_LENGTH   EQU  *-MQIIH
ORG  MQIIH
MQIIH_AREA     DS   CL(MQIIH_LENGTH)

```

### MQIIH için Visual Basic bildirimi

```

Type MQIIH
  StrucId      As String*4 'Structure identifier'
  Version      As Long      'Structure version number'
  StrucLength  As Long      'Length of MQIIH structure'
  Encoding     As Long      'Reserved'
  CodedCharSetId As Long    'Reserved'
  Format       As String*8  'MQ format name of data that follows MQIIH'
  Flags       As Long      'Flags'
  LTermOverride As String*8 'Logical terminal override'
  MFSMapName  As String*8  'Message format services map name'
  ReplyToFormat As String*8 'MQ format name of reply message'
  Authenticator As String*8 'RACF password or passticket'
  TranInstanceId As MQBYTE16 'Transaction instance identifier'

```



```
TranState      As String*1 'Transaction state'  
CommitMode    As String*1 'Commit mode'  
SecurityScope As String*1 'Security scope'  
Reserved      As String*1 'Reserved'  
End Type
```

### ***StrucId (MQCHAR4)***

Bu, yapı tanıtıcısıdır. Değer şu olmalıdır:

#### **MQIIH\_STRUC\_ID**

IMS bilgi üstbilgisi yapısına ilişkin tanıtıcı.

C programlama dili için, aynı zamanda MQIIH\_STRUC\_ID\_ARRAY değişmezi de tanımlanır; bu, MQIIH\_STRUC\_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisidir.

Bu alanın ilk değeri MQIIH\_STRUC\_ID ' dir.

### ***Sürüm (MQUZE)***

Bu, yapı sürüm numarasıdır. Değer şu olmalıdır:

#### **MQIIH\_VERSION\_1**

IMS bilgi üstbilgisi yapısına ilişkin sürüm numarası.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

#### **MQIH\_CURRENT\_VERSION**

IMS bilgi üstbilgisi yapısının yürürlükteki sürümü.

Bu alanın ilk değeri MQIIH\_VERSION\_1' dir.

### ***StrucLength (MQUZE)***

Bu, MQIIH yapısının uzunluğudur. Değer şu olmalıdır:

#### **MQIIH\_LENGTH\_1**

IMS bilgi üstbilgisi yapısının uzunluğu.

Bu alanın ilk değeri MQIIH\_LENGTH\_1' dir.

### ***Kodlama (MQUZE)***

Bu ayrılmış bir alandır; değeri anlamlı değildir. Bu alanın ilk değeri 0 'tır.

MQIIH yapısını izleyen desteklenen yapılara ilişkin kodlama, MQIIH yapısının kendisidir ve önceki herhangi bir MQ üstbilgisinden alınır.

### ***CodedCharSetId (MQHOMER)***

Bu ayrılmış bir alandır; değeri anlamlı değildir. Bu alanın ilk değeri 0 'tır.

MQIIH yapısını izleyen desteklenen yapılara ilişkin Karakter Kümesi Tanıtıcısı, MQIIH yapısının kendisidir ve önceki herhangi bir MQ üstbilgisinden alınır.

### ***Biçim (MQCHAR8)***

Bu, MQIIH yapısından sonra gelen verilerin MQ biçim adını belirtir.

MQPUT ya da MQPUT1 çağrısında, uygulama bu alanı veri için uygun değere ayarlamalıdır.

Bu alanın uzunluğu MQ\_FORMAT\_LENGTH tarafından verilir. Bu alanın ilk değeri MQFMT\_NONE olur.

### ***İşaretler (MQUZE)***

İşaretler değeri şu şekilde olmalıdır:

#### **MQIH\_NONE**

Bayrak yok.

## **MQIIH\_PASS\_EXPIRATION**

Yanıt iletisi şunları içerir:

- İstek iletisine göre aynı süre bitimi raporu seçenekleri
- İstek iletisinden kalan süre bitimi, köprünün işleme süresi için ayarlama yapılmadı.

Bu değer ayarlanmazsa, süre bitim süresi *sınırsız* olarak ayarlanır.

## **MQIH\_REPLY\_FORMAT\_NONE**

Yanıtın MQIIH.Format alanını MQFMT\_NONE olarak ayarlar.

## **MQIH\_IGNORE\_PURG**

Sets the TMAMIPRG indicator in the OTMA prefix, which requests that OTMA ignores PURG calls on the TP PCB for CMO transactions.

## **MQIIH\_CMO\_REQUEST\_RESPONSE**

Commit Mode 0 (CM0) işlemleri için, bu işaret OTMA önekindeki TMAMHRSP göstergesini ayarlar.

Setting this indicator requests that OTMA/IMS generate a DFS2082 RESPONSE MODE TRANSACTION TERMINATED WITHOUT REPLY message when the original IMS application program does not reply to the IOPCB nor message switch to another transaction.

Bu alanın ilk değeri MQIIH\_NONE olur.

## **LTermOverride (MQCHAR8)**

Mantıksal uçbirim geçersiz kılma değeri, GÇ PCB alanına yerleştirilir. Bu isteğe bağlıdır; belirtilmediyse, TPIPE adı kullanılır. İlk baytın boş olması ya da boş değer olması dikkate alınmaz.

Bu alanın uzunluğu MQ\_LTERM\_OVERRIDE\_LENGTH tarafından verilir. Bu alanın ilk değeri 8 boş karakterdir.

## **MFSMapName (MQCHAR8)**

İleti biçimi hizmetleri eşlemi adı, GÇ PCB alanına yerleştirilir. İsteğe bağlıdır. Giriş sırasında, MID 'yi temsil eder ve MOD' yi temsil eder. İlk byte boş ya da boş değerliyse bu değer dikkate alınmaz.

Bu alanın uzunluğu MQ\_MFS\_MAP\_NAME\_LENGTH tarafından verilmektedir. Bu alanın ilk değeri 8 boş karakterdir.

## **ReplyToBiçimi (MQCHAR8)**

Bu, yürürlükteki iletiye yanıt olarak gönderilen yanıt iletisinin MQ biçimi adıdır. Bu alanın uzunluğu MQ\_FORMAT\_LENGTH tarafından verilir. Bu alanın ilk değeri MQFMT\_NONE olur.

Yanıt iletisinde MQGMO\_CONVERT komutunu kullanarak verileri dönüştürmek için, MQIIH.replyToFormat=MQFMT\_STRING ya da MQIIH.replyToFormat=MQFMT\_IMS\_VAR\_STRING belirtin. Bu alanların kullanımına ilişkin açıklamalar için bkz. "[Biçim \(MQCHAR8\)](#)" sayfa 441.

İstek iletisine varsayılan değer (MQIIH.replyToFormat=MQFMT\_NONE) ise ve yanıt iletisi MQGMO\_CONVERT kullanılarak alındıysa, veri dönüştürme gerçekleştirilmez.

## **Kimlik doğrulayıcı (MQCHAR8)**

Bu, RACF parolasıdır ya da PassTicket' dir. Bu isteğe bağlıdır; belirtildiye, güvenlik bağlamı sağlamak üzere IMS ' a gönderilen bir UTOKEN oluşturmak için MQMD güvenlik bağlamındaki kullanıcı kimliğiyle kullanılır. Belirlenmezse, doğrulama işlemi yapılmaksızın kullanıcı kimliği kullanılır. Bu, RACF anahtarlarının ayarına bağlıdır; bu, kimlik doğrulayıcının var olmasını gerektirebilir.

İlk byte boş ya da boş değerliyse bu değer dikkate alınmaz. Aşağıdaki özel değer kullanılabilir:

### **MQIAUT\_NONE**

Kimlik doğrulama yok.

C programlama dili için, değişmez MQIAUT\_NONE\_DIZISI de tanımlanır; bu, MQIAUT\_NONE ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisi olur.

Bu alanın uzunluğu MQ\_AUTHENTICATOR\_LENGTH tarafından verilir. Bu alanın ilk değeri MQIAUT\_NONE olur.

### **TranInstanceTanıtıcısı (MQBYTE16)**

Bu, hareket eşgörünümü tanıtıcısıdır. Bu alan çıkış iletileri tarafından IMS' den kullanılır, bu nedenle ilk girişte yoksayılr. *TranState* 'u MQITS\_IN\_CONVERT olarak ayarladıysanız, IMS ' un iletileri doğru etkileşimde ilişkilendirmesini sağlamak için, sonraki giriş ve sonraki tüm girişler için bu değer sağlanmalıdır. Aşağıdaki özel değeri kullanabilirsiniz:

#### **MQITII\_NONE**

İşlem eşgörünümü tanıtıcısı yok.

C programlama dili için, MQITII\_NONE\_ARRAY değişmezi de tanımlıdır; bu, MQITII\_NONE ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisi olur.

Bu alanın uzunluğu MQ\_TRAN\_INSTANCE\_ID\_LENGTH tarafından verilir. Bu alanın ilk değeri MQITII\_NONE olur.

### **TranState (MQCHAR)**

Bu, IMS etkileşim durumunu gösterir. Bu, ilk girişte yoksayılr, çünkü etkileşim yok. Sonraki girişlerde, bir sohbetin etkin olup olmadığını belirtir. Çıkış sırasında IMStarafından ayarlanır. Değer aşağıdakilerden biri olmalıdır:

#### **MQITS\_IN\_MUHABBET**


Sohbete.

#### **MQITS\_NOT\_IN\_SOHBET**

Konuşmakta değil.

#### **MQITS\_ARCHITECTED**

Hareket durumu verilerini, mimari biçimiyle geri döndürür.

Bu değer yalnızca IMS /DISPLAY TRAN komutu ile kullanılır. Hareket durumu verilerini karakter formu yerine IMS mimarili formdaki verileri döndürür.  Daha fazla bilgi için bkz. [IMS işlem programlarının IBM MQaracılığıyla yazılması](#).

Bu alanın ilk değeri MQITS\_NOT\_IN\_CONVERSACE olur.

### **CommitMode (MQCHAR)**

Bu, IMS kesinleştirme kipidir. IMS kesinleştirme kiplerine ilişkin ek bilgi için *OTMA Reference* belgesine bakın. Değer aşağıdakilerden biri olmalıdır:

#### **MQICM\_COMMIT\_THEN\_SEND**

Gönder ve gönder.

Bu kip, çıkışın iki kez kuyruğa girmesini, ancak bölge doluluk zamanlarının daha kısa olduğunu belirtir. Hızlı yol ve etkileşimli işlemler bu kiple çalışamaz.

#### **MQICM\_SEND\_THEN\_COMMIT**

O zaman teslim et.

Bir MQICM\_SEND\_THEN\_COMMIT kesinleştirme kipinin bir sonucu olarak başlatılan herhangi bir IMS hareketi, hareketin IMS sistem tanımlamasında nasıl tanımlandığından bağımsız olarak (TRANSACT makrosu için MSGTYPE parametresi) çalışır. Bu, bir işlem anahtarı aracılığıyla başlatılan işlemler için geçerlidir.

Bu alanın ilk değeri MQICM\_COMMIT\_THEN\_SEND olarak değerlendirilmektedir.

### **SecurityScope (MQCHAR)**

Bu, IMS güvenlik işleminin gerekli olduğunu gösterir. Aşağıdaki değerler tanımlanır:

## **MQISS\_CHECK**

Güvenlik kapsamını denetleyin: denetim bölgesinde bir ACEE oluşturulur, ancak bağımlı bölgede oluşturulmaz.

## **MQISS\_FULL**

Tam güvenlik kapsamı: Denetim bölgesinde önbelleğe alınmış bir ACEE oluşturulur ve bağımlı bölgede önbelleğe alınmamış bir ACEE oluşturulur. MQISS\_FULL kullanırsanız, ACEE 'nin yerleşik olduğu kullanıcı kimliğinin bağımlı bölgede kullanılan kaynaklara eriştiğinden emin olun.

Bu alan için MQISS\_SEÇE ya da MQISS\_FULL belirtilmediyse, MQISS\_SEVER değeri varsayılır.

Bu alanın ilk değeri MQISS\_REKLE olur.

## **Ayrılmış (MQCHAR)**

Bu ayrılmış bir alandır; boş bırakılmalıdır.

## **MQIMPO-İleti özellik seçeneklerini sorgularken sor**

MQIMPO yapısı, uygulamaların iletilerin özelliklerinin nasıl sorulduğunu denetleyen seçenekleri belirtmesine olanak sağlar. Yapı, MQINQMP çağrısındaki bir giriş değiştirgesidir.

## **Kullanılabilirlik**

Tüm IBM MQ sistemleri ve IBM MQ istemcileri.

## **Karakter kümesi ve kodlama**

MQIMPO içindeki veriler, uygulamanın karakter kümesinde ve uygulamanın kodlamasında (MQENC\_NATIVE) yer almalıdır.

## **Alanlar**

**Not:** Aşağıdaki tabloda, alanlar alfabetik olarak değil, kullanıma göre gruplanır. Alt konular aynı sırayı izler.

<i>Çizelge 499. MQIPMO 'daki Alanlar</i>		
<b>Alan adı ve açıklaması</b>	<b>Değişmezin adı</b>	<b>Değişmezin ilk değeri (varsa)</b>
<u>StrucId</u> (yapı tanıtıcısı)	MQIMPO_STRUC_ID	'IMPO'
<u>Sürüm</u> (yapı sürümü numarası)	MQIMPO_VERSION_1	1
Seçenekler (MQINQMP işlemini denetleyen seçenekler)	MQIMPO_INQ_FIRST	
RequestedEncoding (istenen özelliğin dönüştürüleceği kodlama)	MQENC_NATIVE	
RequestedCCSID (sorulan özelliğin karakter takımı)	MQCCSI_APPL	
ReturnedEncoding (döndürülen değer kodlaması)	MQENC_NATIVE	
<u>ReturnedCCSID</u>	0	
<u>Reserved1</u> (ayrılmış alan)	boşluk karakteri (4 baytlık alan)	
<u>ReturnedName</u> (sorulmuş özelliğin adı)	MQCHARV_DEFAULT	
<u>TypeString</u> (özelliğin veri tipinin dizgi gösterimi)	Boş değerli dizilim ya da boşluklar	

Çizelge 499. MQIPMO 'daki Alanlar (devamı var)

Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
<b>Notlar:</b>		
1. Boş değerli dizilim ya da boşluk değeri, C dizgisindeki boş dizgiyi ve diğer programlama dillerindeki boş karakterleri gösterir.		
2. C programlama dilinde, makro değişkeniMQIPMO_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlar için başlangıç değerleri sağlamak üzere bunu aşağıdaki şekilde kullanın:		
<pre>MQIPMO MyIMPO = {MQIPMO_DEFAULT};</pre>		

## Dil bildirimleri

### MQIPMO için C bildirimini

```
typedef struct tagMQIPMO MQIPMO;
struct tagMQIPMO {
    MQCHAR4  StrucId;           /* Structure identifier */
    MQLONG   Version;          /* Structure version number */
    MQLONG   Options;          /* Options that control the action of
                               MQINQMP */
    MQLONG   RequestedEncoding; /* Requested encoding of Value */
    MQLONG   RequestedCCSID;   /* Requested character set identifier
                               of Value */
    MQLONG   ReturnedEncoding; /* Returned encoding of Value */
    MQLONG   ReturnedCCSID;   /* Returned character set identifier
                               of Value */
    MQCHAR   Reserved1;       /* Reserved field */
    MQCHARV  ReturnedName;    /* Returned property name */
    MQCHAR8  TypeString;      /* Property data type as a string */
};
```

### MQIPMO için COBOL bildirimini

```
** MQIPMO structure
10 MQIPMO.
** Structure identifier
15 MQIPMO-STRUCID PIC X(4).
** Structure version number
15 MQIPMO-VERSION PIC S9(9) BINARY.
** Options that control the action of MQINQMP
15 MQIPMO-OPTIONS PIC S9(9) BINARY.
** Requested encoding of VALUE
15 MQIPMO-REQUESTEDENCODING PIC S9(9) BINARY.
** Requested character set identifier of VALUE
15 MQIPMO-REQUESTEDCCSID PIC S9(9) BINARY.
** Returned encoding of VALUE
15 MQIPMO-RETURNEDENCODING PIC S9(9) BINARY.
** Returned character set identifier of VALUE
15 MQIPMO-RETURNEDCCSID PIC S9(9) BINARY.
** Reserved field
15 MQIPMO-RESERVED1
** Returned property name
15 MQIPMO-RETURNEDNAME.
** Address of variable length string
20 MQIPMO-RETURNEDNAME-VSPTR POINTER.
** Offset of variable length string
20 MQIPMO-RETURNEDNAME-VSOFFSET PIC S9(9) BINARY.
** CCSID of variable length string
20 MQIPMO-RETURNEDNAME-VSCCSID PIC S9(9) BINARY.
** Property data type as string
15 MQIPMO-TYPESTRING PIC S9(9) BINARY.
```

## MQIMPO için PL/I bildirimi

```
dc1
1 MQIMPO based,
3 StrucId          char(4),          /* Structure identifier */
3 Version          fixed bin(31),   /* Structure version number */
3 Options          fixed bin(31),   /* Options that control the
                                   action of MQINQMP */
3 RequestedEncoding fixed bin(31), /* Requested encoding of
                                   Value */
3 RequestedCCSID   fixed bin(31),   /* Requested character set
                                   identifier of Value */
3 ReturnedEncoding fixed bin(31), /* Returned encoding of
                                   Value */
3 ReturnedCCSID    fixed bin(31),   /* Returned character set
                                   identifier of Value */
3 Reserved1        fixed bin(31),   /* Reserved field */
3 ReturnedName,    /* Returned property name */
5 ReturnedName_VSPtr pointer,      /* Address of returned
                                   name */
5 5 ReturnedName_VSOFFSET fixed bin(31), /* Offset of returned
                                   name */
5 5 ReturnedName_VSCCSID fixed bin(31), /* CCSID of returned
                                   name */
3 TypeString       char(8);         /* Property data type as
                                   string */
```

## MQIMPO için High Level Assembler (Üst Düzey Çevirici) bildirimi

```
MQIMPO          DSECT
MQIMPO_STRUCID  DS    CL4  Structure identifier
MQIMPO_VERSION  DS    F    Structure version number
MQIMPO_OPTIONS  DS    F    Options that control the
*               action of MQINQMP
MQIMPO_REQUESTEDENCODING DS F Requested encoding of VALUE
MQIMPO_REQUESTEDCCSID   DS F Requested character set
*               identifier of VALUE
MQIMPO_RETURNEDENCODING DS F Returned encoding of VALUE
MQIMPO_RETURNEDCCSID    DS F Returned character set
*               identifier of VALUE
MQIMPO_RESERVED1       DS    F    Reserved field
MQIMPO_RETURNEDNAME     DS    0F  Force fullword alignment
MQIMPO_RETURNEDNAME_VSPTR DS    F    Address of returned name
MQIMPO_RETURNEDNAME_VSOFFSET DS F Offset of returned name
MQIMPO_RETURNEDNAME_VSLENGTH DS F Length of returned name
MQIMPO_RETURNEDNAME_VSCCSID DS F CCSID of returned name
MQIMPO_RETURNEDNAME_LENGTH EQU  *-MQIMPO_RETURNEDNAME
MQIMPO_RETURNEDNAME_AREA DS    CL(MQIMPO_RETURNEDNAME_LENGTH)
*
MQIMPO_TYPESTRING      DS    CL8  Property data type as string
MQIMPO_LENGTH          EQU  *-MQIMPO
MQIMPO_AREA            DS    CL(MQIMPO_LENGTH)
```

### **StrucId (MQCHAR4)**

Sorgu iletişi özellik seçenekleri yapısı- StrucId alanı

Bu, yapı tanıtıcısıdır. Değer şu olmalıdır:

#### **MQIMPO\_STRUC\_ID**

Sorgu iletişi özellik seçenekleri yapısının tanıtıcısı.

C programlama dili için, aynı zamanda MQIMPO\_STRUC\_ID\_ARRAY değişmezi de tanımlanır; bu, MQIMPO\_STRUC\_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisi olur.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQIMPO\_STRUC\_ID 'dir.

### **Sürüm (MQUZE)**

Sorgu iletişi özellik seçenekleri yapısı-Sürüm alanı

Bu, yapı sürüm numarasıdır. Değer şu olmalıdır:

## **MQIMPO\_VERSION\_1**

Sorgu iletisi özellik seçenekleri yapısı için sürüm numarası.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

## **MQIMPO\_CURRENT\_VERSION**

Sorgu iletisi özellik seçenekleri yapısının yürürlükteki sürümü.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQIMPO\_VERSION\_1' dir.

## **Seçenekler (MQUZE)**

Sorgu iletisi özellik seçenekleri yapısı-Seçenekler alanı

İzleyen seçenekler MQINQMP ' nin işlemini denetler. Bu seçeneklerden birini ya da birkaçını belirleyebilirsiniz. birden çok seçenek belirtmek için, değerleri bir araya ekleyin (aynı değişmezi bir kereden fazla eklemeyin) ya da bit ya da işlemi kullanarak değerleri birleştirin (programlama dili, bit işlemleri destekliyse).

Geçerli olmayan seçenek birleşimleri dikkat edilir; diğer tüm birleşimler geçerlidir.

**Değer verisi seçenekleri:** Aşağıdaki seçenekler, özellik iletiden alınınca değer verilerinin işlenmesiyle ilgilidir.

## **MQIMPO\_CONVERT\_VALUE**

Bu seçenek, özellik değerinin, MQINQMP çağrısından önce belirtilen *RequestedCCSID* ve *RequestedEncoding* değerlerine, *Value* alanındaki özellik değerini döndürmesi için dönüştürülmesini ister.

- Dönüştürme başarılı olursa, *ReturnedCCSID* ve *ReturnedEncoding* alanları, MQINQMP çağrısından dönüşte *RequestedCCSID* ve *RequestedEncoding* ile aynı şekilde ayarlanır.
- Dönüştürme başarısız olursa, ancak MQINQMP çağrısı hatasız tamamlanırsa, özellik değeri dönüştürülmez olarak döndürülür.

Özellik bir dizelse, *ReturnedCCSID* ve *ReturnedEncoding* alanları, dönüştürülenmemiş dizginin karakter takımı ve kodlamasına ayarlanır.

Tamamlanma kodu bu durumda MQCC\_UYARI olur; neden kodu MQRC\_PROP\_VALUE\_NOT\_CONVERTED neden kodu. Özellik imleci döndürülen özelliğe iletilir.

Özellik değeri dönüştürme sırasında genişler ve **Value** parametresinin boyutunu aşarsa, MQCC\_FAILED tamamlanma kodu ile değer dönüştürülemez olarak döndürülür; neden kodu MQRC\_PROPERTY\_VALUE\_TOO\_BIG olarak ayarlanır.

MQINQMP çağrısının **DataLength** değiştirgesi, uygulamanın dönüştürülen özellik değerini sığdırmak için gereken arabelleğin büyüklüğünü belirlemesine olanak vermek için, özellik değerinin dönüştürüleceğini döndürür. Özellik imleci değişmeden kalır.

Bu seçenek aşağıdaki bilgileri de ister:

- Özellik adı bir genel arama karakteri içeriyorsa ve
- *ReturnedName* alanı, döndürülen ad için bir adres ya da görelî konum ile başlatılır.

Döndürülen ad, *RequestedCCSID* ve *RequestedEncoding* değerlerine uygun olarak dönüştürülür.

- Dönüştürme başarılı olursa, *ReturnedName* ' in *VSCCSID* alanı ve döndürülen adın kodlaması, *RequestedCCSID* ve *RequestedEncoding* giriş değerine ayarlanır.
- Dönüştürme başarısız olursa, ancak MQINQMP çağrısı hata ya da uyarı olmadan tamamlanırsa, döndürülen ad dönüştürülmez. Tamamlanma kodu bu durumda MQCC\_UYARI olur; neden kodu MQRC\_PROP\_NAME\_NOT\_CONVERTED neden kodu.

Özellik imleci döndürülen özelliğe iletilir. Hem değer, hem de ad dönüştürülmezse, MQRC\_PROP\_VALUE\_NOT\_CONVERTED döndürülür.

Döndürülen ad dönüştürme sırasında genişler ve *RequestedName* alanının *VSBufsize* alanının büyüklüğünü aşarsa, döndürülen dizgi dönüştürülemez olarak bırakılır; tamamlanma kodu MQCC\_FAILED ve neden kodu MQRC\_XX\_ENCODE\_CASE\_ONE properties \_NAME\_TOO\_BÜYÜK olarak ayarlıdır.

MQCHARV yapısının *VSLength* alanı, uygulamanın dönüştürülen özellik değerini barındırmak için gereken arabelleğin büyüklüğünü belirlemesine olanak vermek için, özellik değerinin dönüştürüleceğini döndürür. Özellik imleci değişmeden kalır.

### **MQIMPO\_CONVERT\_TYPE**

Bu seçenek, özelliğin değerinin yürürlükteki veri tipinden, MQINQMP çağrısının **Type** parametresinde belirtilen veri tipine dönüştürülmesini ister.

- Dönüştürme başarılı olursa, MQINQMP çağrısının geri döndürülmesi sırasında **Type** parametresi değişmez.
- Dönüştürme başarısız olursa, ancak MQINQMP çağrısı hatasız olarak tamamlanırsa, çağrı başarısız olur; neden MQRC\_PROP\_CONV\_NOT\_SUPPORTED nedeniyle başarısız olur. Özellik imleci değişmeden kalır.

Veri tipinin dönüştürülmesi, dönüştürme sırasında değer genişletilmesine neden olursa ve dönüştürülen değer, **Value** parametresinin boyutunu aşarsa, değer geri çevrilmeden geri döndürülür; tamamlanma kodu MQCC\_FAILED ve neden kodu MQRC\_PROPERTY\_VALUE\_TOO\_BIG olarak ayarlanır.

MQINQMP çağrısının **DataLength** değiştirgesi, uygulamanın dönüştürülen özellik değerini sığdırmak için gereken arabelleğin büyüklüğünü belirlemesine olanak vermek için, özellik değerinin dönüştürüleceğini döndürür. Özellik imleci değişmeden kalır.

MQINQMP çağrısının **Type** değiştirgesinin değeri geçerli değilse, çağrı neden MQRC\_XX\_ENCODE\_CASE\_ONE property\_type\_error ile başarısız olur.

İstenen veri tipi dönüşümü desteklenmiyorsa, çağrıya MQRC\_PROP\_CONV\_NOT\_SUPPORTED neden başarısız olur. Aşağıdaki veri tipi dönüşümleri desteklenmektedir:

Özellik veri tipi	Desteklenen hedef veri tipleri
MQTYPE_BOOLEAN	MQTYPE_STRING, MQTYPE_INT8, MQTYPE_INT16, MQTYPE_INT32, MQTYPE_INT64
MQTYPE_BYTE_STRING	MQTYPE_STRING
MQTYPE_INT8	MQTYPE_STRING, MQTYPE_INT16, MQTYPE_INT32, MQTYPE_INT64
MQTYPE_INT16	MQTYPE_STRING, MQTYPE_INT32, MQTYPE_INT64
MQTYPE_INT32	MQTYPE_STRING, MQTYPE_INT64
MQTYPE_INT64	MQTYPE_STRING
MQTYPE_FLOAT32	MQTYPE_STRING, MQTYPE_FLOAT64
MQTYPE_FLOAT64	MQTYPE_STRING
MQTYPE_STRING	MQTYPE_BOOLEAN, MQTYPE_INT8, MQTYPE_INT16, MQTYPE_INT32, MQTYPE_INT64, MQTYPE_FLOAT32, MQTYPE_FLOAT64
MQTYPE_NULL	Yok

Desteklenen dönüşümleri yöneten genel kurallar aşağıdaki gibidir:

- Sayısal özellik değerleri, dönüştürme sırasında hiçbir veri kaybolmaması koşuluyla, bir veri tipinden diğerine dönüştürülebilir.



For example, the value of a property with data type MQTYPE\_INT32 can be converted into a value with data type MQTYPE\_INT64, but cannot be converted into a value with data type MQTYPE\_INT16.

- Herhangi bir veri tipindeki bir özellik değeri bir dizgiye dönüştürülebilir.
- Bir dizgi özelliği değeri, dizginin dönüştürme için doğru şekilde biçimlendirilebilmesi koşuluyla, diğer herhangi bir veri tipine dönüştürülebilir. Bir uygulama, biçimlendirilmemiş bir dizgi özelliği değerini doğru şekilde dönüştürmeye çalışırsa, IBM MQ neden kodu MQRC\_PROP\_NUMBER\_FORMAT\_ERROR neden kodunu döndürür.
- Bir uygulama, desteklenmeyen bir dönüştürme girişiminde bulunursa, IBM MQ neden kodu MQRC\_PROP\_CONV\_NOT\_DESPORTED neden kodunu döndürür.

Bir özellik değerini bir veri tipinden diğerine dönüştürmeye ilişkin belirli kurallar şunlardır:

- Bir MQTYPE\_BOOLEAN özellik değerini dizgiye dönüştürürken, TRUE değeri "TRUE" dizgisine dönüştürülür ve false değeri "FALSE" dizgisine dönüştürülür.
- Bir MQTYPE\_BOOLEAN özellik değeri sayısal bir veri tipine dönüştürülürken, TRUE değeri bir değere dönüştürülür ve FALSE değeri sıfıra dönüştürülür.
- Bir dizgi özelliği değerini MQTYPE\_BOOLEAN değerine dönüştürürken, "TRUE" ya da "1" dizgisi TRUE olarak dönüştürülür ve "FALSE" ya da "0" dizgisi FALSE olarak dönüştürülür.

"TRUE" ve "FALSE" terimlerinin büyük ve küçük harfe duyarlı olmadığını unutmayın.

Başka bir dizgi dönüştürülemez; IBM MQ , MQRC\_PROP\_NUMBER\_FORMAT\_ERROR neden kodunu döndürür.

- Bir dizgi özelliği değerini MQTYPE\_INT8, MQTYPE\_INT16, MQTYPE\_INT32 ya da MQTYPE\_INT64 veri tipli bir değere dönüştürürken, dizginin şu biçimde olması gerekir:

```
[blanks][sign]digits
```

Dizginin bileşenlerinin anlamları aşağıdaki gibidir:

#### **blanks**

İsteğe bağlı olarak kullanılan boş karakterler

#### **sign**

İsteğe bağlı artı işareti (+) ya da eksi işareti (-) karakteri.

#### **digits**

Ardışık rakam karakterleri dizisi (0-9). En az bir sayı karakteri var olmalıdır.

Sayı karakterleri sırasından sonra dizgi, rakam karakteri olmayan diğer karakterleri içerebilir, ancak bu karakterlerin ilk kez ulaşıldığı anda dönüştürme durakları durur. Dizilimin ondalık bir tamsayıyı temsil ettiği varsayılır.

IBM MQ , dizgi doğru şekilde biçimlendirilmediyse, MQRC\_PROP\_NUMBER\_FORMAT\_ERROR neden kodunu döndürür.

- Bir dizgi özelliği değerini MQTYPE\_FLOAT32 ya da MQTYPE\_FLOAT64 veri tipine sahip bir değere dönüştürürken, dizginin şu biçimde olması gerekir:

```
[blanks][sign]digits[.digits][e_char[e_sign]e_digits]
```

Dizginin bileşenlerinin anlamları aşağıdaki gibidir:

#### **blanks**

İsteğe bağlı olarak kullanılan boş karakterler

#### **sign**

İsteğe bağlı artı işareti (+) ya da eksi işareti (-) karakteri.

#### **digits**

Ardışık rakam karakterleri dizisi (0-9). En az bir sayı karakteri var olmalıdır.

### **e\_char**

"E" ya da "e" olan bir üstel karakter.

### **e\_sign**

Üstel için isteğe bağlı artı işareti (+) ya da eksi işareti (-) karakteri.

### **e\_digits**

Üs için bitişik sayı karakterleri dizisi (0-9). Dizgi bir üstel karakter içeriyorsa en az bir sayı karakteri var olmalıdır.

Sayı karakterleri sırasından sonra ya da bir üssü temsil eden isteğe bağlı karakterler sonra, dizgi sayı karakteri olmayan diğer karakterleri içerebilir, ancak bu karakterlere ilk ulaşıldığında dönüştürme durakları durur. Dizginin, 10 'un gücü olan bir üstel bir ondalık kayan noktalı sayıyı temsil ettiği varsayılır.

IBM MQ , dizgi doğru şekilde biçimlendirilmediyse, MQRC\_PROP\_NUMBER\_FORMAT\_ERROR neden kodunu döndürür.

- Bir sayısal özellik değerini bir dizgiye dönüştürürken, değer, bu değere ilişkin ASCII karakterini içeren dizgi değil, değer dizgi gösterimine ondalık sayı olarak dönüştürülür. Örneğin, 65 tamsayı, "A" dizgisine değil, "65" dizgisine dönüştürülür.
- Bir bayt dizilimi özellik değerini bir dizgiye dönüştürürken, her bayt bayt 'ı temsil eden iki onaltılı karaktere dönüştürülür. Örneğin, bayt dizisi {0xF1, 0x12, 0x00, 0xFF} , "F11200FF" dizgisine dönüştürülür.

### **MQIMPO\_QUERY\_LENGTH**

Özellik değerinin tipini ve uzunluğunu sorgulayın. MQINQMP çağrısının **DataLength** parametresine uzunluk değeri döndürülür. Özellik değeri döndürülmez.

Bir **ReturnedName** arabelleği belirtilirse, MQCHARV yapısının *VSLength* alanı, özellik adının uzunluğuna uygun olarak doldurulur. Özellik adı döndürülmez.

**Yineleme seçenekleri:** Aşağıdaki seçenekler, genel arama karakteri içeren bir ad kullanarak, özellikler üzerinde yineleme ile ilgilidir.

### **MQIMPO\_INQ\_FIRST**

Belirtilen adla eşleşen ilk özelliğe ilişkin bilgi edinmek için. Bu çağrıdan sonra, döndürülen özellik üzerinde bir geçici çizelge oluşturulur.

Bu varsayılan değerdir.

MQIMPO\_INQ\_PROP\_UNDER\_CURSOR seçeneği daha sonra bir MQINQMP çağrısıyla birlikte kullanılabilir (gerekliyorsa, aynı özelliği yeniden sorgulamak için).

Yalnızca bir özellik imlecinin olduğunu unutmayın; bu nedenle, MQINQMP çağrısında belirlenen özellik adı belirlenirse, imleç ilk durumuna getirilir.

Bu seçenek, aşağıdaki seçeneklerin herhangi biri ile geçerli değildir:

MQIMPO\_INQ\_Next (SONRAKI)  
MQIMPO\_INQ\_PROP\_UNDER\_CURSOR

### **MQIMPO\_INQ\_Next (SONRAKI)**

Belirtilen adla eşleşen sonraki özelliği sorgular, aramayı özellik imlecinden devam ettirir. İmleç, döndürülen özelliğe ilerletilir.

Bu, belirtilen ad için ilk MQINQMP çağrıyla, belirtilen ad ile eşleşen ilk özellik döndürülür.

MQIMPO\_INQ\_PROP\_UNDER\_CURSOR seçeneği, gerekirse bir MQINQMP çağrısıyla, aynı özelliği yeniden sorgulamak için kullanılabilir.

İmlecini altındaki özellik silindiyse, MQINQMP, silinen bir sonraki özellik değerini döndürür.

Genel arama karakteriyle eşleşen bir özellik eklenirse, bir yineleme devam ederken, bu özellik yinelemenin tamamlanması sırasında döndürülmeyebilir ya da iade edilmeyebilir. MQIMPO\_INQ\_FIRST kullanılarak yineleme yeniden başlatıldıktan sonra özellik döndürülmektedir.

Silinme işlemi devam ederken silinen genel arama karakteriyle eşleşen bir özellik, silme işleminin ardından döndürülmez.

Bu seçenek, aşağıdaki seçeneklerin herhangi biri ile geçerli değildir:

MQIMPO\_INQ\_FIRST  
MQIMPO\_INQ\_PROP\_UNDER\_CURSOR

#### **MQIMPO\_INQ\_PROP\_UNDER\_CURSOR**

Özellik imlecine işaret edilen özelliğin değerini alır. Özellik imlecinin gösterdiği özellik, MQIMPO\_INQ\_FIRST ya da MQIMPO\_INQ\_Next seçeneğini kullanarak son olarak sorgulanandır.

İleti tanıtıcısı yeniden kullanıldığında, bir MQGET çağrısındaki MQGMO ' nun *MsgHandle* alanında ileti tanıtıcısı belirtildiğinde ya da bir MQPUT çağrısındaki MQPMO yapısının *OriginalMsgHandle* ya da *NewMsgHandle* alanlarında ileti tanıtıcısı belirtildiğinde, özellik imleci sıfırlanır.

Bu seçenek, özellik imleci henüz oluşturulmadığında ya da özellik imlecinin gösterdiği özellik silindiğinde kullanılırsa, çağrı tamamlanma kodu MQCC\_FAILED ile başarısız olur ve MQRC\_PROPERTY\_NOT\_AVAISLE neden olur.

Bu seçenek, aşağıdaki seçeneklerin herhangi biri ile geçerli değildir:

MQIMPO\_INQ\_FIRST  
MQIMPO\_INQ\_Next (SONRAKI)

Önceden tanımlanan seçeneklerden hiçbiri gerekli değilse, aşağıdaki seçenek kullanılabilir:

#### **MQIMPO\_NONE**

Başka bir seçenek belirtilmemeyi belirtmek için bu değeri kullanın; tüm seçenekler varsayılan değerlerini kabul eder.

MQIMPO\_NONE aids programı belgeleri; bu seçeneğin diğeriyle birlikte kullanılması amaçlanmaz, ancak değeri sıfır olduğu için, bu tür bir kullanım saptanamaz.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQIMPO\_INQ\_FIRST ' dir.

#### **RequestedEncoding (MQUZE)**

Sorgu iletisi özellik seçenekleri yapısı- RequestedEncoding alanı

Bu, sorgulayan özellik değerinin MQIMPO\_CONVERT\_VALUE ya da MQIMPO\_CONVERT\_TYPE belirtildiğinde dönüştürüleceği kodlamadır.

Bu alanın ilk değeri MQENC\_NATIVE 'dir.

#### **RequestedCCSID (MQUZE)**

Sorgu iletisi özellik seçenekleri yapısı- RequestedCCSID alanı

Değer, bir karakter dizgisiyse, sorgulayan özellik değerinin dönüştürülmesini sağlar. Bu, aynı zamanda, MQIMPO\_CONVERT\_VALUE ya da MQIMPO\_CONVERT\_TYPE belirtildiğinde *ReturnedName* ' in dönüştürüleceği karakter kümesidir.

Bu alanın ilk değeri MQCCSI\_APPL 'dir.

#### **ReturnedEncoding (MQlong)**

İleti özelliği seçenekleri yapısı- ReturnedEncoding alanı

Çıkışta, döndürülen değer kodlaması bu.

MQIMPO\_CONVERT\_VALUE seçeneği belirtilirse ve dönüştürme başarılı olursa, dönüş sırasında *ReturnedEncoding* alanı, geçirilen değerle aynı değerde olur.

Bu alanın ilk değeri MQENC\_NATIVE 'dir.

## **ReturnedCCSID (MQHOT)**

İleti özelliği seçenekleri yapısı- ReturnedCCSID alanı

Çıkışta, MQINQMP çağrısının **Type** değiştirgesi MQTYPE\_STRING ise, döndürülen değer karakter takılıdır.

MQIMPO\_CONVERT\_VALUE seçeneği belirtilirse ve dönüştürme başarılı olursa, dönüş sırasında *ReturnedCCSID* alanı, geçirilen değerle aynı değerde olur.

Bu alanın ilk değeri sıfır.

## **Reserved1 (MQCHAR)**

Bu ayrılmış bir alandır. Bu alanın ilk değeri boşluk karakteridir (4 byte 'lık alan).

## **ReturnedName (MQCHARV)**

Sorgu iletisi özellik seçenekleri yapısı- ReturnedName alanı

Sorgulamanın gerçek adı.

Girişteki bir dizgi arabelleği, MQCHARV yapısının *VSPtr* ya da *VSOffset* alanı kullanılarak aktarılabilir. Dizgi arabelleğindeki uzunluk, MQCHARV yapısının *VSBuFSIZE* alanı kullanılarak belirtilir.

MQINQMP çağrısından geri dönerek, dizgi arabelleğinin adı tam olarak içermeye yetecek kadar uzun olması koşuluyla, dizgi arabelleği sorgulayan özelliğin adıyla tamamlanır. MQCHARV yapısına ilişkin *VSLength* alanı, özellik adının uzunluğuna sahip olarak doldurulur. Döndürülen adın karakter kümesini belirtmek için, MQCHARV yapısına ilişkin *VSCCSID* alanı doldurulur. Bu ad, adın dönüştürülmesinin başarısız olup olmadığını gösterir.

Bu bir giriş/çıkış alanıdır. Bu alanın ilk değeri MQCHARV\_DEFAULT 'tır.

## **TypeString (MQCHAR8)**

Sorgu iletisi özellik seçenekleri yapısı- TypeString alanı

Özelliğin veri tipinin dizgi gösterimi.

Özellik bir MQRFH2 üstbilgisinde belirtildiyse ve MQRFH2 dt özniteliği tanınmadıysa, özelliğin veri tipini saptamak için bu alan kullanılabilir. *TypeString* , 1208 (UTF-8) kodlu karakter kümesinde döndürülür ve tanınması başarısız olan özelliğin dt özniteliğinin ilk sekiz bayttır.

Bu her zaman bir çıkış alanıdır. Bu alanın başlangıç değeri, C programlama dilinde boş değer, diğer programlama dillerindeki 8 boş karakter dizilimidir.

## **MQMD-İleti tanımlayıcı**

MQMD yapısı, bir ileti gönderme ve alma uygulamaları arasında dolaşırken uygulama verileriyle birlikte gönderilen denetim bilgilerini içerir. Yapı, MQGET, MQPUT ve MQPUT1 çağrılarında bir giriş/çıkış değiştirgesidir.

## **Kullanılabilirlik**

Tüm IBM MQ sistemleri ve bu sistemlere bağlı IBM MQ MQI clients .

## **Sürüm**

MQMD ' nin yürürlükteki sürümü: MQMD\_VERSION\_2. Birkaç ortam arasında taşınabilir olması amaçlanan uygulamalar, ilgili tüm ortamlarda MQMD ' nin gerekli sürümünün desteklendiğinden emin olmalıdır. Yapının yalnızca daha yeni sürümlerinde bulunan alanlar, izleyen açıklamalarda bu şekilde tanımlanır.

Desteklenen programlama dilleri için sağlanan üstbilgi, COPY ve INCLUDE dosyaları, ortam tarafından desteklenen en son MQMD sürümünü içerir, ancak *Version* alanının ilk değeri MQMD\_VERSION\_1olarak ayarlanmıştır. version-1 yapısında var olmayan alanları kullanmak için uygulamanın *Version* alanını gerekli sürümün sürüm numarasına ayarlaması gerekir.

version-1 yapısına ilişkin bir bildirim MQMD1adıyla kullanılabilir.

## Karakter kümesi ve kodlama

MQMD ' deki veriler, yerel kuyruk yöneticisinin karakter kümesi ve kodlamasında olmalıdır; bunlar **CodedCharSetId** kuyruk yöneticisi özneliği ve MQENC\_NATIVE ile verilir. Ancak, uygulama IBM MQ MQI clientolarak çalışıyorsa, yapı istemcinin karakter kümesinde ve kodlamasında olmalıdır.

Gönderen ve alan kuyruk yöneticileri farklı karakter kümeleri ya da kodlamalar kullanıyorsa, MQMD ' deki veriler otomatik olarak dönüştürülür. Uygulamanın MQMD ' yi dönüştürmesi gerekli değildir.

## MQMD ' nin farklı sürümlerini kullanma

version-2 MQMD, version-1 MQMD ' nin kullanılmasıyla ve ileti verilerinin MQMDE yapısıyla önceden düzeltilmesiyle eşdeğerdir. Ancak, MQMDE yapısındaki tüm alanların varsayılan değerleri varsa, MQMDE atlanabilir. version-1 MQMD ve MQMDE aşağıda açıklandığı gibi kullanılır:

- MQPUT ve MQPUT1 çağrılarında uygulama bir version-1 MQMD sağlarsa, uygulama isteğe bağlı olarak ileti verilerine bir MQMDE öneki gönderebilir ve MQMDE 'nin var olduğunu göstermek için MQMD ' deki *Format* alanını MQFMT\_MD\_EXTENSION olarak ayarlayabilir. Uygulama bir MQMDE sağlamazsa, kuyruk yöneticisi MQMDE ' deki alanlar için varsayılan değerleri varsayar.

**Not:** version-2 MQMD 'de var olan, ancak version-1 MQMD ' de olmayan alanlar, MQPUT ve MQPUT1 çağrılarındaki giriş/çıkış alanlarıdır. Ancak, kuyruk yöneticisi MQPUT ve MQPUT1 çağrılarının çıkışında MQMDE ' deki eşdeğer alanlarda herhangi bir değer döndürmez; uygulama bu çıkış değerlerini gerektiriyorsa, bir version-2 MQMD kullanmalıdır.

- MQGET çağrısında, uygulama version-1 MQMD sağlarsa, kuyruk yöneticisi MQMDE ile döndürülen iletiyi örnekler, ancak yalnızca MQMDE ' deki alanlardan birinin ya da daha fazlasının varsayılan olmayan bir değeri varsa. MQMD 'deki *Format* alanı, MQMDE ' nin var olduğunu belirtmek için MQFMT\_MD\_EXTENSION değerine sahip olacaktır.

Kuyruk yöneticisinin MQMDE ' deki alanlar için kullandığı varsayılan değerler, [Çizelge 504 sayfa 467](#) içinde gösterilen bu alanların ilk değerleriyle aynıdır.

Bir ileti iletim kuyruğunda olduğunda, MQMD ' deki bazı alanlar belirli değerlere ayarlanır; ayrıntılar için bkz. [“MQXQH-İletim kuyruğu üstbilgisi” sayfa 608](#) .

## İleti bağılamı

MQMD ' deki bazı alanlar ileti bağılamını içerir. İki tip ileti bağılamı vardır: *kimlik bağılamı* ve *kaynak bağılamı*. Genellikle:

- Kimlik bağılamı, iletiyi *aslen* koyan uygulamayla ilgilidir
- Kaynak bağılamı, iletiyi koyan *en son* uygulamayla ilgilidir.

Bu iki uygulama aynı uygulama olabilir, ancak farklı uygulamalar da olabilir (örneğin, bir ileti bir uygulamadan diğerine iletildiğinde).

Kimlik ve kaynak bağılamın genellikle açıklanan anlamları olsa da, MQMD ' deki her iki tip bağılam alanının içeriği, ileti konduğunda belirtilen MQPMO\_ \* \_CONTEXT seçeneklerine bağlıdır. Sonuç olarak, kimlik bağılamı, ilk olarak iletiyi koyan uygulamayla ilgili olmak zorunda değildir ve kaynak bağılamı, mesajı en son koyan uygulamayla ilişkili olmak zorunda değildir; uygulama paketinin tasarımına bağlıdır.

İleti kanalı aracısı (MCA) hiçbir zaman ileti bağılamını değiştirir. Uzak kuyruk yöneticilerinden ileti alan MCA ' lar MQPUT ya da MQPUT1 çağrısında MQPMO\_SET\_ALL\_CONTEXT bağılam seçeneğini kullanır. Bu, alan MCA 'nın gönderen MCA' dan gelen iletiyle birlikte gönderilen ileti bağılamını tam olarak korumasını sağlar. Ancak sonuç, kaynak bağılamın iletiyi gönderen ve alan MCA ' ların herhangi biriyle ilgili olmamasıdır.

Kaynak bağlam, iletiyi koyan daha önceki bir uygulamaya gönderme yapıyor. Tüm ara uygulamalar ileti bağlamını geçtiyse, kaynak bağlam kaynak uygulamanın kendisini gösterir.

Tanımlarda, bağlam alanları daha önce anlatıldığı gibi kullanılır. İleti bağlamıyla ilgili daha fazla bilgi için bkz. [İleti bağlamı](#).

## Alanlar

**Not:** Aşağıdaki tabloda, alanlar alfabetik olarak değil, kullanıma göre gruplanır. Alt konular aynı sırayı izler.

Çizelge 501. MQMD ' de MQMD için alanlar		
Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
<a href="#">StrucId</a> (yapı tanıtıcısı)	MQMD_STRUC_ID	' MD '
<a href="#">Sürüm</a> (yapı sürümü numarası)	MQMD_VERSION_1	1
<a href="#">Rapor</a> (rapor iletileri için seçenekler)	MQRO_NONE	0
<a href="#">MsgType</a> (ileti tipi)	MQMT_DATAGRAM	8
<a href="#">MQMD-Süre bitimi alanı</a> (ileti geçerlik süresi)	MQEI_UNLIMITED	-1
<a href="#">MQMD-Geribildirim alanı</a> (geribildirim ya da neden kodu)	MQFB_NONE	0
<a href="#">Kodlama</a> (ileti verilerinin sayısal kodlaması)	MQENC_NATIVE	Ortama bağlıdır
<a href="#">CodedCharSetId</a> (ileti verilerinin karakter kümesi tanıtıcısı)	MQCCSI_Q_MGR	0
<a href="#">Biçim</a> (ileti verilerinin biçim adı)	MQFMT_NONE	Boşluklar
<a href="#">Öncelik</a> (ileti önceliği)	MQPRI_PRIORITY_AS_Q_DEF	-1
<a href="#">Kalıcılık</a> (ileti kalıcılığı)	MQPER_PERSISTENCE_AS_Q_DEF	2
<a href="#">MQMD- MsgId alanı</a> (ileti tanıtıcısı)	MQMI_NONE	Boş Değerler
<a href="#">CorrelId</a> (ilinti tanıtıcısı)	MQCI_NONE	Boş Değerler
<a href="#">BackoutCount</a> (geriletme sayacı)	Yok	0
<a href="#">ReplyToQ</a> (yanıt kuyruğunun adı)	Yok	Boş değerli dizilim ya da boşluklar
<a href="#">ReplyToQMgr</a> (yanıt kuyruğu yöneticisinin adı)	Yok	Boş değerli dizilim ya da boşluklar
<a href="#">UserIdentifier</a> (kullanıcı kimliği)	Yok	Boş değerli dizilim ya da boşluklar
<a href="#">AccountingToken</a> (muhasebe simgesi)	MQACT_NONE	Boş Değerler
<a href="#">ApplIdentityVerileri</a> (kimlikle ilgili uygulama verileri)	Yok	Boş değerli dizilim ya da boşluklar
<a href="#">PutApplTip</a> (iletiyi koyan uygulamanın tipi)	MQAT_NO_CONTEXT	0
<a href="#">PutApplAdı</a> (iletiyi koyan uygulamanın adı)	Yok	Boş değerli dizilim ya da boşluklar
<a href="#">PutDate</a> (iletinin gönderildiği tarih)	Yok	Boş değerli dizilim ya da boşluklar

Çizelge 501. MQMD ' de MQMD için alanlar (devamı var)

Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
<u>PutTime</u> (iletinin konma zamanı)	Yok	Boş değerli dizilim ya da boşluklar
<u>ApplOriginVerileri</u> (kaynak ile ilgili uygulama verileri)	Yok	Boş değerli dizilim ya da boşluklar
<b>Not:</b> <i>Version</i> , MQMD_VERSION_2' den küçükse kalan alanlar yoksayılır.		
<u>GroupId</u> (grup tanıtıcısı)	MQGI_NONE	Boş Değerler
<u>MsgSeqNumber</u> (grup içindeki mantıksal iletinin sıra numarası)	Yok	1
<u>Görelî Konum</u> (mantıksal iletinin başlangıcından fiziksel iletideki verilerin görelî konumu)	Yok	0
<u>MQMD- MsgFlags alanı</u> (ileti işaretleri)	MQMF_NONE	0
<u>OriginalLength</u> (özgün iletinin uzunluğu)	MQOL_UNDEFINED	-1
<p><b>Notlar:</b></p> <ol style="list-style-type: none"> <li>1. Boş değerli dizilim ya da boşluk değeri, C dizgisindeki boş dizgiyi ve diğer programlama dillerindeki boş karakterleri gösterir.</li> <li>2. C programlama dilinde, makro değişkeniMQMD_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlar için başlangıç değerleri sağlamak üzere aşağıdaki şekilde kullanılabilir:</li> </ol> <pre style="background-color: #f0f0f0; padding: 10px;">MQMD MyMD = {MQMD_DEFAULT};</pre>		

## Dil bildirimleri

### MQMD için C bildirimi

```
typedef struct tagMQMD MQMD;
struct tagMQMD {
    MQCHAR4  StrucId;           /* Structure identifier */
    MQLONG   Version;         /* Structure version number */
    MQLONG   Report;         /* Options for report messages */
    MQLONG   MsgType;        /* Message type */
    MQLONG   Expiry;         /* Message lifetime */
    MQLONG   Feedback;       /* Feedback or reason code */
    MQLONG   Encoding;       /* Numeric encoding of message data */
    MQLONG   CodedCharSetId; /* Character set identifier of message
                             data */
    MQCHAR8  Format;          /* Format name of message data */
    MQLONG   Priority;        /* Message priority */
    MQLONG   Persistence;    /* Message persistence */
    MQBYTE24 MsgId;          /* Message identifier */
    MQBYTE24 CorrelId;       /* Correlation identifier */
    MQLONG   BackoutCount;   /* Backout counter */
    MQCHAR48 ReplyToQ;       /* Name of reply queue */
    MQCHAR48 ReplyToQMGr;    /* Name of reply queue manager */
    MQCHAR12 UserIdentifier; /* User identifier */
    MQBYTE32 AccountingToken; /* Accounting token */
    MQCHAR32 ApplIdentityData; /* Application data relating to
                             identity */
    MQLONG   PutAppType;     /* Type of application that put the
                             message */
    MQCHAR28 PutAppName;    /* Name of application that put the
                             message */
    MQCHAR8  PutDate;       /* Date when message was put */
    MQCHAR8  PutTime;       /* Time when message was put */
    MQCHAR4  ApplOriginData; /* Application data relating to origin */
};
```

```

MQBYTE24  GroupId;          /* Group identifier */
MQLONG    MsgSeqNumber;    /* Sequence number of logical message
                           within group */
MQLONG    Offset;          /* Offset of data in physical message
                           from start of logical message */
MQLONG    MsgFlags;        /* Message flags */
MQLONG    OriginalLength;  /* Length of original message */
};

```

## MQMD için COBOL bildirimi

```

** MQMD structure
 10 MQMD.
** Structure identifier
 15 MQMD-STRUCID          PIC X(4).
** Structure version number
 15 MQMD-VERSION         PIC S9(9) BINARY.
** Options for report messages
 15 MQMD-REPORT          PIC S9(9) BINARY.
** Message type
 15 MQMD-MSGTYPE         PIC S9(9) BINARY.
** Message lifetime
 15 MQMD-EXPIRY          PIC S9(9) BINARY.
** Feedback or reason code
 15 MQMD-FEEDBACK        PIC S9(9) BINARY.
** Numeric encoding of message data
 15 MQMD-ENCODING        PIC S9(9) BINARY.
** Character set identifier of message data
 15 MQMD-CODEDCHARSETID PIC S9(9) BINARY.
** Format name of message data
 15 MQMD-FORMAT          PIC X(8).
** Message priority
 15 MQMD-PRIORITY        PIC S9(9) BINARY.
** Message persistence
 15 MQMD-PERSISTENCE     PIC S9(9) BINARY.
** Message identifier
 15 MQMD-MSGID           PIC X(24).
** Correlation identifier
 15 MQMD-CORRELID        PIC X(24).
** Backout counter
 15 MQMD-BACKOUTCOUNT   PIC S9(9) BINARY.
** Name of reply queue
 15 MQMD-REPLYTOQ        PIC X(48).
** Name of reply queue manager
 15 MQMD-REPLYTOQMGR     PIC X(48).
** User identifier
 15 MQMD-USERIDENTIFIER  PIC X(12).
** Accounting token
 15 MQMD-ACCOUNTINGTOKEN PIC X(32).
** Application data relating to identity
 15 MQMD-APPLIDENTITYDATA PIC X(32).
** Type of application that put the message
 15 MQMD-PUTAPPLTYPE     PIC S9(9) BINARY.
** Name of application that put the message
 15 MQMD-PUTAPPLNAME     PIC X(28).
** Date when message was put
 15 MQMD-PUTDATE         PIC X(8).
** Time when message was put
 15 MQMD-PUTTIME         PIC X(8).
** Application data relating to origin
 15 MQMD-APPLORIGINDATA  PIC X(4).
** Group identifier
 15 MQMD-GROUPID         PIC X(24).
** Sequence number of logical message within group
 15 MQMD-MSGSEQNUMBER    PIC S9(9) BINARY.
** Offset of data in physical message from start of logical message
 15 MQMD-OFFSET          PIC S9(9) BINARY.
** Message flags
 15 MQMD-MSGFLAGS        PIC S9(9) BINARY.
** Length of original message
 15 MQMD-ORIGINALLENGTH  PIC S9(9) BINARY.

```

## MQMD için PL/I bildirimi

```

dcl
  1 MQMD based,
  3 StrucId          char(4),          /* Structure identifier */

```



```

3 Version          fixed bin(31), /* Structure version number */
3 Report           fixed bin(31), /* Options for report messages */
3 MsgType          fixed bin(31), /* Message type */
3 Expiry           fixed bin(31), /* Message lifetime */
3 Feedback         fixed bin(31), /* Feedback or reason code */
3 Encoding         fixed bin(31), /* Numeric encoding of message
                               data */
3 CodedCharSetId  fixed bin(31), /* Character set identifier of
                               message data */
3 Format            char(8), /* Format name of message data */
3 Priority          fixed bin(31), /* Message priority */
3 Persistence      fixed bin(31), /* Message persistence */
3 MsgId            char(24), /* Message identifier */
3 CorrelId         char(24), /* Correlation identifier */
3 BackoutCount     fixed bin(31), /* Backout counter */
3 ReplyToQ         char(48), /* Name of reply queue */
3 ReplyToQMgr      char(48), /* Name of reply queue manager */
3 UserIdentifier   char(12), /* User identifier */
3 AccountingToken  char(32), /* Accounting token */
3 ApplIdentityData char(32), /* Application data relating to
                               identity */
3 PutApplType      fixed bin(31), /* Type of application that put the
                               message */
3 PutApplName      char(28), /* Name of application that put the
                               message */
3 PutDate          char(8), /* Date when message was put */
3 PutTime          char(8), /* Time when message was put */
3 ApplOriginData  char(4), /* Application data relating to
                               origin */
3 GroupId          char(24), /* Group identifier */
3 MsgSeqNumber     fixed bin(31), /* Sequence number of logical
                               message within group */
3 Offset           fixed bin(31), /* Offset of data in physical
                               message from start of logical
                               message */
3 MsgFlags         fixed bin(31), /* Message flags */
3 OriginalLength   fixed bin(31); /* Length of original message */

```

#### MQMD için High Level Assembler (Üst Düzey Çevirici) bildirimi

```

MQMD              DSECT
MQMD_STRUCID      DS   CL4   Structure identifier
MQMD_VERSION      DS   F     Structure version number
MQMD_REPORT       DS   F     Options for report messages
MQMD_MSGTYPE      DS   F     Message type
MQMD_EXPIRY       DS   F     Message lifetime
MQMD_FEEDBACK     DS   F     Feedback or reason code
MQMD_ENCODING     DS   F     Numeric encoding of message data
MQMD_CODEDCHARSETID DS   F   Character set identifier of message
*                data
MQMD_FORMAT       DS   CL8   Format name of message data
MQMD_PRIORITY     DS   F     Message priority
MQMD_PERSISTENCE  DS   F     Message persistence
MQMD_MSGID        DS   XL24  Message identifier
MQMD_CORRELID     DS   XL24  Correlation identifier
MQMD_BACKOUTCOUNT DS   F     Backout counter
MQMD_REPLYTOQ     DS   CL48  Name of reply queue
MQMD_REPLYTOQMGR  DS   CL48  Name of reply queue manager
MQMD_USERIDENTIFIER DS   CL12 User identifier
MQMD_ACCOUNTINGTOKEN DS   XL32 Accounting token
MQMD_APPLIDENTITYDATA DS   CL32 Application data relating to identity
MQMD_PUTAPPLTYPE  DS   F     Type of application that put the
*                message
MQMD_PUTAPPLNAME  DS   CL28  Name of application that put the
*                message
MQMD_PUTDATE      DS   CL8   Date when message was put
MQMD_PUTTIME      DS   CL8   Time when message was put
MQMD_APPLORIGINDATA DS   CL4  Application data relating to origin
MQMD_GROUPID      DS   XL24  Group identifier
MQMD_MSGSEQNUMBER DS   F     Sequence number of logical message
*                within group
MQMD_OFFSET       DS   F     Offset of data in physical message
*                from start of logical message
MQMD_MSGFLAGS     DS   F     Message flags
MQMD_ORIGINALLENGTH DS   F   Length of original message
*
MQMD_LENGTH       EQU   *-MQMD
                  ORG   MQMD
MQMD_AREA         DS   CL(MQMD_LENGTH)

```

## MQMD için Visual Basic bildirimi

```
Type MQMD
  StrucId      As String*4  'Structure identifier'
  Version     As Long      'Structure version number'
  Report      As Long      'Options for report messages'
  MsgType     As Long      'Message type'
  Expiry      As Long      'Message lifetime'
  Feedback    As Long      'Feedback or reason code'
  Encoding    As Long      'Numeric encoding of message data'
  CodedCharSetId As Long      'Character set identifier of message'
  'data'
  Format      As String*8  'Format name of message data'
  Priority    As Long      'Message priority'
  Persistence As Long      'Message persistence'
  MsgId      As MQBYTE24  'Message identifier'
  CorrelId   As MQBYTE24  'Correlation identifier'
  BackoutCount As Long      'Backout counter'
  ReplyToQ   As String*48  'Name of reply queue'
  ReplyToQMgr As String*48  'Name of reply queue manager'
  UserIdentifier As String*12 'User identifier'
  AccountingToken As MQBYTE32 'Accounting token'
  ApplIdentityData As String*32 'Application data relating to identity'
  PutApplType As Long      'Type of application that put the'
  'message'
  PutApplName As String*28 'Name of application that put the'
  'message'
  PutDate     As String*8  'Date when message was put'
  PutTime     As String*8  'Time when message was put'
  ApplOriginData As String*4  'Application data relating to origin'
  GroupId     As MQBYTE24  'Group identifier'
  MsgSeqNumber As Long      'Sequence number of logical message'
  'within group'
  Offset      As Long      'Offset of data in physical message'
  'from start of logical message'
  MsgFlags    As Long      'Message flags'
  OriginalLength As Long      'Length of original message'
End Type
```

### **StrucId (MQCHAR4)**

Bu, yapı tanıttıcısıdır ve aşağıdaki olmalıdır:

#### **MQMD\_STRUC\_ID**

İleti tanımlayıcı yapısına ilişkin tanıttıcı.

C programlama dili için, MQMD\_STRUC\_ID\_ARRAY değişmezi de tanımlanır; bu, MQMD\_STRUC\_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisidir.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQMD\_STRUC\_ID ' dir.

### **Sürüm (MQUZE)**

Bu, yapı sürüm numarasıdır ve aşağıdakilerden biri olmalıdır:

#### **MQMD\_VERSION\_1**

Version-1 ileti tanımlayıcı yapısı.

Bu sürüm, tüm ortamlarda desteklenir.

#### **MQMD\_VERSION\_2**

Version-2 ileti tanımlayıcı yapısı.

Bu sürüm, tüm IBM MQ V6.0 ortamlarında ve daha sonraki ortamlarda ve bu sistemlere bağlı olarak IBM MQ MQI clients ile desteklenir.

**Not:** Bir version-2 MQMD (MQMD) kullanıldığında, kuyruk yöneticisi, uygulama iletisi verilerinin başında var olabilecek herhangi bir MQ üstbilgi yapısıyla ilgili ek denetimler gerçekleştirir; daha fazla ayrıntı için, MQPUT çağrısına ilişkin kullanım notlarına bakın.

Yalnızca yapının daha yeni sürümünde var olan alanlar, alanların açıklamalarında olduğu gibi tanımlanır. Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

## **MQMD\_CURRENT\_VERSION**

İleti tanımlayıcı yapısının yürürlükteki sürümü.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQMD\_VERSION\_1' dir.

## **Rapor (MQLONG)**

Rapor iletisi, bir uygulamayı, özgün iletiyle ilgili beklenen ya da beklenmeyen olaylarla ilgili bilgilendirmek için kullanılan başka bir iletiyle ilgili bir iletidir. *Report* alanı, uygulamanın, hangi rapor iletilerinin gerekli olduğunu, uygulama iletisi verilerinin bunlara dahil edilip edilmeyeceğini ve ayrıca (hem raporlar hem de yanıtlar için) ileti ya da yanıt iletisinde ileti ve ilinti tanıtıcılarının nasıl ayarlanacağını belirlemek için özgün iletiyi göndermesini sağlar. Aşağıdaki tip rapor iletilerinin herhangi biri ya da tümü (ya da hiçbiri) istenebilir:

- Özel durum
- Süre Bitim Tarihi
- Varışta onayla (COA)
- Teslim edilmeyi onayla (COD)
- Pozitif işlem bildirim (PAN)
- Negatif işlem bildirim (NAN)

Bu seçeneklerden birini ya da birkaçını belirleyebilirsiniz. birden çok seçenek belirtmek için, değerleri bir araya ekleyin (aynı değişmezi bir kereden fazla eklemeyin) ya da bit ya da işlemi kullanarak değerleri birleştirin (programlama dili, bit işlemleri destekliyse).

Rapor iletisini alan uygulama, MQMD ' deki *Feedback* alanı incelenerek raporun oluşturulduğu nedeni belirleyebilir; ek ayrıntılar için *Feedback* alanına bakın.

Bir konuya ileti yerleştirilirken rapor seçeneklerinin kullanılması sıfır, bir ya da çok sayıda rapor iletisinin oluşturulmasına ve uygulamaya gönderilmesine neden olabilir. Bunun nedeni, yayın iletisinin sıfır, bir ya da daha fazla abone olma uygulamasına gönderilebileceği içindir.

**Kural dışı durum seçenekleri:** Bir kural dışı durum raporu iletisi istemek için listelenen seçeneklerden birini belirtin.

## **MQRO\_EXCEPTION**

Başka bir kuyruk yöneticisine bir ileti gönderildiğinde ve ileti belirtilen hedef kuyruğa teslim edilemediğinde, bir ileti kanalı aracısı bu rapor tipini oluşturur. Örneğin, hedef kuyruk ya da ara iletim kuyruğu dolu olabilir ya da ileti kuyruk için çok büyük olabilir.

Kural dışı durum raporu iletisinin oluşturulması, özgün iletinin kalıcılarına ve özgün iletinin seyahat ettiği ileti kanalının (olağan ya da hızlı) hıza bağlıdır.

- Tüm kalıcı iletiler ve normal ileti kanalları üzerinden seyahat eden kalıcı olmayan iletiler için, kural dışı durum raporu yalnızca, hata koşuluna ilişkin gönderme uygulaması tarafından belirtilen işlemin başarıyla tamamlanabilmesi için üretilir. Gönderme uygulaması, hata koşulu ortaya çıkan özgün iletinin durumunu denetlemek için aşağıdaki işlemlerden birini belirtebilir:
  - MQRO\_DEAD\_LETTER\_Q (bu, özgün iletiyi ölü harf kuyruğunda yerleştirir).
  - MQRO\_DISCARD\_MSG (bu ileti özgün iletiyi atar).

Gönderme uygulaması tarafından belirtilen işlem başarıyla tamamlanamazsa, özgün ileti iletim kuyruğunda bırakılır ve kural dışı durum raporu iletisi oluşturulamaz.

- Hızlı ileti kanalları üzerinden seyahat eden kalıcı olmayan iletiler için, özgün ileti iletim kuyruğundan kaldırılır ve kural dışı durum raporu üretildi *sa bile* hata koşulu için belirtilen işlem başarıyla tamamlanamaz. Örneğin, MQRO\_DEAD\_LETTER\_Q belirtilirse, ancak kuyruk dolu olduğu için özgün ileti, ölü-harfli kuyruğuna yerleştirilemez; kural dışı durum raporu iletisi oluşturulur ve özgün ileti atılır.

Normal ve hızlı ileti kanallarına ilişkin daha fazla bilgi için bkz. [Kalıcı olmayan ileti hızı \(NPMSPEED\)](#).

Özgün iletiyi koyan uygulama, sorunun zamanuyumlu olarak MQPUT ya da MQPUT1 çağrısı tarafından döndürülmesi yoluyla sorunun bildirilebileceğini bildiren bir kural dışı durum raporu oluşturulmaz.

Uygulamalar, bir iletinin işlenemeyeceğini belirtmek için (örneğin, hesabın kredi limitini aşmasına neden olacak bir borç işlemi olduğu için) özel durum raporları gönderebilir.

Özgün iletiden ileti verileri, rapor iletisine dahil edilmiyor.

Birden fazla MQRO\_EXCEPTION, MQRO\_EXCEPTION\_WITH\_DATA ve MQRO\_EXCEPTION\_WITH\_FULL\_DATA değerini belirtmeyin.

#### **MQRO\_EXCEPTION\_WITH\_DATA**

Bu, özgün iletiden gelen uygulama iletisi verilerinin ilk 100 baytı rapor iletisine dahil etmek dışında, MQRO\_EXCEPTION ile aynıdır. Özgün ileti bir ya da daha çok MQ üstbilgi yapıları içeriyorsa, bunlar rapor iletisine dahil edilir ve 100 baytlık uygulama verileri eklenir.

Birden fazla MQRO\_EXCEPTION, MQRO\_EXCEPTION\_WITH\_DATA ve MQRO\_EXCEPTION\_WITH\_FULL\_DATA değerini belirtmeyin.

#### **MQRO\_EXCEPTION\_WITH\_FULL\_DATA**

Tam veri gerekli olan kural dışı durum raporları.

Bu, özgün iletiden gelen tüm uygulama iletisi verilerinin rapor iletisine dahil olması dışında, MQRO\_EXCEPTION ile aynıdır.

Birden fazla MQRO\_EXCEPTION, MQRO\_EXCEPTION\_WITH\_DATA ve MQRO\_EXCEPTION\_WITH\_FULL\_DATA değerini belirtmeyin.

**Süre bitim seçenekleri:** Bir süre bitimi rapor iletisi istemek için listelenen seçeneklerden birini belirleyin.

#### **MQRO\_EXPIRATION**

This type of report is generated by the queue manager if the message is discarded before delivery to an application because its expiry time has passed (see the *Expiry* field). Bu seçenek belirlenmezse, bu nedenden dolayı bir ileti atılırsa (MQRO\_EXCEPTION\_ \* seçeneklerinden birini belirtseniz bile) herhangi bir rapor iletisi oluşturulmaz.

Özgün iletiden ileti verileri, rapor iletisine dahil edilmiyor.

Birden fazla MQRO\_BITIM, MQRO\_EXPIRATION\_WITH\_DATA ve MQRO\_EXPIRATION\_WITH\_FULL\_DATA belirtmeyin.

#### **MQRO\_EXPIRATION\_WITH\_DATA**

Bu, özgün iletiden gelen uygulama iletisi verilerinin ilk 100 baytı rapor iletisine dahil olması dışında, MQRO\_EXPIRATION ile aynıdır. Özgün ileti bir ya da daha çok MQ üstbilgi yapıları içeriyorsa, bunlar rapor iletisine dahil edilir ve 100 baytlık uygulama verileri eklenir.

Birden fazla MQRO\_BITIM, MQRO\_EXPIRATION\_WITH\_DATA ve MQRO\_EXPIRATION\_WITH\_FULL\_DATA belirtmeyin.

#### **MQRO\_EXPIRATION\_WITH\_FULL\_DATA**

Bu, özgün iletiden gelen tüm uygulama iletisi verilerinin rapor iletisinde yer aldığından, MQRO\_EXPIRATION ile aynıdır.

Birden fazla MQRO\_BITIM, MQRO\_EXPIRATION\_WITH\_DATA ve MQRO\_EXPIRATION\_WITH\_FULL\_DATA belirtmeyin.

**Confirm-on-invrival options**(Gelir-geliş seçeneklerini doğrula): Gelir-geliş raporu iletisi istemek için listelenen seçeneklerden birini belirleyin.

#### **MQRO\_COA**

Bu tip bir rapor, hedef kuyruğa ileti yerleştirildiğinde hedef kuyruğa sahip olan kuyruk yöneticisi tarafından oluşturulur. Özgün iletiden ileti verileri, rapor iletisine dahil edilmiyor.

İleti bir iş biriminin bir parçası olarak konursa ve hedef kuyruk yerel bir kuyruksa, kuyruk yöneticisi tarafından oluşturulan COA rapor iletisi, yalnızca iş birimi kesinleştirildiyse alınabilir.

İleti tanımlayıcısındaki *Format* alanı MQFMT\_XMIT\_Q\_HEADER ya da MQFMT\_DEAD\_LETTER\_HEADER ise, COA raporu oluşturulmaz. Bu işlem, ileti bir iletim kuyruğuna konursa ya da teslim edilemediyse ve bir ölü-mektup kuyruğuna konursa COA raporunun oluşturulmasını önler.

Bir IMS köprü kuyruğu olması durumunda, ileti IMS kuyruğuna ulaştığında ( IMS ' tan alınan alındı bildirimi) COA raporu oluşturulur. ve ileti MQ köprü kuyruğuna konduğunda değil. Bu, IMS etkin değilse, IMS başlatılıncaya kadar COA raporu oluşturulmaz ve IMS kuyruğunda bir ileti kuyruğa alınır.

MQMD.Report= MQRO\_COA ' nın yanıt kuyruğunda + passid yetkisi olması gerekir. Kullanıcı + passid yetkisine sahip değilse, COA rapor iletisi yanıt kuyruğuna ulaşmaz. Rapor iletisini ölü mektup kuyruğuna koymak için bir girişimde bulunuldu.

MQRO\_COA, MQRO\_COA\_WITH\_DATA ve MQRO\_COA\_WITH\_FULL\_DATA dışında birden çok MQRO\_COA belirtmeyin.

### **MQRO\_CO\_WITH\_DATA**

Bu, MQRO\_COA ile aynıdır, ancak özgün iletiden gelen uygulama iletisi verilerinin ilk 100 baytı rapor iletisine dahil edilir. Özgün ileti bir ya da daha çok MQ üstbilgi yapıları içeriyorsa, bunlar rapor iletisine dahil edilir ve 100 baytlık uygulama verileri eklenir.

MQRO\_COA, MQRO\_COA\_WITH\_DATA ve MQRO\_COA\_WITH\_FULL\_DATA dışında birden çok MQRO\_COA belirtmeyin.

### **MQRO\_COA\_WITHL\_FULL\_DATA**

Bu, özgün iletiden gelen tüm uygulama iletisi verilerinin rapor iletisine dahil olması dışında, MQRO\_COA ile aynıdır.

MQRO\_COA, MQRO\_COA\_WITH\_DATA ve MQRO\_COA\_WITH\_FULL\_DATA dışında birden çok MQRO\_COA belirtmeyin.

**Confirm-on-delivery options**(Doğrulama-Teslim seçenekleri): Bir teslim onayı rapor iletisi istemek için listelenen seçeneklerden birini belirleyin.

### **MQRO\_COD**

Bu tip bir rapor, kuyruk yöneticisi tarafından bir uygulama hedef kuyruktan iletiyi, kuyruktan iletiyi silen bir yöntemle aldıklarında oluşturulur. Özgün iletiden ileti verileri, rapor iletisine dahil edilmiyor.

İleti bir iş biriminin bir parçası olarak alınırsa, rapor iletisi aynı iş birimi içinde oluşturulur; böylece, iş birimi kesinleştirilinceye kadar rapor kullanılabilir değildir. İş birimi yedeklendiyse, rapor gönderilmez.

MQGMO\_MARK\_SKIP\_BACKUT seçeneğiyle bir ileti alınırsa, COD raporu her zaman oluşturulmaz. Birincil iş birimi yedeklenirse, ancak ikincil iş birimi kesinleştirilirse, ileti kuyruktan kaldırılır, ancak COD raporu oluşturulmaz.

İleti tanımlayıcısındaki *Format* alanı MQFMT\_DEAD\_LETTER\_HEADER ise bir COD raporu oluşturulmaz. Bu işlem, iletinin teslim edilememesi ve kuyruksuz bir kuyruğa konması durumunda COD raporunun oluşturulmasını önler.

Hedef kuyruk bir XCF kuyruğysa MQRO\_COD geçerli değildir.

MQRO\_COD, MQRO\_COD\_WITH\_DATA ve MQRO\_COD\_WITH\_FULL\_DATA için birden çok MQRO\_COD belirtmeyin.

### **MQRO\_COD\_WITH\_DATA**

Bu, MQRO\_COD ile aynıdır, ancak özgün iletiden gelen uygulama iletisi verilerinin ilk 100 baytı rapor iletisine dahil edilir. Özgün ileti bir ya da daha çok MQ üstbilgi yapıları içeriyorsa, bunlar rapor iletisine dahil edilir ve 100 baytlık uygulama verileri eklenir.

Özgün ileti için MQGET çağrısında MQGMO\_ACCEPT\_TRUNCATED\_MSG belirtilirse ve alınan ileti kısaltılırsa, rapor iletisine yerleştirilen uygulama iletisi verilerinin miktarı ortama bağlıdır:

- z/OS üzerinde, aşağıdakiler en az:
  - Özgün iletinin uzunluğu
  - İletiyi almak için kullanılan arabelleğin uzunluğu

- 100 bayt.
- Diğer ortamlarda, aşağıdakiler en az aşağıdakilerden biri olabilir:
  - Özgün iletinin uzunluğu
  - 100 bayt.

Hedef kuyruk bir XCF kuyruğysa, MQRO\_COD\_WITH\_DATA geçerli değildir.

MQRO\_COD, MQRO\_COD\_WITH\_DATA ve MQRO\_COD\_WITH\_FULL\_DATA için birden çok MQRO\_COD belirtmeyin.

### **MQRO\_COD\_WITH\_FULL\_DATA**

Bu, özgün iletiden gelen tüm uygulama iletisi verilerinin rapor iletisine dahil olması dışında, MQRO\_COD ile aynıdır.

Hedef kuyruk bir XCF kuyruğysa, MQRO\_COD\_WITH\_FULL\_DATA geçerli değildir.

MQRO\_COD, MQRO\_COD\_WITH\_DATA ve MQRO\_COD\_WITH\_FULL\_DATA için birden çok MQRO\_COD belirtmeyin.

**İşlem-bildirim seçenekleri:** Alma uygulamasının olumlu ya da olumsuz işlem rapor iletisi göndermesini istemek için listelenen seçeneklerden birini ya da her ikisini belirtin.

### **MQRO\_PAN**

Bu rapor tipi, iletii alan ve üzerinde işlem yapan uygulama tarafından oluşturulur. İletide istenen işlemin başarılı bir şekilde gerçekleştirildiğini gösterir. Raporu oluşturan uygulama, raporla birlikte herhangi bir verinin dahil edilip edilmeyeceğini belirler.

İletii alan uygulamaya bu isteği iletmekten başka bir seçenek dışında, kuyruk yöneticisi bu seçeneğe dayalı olarak herhangi bir işlem gerekmez. Uygun olduğunda, alma uygulamasının raporu oluşturması gerekir.

### **MQRO\_NAN**

Bu rapor tipi, iletii alan ve üzerinde işlem yapan uygulama tarafından oluşturulur. İletide istenen işlemin başarılı bir şekilde gerçekleştirilmediğini gösterir. Raporu oluşturan uygulama, raporla birlikte herhangi bir verinin dahil edilip edilmeyeceğini belirler. Örneğin, isteğin neden gerçekleştirilemediğini belirtmek için bazı verileri dahil etmek isteyebilirsiniz.

İletii alan uygulamaya bu isteği iletmekten başka bir seçenek dışında, kuyruk yöneticisi bu seçeneğe dayalı olarak herhangi bir işlem gerekmez. Uygun olduğunda, alma uygulamasının raporu oluşturması gerekir.

Uygulama, hangi koşulların olumlu bir eyleme karşılık geldiğini ve hangilerinin olumsuz bir işleme karşılık geldiğini belirlemelidir. Ancak, istek kısmen gerçekleştirildiyse, istenirse PAN raporu yerine bir NAN raporu oluşturun. Olası her koşul, olumlu bir işleme ya da olumsuz bir işleme karşılık gelmelidir, ancak her ikisi birden olmaz.

**İleti tanıtıcısı seçenekleri:** Rapor iletisinin (ya da yanıt iletisinin) *MsgId* 'inin nasıl ayarlanacağını denetlemek için listelenen seçeneklerden birini belirleyin.

### **MQRO\_NEW\_MSG\_ID**

Bu, varsayılan işlemdir ve bir rapor ya da yanıt bu iletinin sonucu olarak oluşturulursa, rapor ya da yanıt iletisi için yeni bir *MsgId* oluşturulduğunu belirtir.

### **MQRO\_PASS\_MSG\_ID**

Bir rapor ya da yanıt bu iletinin bir sonucu olarak oluşturulduysa, bu iletinin *MsgId* 'i rapor ya da yanıt iletisinin *MsgId* ' e kopyalanır.

Bir yayın iletisinin *MsgId* , yayının bir kopyasını alan her abone için farklı olur; bu nedenle, rapor ya da yanıt iletisine kopyalanan *MsgId* her bir abone için farklı olur.

Bu seçenek belirlenmezse, MQRO\_NEW\_MSG\_ID varsayılan değer olarak kabul edilir.

**İlinti tanıtıcısı seçenekleri:** Rapor iletisinin (ya da yanıt iletisinin) *CorrelId* 'inin nasıl ayarlanacağını denetlemek için listelenen seçeneklerden birini belirleyin.

### **MQRO\_COPY\_MSG\_ID\_TO\_CORREL\_ID**

Bu, varsayılan işlemdir ve bir rapor ya da yanıt bu iletinin sonucu olarak oluşturulursa, bu iletinin *MsgId* 'inin rapor ya da yanıt iletisinin *CorrelId* ' e kopyalandığını belirtir.

The *MsgId* of a publication message will be different for each subscriber that receives a copy of the publication and therefore the *MsgId* copied into the *CorrelId* of the report or reply message will be different for each one.

### **MQRO\_PASS\_COREL\_ID**

Bir rapor ya da yanıt bu iletinin bir sonucu olarak oluşturulduysa, bu iletinin *CorrelId* 'i rapor ya da yanıt iletisinin *CorrelId* ' e kopyalanır.

Bir yayın iletisinin *CorrelId* , MQSO\_SET\_CORIL\_ID seçeneğini kullanmadığı ve MQSD ' deki SubCorrelId alanını MQCI\_NONE olarak ayarlamadıkça bir aboneye özgü olur. Bu nedenle, rapor ya da yanıt iletisinin *CorrelId* 'ine kopyalanan *CorrelId* ' nin her biri için farklı olması mümkündür.

Bu seçenek belirlenmezse, MQRO\_COPY\_MSG\_ID\_TO\_CORREL\_ID varsayılır.

Özgün iletide MQRO\_PASS\_MSG\_ID ya da MQRO\_PASS\_COREL\_ID seçeneklerinin belirlenip belirlenmediğini denetlemek ya da rapor iletileri oluşturmak için yanıt veren sunucular denetlenmelidir. Bu durumda, sunucular, bu seçenekler için açıklanan işlemi kabul etmelidir. İkisi de ayarlanmazsa, sunucuların karşılık gelen varsayılan eylemi alması gerekir.

**Yok etme seçenekleri:** Hedef kuyruğa teslim edilemediğinde özgün iletinin yok edilmesini denetlemek için listelenen seçeneklerden birini belirtin. Uygulama, yok etme seçeneklerini, kural dışı durum raporlarından bağımsız olarak ayarlayabilir.

### **MQRO\_DEAD\_LETTER\_Q**

Bu, varsayılan işlemdir ve ileti hedef kuyruğa teslim edilemezse, iletiyi ölü-mektup kuyruğuna yerleştirir. Bu, aşağıdaki durumlarda gerçekleşir:

- Özgün iletiyi koyan uygulama, MQPUT ya da MQPUT1 çağrısının döndürdüğü neden kodu yoluyla sorunun zamanuyumlu olarak bildirilememesini sağlar. Gönderen tarafından istendiyse, bir kural dışı durum raporu iletisi üretilir.
- Özgün iletiyi koyan uygulama bir konuya yerleştirildi

### **MQRO\_DISCARD\_MSG**

Bu işlem, hedef kuyruğa teslim edilemiyorsa, bu iletiyi atar. Bu, aşağıdaki durumlarda gerçekleşir:

- Özgün iletiyi koyan uygulama, MQPUT ya da MQPUT1 çağrısının döndürdüğü neden kodu yoluyla sorunun zamanuyumlu olarak bildirilememesini sağlar. Gönderen tarafından istendiyse, bir kural dışı durum raporu iletisi üretilir.
- Özgün iletiyi koyan uygulama bir konuya yerleştirildi

Özgün iletiyi gönderene geri vermek istiyorsanız, özgün ileti çıkış kuyruğu kuyruğuna yerleştirilirse, gönderen MQRO\_DISCARD\_MSG ile MQRO\_EXCEPTION\_WITH\_FULL\_DATA değerini belirtmelidir.

### **MQRO\_PASS\_DISCARD\_AND\_IFADESI**

Bu seçenek bir iletiyle ayarlanırsa ve bir rapor ya da yanıt bunun nedeniyle oluşturulduysa, raporun ileti açıklayıcısı devralır:

- MQRO\_DISCARD\_MSG ayarlandıysa.
- İletinin kalan süre bitimi (bu süre bitimi raporu değilse). Bu süre bitimi raporu ise, süre sonu süresi 60 saniye olarak ayarlanır.

### **Etkinlik seçeneği**

#### **MQRO\_ACTIVITY**

Bu değer kullanılması, **any** iletisinin bir kuyruk yöneticisi ağı boyunca izlenmesini sağlar. Rapor seçeneği herhangi bir geçerli kullanıcı iletisine belirtilebilir ve iletinin rotasını ağ üzerinden hesaplamaya başlamanıza olanak tanır.

İletiyi oluşturan uygulama, etkinlik raporu oluşturmayı etkinleştiremezse, raporlama, kuyruk yöneticisi denetimcileri tarafından sağlanan bir API geçiş çıkışı kullanılarak etkinleştirilebilir.

#### **Not:**

1. Ağ üzerinde etkinlik raporları oluşturabilen kuyruk yöneticisi sayısı ne kadar azsa, rotayı daha az ayrıntılandırır.
2. Etkinlik raporları, alınan rotayı belirlemek için doğru sırada yer almak zor olabilir.
3. Etkinlik raporları, istenen varış noktalarına giden bir rota bulamayabilir.
4. Bu rapor seçeneği ayarına sahip iletiler, herhangi bir kuyruk yöneticisi tarafından kabul edilmesi gerekir, ancak bu seçenek bu seçeneği anlamasa bile. Bu, rapor seçeneğinin IBM WebSphere MQ 6.0 dışı ya da daha sonraki bir kuyruk yöneticisi tarafından işlenmiş olsa bile, herhangi bir kullanıcı iletilerine ayarlanmasını sağlar.
5. Bir süreç, bir kuyruk yöneticisi ya da kullanıcı süreci, bu seçenek kümesi olan bir iletilerde bir etkinlik gerçekleştirirse, bir etkinlik raporu oluşturmayı ve bu raporu yerleştirmeyi seçebilir.

**Varsayılan seçenek:** Rapor seçeneği gerekmiyorsa, aşağıdakileri belirtin:

#### **MQRO\_NONE**

Başka bir seçenek belirtilmemeyi belirtmek için bu değeri kullanın. MQRO\_NONE, program belgelerine yardımcı olmak için tanımlıdır. Bu seçeneğin başka biriyle birlikte kullanılması amaçlanmaz, ancak değeri sıfır olduğu için bu tür bir kullanım saptanamaz.

#### **Genel bilgiler:**

1. Gerekli tüm rapor tiplerinin, özgün iletiyi gönderen uygulama tarafından özel olarak istenmesi gerekir. Örneğin, bir COA raporu istenirse, ancak bir kural dışı durum raporu yoksa, ileti hedef kuyruğa yerleştirildiğinde COA raporu oluşturulur, ancak ileti oraya ulaştığında hedef kuyruk dolduysa herhangi bir kural dışı durum raporu oluşturulmaz. Hiçbir *Report* seçeneği ayarlanmazsa, kuyruk yöneticisi ya da ileti kanalı aracı (MCA) tarafından hiçbir rapor iletileri oluşturulmaz.

Bazı rapor seçenekleri, yerel kuyruk yöneticisi bunları tanımamış olsa da belirtilebilir; bu seçenek, seçenek *hedef* kuyruk yöneticisi tarafından işlenmek üzere işlendiğinde yararlı olur. Daha ayrıntılı bilgi için bkz. "[Rapor seçenekleri ve ileti işaretleri](#)" sayfa 887 .

Bir rapor iletileri istenirse, raporun gönderileceği kuyruğun adı, *ReplyToQ* alanında belirtilmeli. Bir rapor iletileri alındığında, raporun niteliği, ileti tanımlayıcısındaki *Feedback* alanı incelenerek saptanabilir.

2. Bir rapor iletileri oluşturan kuyruk yöneticisi ya da MCA, rapor iletilerini yanıt kuyruğuna koyamıyorsa (örneğin, yanıt kuyruğu ya da iletiler kuyruğu dolu olduğu için), rapor iletileri ölü harf kuyruğunda yerine yerleştirilir. *Aynı zamanda* da başarısız olursa ya da herhangi bir ölü harf kuyruğu yoksa, işlem, rapor iletilerinin tipine bağlıdır:

- Rapor iletileri bir kural dışı durum bildirimiyse, kural dışı durum raporunu oluşturan ileti iletiler kuyruğunda bırakılır; bu, iletilerin kaybolmamasını sağlar.
- Diğer tüm rapor tipleri için, rapor iletileri atılır ve işleme olağan şekilde devam eder. Bu, özgün ileti güvenli bir şekilde (COA ya da COD rapor iletileri için) güvenli bir şekilde teslim edildiğinden ya da artık herhangi bir ilgi göstermediği için yapılır (bir süre sonu rapor iletileri için).

Bir ileti kuyruğunda (hedef kuyruk ya da ara iletiler kuyruğu) başarıyla yerleştirildikten sonra, ileti artık özel işleme tabi değildir; diğer iletiler gibi işlem görür.

3. Rapor oluşturulduğunda, *ReplyToQ* kuyruğu açılır ve aşağıdaki durumlar dışında, rapora neden olan iletilerin MQMD ' de *UserIdentifier* yetki yetkisini kullanarak rapor iletileri açılır:
  - Alıcı bir MCA tarafından oluşturulan kural dışı durum raporları, rapora neden olan iletiyi yerleştirmeye çalışıldığında, MCA ' nın kullandığı yetki ile birlikte ortaya çıkarılır.
  - Kuyruk yöneticisi tarafından oluşturulan COA raporları, rapora neden olan ileti, raporu oluşturan kuyruk yöneticisi üzerine konulduğunda kullanılan herhangi bir yetki ile konabilir. Örneğin, ileti MCA 'nın kullanıcı tanımlayıcısı kullanılarak bir alıcı MCA tarafından konulduysa, kuyruk yöneticisi, MCA' nın kullanıcı kimliğini kullanarak COA raporunu koyar.

Rapor oluşturan uygulamaların, yanıt oluşturmak için kullandıkları yetkiyi kullanmaları gerekir; bu genellikle özgün iletilerde kullanıcı kimliğinin yetkidir.

Eğer rapor uzak bir hedefe gitmek zorunda ise, gönderenler ve alıcılar diğer mesajlarla aynı şekilde bunu kabul edip etmeyeceğine karar verebilirler.



#### 4. Veri içeren bir rapor iletisi istenirse:

- Rapor iletisi her zaman, özgün iletinin göndereni tarafından istenen veri miktarlarıyla oluşturulur. Rapor iletisi yanıt kuyruğu için çok büyükse, yukarıda açıklanan işleme gerçekleşir; rapor iletisi hiçbir zaman yanıt kuyruğuna sığmayacak şekilde kısaltılır.
- Özgün iletinin *Format* 'si MQFMT\_XMIT\_Q\_HEADER ise, raporda yer alan veriler MQXQH' yi içermez. Rapor verileri, özgün iletide MQXQH ' nin ötesindeki ilk veri byte 'ını kullanarak başlar. Bu durum, kuyruğun iletim kuyruğu olup olmadığını ortaya çıkar.

#### 5. Yanıt kuyruğunda bir COA, COD ya da süre bitimi raporu alındıysa, özgün iletinin teslim edildiği, teslim edildiği ya da gerektiği gibi süresi dolduğu garanti edilir. Ancak, bu rapor iletilerinden biri ya da daha fazlası istenirse ve alınmazsa, aşağıdakilerden biri gerçekleşmiş olabileceği için ters varsayılmaz.

- a. Bir bağlantı kapalı olduğu için rapor iletisi tutuluyor.
- b. Rapor iletisi, bir ara iletim kuyruğunda ya da yanıt kuyruğunda bir engelleme koşulu var olduğundan (örneğin, kuyruğun tam ya da engellenmiş olduğu bir kuyruk) olduğu için, bu ileti iletisi tutuluyor.
- c. Rapor iletisi, bir ölü-mektup kuyruğunda.
- d. Kuyruk yöneticisi, rapor iletisini oluşturmaya çalışırken, iletiyi uygun kuyruğa ya da ölü mektup kuyruğuna yerleştiremedi; böylece, rapor iletisi oluşturulamadı.
- e. Raporlanan işlem (geliş, teslim ya da süre bitimi) ve karşılık gelen rapor iletisinin oluşturulması arasında kuyruk yöneticiliğinin bir hatası oluştu. (Uygulama bir iş birimi içinde özgün iletiyi aldıysa, COD rapor iletisi aynı iş birimi içinde oluşturulduğunda COD rapor iletileri için bu gerçekleşmez.)

Kural dışı durum raporu iletileri, yukarıdaki 1, 2 ve 3. nedenlerle aynı şekilde tutulabilir. Ancak, bir MCA bir özel durum raporu iletisi oluşturamadığında (rapor iletisi yanıt kuyruğuna ya da ölü-mektup kuyruğuna konulamaz), özgün ileti göndericindeki iletim kuyruğunda kalır ve kanal kapatılır. Bu durum, rapor iletisinin, kanalın gönderme ya da alma sırasında oluşturulup oluşturulmayacağı dikkate alınmalıdır.

#### 6. Özgün ileti geçici olarak engellendiyse (oluşturulmakta olan bir kural dışı durum raporu iletisi ve özgün ileti bir ölü-mektup kuyruğuna konursa), ancak tıkanıklık temizlenir ve bir uygulama, özgün iletiyi ölü harf kuyruğundan okur ve bu iletiyi yeniden hedefine yerleştirir, aşağıdaki durumlar oluşabilir:

- Bir kural dışı durum raporu iletisi oluşturulsa da, özgün ileti sonunda hedefine başarıyla ulaşır.
- Özgün ileti başka bir engelleme daha sonra karşılaşılabileceğinden, tek bir özgün iletiye göre birden çok kural dışı durum raporu iletisi oluşturulur.

#### **Konu bir konuya yerleştirilirken iletileri raporla:**

1. Bir konuya ileti yerleştirilirken raporlar oluşturulabilir. Bu ileti tüm abonelere, sıfır, bir ya da daha fazla olan konuya gönderilecektir. Sonuç olarak birçok rapor iletisi oluşturulabilirken, rapor seçeneklerini kullanma seçilirken bu işlem dikkate alınmalıdır.
2. Bir konuya ileti yerleştirilirken, iletinin bir kopyasını verilecek birçok hedef kuyrukları olabilir. Bu hedef kuyruklardan bazılarının kuyruk dolu gibi bir sorunu varsa, MQPUT ' un başarıyla tamamlanması, NPMSGDLV ya da PMSGDLV (iletinin kalıcısına bağlı olarak) ayarına bağlıdır. Ayar, hedef kuyruğa ileti tesliminin başarılı olması gerekiyorsa (örneğin, kalıcı bir aboneye kalıcı bir iletidir ve PMSGDLV ALL ya da ALLDIR olarak ayarlanır), başarı aşağıdaki ölçütlerden biri olarak tanımlıdır:

- Abone kuyruğuna başarılı bir şekilde yerleştirilecek
- Abone kuyruğu iletiyi alamıyorsa, MQRO\_DEAD\_LETTER\_Q ve Başarılı bir ileti kuyruğuna başarılı bir şekilde konun.
- Abone kuyruğu iletiyi alamıyorsa MQRO\_DISCARD\_MSG ' nin kullanılması.

#### **İleti bölümleri için rapor iletileri:**

1. İzin verilen bölümlenmeye sahip iletiler için rapor iletileri istenebilir (MQMF\_SEGMENTATION\_ALLOWALIZD işaretinin açıklamasına bakın). Kuyruk yöneticisi iletiyi bölümlenmek için gerekli bulursa, ilgili koşulla daha sonra karşılaştıran her bir kesim için bir rapor iletisi oluşturulabilir. İstenen her rapor iletisi tipi için birden çok rapor iletisi almak üzere uygulamalar

hazırlanmalıdır. Çoklu raporları özgün iletinin grup tanıtıcısıyla ilişkilendirmek için rapor iletisinde *GroupId* alanını kullanın ve *Feedback* alanı, her bir rapor iletisinin tipini tanımlar.

2. Kesimlere ilişkin rapor iletilerini almak için MQGMO\_LOGICAL\_ORDER kullanılıyorsa, art arda gelen MQGET çağrılarının tarafından *farklı tip* raporlarının döndürülebileceğini göz aþın. Örneğin, kuyruk yöneticisi tarafından bölümlemiş bir ileti için hem COA hem de COD raporları istenirse, rapor iletileri için MQGET çağrılarının, COA ve COD rapor iletilerinin öngörülemez bir şekilde serpiştirilmiş olarak dönmesine neden olabilir. MQGMO\_COMPLE\_MSG seçeneğini kullanarak (isteğe bağılı olarak MQGMO\_ACCEPT\_TRUNCATED\_MSG ile) bunu engelle. MQGMO\_COMPLE\_MSG, kuyruk yöneticisinin aynı rapor tipine sahip rapor iletilerini yeniden bir araya getirmesine neden olur. Örneğin, ilk MQGET çağrısı özgün iletiyle ilgili tüm COA iletilerini yeniden birleştirebilir ve ikinci MQGET çağrısı tüm COD iletilerini yeniden birleştirebilir. İlk olarak yeniden birleştirilen, kuyruğun ilk olarak hangi tip rapor iletisine geçeceği bağılıdır.
3. Kesimler yerleştiren uygulamalar, her bir bölüm için farklı rapor seçenekleri belirtebilir. Ancak, aşağıdaki noktalara dikkat edin:
  - Kesimler MQGMO\_COMPLE\_MSG seçeneği kullanılarak alınır, yalnızca *ilk* bölümündeki rapor seçenekleri kuyruk yöneticisi tarafından onurlandırılır.
  - Kesimler birer birer alındıysa ve bunların çoğu MQRO\_COD\_ \* seçeneklerinden birine sahip olur, ancak en az bir kısım yok, tek bir MQGET çağrısıyla rapor iletilerini almak için MQGMO\_COMPLE\_MSG seçeneğini kullanamazsınız ya da tüm rapor iletilerinin ne zaman geldiğini saptamak için MQGMO\_ALL\_SEGMENTS\_AVAILLLLLLLLLLK seçeneğini kullanabilirsiniz.
4. Bir MQ ağında kuyruk yöneticileri farklı yeteneklere sahip olabilir. Segmentasyonu desteklemeyen bir kuyruk yöneticisi ya da MCA tarafından bir kesime ilişkin bir rapor iletisi oluşturulursa, kuyruk yöneticisi ya da MCA varsayılan olarak rapor iletisinde gerekli bölüm bilgilerini içermiyorsa ve bu durum, raporun oluşturulmasını neden olan özgün iletinin tanınmasını zorlaştırabilir. Uygun MQRO\_ \* \_WITH\_DATA ya da MQRO\_ \* WITH\_WITH\_DATA seçeneklerini belirterek, rapor iletisiyle veri isteyerek bu zorluğu önlemekten kaçınınız. Ancak, MQRO\_ \* \_WITH\_DATA belirtilirse, rapor iletisi bir kuyruk yöneticisi tarafından oluşturulduysa ya da bölümleme desteğini desteklemeyen MCA tarafından oluşturulduysa, rapor iletisini alan uygulamaya *küçüktür* 100 bayt uygulama iletisi verisi döndürülebileceğini unutmayınız.

**Rapor iletisi için ileti tanımlayıcısının içeriği:** Kuyruk yöneticisi ya da ileti kanalı aracısı (MCA) bir rapor iletisi oluşturduğunda, ileti tanımlayıcısındaki alanları aşağıdaki değerlere ayarlar ve sonra iletiyi olağan şekilde yerleştirir.

*Çizelge 502. Bir rapor iletisi sistem tarafından üretildiğinde, MQMD alanları için kullanılan değerler*

MQMD ' de alan	Kullanılan değer
<i>StrucId</i>	MQMD_STRUC_ID
<i>Version</i>	MQMD_VERSION_2
<i>Report</i>	MQRO_NONE
<i>MsgType</i>	MQMT_REPORT
<i>Expiry</i>	MQE_UNSENSITIVE
<i>Feedback</i>	Raporun niteine uygun olarak (MQFB_COA, MQFB_COD, MQFB_EXPIRATION ya da bir MQRC_ * değeri)
<i>Encoding</i>	Özgün ileti tanımlayıcısından kopyalandı
<i>CodedCharSetId</i>	Özgün ileti tanımlayıcısından kopyalandı
<i>Format</i>	Özgün ileti tanımlayıcısından kopyalandı
<i>Priority</i>	Özgün ileti tanımlayıcısından kopyalandı
<i>Persistence</i>	Özgün ileti tanımlayıcısından kopyalandı
<i>MsgId</i>	Özgün ileti tanımlayıcısındaki rapor seçenekleri tarafından belirlendiği gibi

MQMD ' de alan	Kullanılan değer
<i>CorrelId</i>	Özgün ileti tanımlayıcısındaki rapor seçenekleri tarafından belirlendiği gibi
<i>BackoutCount</i>	0
<i>ReplyToQ</i>	Boşluklar
<i>ReplyToQMGr</i>	Kuyruk yöneticisinin adı
<i>UserIdentifier</i>	MQPMO_PASS_IDENTITY_CONTEXT seçeneği tarafından belirlendiği gibi
<i>AccountingToken</i>	MQPMO_PASS_IDENTITY_CONTEXT seçeneği tarafından belirlendiği gibi
<i>AppIdentityData</i>	MQPMO_PASS_IDENTITY_CONTEXT seçeneği tarafından belirlendiği gibi
<i>PutAppType</i>	MQAT_QMGR ya da ileti kanalı aracısının uygun olduğu şekilde
<i>PutAppName</i>	Kuyruk yöneticisi adının ya da ileti kanalı aracısı adının ilk 28 byte 'ı. For report messages generated by the IMS bridge, this field contains the XCF group name and XCF member name of the IMS system to which the message relates.
<i>PutDate</i>	Rapor iletisinin gönderildiği tarih
<i>PutTime</i>	Rapor iletisinin gönderildiği saat
<i>AppOriginData</i>	Boşluklar
<i>GroupId</i>	Özgün ileti tanımlayıcısından kopyalandı
<i>MsgSeqNumber</i>	Özgün ileti tanımlayıcısından kopyalandı
<i>Offset</i>	Özgün ileti tanımlayıcısından kopyalandı
<i>MsgFlags</i>	Özgün ileti tanımlayıcısından kopyalandı
<i>OriginalLength</i>	MQOL_UNDEFINED değilse, özgün ileti tanımlayıcısından kopyalanmış ve özgün ileti verilerinin uzunluğuna ayarlanmışsa

Aşağıdakilerin dışında, benzer değerleri ayarlamak için rapor oluşturan bir uygulama önerilir:

- *ReplyToQMGr* alanı boşluk olarak ayarlanabilir (kuyruk yöneticisi, ileti konduğunda yerel kuyruk yöneticisinin adına bunu değiştirir).
- Bir yanıt için kullanılan seçeneği kullanarak bağlam alanlarını ayarlayın; olağan durumda MQPMO\_PASS\_IDENTITY\_CONTEXT.

**Rapor alanının çözümlenmesi:** *Report* alanı alt alanlar içerir; bunun nedeni, belirli bir rapor istenen iletinin gönderenin "Rapor alanının çözümleniyor" sayfa 889' de açıklanan tekniklerden birini kullanması gerekip gerekmediğini denetlememesi gereken uygulamalar.

Bu, MQGET çağrısına ilişkin bir çıkış alanıdır ve MQPUT ve MQPUT1 çağrılarına ilişkin bir giriş alanı. Bu alanın ilk değeri MQRO\_NONE olur.

### **MsgType (MQUZE)**

Bu, iletinin tipini gösterir. İleti tipleri şu şekilde gruplanır:

#### **MQMT\_SYSTEM\_FIRST**

Sistem tarafından tanımlanan ileti tipleri için en düşük değer.

#### **MQMT\_SYSTEM\_SON**

Sistem tarafından tanımlanan ileti tipleri için en yüksek değer.

Aşağıdaki değerler sistem aralığı içinde tanımlanandır:

## **MQMT\_DATAGRAM**

İleti, yanıt gerektirmeyen bir iletidir.

## **MQMT\_REQUEST**

İleti, yanıt gerektiren bir iletidir.

*ReplyToQ* alanına, yanıtın gönderileceği kuyruğun adını belirtin. *Report* alanı, yanıtın *MsgId* ve *CorrelId* ' in nasıl ayarlanacak olduğunu gösterir.

## **MQMT\_REPLY**

İleti, daha önceki bir istek iletisine (MQMT\_REQUEST) yanıt iletilir. İleti, istek iletisinin *ReplyToQ* alanı tarafından belirtilen kuyruğa gönderilmelidir. Yanıtın *MsgId* ve *CorrelId* ' in nasıl ayarlanacak şekilde ayarlanacak şekilde denetlenmesi için isteğin *Report* alanını kullanın.

**Not:** Kuyruk yöneticisi, istek-yanıt ilişkisini uygulamaz; bu bir uygulama sorumluluğunda.

## **MQMT\_REPORT**

İleti, genellikle başka bir iletiyle ilgili (örneğin, geçerli olmayan veri içeren bir istek iletisi alındı), beklenen ya da beklenmeyen bazı oluşumlarla ilgili raporlama içerir. İletiyi, özgün iletinin ileti tanımlayıcısının *ReplyToQ* alanı tarafından belirtilen kuyruğa gönderin. Raporun niteliyi belirtmek için *Feedback* alanını ayarlayın. Rapor iletisinin *MsgId* ve *CorrelId* ' in nasıl ayarlanacak şekilde ayarının denetlenmesini denetlemek için özgün iletinin *Report* alanını kullanın.

Kuyruk yöneticisi ya da ileti kanalı aracısı tarafından oluşturulan rapor iletileri, her zaman *Feedback* ve *CorrelId* alanları yukarıda açıklandığı şekilde ayarlanmış şekilde, *ReplyToQ* kuyruğuna gönderilir.

Uygulama tanımlı değerler de kullanılabilir. Bunlar aşağıdaki aralık içinde olmalıdır:

## **MQMT\_APPL\_FIRST**

Uygulama tanımlı ileti tipleri için en düşük değer.

## **MQMT\_APPL\_SON**

Uygulama tanımlı ileti tipleri için en yüksek değer.

MQPUT ve MQPUT1 çağrılarını için, *MsgType* değerinin sistem tanımlı aralıkta ya da uygulama tanımlı aralıkta olması gerekir; değilse, çağrı neden kodu MQRC\_MSG\_TYPE\_ERROR ile başarısız olur.

Bu, MQGET çağrısına ilişkin bir çıkış alanıdır ve MQPUT ve MQPUT1 çağrılarını için bir giriş alanı. Bu alanın ilk değeri MQMT\_DATAGRAM ' tır.

## **Süre Bitimi (MQUZE)**

Bu, iletiyi yerleştiren uygulama tarafından belirlenen saniyenin onda biri olarak ifade edilen bir dönemdir. İleti, bu süre geçtikten sonra hedef kuyruktan kaldırılmamışsa, atılabilir duruma gelir.

Örneğin, süre bitimi için bir dakika ayarlamak için **MQMD**' yi ayarlamanız gerekir. **Expiry** -600.

Bu değer, iletinin hedef kuyrukte geçirdiği süreyi ve uzak bir kuyruğa konursa ara iletim kuyruklarında da yer alan süreyi yansıtabilecek şekilde belirlenir. İleti kanalı araçları, iletim sürelerini yansıtmak için de azaltılabilir; bunlar önemli olabilir. Benzer şekilde, bu iletiyi başka bir kuyruğa ileten bir uygulama, iletiyi önemli bir süre alırsa, gerekirse değeri azaltılabilir. Ancak, süre bitim süresi yaklaşık olarak kabul edilir ve değer, küçük zaman aralıklarını yansıtabilecek şekilde azaltılmamalı.

İleti, MQGET çağrısını kullanan bir uygulama tarafından alındığında, *Expiry* alanı kalan süre bitimi süresini gösterir.

Bir iletinin süre bitimi süresi dolduktan sonra, kuyruk yöneticisi tarafından atılır. Bir göz atma ya da göz atma olmayan bir MQGET çağrısı gerçekleştiğinde, iletiyi döndürecek olan bir MQGET çağrısı gerçekleştiğinde bu ileti atılır. Örneğin, MQGMO ' daki *MatchOptions* alanıyla MQMO\_NONE değerine ayarlanmış olmayan bir MQGET çağrısı, bir FIFO sıralı kuyruğundan okunurken, süresi dolan ilk iletiye kadar süresi dolan tüm iletileri atar. Aynı çağrı, öncelikli olarak sipariş edilen bir kuyruk ile, süresi dolan daha yüksek önceliğe sahip iletileri ve ilk süresi dolmadan önce kuyruğa gelen eşit önceliğe sahip iletileri atar.

Süresi dolmuş bir ileti bir uygulamaya (bir göz atma yoluyla ya da bir MQGET dışı çağrıyla) döndürülmez; bu nedenle, başarılı bir MQGET çağrısının ardından ileti tanımlayıcısının *Expiry* alanındaki değer sıfırdan büyük ya da MQE\_UNESSIRMA özel değeri.

Bir ileti uzak bir kuyruğa konursa, ileti bir ara iletim kuyruğunda olduğunda, ileti hedef kuyruğa ulaşmadan önce sona erebilir (ve atılabilir).

İleti süresi dolmuş bir ileti atıldığında, iletinin MQRO\_EXPIRATION\_\* rapor seçeneklerinden birini belirtmesi durumunda oluşturulur. Bu seçeneklerden hiçbiri belirtilmediyse, böyle bir rapor oluşturulmaz; iletinin bu zaman döneminden sonra ilgili olmadığı kabul edilir (daha sonraki bir iletinin yerine geçtiğinden olabilir).

Bir iletinin uyumluluk noktası (syncpoint) içine konması için, süre bitimi aralığı, ileti konacağı sırada başlar, uyumluluk noktasının kesinleştirildiği süre değildir. syncpoint kesinleştirilmeden önce süre bitimi aralığı geçirilebilir. Bu durumda ileti, kesinleştirme işleminden sonra bir süre sonra atılır ve MQGET işlemine yanıt olarak bir uygulamaya ileti döndürülmez.

Son kullanma zamanına dayalı iletileri atan başka bir program da istendiyse, uygun bir rapor iletisi göndermelidir.

### Notlar:

1. Bir ileti *Expiry* saati sıfır ya da 999 999 999 'dan büyük bir sayıyla konursa, MQPUT ya da MQPUT1 çağrısı MQRC\_EXPIRY\_ERROR; neden koduyla başarısız olur; bu durumda hiçbir rapor iletisi oluşturulamaz.

Neden kodu 2013 'i etkinleştirmek için, MQRC\_EXPIRY\_ERROR, AMQ\_ENFORCE\_MAX\_EXPIRY\_ERROR ortam değişkenini etkinleştirmeniz gerekir.

Aşağıda Linux için bir örnek kullanılmıştır:

```
$ export AMQ_ENFORCE_MAX_EXPIRY_ERROR=True
```

Aşağıdakileri dikkate alın:

- Önemli olan değişken ihraç etmek.
  - Gerçek değer yoksayıdır, ancak True kullanılması, kurulumu gözden geçirirken yararlı olabilir.
2. Süresi sona erme süresi olan bir ileti daha sonra atılamayacağından, bir kuyruğun süre bitimi süresini geçmiş ve bu nedenle alma için uygun olmayan iletiler atılamayabilir. Bu iletiler yine de, derinlik tetikleme de dahil olmak üzere, kuyruğun tüm amaçlar için kuyruklardaki ileti sayısına kadar sayılabilir.
- Bir abone/tüketici (istemci) bir ileti almayı denediğinde ve bu iletinin süresi dolduğunda, ileti çok eski olduğu için ileti atıldığı için istemci hiçbir şey almaz. Ayrıca, istemci herhangi bir hata iletisi almaz.
3. Bir süre bitim raporu oluşturulur (istenirse), ileti atıldığında değil, atıldığında değil olarak atılır.
  4. Süresi dolmuş bir iletinin atılması ve istenirse bir süre bitimi raporu oluşturulması, bir iş birimi içinde çalışan bir MQGET çağrısının sonucu olarak atma için zamanlanan ileti olsa bile, uygulamanın iş biriminin hiçbir zaman parçası değildir.
  5. Bir iş birimi içinde MQGET çağrısıyla yaklaşık süresi dolan bir ileti alınır ve iş birimi daha sonra gerilettiyse, ileti yeniden alınmadan önce atılabilir duruma gelebilir.
  6. Süresi dolan bir ileti MQGMO\_LOCK ile MQGET çağrısıyla kilitlendiyse, ileti MQGMO\_MSG\_UNDER\_CURSORSE ile bir MQGET çağrısıyla alınmadan önce atılabilir duruma gelebilir; neden kodu MQRC\_NO\_MSG\_UNDER\_CURSOR ile birlikte bu sonraki MQGET çağrısına döndürülmesi durumunda bu ileti atılabilir.
  7. Sıfırdan büyük bir süre bitimi içeren bir istek iletisi alındığında, uygulama yanıt iletisini gönderdiğinde aşağıdaki işlemlerden birini alabilir:
    - Kalan süre bitimini istek iletisinden yanıt iletisine kopyalayın.
    - Yanıt iletisinde süre bitimi süresini sıfırdan büyük bir açık değere ayarlayın.
    - Yanıt iletisinde süre bitimi zamanını MQE\_UNINSINLI olarak ayarlayın.

Yapılması gereken işlem, uygulamanın tasarımına bağlıdır. Ancak, iletileri bir ölü harfe (teslim edilmemiş ileti) koymak için varsayılan işlem, iletinin kalan süre bitimini korumak ve azaltmaya devam etmek için olmalıdır.

8. Tetikleme iletileri her zaman MQEL\_UNESSİSIT ile oluşturulur.
9. MQFMT\_XMIT\_Q\_HEADER *Format* adına sahip bir ileti (genellikle iletim kuyruğunda), MQXQH içinde ikinci bir ileti tanımlayıcısına sahiptir. Bu nedenle, ilişkili iki *Expiry* alanı vardır. Bu durumda aşağıdaki ek noktalar belirtilmeli:
  - Bir uygulama uzak bir kuyruğa ileti yerleştirdiğinde, kuyruk yöneticisi iletiyi başlangıçta yerel bir iletim kuyruğuna yerleştirir ve uygulama iletilerinin örneklerini MQXQH yapısıyla ekler. Kuyruk yöneticisi, iki *Expiry* alanının değerlerini, uygulama tarafından belirtilen değerle aynı olacak şekilde ayarlar.

Bir uygulama doğrudan yerel iletim kuyruğuna bir ileti yerleştirirse, ileti verileri önceden MQXQH yapısıyla başlamalı ve biçim adı MQFMT\_XMIT\_Q\_HEADER olmalıdır. Bu durumda, uygulamanın bu iki *Expiry* alanının değerlerini aynı olacak şekilde ayarlamaması gerekir. (Kuyruk yöneticisi, MQXQH içindeki *Expiry* alanının geçerli bir değer içerdiğini ve ileti verilerinin bu değeri içermek için yeterince uzun olduğunu denetler). Doğrudan iletim kuyruğuna yazabilen bir uygulama için, uygulamanın gömülü ileti tanımlayıcısıyla bir iletim kuyruğu üstbilgisi yaratması gerekir. Ancak, iletim kuyruğuna yazılan ileti tanımlayıcısındaki süre bitimi değeri, yerleşik ileti tanımlayıcısındaki değerle tutarsız olursa, süre bitimi hatası reddedilir.
  - When a message with a *Format* name of MQFMT\_XMIT\_Q\_HEADER is retrieved from a queue (whether this is a normal or a transmission queue), the queue manager decrements *her ikisi* these *Expiry* fields with the time spent waiting on the queue. İleti verileri MQXQH 'deki *Expiry* alanını içerecek kadar uzun değilse, hata ortaya çıktı.
  - Kuyruk yöneticisi, iletinin atılmasına uygun olup olmadığını test etmek için ayrı bir ileti tanımlayıcısında (MQXQH yapısı içinde gömülü olan ileti tanımlayıcısında yer alan değil) *Expiry* alanını kullanır.
  - İki *Expiry* alanının başlangıç değerleri farklıysa, ileti alındığında ayrı ileti tanımlayıcısındaki *Expiry* saati sıfırdan büyük olabilir (bu nedenle, MQXQH 'deki *Expiry* alanına göre ileti atılmasına hak kazanmaz). Bu durumda, MQXQH 'deki *Expiry* alanı sıfır olarak ayarlanır.
10. MQIIH İşaretleri alanında MQIIH\_PASS\_EXPIRATION belirlenmezse, IMS köprüsünden döndürülen bir yanıt iletileri için süre bitimi sınırsız olur. Ek bilgi için İşaretler konusuna bakın.

Şu özel değer tanıandı:

#### **MQE\_UNSSİSİZ**

İletin süre bitimi sınırsız.

Bu, MQGET çağrısına ilişkin bir çıkış alanıdır ve MQPUT ve MQPUT1 çağrılarında ilişkin bir giriş alanı. Bu alanın ilk değeri MQII\_UNESSİSİNİDİR.

*z/OS' ta süresi dolmuş iletiler*

IBM MQ for z/OS' ta, süresi dolmuş iletiler bir sonraki uygun MQGET çağrısıyla atılır.

Ancak, böyle bir çağrı gerçekleşmezse, süresi dolan ileti atılmaz ve bazı kuyruklar için çok sayıda süresi dolan ileti birikebilir. Bunu düzeltmek için, kuyruk yöneticisini kuyrukları düzenli aralıklarla tarar ve bir ya da daha çok kuyrukta süresi dolan iletileri aşağıdaki yollardan birini kullanarak atar:

#### **Periyodik tarama**

EXPYINT (süre bitimi aralığı) kuyruk yöneticisi özniteliğini kullanarak bir dönem belirtebilirsiniz. Son kullanma tarihi aralığına ulaşıldığında, kuyruk yöneticisi süresi dolmuş iletileri atmak için taramaya değer aday kuyruklarını arar.

Kuyruk yöneticisi, her kuyruğun süresi dolmuş iletilerine ilişkin bilgileri tutar ve süresi dolan iletiler için bir taramanın değerli olup olmadığını bilir. Bu nedenle, herhangi bir zamanda yalnızca kuyruklar seçimi taranır.

Paylaşılan kuyruklar, bir kuyruk paylaşım grubunda yalnızca bir kuyruk yöneticisi tarafından taranır. Genellikle, ilk kez yeniden başlatılacak ilk kuyruk yöneticidir ya da ilk EXPYINT ayarına sahip olan ilk

kuyruk yöneticidir. Bu kuyruk yöneticisi sona ererse, kuyruk paylaşım grubundaki başka bir kuyruk yöneticisi kuyruk taramasını devralır. Bir kuyruk paylaşım grubu içindeki tüm kuyruk yöneticileri için süre bitimi aralığı değerini aynı değere ayarlayın.

Bir kuyruk yöneticisi yeniden başlatıldığında, EXPYYINT ayarından bağımsız olarak, her kuyruk için süre bitimi işleminin gerçekleşeceğini unutmayın.

### **Belirtik istek**

Taranmasını istediğiniz kuyruğu ya da kuyrukları belirterek REFRESH QMGR TYPE (EXPIRY) komutunu verin.

### *Düşük süre bitim süreleri zorlamalı*

Yöneticiler, kuyruğa ya da konudaki **CUSTOM** özniteisinde belirtilen **CAPEXPY** özniteliğini kullanarak, bir kuyruğa ya da konuya konarak herhangi bir iletinin süre bitimi süresini sınırlayabilirler.

MQMD 'nin **Expiry** alanında belirtilen süre bitimi, bir uygulama tarafından, kuyruğa ya da baştaki **CUSTOM** özniteisinde belirtilen **CAPEXPY** değerinden daha büyük bir süre, bu **CAPEXPY** değeriyle değiştirilir. Bir uygulama tarafından belirtilen süre bitimi ( **CAPEXPY** değerinden daha düşük bir süre kullanılır).

**CAPEXPY** değerinin saniye cinsinden ifade edildiğine dikkat edin. Bu nedenle, bir dakikanın 600 değeri vardır.

Çözüm yolunda birden çok nesne kullanılırsa, örneğin, bir ileti bir diğer ada ya da uzak kuyruğa bulunduğu, ileti süre bitimi için üst sınır olarak tüm **CAPEXPY** değerlerinin en düşük değeri kullanılır.

**CAPEXPY** değerlerinde yapılan değişiklikler hemen yürürlüğe girmektedir. Süre bitimi değeri, her bir kuyruğa koyma değeri bir kuyruğa ya da konuya yönelik olarak değerlendirilir ve bu nedenle, her bir put işlemi arasında farklı olabilecek nesne çözünürlüğe duyarlıdır.

However, note that existing messages in the queue, prior to a change in **CAPEXPY**, are not affected by the change (that is, their expiry time remains intact). Only new messages that are put into the queue after the change in **CAPEXPY** have the new expiry time.

Örneğin, bir put işleminin MQOO\_BIND\_NOT\_FIXED ile açıldığı bir kümede, ileti seçilen hedef kuyruk yöneticisine iletiyi gönderen, kanal tarafından kullanılan iletim kuyruğu için ayarlanan **CAPEXPY** değerine bağlı olarak, her bir put üzerinde farklı süre bitimi değerleri atanabilir.

Teslim gecikmesi, hedef kuyruk ya da konu için çözümlenen süre bitimi süresiyse, MQRC\_EXPIRY\_ERROR ile bir teslim gecikmesi belirten bir JMS uygulaması tarafından bir kuyruğa ya da konuya ilişkin bir put işlemi başarısız olduğunu unutmayın. Bir JMS hedefi için çözülen bir kuyruğun üzerinde ayarlanan **CAPEXPY** özniteliği bu hataya neden olabilir.

**Not: CAEXPY** must not be used on any queues that will hold IBM MQ internally-generated messages such as any SYSTEM.CLUSTER.\* kuyruğu ve SYSTEM.PROTECTION.POLICY.QUEUE.

### **İlgili başvurular**

[TANIMLAR](#)

[TANITIM](#)

### **Geribildirim (MQUZE)**

Geribildirim alanı, raporun niteliyi belirtmek için MQMT\_REPORT tipinde bir iletiyle birlikte kullanılır ve yalnızca bu ileti tipiyle anlamlıdır.

Bu alan, MQFB\_ \* değerlerinden birini ya da MQRC\_ \* değerlerinden birini içerebilir. Geribildirim kodları aşağıdaki gibi gruplanır:

#### **MQFB\_YOK**

Geri bildirim sağlanmadı.

#### **MQFB\_SYSTEM\_FIRST**

Sistem tarafından oluşturulan geribildirim için en düşük değer.

#### **MQFB\_SYSTEM\_SON**

Sistem tarafından oluşturulan geribildirimlere ilişkin en yüksek değer.

MQFB\_SYSTEM\_FIRST ile MQFB\_SYSTEM\_SON ile sistem tarafından oluşturulan geribildirim kodlarının aralığı, bu konuda (MQFB\_\*) listelenen genel geri bildirim kodlarını ve ayrıca ileti hedef kuyruğa konamadığında oluşabilecek neden kodlarının (MQRC\_\*) yer alır.

#### **MQFB\_APPL\_FIRST**

Uygulama tarafından oluşturulan geribildirim için en düşük değer.

#### **MQFB\_APPL\_SON**

Uygulama tarafından oluşturulan geribildirimlere ilişkin en yüksek değer.

Rapor iletileri oluşturan uygulamalar, kuyruk yöneticisi ya da ileti kanalı aracısı tarafından oluşturulan rapor iletilerinin benzetimini yapmak istemedikleri sürece, sistem aralığında (MQFB\_QUIT dışında) geribildirim kodlarını kullanmamalıdır.

MQPUT ya da MQPUT1 çağrılarında, belirtilen değer MQFB\_NONE olması ya da sistem aralığı ya da uygulama aralığı içinde olması gerekir. Bu, *MsgType* değeri ne olursa olsun işaretlendi.

#### **Genel geri bildirim kodları:**

##### **MQFB\_COA**

Hedef kuyruğa gelmenin onayı (MQRO\_COA 'ya bakın).

##### **MQFB\_COD**

Alma uygulamasına teslim edilme onayı (MQRO\_COD 'ye bakın).

##### **MQFB\_BITIM**

İleti, süre bitimi süresinin dolmasından önce hedef kuyruktan kaldırılmadığından atıldı.

##### **MQFB\_PAN**

Pozitif işlem bildirim (MQRO\_PAN 'a bakın).

##### **MQFB\_NAN**

Negatif işlem bildirim (MQRO\_NAN 'a bakın).

##### **MQFB\_QUIT**

Uygulamayı sona erdirin.

Bu, çalışmakta olan bir uygulama programının eşgörünümlerinin denetlenmesini denetlemek için bir iş yükü zamanlama programı tarafından kullanılabilir. Uygulama programının bir örneğine bu geribildirim koduyla bir MQMT\_REPORT ileti gönderilmesi, o yönetim ortamına işlemi durdurması gerektiğini belirtir. Ancak, bu sözleşmeye uyulması uygulama için önemli bir maddedir; kuyruk yöneticisi tarafından uygulanmaz.

#### **Kanal geribildirim kodları:**

##### **MQFB\_CHANNEL\_COMPLETED**

Bir kanal normal şekilde sona erdi.

##### **MQFB\_CHANNEL\_FAIL**

Bir kanal olağandışı biçimde sona erdirildi ve DURDURULDU durumuna geçiyor.

##### **MQFB\_CHANNEL\_FAIL\_RETRY**

Bir kanal olağandışı biçimde sona erdirildi ve RETRY durumuna geçiyor.

#### **IMS-bridge geribildirim kodları**

Beklenmeyen bir IMS-OTMA algılama kodu alındığında bu kodlar kullanılır. Algılama kodu ya da algılama kodu 0x1A bu algılama koduyla ilişkili neden kodu ise *Geribildirim* ile gösterilir.

1. *Geribildirim* kodları için MQFB\_IMS\_FIRST (300) ile MQFB\_IMS\_SON (399) aracılığıyla, 0x1A dışında bir algılama kodu alındı. *algılama kodu*, ifade tarafından verilir (*Geribildirim* - MQFB\_IMS\_FIRST+1)
2. For *Geribildirim* codes in range MQFB\_IMS\_NACK\_1A\_REASON\_FIRST (600) through MQFB\_IMS\_NACK\_1A\_REASON\_LAST (855), a sense code of 0x1A was received. Algılama koduyla ilişkilendirilen *neden kodu* ifadesi (*Geribildirim* - MQFB\_IMS\_NACK\_1A\_REASON\_FIRST)

IMS-OTMA durum kodlarının ve bunlara karşılık gelen neden kodlarının anlamı, *Open Transaction Manager Access Guide and Reference* adlı yayında açıklanmıştır.

IMS köprüsü tarafından aşağıdaki geri bildirim kodları oluşturulabilir:



**MQFB\_DATA\_LENGTH\_ZERO**

İletinin uygulama verilerinde bir kesim uzunluğu sıfırdı.

**MQFB\_DATA\_LENGTH\_NEGATIF**

İletinin uygulama verilerinde bir bölüm uzunluğu negatif oldu.

**MQFB\_DATA\_LENGTH\_TOO\_BIG**

İletinin uygulama verilerinde bir bölüm uzunluğu çok büyük.

**MQFB\_BUFFER\_OVERFLOW**

Uzunluk alanlarından birinin değeri, verilerin ileti arabelleğindeki taşmasına neden olur.

**MQFB\_LENGTH\_OFF\_BY\_ONE**

Uzunluk alanlarından birinin değeri 1 byte 'ı çok kısa.

**MQFB\_IH\_ERROR**

MQMD 'deki *Format* alanı MQFMT\_IMS' i belirtir, ancak ileti geçerli bir MQIIH yapısıyla başlamaz.

**MQFB\_NOT\_AUTHORIZED\_FOR\_IMS**

MQMD ileti tanımlayıcısında bulunan kullanıcı kimliği ya da MQIIH yapısındaki *Authenticator* alanında bulunan parola, IMS köprüsü tarafından gerçekleştirilen geçerlilik denetiminde başarısız oldu. Sonuç olarak ileti IMS' e iletilmedi.

**MQFB\_IMS\_ERROR**

IMStarafından beklenmeyen bir hata döndürüldü. Hatayla ilgili daha fazla bilgi için IMS köprüsünün bulunduğu sistemde IBM MQ hata günlüğüne bakın.

**MQFB\_IMS\_FIRST**

IMS-OTMA algılama kodu 0x1Adeğilse, IMStarafından oluşturulan geribildirim kodları MQFB\_IMS\_FIRST (300) aralığında MQFB\_IMS\_SON (399) ile olur. IMS-OTMA durum kodunun kendisi, *Feedback* eksi MQFB\_IMS\_ERROR ' tır.

**MQFB\_IMS\_SON**

Algılama kodu 0x1Adeğilse, IMS-tarafından oluşturulan geribildirim için en yüksek değer.

**MQFB\_IMS\_NACK\_1A\_REASON\_FIRST**

Algılama kodu 0x1Aolduğunda, IMStarafından oluşturulan geribildirim kodları MQFB\_IMS\_NACK\_1A\_REASON\_FIRST (600) ile MQFB\_IMS\_NACK\_1A\_REASON\_LAST (855) aralığında yer alıyor.

**MQFB\_IMS\_NACK\_1A\_REASON\_LAST**

Algılama kodu 0x1Aolduğunda, IMStarafından oluşturulan geribildirime ilişkin en yüksek değer

**CICS-bridge geribildirim kodları:** Aşağıdaki bildirim kodları CICS bridgetarafından oluşturulabilir:

**MQFB\_CICS\_APPL\_ABENLENDİ**

İletide belirtilen uygulama programı olağandışı sona erdi. Bu geribildirim kodu, yalnızca MQDLH yapısının *Reason* alanında gerçekleşir.

**MQFB\_CICS\_APPL\_NOT\_STARTED**

İletide belirtilen uygulama programına ilişkin EXEC CICS LINK başarısız oldu. Bu geribildirim kodu, yalnızca MQDLH yapısının *Reason* alanında gerçekleşir.

**MQFB\_CICS\_BRIDGE\_FAILURE**

CICS bridge olağan dışı hata işlemeyi tamamlamadan olağan dışı bir şekilde sonlandırıldı.

**MQFB\_CICS\_CCSID\_ERROR**

Karakter takımı tanıtıcısı geçerli değil.

**MQFB\_CICS\_CIH\_ERROR**

CICS bilgi üstbilgisi yapısı eksik ya da geçerli değil.

**MQFB\_CICS\_COMMAREA\_ERROR**

CICS COMMAREA uzunluğu geçerli değil.

**MQFB\_CICS\_CORREL\_ID\_ERROR**

İlinti tanıtıcısı geçerli değil.

**MQFB\_CICS\_DLQ\_ERROR**

CICS bridge görevi, bu istekle gönderilen bir yanıtı ölü-mektup kuyruğuna kopyalayamadı. İstek geriletildi.

**MQFB\_CICS\_ENCODING\_ERROR**

Kodlama geçerli değil.

**MQFB\_CICS\_INTERNAL\_ERROR**

CICS bridge beklenmeyen bir hata saptadı.

Bu geribildirim kodu, yalnızca MQDLH yapısının *Reason* alanında gerçekleşir.

**MQFB\_CICS\_NOT\_YETKILI**

Kullanıcı kimliği yetkili değil ya da parola geçerli değil.

Bu geribildirim kodu, yalnızca MQDLH yapısının *Reason* alanında gerçekleşir.

**MQFB\_CICS\_UOW\_BACKED\_OUT**

İş birimi geriletildi; bunun nedeni aşağıdakilerden biri olabilir:

- Aynı iş birimi içinde başka bir istek işlenirken bir hata saptandı.
- Bir CICS olağandışı sonu, iş birimi devam ederken oluştu.

**MQFB\_CICS\_UOW\_ERROR**

İş birimi denetim alanı *UOWControl* geçerli değil.

**İzleme rotası ileti geribildirim kodları:****MQFB\_ACTIVITY**

Etkinlik raporlarını izleyen kullanıcı verileri seçeneğine izin vermek için, MQFMT\_EMBEDDED\_PCF biçimiyle kullanılır.

**MQFB\_MAX\_ACTIVITIES**

İletin ilişkili olduğu etkinliklerin sayısı maksimum etkinlik sınırını aştığından izleme rotası ileti atıldığında döndürülür.

**MQFB\_NOT\_ILETILDI**

İzleme rotası ileti atıldığında, izleme rotası iletilerini desteklemeyen uzak bir kuyruk yöneticisine gönderilmek üzere atıldığında döndürülür.

**MQFB\_NOT\_TESLIM**

İzleme rotası ileti yerel bir kuyruğa konmak üzere olduğu için atıldığında döndürülür.

**MQFB\_UNSUPPORTED\_FORWARDING**

İletme (iletme) değiştirgesindeki bir değer tanınmadığı için, izleme rotası ileti atıldığında ve reddedilen bit maskesinde yer aldığından bu ileti yayınlanır.

**MQFB\_UNSUPPORTED\_DELIVERY**

Teslim parametresindeki bir değer tanınmadığı için izleme rotası ileti atıldığında ve reddedilen bit maskesinde yer aldığından, döndürülen rota ileti yayınlanır.

**IBM MQ neden kodları:** Kural dışı durum raporu iletileri için *Feedback* , bir IBM MQ neden kodu içerir. Olası neden kodları arasında şunlar yer alıyor:

**MQRC\_PUT\_INHIBITED**

(2051, X'803 ') Kuyruk araması engellenmiş olarak çağrılıyor.

**MQRC\_Q\_FULL**

(2053, X'805 ') Kuyruk, ileti sayısı üst sınırını zaten içeriyor.

**MQRC\_NOT\_YETKILI**

(2035, X'7F3') Erişim için yetkili değil.

**MQRC\_Q\_SPACE\_NOT\_VAR**

(2056, X'808 ') Kuyruğun diskte kullanılabilir alan yok.

**MQRC\_PERSISTENT\_NOT\_ALLOWED**

(2048, X'800 ') Kuyruk kalıcı iletileri desteklemiyor.

## **MQRC\_MSG\_TOO\_BIG\_FOR\_Q\_MGR**

(2031, X'7EF') İleti uzunluğu kuyruk yöneticisi için üst sınırdan büyük.

## **MQRC\_MSG\_TOO\_BIG\_FOR\_Q**

(2030, X'7EE') İleti uzunluğu, kuyruk için üst sınırdan büyük.

Neden kodlarının tam listesi için aşağıdaki başlara bakın:

- IBM MQ for z/OS için bkz. [API tamamlama ve neden kodları](#).
- Diğer tüm platformlar için bkz. [API tamamlama ve neden kodları](#).

Bu, MQGET çağrısına ilişkin bir çıkış alanıdır ve MQPUT ve MQPUT1 çağrıları için bir giriş alanı. Bu alanın ilk değeri MQFB\_NONE olur.

## **Kodlama (MQUZE)**

Bu, iletteki sayısal verilerin sayısal kodlamasını belirtir; MQMD yapısındaki sayısal veriler için geçerli değildir. Sayısal kodlama, ikili tamsayılar, paketlenmiş ondalık tamsayılar ve kayan noktalı sayılar için kullanılan gösterimi tanımlar.

z/OS' ta, ilgili karakter takımı tanıtıcısı, karakter kümesinin gösteriminin ikili tamsayılar için kullanılan kodlamaya bağımlı olduğunu gösterdiğinde, ileti gövdesindeki karakter verilerinin tamsayı kodlamasını belirtmek için de Encoding alanının ikili tamsayı bölümü kullanılır. Bu, yalnızca bazı çok baytlı karakter kümelerini (örneğin, UTF-16 karakter takımları) etkiler.

MQPUT ya da MQPUT1 çağrısında, uygulama bu alanı veri için uygun değere ayarlamalıdır. Kuyruk yöneticisi, alanın geçerli olup olmadığını denetmez. Aşağıdaki özel değer tanımlıdır:

## **MQENC\_NATIVE**

Kodlama, uygulamanın çalışmakta olduğu programlama dili ve makinesi için varsayılan değerdir.

**Not:** Bu değişimin değeri, programlama diline ve ortama bağlıdır. Bu nedenle, uygulamaların, uygulamanın çalışacağı ortama uygun olan üstbilgi, makro, COPY ya da INCLUDE dosyaları kullanılarak derlenmeleri gerekir.

İleti yerleştiren uygulamalar genellikle MQENC\_NATIVE değerini belirtir. İletileri alan uygulamaların bu alanı MQENC\_NATIVE; değeriyle karşılaştırmaları gerekir; değerler farklı olursa, uygulamanın sayısal verileri iletiyle dönüştürmesi gerekebilir. Kuyruk yöneticisini, MQGET çağrısının işlenmesinin bir parçası olarak dönüştürmek üzere kuyruk yöneticisini istemek için MQGMO\_CONVERT seçeneğini kullanın. Encoding alanının nasıl oluşturulabilmesiyle ilgili ayrıntılar için bkz. [“Makine kodlamaları” sayfa 884](#) .

MQGET çağrısında MQGMO\_CONVERT seçeneğini belirtirseniz, bu alan bir giriş/çıkış alanıdır. Uygulama tarafından belirtilen değer, gerekiyorsa ileti verilerinin dönüştürülebileceği kodlamadır. Dönüştürme başarılı ya da gereksizse, değer değişmeden kalır. Dönüştürme başarısız olursa, MQGET çağrısından sonraki değer, uygulamaya döndürülen dönüştürülmemiş iletinin kodlamasını temsil eder.

Diğer durumlarda, bu, MQGET çağrısına ilişkin bir çıkış alanıdır ve MQPUT ve MQPUT1 çağrılarına ilişkin bir giriş alanı. Bu alanın ilk değeri MQENC\_NATIVE 'dir.

## **CodedCharSetId (MQHOMER)**

Bu alan, ileti gövdesinde karakter verilerinin karakter kümesi tanıtıcısını belirtir.

**Not:** MQMD ' deki karakter verileri ve çağrılarda değiştirge olan diğer MQ veri yapıları, kuyruk yöneticisinin karakter kümesinde bulunmalıdır. Bu, kuyruk yöneticisinin **CodedCharSetId** özneliği tarafından tanımlanır; bu özneliğe ilişkin ayrıntılar için [“Kuyruk yöneticisine ilişkin öznelikler” sayfa 785](#) ' e bakın.

Bu alan, seçeneklerde MQGMO\_CONVERT ile MQGET çağrılırken MQCCSI\_Q\_MGR olarak ayarlanırsa, istemci ile sunucu uygulamaları arasında davranış farklıdır. Sunucu uygulamaları için, karakter dönüştürme işlemi için kullanılan kod sayfası kuyruk yöneticisinin CodedCharSetId 'sidir; istemci uygulamaları için, karakter dönüştürme için kullanılan kod sayfası yürürlükteki ülke değeri kod sayfasıdır.

İstemci uygulamaları için, MQCCSI\_Q\_MGR, kuyruk yöneticisinde bulunan istemcinin yerine istemcinin yerel ayarlarına dayalı olarak doldurulur. Bu kuralın kural dışı durumu, bir iletiyi bir IMS köprü kuyruğuna koyduğunuzda; döndürülen, MQMD ' nin *CodedCharSetId* alanında, kuyruk yöneticisinin CCSID 'sidir.

Aşağıdaki özel değeri kullanmamanız gerekir:

#### **MQCCSI\_APPL**

Bu, MQMD ' nin *CodedCharSetId* alanında yanlış bir değerle sonuçlanır ve MQRC\_SOURCE\_CCSID\_ERROR dönüş koduna (ya da z/OS için MQRC\_FORMAT\_ERROR) neden olur. İleti, MQGO\_CONVERT seçeneğiyle MQGET çağrısını kullanarak alındığında.

Aşağıdaki özel değerleri kullanabilirsiniz:

#### **MQCCSI\_Q\_MGR**

İletideki karakter verileri, kuyruk yöneticisinin karakter takımında yer alıyor.

MQPUT ve MQPUT1 çağrılarında, kuyruk yöneticisi bu değeri, kuyruk yöneticisinin true karakter kümesi tanıtıcısına iletiyle gönderilen MQMD ' de değiştirir. Sonuç olarak, MQCCSI\_Q\_MGR değeri MQGET çağrısıyla hiçbir zaman döndürülmez.

#### **MQCCSI\_VARSA**

*String* alanındaki verilerin *CodedCharSetId* değeri, MQCFH yapısından önce gelen üstbilgi yapısındaki *CodedCharSetId* alanı tarafından ya da MQCFH iletinin başlangıcındaki MQMD ' deki *CodedCharSetId* alanı tarafından tanımlanır.

#### **MQCCSI\_INHERIT**

İletideki karakter verileri, bu yapıyla aynı karakter takımında yer alıyor; bu, kuyruk yöneticisinin karakter takısıdır. (Yalnızca MQMD için, MQCCSI\_INHERIT, MQCCSI\_Q\_MGR ile aynı anlamlara sahiptir).

Kuyruk yöneticisi, MQMD 'nin gerçek karakter kümesi tanıtıcısına iletiyle gönderilen MQMD' de bu değeri değiştirir. Herhangi bir hata oluşmaması koşuluyla, MQCCSI\_INHERIT değeri MQGET çağrısıyla döndürülmez.

MQMD ' deki *PutAppLType* alanının değeri MQAT\_BROKER ise, MQCCSI\_INHERIT kullanmayın.

#### **MQCCSI\_EMBEDDED**

İletideki karakter verileri, ileti verilerinin kendisinde yer alan tanıtıcıyı içeren bir karakter kümesinde bulunur. Verilerin farklı kısımlarına uygulanırken, ileti verileri içine yerleştirilmiş herhangi bir sayıda karakter kümesi tanıtıcısı olabilir. Bu değer, karakter kümelerinin bir karışımında veri içeren PCF iletileri (MQFMT\_ADMIN, MQFMT\_EVENT ya da MQFMT\_PCF biçimiyle) için kullanılmalıdır. PCF iletileri içinde yer alan her MQCFST, MQCFSL ve MQCFSTF yapısının belirtilmiş bir belirtik karakter kümesi tanıtıcısı olmalıdır; MQCCSI\_VARSAYILANSICISINIR değil.

MQFMT\_EMBEDDED\_PCF biçiminde bir ileti, karakter kümelerinin karışımında veri içerecekse, MQCCSI\_EMBEDDED kullanmayın. Bunun yerine, MQEPH yapısındaki İşaretler alanına MQPH\_CCSID\_EMBEDDED olarak ayarlayın. Bu, önceki yapıdaki MQCCSI\_EMBEDDED ayarını ayarlamaya eşdeğerdir. PCF iletileri içinde bulunan her MQCFST, MQCFSL ve MQCFSTF yapısının belirtilmiş bir belirtik karakter kümesi tanıtıcısı olmalıdır ve MQCCSI\_XX\_ENCODE\_CASE\_ONE default belirlenmemiş olmalıdır. MQEPH yapısı hakkında daha fazla bilgi için bkz. [“MQEPH-Gömülü PCF üstbilgisi” sayfa 359.](#)

Bu değeri yalnızca MQPUT ve MQPUT1 çağrılarında belirtin. MQGET çağrısında belirtilirse, iletinin dönüştürülmesini önler.

MQPUT ve MQPUT1 çağrılarında, kuyruk yöneticisi, yukarıda açıklandığı gibi, iletiyle birlikte gönderilen MQCCSI\_Q\_MGR ve MQCCSI\_INHERIT değerlerini, MQPUT ya da MQPUT1 çağrısında belirtilen MQMD ' yi değiştirmiyor. Belirtilen değerde başka bir denetim gerçekleştirilmez.

İletileri alan uygulamaların bu alanı uygulamanın beklediği değerle karşılaştırmaları gerekir; değerler farklı olursa, uygulamanın karakter verilerini iletide dönüştürmesi gerekebilir.

On z/OS, the Encoding field of the MQMD is used to specify the integer encoding of character data in the message body, when the *CodedCharSetId* field of the MQMD indicates that the representation of the character set is dependent on the encoding used for binary integers. Multiplatformsüzerinde, karakter

verilerinin byte sırası, kuyruk yöneticisinin çalışmakta olduğu altyapıya ilişkin yerli tamsayı kodlamasıyla aynı olmalıdır. Bu yalnızca bazı çok baytlı karakter kümelerini etkiler (örneğin, UTF-16 karakter kümeleri).

MQGET çağrısında MQGMO\_CONVERT seçeneğini belirtirseniz, bu alan bir giriş/çıkış alanıdır. Uygulama tarafından belirlenen değer, gerektiğinde ileti verilerinin dönüştürülebileceği kodlanmış karakter takımı tanıtıcısıdır. Dönüştürme başarılı ya da gereksizse, değer değişmeden kalır (MQCCSI\_Q\_MGR ya da MQCCSI\_INHERHERT değeri gerçek değere dönüştürülmüş olur). Dönüştürme başarısız olursa, MQGET çağrısından sonraki değer, uygulamaya döndürülen dönüştürülmemiş iletinin kodlanmış karakter takımı tanıtıcısını gösterir.

Tersi durumda, bu, MQGET çağrısına ilişkin bir çıkış alanıdır ve MQPUT ve MQPUT1 çağrılarına ilişkin bir giriş alanı olur. Bu alanın ilk değeri MQCCSI\_Q\_MGR ' dir.

### **Biçim (MQCHAR8)**

Bu ad, iletiyi gönderenin, iletteki verilerin doğasını alıcıya göstermek için kullandığı bir addir. Bu ad için, kuyruk yöneticisinin karakter kümesinde yer alan karakterler belirtilebilir; ancak, adı aşağıdakine göre sınırlamanız gerekir:

- A-Z arasındaki büyük harf
- 0-9 arası sayısal rakamlar

Diğer karakterler kullanılırsa, gönderme ve alma kuyruğu yöneticilerinin karakter kümeleri arasındaki adı çevirmek olanaklı olmayabilir.

Adı, alanın uzunluğuna doğru boşluklarla doldur ya da alanın sonundan önce sonlandırmak için boş değerli bir karakter kullanın; boş değer ve sonraki karakterler boşluk olarak kabul edilir. Baştaki ya da gömülü boşluklar içeren bir ad belirlemeyin. MQGET çağrısına ilişkin kuyruk yöneticisi, boşlukla doldurulan adı alanın uzunluğuna göre döndürür.

Kuyruk yöneticisi, adın yukarıda açıklanan önerilere uygun olup olmadığını denetleyemez.

Üst, alt ve karışık büyüklerde MQ ' un başındaki adlar, kuyruk yöneticisi tarafından tanımlanan anlamlara sahiptir; kendi biçimleriniz için bu harflerle başlayan adları kullanmayın. Kuyruk yöneticisi yerleşik biçimleri şunlardır:

#### **MQFMT\_NONE**

Verilerin türü tanımsız: MQGMO\_CONVERT seçeneği kullanılarak bir kuyruktan ileti alındığında veriler dönüştürülemez.

MQGET çağrısında MQGMO\_CONVERT ögesini ve iletteki verilerin karakter kümesini ya da kodlamasını, **MsgDesc** değiştirgesinde belirtilenden farklıysa, ileti aşağıdaki tamamlanma ve neden kodlarıyla döndürülür (başka hata varsayılarak):

- MQFMT\_NONE verileri iletinin başında yer aldıysa, MQCC\_UYARI tamamlanma kodu ve MQRC\_FORMAT\_ERROR neden kodu.
- MQFMT\_NONE verileri iletinin sonunda (yani bir ya da daha çok MQ üstbilgi yapısından önce gelen), MQCC\_OK ve neden kodu MQRC\_NONE tamamlanma kodu. MQ üstbilgi yapıları, bu durumda istenen karakter kümesine ve kodlamaya dönüştürüldü.

C programlama dili için, sabit MQFMT\_NONE\_DIZISI de tanımlıdır; bu, MQFMT\_NONE ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisi olur.


#### **MQFMT\_ADMIN**

İleti, programlanabilir komut biçiminde (PCF) komut sunucusu isteği ya da yanıt iletidir. MQGET çağrısında MQGMO\_CONVERT seçeneği belirtildiyse, bu biçimdeki iletiler dönüştürülebilmektedir. Programlanabilir komut biçimi iletileri kullanmaya ilişkin ek bilgi için [Programlanabilir Komut Biçimlerinin Kullanılması](#) konusuna bakın.

C programlama dili için, MQFMT\_ADMIN\_ARRAY değişmezi de tanımlanır; bu değer, MQFMT\_ADMIN ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisidir.

## **MQFMT\_CICS**

The message data begins with the CICS information header MQCIH, followed by the application data. Uygulama verilerine ilişkin biçim adı, MQCIH yapısındaki *Format* alanı tarafından verilir.

 z/OS üzerinde, MQFMT\_CICS biçimine sahip iletileri dönüştürmek için MQGET çağrısında MQGMO\_CONVERT seçeneğini belirtin.

C programlama dili için, MQFMT\_CICS\_ARRAY değişmezi de tanımlıdır; bu, MQFMT\_CICS ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisi olur.

## **MQFMT\_COMMAND\_1**

İleti, nesne sayısını, tamamlanma kodunu ve neden kodunu içeren bir MQSC komut sunucusu yanıt iletidir. MQGET çağrısında MQGMO\_CONVERT seçeneği belirtildiyse, bu biçimdeki iletiler dönüştürülebilmektedir.

For the C programming language, the constant MQFMT\_COMMAND\_1\_ARRAY is also defined; this has the same value as MQFMT\_COMMAND\_1, but is an array of characters instead of a string.

## **MQFMT\_COMMAND\_2**

İleti, istenen nesnelere ilgili bilgileri içeren bir MQSC komut sunucusu yanıt iletidir. MQGET çağrısında MQGMO\_CONVERT seçeneği belirtildiyse, bu biçimdeki iletiler dönüştürülebilmektedir.

For the C programming language, the constant MQFMT\_COMMAND\_2\_ARRAY is also defined; this has the same value as MQFMT\_COMMAND\_2, but is an array of characters instead of a string.

## **MQFMT\_DEAD\_LETTER\_HEADER**

İleti verileri, MQDLH adlı ölü-harf üstbilgisi ile başlar. Özgün iletimden gelen veriler, MQDLH yapısından hemen sonra gelir. Özgün ileti verilerinin biçim adı, MQDLH yapısındaki *Format* alanı tarafından verilir; bu yapının ayrıntıları için [“MQDLH-Harf üstbilgisi” sayfa 347](#) ' e bakın. MQGET çağrısında MQGMO\_CONVERT seçeneği belirtildiyse, bu biçimdeki iletiler dönüştürülebilmektedir.

*Format* MQFMT\_DEAD\_LETTER\_HEADER alan iletiler için COA ve COD raporları oluşturulmaz.

C programlama dili için, sabit MQFMT\_DEAD\_LETTER\_HEADER\_ARRAY değişmezi de tanımlanır; bu değer MQFMT\_DEAD\_LETTER\_HEADER ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisi olur.

## **MQFMT\_DIST\_HEADER**

İleti verileri, dağıtım listesi üstbilgisiyle (MQDH) başlar; bu, MQOR ve MQPMR kayıtlarının dizilerini içerir. Dağıtım listesi üstbilgisinde ek veriler de izlenebilir. Ek verilerin biçimi (varsa), MQDH yapısındaki *Format* alanı tarafından verilir; bu yapının ayrıntıları için [“MQDH-Dağıtım üstbilgisi” sayfa 341](#) ' e bakın. MQGET çağrısında MQGMO\_CONVERT seçeneği belirtildiyse, MQFMT\_DIST\_HEADER biçimi olan iletiler dönüştürülebilmektedir.

Bu biçim aşağıdaki ortamlarda desteklenir:

-  AIX
-  IBM i
-  Linux
-  Windows

ve bu sistemlere bağlı IBM MQ MQI clients için.

C programlama dili için, MQFMT\_DIST\_HEADER\_ARRAY değişmezi de tanımlanır; bu değer MQFMT\_DIST\_HEADER ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisidir.

## **MQFMT\_EMBEDDED\_PCF**

PCF komut değerinin MQCMD\_TRACE\_ROUTE olarak ayarının sağlanması koşuluyla, izleme rotası ileti biçimi. Bu biçimle, uygulamalarının önceki PCF parametreleriyle başa çıkabilmesi koşuluyla, kullanıcı verilerinin izleme rotası iletiyle birlikte gönderilmesine izin verilir.

PCF üstbilgisi ilk üstbilgi olmalıdır ya da ileti, izleme rotası iletisi olarak işlenmez. Bu, iletinin bir grupta yer almayacağı ve izleme rotası iletilerinin kesimlenemeyeceği anlamına gelir. Bir grup içinde izleme rotası iletisi gönderilirse, ileti, MQRC\_MSG\_NOT\_ALLOWED\_IN\_GROUP neden koduyla reddedilir.

MQFMT\_ADMIN, izleme rotası iletisinin biçimi için de kullanılabilir; ancak, bu durumda izleme rotası iletisiyle birlikte herhangi bir kullanıcı verisi gönderilemez.

### **MQFMT\_OLAY**

İleti, oluşan bir olayı bildiren bir MQ olay iletidir. Olay iletileri programlanabilir komutlarla aynı yapıya sahiptir; bu yapıyla ilgili ek bilgi için [PCF komut iletileri](#) ve olaylarla ilgili bilgi için [Olay izleme](#) başlıklı konuya bakın.

MQGET çağrısında MQGMO\_CONVERT seçeneği belirtilirse, Version-1 olay iletileri tüm ortamlarda dönüştürülebilir. Version-2 olay iletileri yalnızca z/OS' ta dönüştürülebilir.

C programlama dili için, sabit MQFMT\_EVENT\_ARRAY değişmezi de tanımlanır; bu, MQFMT\_EVENT ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisi olur.

### **MQFMT\_IMS**

İleti verileri, uygulama verileri tarafından izlenen IMS bilgi üstbilgisi MQIIH (MQIIH) ile başlar. Uygulama verilerine ilişkin biçim adı, MQIIH yapısındaki Format alanı tarafından verilir.

MQGMO\_CONVERT ile MQGET kullanılırken MQIIH yapısının nasıl ele alınmasıyla ilgili ayrıntılar için ["Biçim \(MQCHAR8\)" sayfa 405](#) ve ["ReplyToBiçimi \(MQCHAR8\)" sayfa 406](#) konusuna bakın.

C programlama dili için, MQFMT\_IMS\_ARRAY değişmezi de tanımlanır; bu, MQFMT\_IMS ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisidir.

### **MQFMT\_IMS\_VAR\_STRING**

İleti, 11zzccc biçiminin bir dizgisi olan IMS değişkeni dizgisidir; burada:

#### **11**

IMS değişken dizgi ögesinin toplam uzunluğunu belirten 2 baytlık bir uzunluk alanıdır. Bu uzunluk, 11 (2 byte) uzunluğuna eşittir ve zz (2 byte) uzunluğunun yanı sıra, karakter dizgisinin uzunluğuna eşittir. 11 , Encoding alanı tarafından belirtilen kodlamadaki 2 baytlık ikili tamsayıdır.

#### **zz**

IMS için önemli olan işaretleri içeren 2 baytlık bir alandır. zz , iki MQBYTE alanından oluşan bir bayt dizilimidir ve göndericiden alıcıya değişiklik yapılmadan iletilir (yani, zz herhangi bir dönüştürmeye tabi değildir).

#### **ccc**

11-4 karakterlerini içeren değişken uzunluklu bir karakter dizilimidir. ccc , CodedCharSetId alanı tarafından belirtilen karakter kümesinde yer alıyor.

On z/OS, the message data can consist of a sequence of IMS variable strings butted together, with each string being of the form 11zzccc. Ardışık IMS değişken dizgileri arasında atlanan byte olmamalıdır. Bu, ilk dizilimin tek bir uzunluğu varsa, ikinci dizginin yanlış hizalanacağı anlamına gelir; yani, iki tane birden çok olan bir sınırda başlamayacaktır. Temel veri tiplerinin hizalamasını gerektiren makinelerde bu tür dizgileri oluştururken dikkatli olun.

MQFMT\_IMS\_VAR\_STRING biçimine sahip iletileri dönüştürmek için MQGET çağrısında MQGMO\_CONVERT seçeneğini kullanın.

C programlama dili için, MQFMT\_IMS\_VAR\_STRING\_ARRAY değişmezi de tanımlıdır; bu, MQFMT\_IMS\_VAR\_STRING ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisi olur.

### **MQFMT\_MD\_EXTENSION**

İleti verileri, MQMDE ileti tanımlayıcısı uzantısıyla başlar ve isteğe bağlı olarak diğer veriler de (genellikle uygulama iletisi verileri) izler. MQMDE 'yi izleyen verilerin biçim adı, karakter kümesi ve kodlaması, MQMDE' deki Format, CodedCharSetId ve Encoding alanları tarafından verilir. Bu yapıyla ilgili ayrıntılar için bkz. ["MQMDE-İleti tanımlayıcı uzantısı" sayfa 464](#) . MQGET çağrısında MQGMO\_CONVERT seçeneği belirtildiyse, bu biçimdeki iletiler dönüştürülebilmektedir.

C programlama dili için, sabit MQFMT\_MD\_EXTENSION\_ARRAY değişmezi de tanımlanır; bu değer MQFMT\_MD\_EXTENSION ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisidir.



## MQFMT\_PCF

İleti, programlanabilir komut biçimi (PCF) iletilerinin yapısına uygun, kullanıcı tanımlı bir iletidir. MQGET çağrısında MQGMO\_CONVERT seçeneği belirtildiyse, bu biçimdeki iletiler dönüştürülebilmektedir. Programlanabilir komut biçimi iletileri kullanmaya ilişkin ek bilgi için [Programlanabilir Komut Biçimlerinin Kullanılması](#) konusuna bakın.

C programlama dili için, aynı zamanda MQFMT\_PCF\_ARRAY değişmezi de tanımlanır; bu, MQFMT\_PCF ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisi olur.

## MQFMT\_REF\_MSG\_HEADER

İleti verileri, MQRMH başvuru iletisi üstbilgisiyle başlar ve isteğe bağlı olarak diğer veriler tarafından takip edilir. Verilerin biçim adı, karakter kümesi ve kodlaması, MQRMH 'deki Format, CodedCharSetIdve Encoding alanları tarafından verilir. Bu yapıyla ilgili ayrıntılar için bkz. [“MQRMH-Başvuru iletisi üstbilgisi” sayfa 540](#) . MQGET çağrısında MQGMO\_CONVERT seçeneği belirtildiyse, bu biçimdeki iletiler dönüştürülebilmektedir.

Bu biçim aşağıdaki ortamlarda desteklenir:

-  AIX
-  IBM i
-  Linux
-  Windows

ve bu sistemlere bağlı IBM MQ MQI clients için.

C programlama dili için, MQFMT\_REF\_MSG\_HEADER\_ARRAY değişmezi de tanımlıdır; bu değer MQFMT\_REF\_MSG\_HEADER ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisi olur.

## MQFMT\_RF\_HEADER

İleti verileri, kural ve biçimlendirme üstbilgisi MQRFH ile başlar ve isteğe bağlı olarak diğer veriler tarafından takip edilir. Verilerin biçim adı, karakter kümesi ve kodlanması (varsa), MQRFH içindeki Format, CodedCharSetIdve Encoding alanları tarafından verilir. MQGET çağrısında MQGMO\_CONVERT seçeneği belirtildiyse, bu biçimdeki iletiler dönüştürülebilmektedir.

C programlama dili için, aynı zamanda MQFMT\_RF\_HEADER\_ARRAY değişmezi de tanımlanır; bu, MQFMT\_RF\_HEADER ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisi olur.

## MQFMT\_RF\_HEADER\_2

İleti verileri, version-2 kuralları ve biçimlendirme üstbilgisi MQRFH2ile başlar ve isteğe bağlı olarak diğer veriler tarafından takip edilir. İsteğe bağlı verilerin biçim adı, karakter kümesi ve kodlanması (varsa), MQRFH2içinde Format, CodedCharSetIdve Encoding alanları tarafından verilir. MQGET çağrısında MQGMO\_CONVERT seçeneği belirtildiyse, bu biçimdeki iletiler dönüştürülebilmektedir.

For the C programming language, the constant MQFMT\_RF\_HEADER\_2\_ARRAY is also defined; this has the same value as MQFMT\_RF\_HEADER\_2, but is an array of characters instead of a string.

## MQFMT\_STRING

Uygulama iletisi verileri bir SBCS dizgisi (tek baytlık karakter takımı) ya da bir DBCS dizilimi (çift baytlık karakter takımı) olabilir. MQGET çağrısında MQGMO\_CONVERT seçeneği belirtildiyse, bu biçimdeki iletiler dönüştürülebilmektedir.

C programlama dili için, sabit MQFMT\_STRING\_ARRAY değişmezi de tanımlıdır; bu, MQFMT\_STRING ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisidir.

## MQFMT\_TETIKLEYICISI


İleti, MQTM yapısı tarafından açıklanan bir tetikleme iletidir; bu yapının ayrıntıları için [“MQTM-Tetikleyici iletisi” sayfa 589](#) konusuna bakın. MQGET çağrısında MQGMO\_CONVERT seçeneği belirtildiyse, bu biçimdeki iletiler dönüştürülebilmektedir.

C programlama dili için, MQFMT\_TRIGGER\_ARRAY değişmezi de tanımlanır; bu değer, MQFMT\_TRIGGER ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisidir.



## **MQFMT\_WORK\_INFO\_HEADER**

İleti verileri, uygulama verileri tarafından izlenen iş bilgisi üstbilgisi MQWIIH ile başlar. Uygulama verilerine ilişkin biçim adı, MQWIIH yapısındaki Format alanı tarafından verilir.

 z/OS üzerinde, kullanıcı verilerini MQFMT\_WORK\_INFO\_HEADER biçiminde olan iletilerde dönüştürmek için MQGET çağrısında MQGMO\_CONVERT seçeneğini belirtin. Ancak, MQWIIH yapısının kendisi her zaman kuyruk yöneticisinin karakter kümesi ve kodlamasında döndürülür (yani, MQWIIH yapısı, MQGMO\_CONVERT seçeneğinin belirlenip belirlenmeyeceği dönüştürülür).

C programlama dili için, MQFMT\_WORK\_INFO\_HEADER\_ARRAY değişmezi de tanımlanır; bu değer, MQFMT\_WORK\_INFO\_HEADER ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisi olur.

## **MQFMT\_XMIT\_Q\_HEADER**

İleti verileri, MQXQH iletim kuyruğu üstbilgisiyle başlar. Özgün iletimden gelen veriler, MQXQH yapısından hemen sonra gelir. Özgün ileti verilerinin biçim adı, MQXQH iletim kuyruğu üstbilgisinin bir parçası olan MQMD yapısındaki Format alanı tarafından verilir. Bu yapıyla ilgili ayrıntılar için bkz. [“MQXQH-İletim kuyruğu üstbilgisi” sayfa 608](#).

COA and COD reports are not generated for messages that have a Format of MQFMT\_XMIT\_Q\_HEADER.

C programlama dili için, sabit MQFMT\_XMIT\_Q\_HEADER\_ARRAY değişmezi de tanımlanır; bu değer MQFMT\_XMIT\_Q\_HEADER ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisi olur.

Bu, MQGET çağrısına ilişkin bir çıkış alanıdır ve MQPUT ve MQPUT1 çağrılarında ilişkin bir giriş alanı. Bu alanın uzunluğu MQ\_FORMAT\_LENGTH tarafından verilir. Bu alanın ilk değeri MQFMT\_NONE olur.

## **Öncelik (MQUBE)**

MQPUT ve MQPUT1 çağrıları için, değer in sıfırdan büyük ya da sıfıra eşit olması gerekir; sıfır, en düşük önceliğe sahip olmalıdır. Aşağıdaki özel değer de kullanılabilir:

### **MQPRI\_PRIORITY\_AS\_Q\_DEF**

- Kuyruk bir küme kuyruğıysa, iletiye ilişkin öncelik, iletimin yerleştirdiği kuyruğun belirli bir örneğinin sahibi olan *hedef* kuyruk yöneticisinde tanımlanan **DefPriority** özniteliğinden alınır.

Küme kuyruğunun birden çok eşgörünümü varsa ve bu öznitelikte farklılık gösterdiklerinde, bunlardan birinin tane seçkullanım seçtiği değer tahmin edilemez. Bu nedenle, bu özniteliği tüm eşgörünümelerde aynı değere ayarlamalısınız. Böyle bir durumda değilse, kuyruk yöneticisi günlüklerine AMQ9407 hata ileti gönderilir. Ayrıca bkz. [Diğer adlar, uzak kuyruklar ve küme kuyrukları için hedef nesne öznitelikleri nasıl çözülür?](#)

İleti hedef kuyruğa yerleştirildiğinde, *DefPriority* değeri *Priority* alanına kopyalanır. *DefPriority* daha sonra değiştirilirse, kuyruğa önceden yerleştirmiş olan iletiler etkilenmez.

- Kuyruk bir küme kuyruğu değilse, ileti için öncelik, hedef kuyruk yöneticisi uzak olsa bile, *yerel* kuyruk yöneticisinde tanımlanan **DefPriority** özniteliğinden alınır.

Kuyruk adı çözümleme yolunda birden çok tanımlama varsa, yoldaki *ilk* tanımlamasındaki bu özniteliğin değerinden varsayılan öncelik alınır. Bu durumda şunlar olabilir:

- Diğer ad kuyruğu
- Yerel kuyruk
- Uzak kuyruğun yerel tanımlaması
- Kuyruk yöneticisi diğer adı
- Bir iletim kuyruğu (örneğin, *DefXmitQName* kuyruğu)

İleti konduğunda, *DefPriority* değeri *Priority* alanına kopyalanır. *DefPriority* daha sonra değiştirilirse, önceden yerleştirmiş olan iletiler bundan etkilenmez.

MQGET çağrısının döndürdüğü değer her zaman sıfırdan büyük ya da bu değere eşittir; MQPRI\_PRIORITY\_AS\_Q\_DEF değeri hiçbir zaman döndürülmez.

Bir ileti, yerel kuyruk yöneticisi tarafından desteklenen üst sınırdan daha büyük bir önceliğe sahip olursa (bu üst sınır, **MaxPriority** kuyruk yöneticisi özneliği tarafından verilir), ileti kuyruk yöneticisi tarafından kabul edilir, ancak kuyruk yöneticisinin en yüksek önceliğindeki kuyruğa yerleştirilir; MQPUT ya da MQPUT1 çağrısı MQCC\_UYARI ile tamamlanır ve neden kodu MQRC\_PRIORITY\_EXCEEDS\_MAXIMUM. Ancak, *Priority* alanı, iletiyi koyan uygulama tarafından belirtilen değeri korur.

z/OS' ta, MsgSeqnumarasına sahip bir ileti, ileti teslim sırası MQMDS\_PRIORITY ve bir MQIT\_GROUP\_ID dizin tipine sahip bir kuyruğa konursa, kuyruk, iletiyi farklı bir önceliğe sahip olarak değerlendirebilir. İleti, kuyruk üzerinde 0 ya da 1 önceliğiyle yerleştirildiyse, öncelik 2 'nin önceliği olduğu gibi işlenir. Bunun nedeni, bu kuyruk tipine yerleştirilecek iletilerin sırası, verimli grup tamamlama sınamalarını etkinleştirmek için eniyilenmiş olur. İleti teslimi sırası MQMDS\_PRIORITY ve MQIT\_GROUP\_ID dizin tipi ile ilgili ek bilgi için [MsgDeliverySequence](#) attributebaşlıklı konuya bakın.

Bir iletiyi yanıtlarken, uygulamaların yanıt iletisine ilişkin istek iletisinin önceliğini kullanması gerekir. Diğer durumlarda, MQPRI\_PRIORITY\_AS\_Q\_DEF değerinin belirlenmesi, uygulamanın değiştirilmeden önce gerçekleştirilmesine izin verir.

Bu, MQGET çağrısına ilişkin bir çıkış alanıdır ve MQPUT ve MQPUT1 çağrılarında ilişkin bir giriş alanı. Bu alanın ilk değeri MQPRI\_PRIORITY\_AS\_Q\_DEF olur.

### **Kalıcılık (MQUZE)**

Bu, iletinin kuyruk yöneticisinin sistem hatalarından ve yeniden başlatılıp yeniden başlatılıp başlatılmayacağını belirtir. MQPUT ve MQPUT1 çağrıları için, değerin aşağıdakilerden biri olması gerekir:

#### **MQPER\_PERISPER**

İleti, kuyruk yöneticisinin sistem hatalarından ve yeniden başlatılabilmesinin ardından devam eder. İleti konulduktan sonra, işlendiği iş birimi kesinleştirildikten sonra (ileti bir iş biriminin bir parçası olarak gönderildiyse), yardımcı saklama alanında saklanır. İleti kuyruktan kaldırılınca kadar ve içinde bulunduğu iş birimi kesinleştirilinceye kadar kalır (ileti, bir iş biriminin bir parçası olarak alındıysa).

Uzak bir kuyruğa kalıcı bir ileti gönderildiğinde, ileti bir sonraki kuyruk yöneticisine vardığı bilininceye kadar, saklama ve iletme mekanizmasının her bir kuyruk yöneticisinde hedefe giden yol boyunca bir ileti bulundurulur.

Kalıcı iletiler üzerine yerleştirilemiyor:

- Geçici dinamik kuyruklar
- CFLEVEL (2) ya da aşağısında bir CFSTRUCT nesnesiyle eşlenen ya da CFSTRUCT nesnesinin RECOVER (NO) olarak tanımlı olduğu kuyruklar paylaşılır.

Kalıcı iletiler kalıcı dinamik kuyruklara ve önceden tanımlanmış kuyruklara yerleştirilebilir.

#### **MQPER\_NOT\_PERSISTENT**

Bu ileti genellikle sistem hatalarından ya da kuyruk yöneticisi yeniden başlatılmadığından emin değildir. Bu, kuyruk yöneticisi yeniden başlatıldığında yardımcı belleğimde, iletinin el değmemiş bir kopyası bulunsa bile geçerlidir.

NPMCLASS (HIGH) kuyrukları kalıcı olmayan iletileri, olağan bir kuyruk yöneticisi kapanması ve yeniden başlatıp atlatılabilmesinse, devam eder.

Paylaşılan kuyruklar durumunda, kalıcı olmayan iletiler, kuyruk paylaşımı grubundaki kuyruk yöneticisinde yeniden başlar, ancak paylaşılan kuyruklarda iletileri saklamak için kullanılan bağlaşım tesisinin başarısızlıklarını atlatmaz.

#### **MQPER\_PERSISTENCE\_AS\_Q\_DEF**

- Kuyruk bir küme kuyruğıysa, iletinin kalıcılığı, iletinin yerleştiği kuyruğun belirli bir örneğinin sahibi olan *hedef* kuyruk yöneticisinde tanımlanan **DefPersistence** özneliğinden alınır.

Küme kuyruğunun birden çok eşgörünümü varsa ve bu öznelikte farklılık gösterdiklerinde, bunlardan birinin tane seçkullanım seçtiği değer tahmin edilemez. Bu nedenle, bu özneliği tüm eşgörünümelerde aynı değere ayarlamalısınız. Böyle bir durumda değilse, kuyruk yöneticisi günlüklerine AMQ9407 hata iletisi gönderilir. Ayrıca bkz. [Diğer adlar, uzak kuyruklar ve küme kuyrukları için hedef nesne öznelikleri nasıl çözülür?](#)

İleti hedef kuyruğa yerleştirildiğinde, *DefPersistence* değeri *Persistence* alanına kopyalanır. *DefPersistence* daha sonra değiştirilirse, kuyruğa önceden yerleştirmiş olan iletiler etkilenmez.

- If the queue is not a cluster queue, the persistence of the message is taken from the **DefPersistence** attribute defined at the *yere*l queue manager, even if the destination queue manager is remote.

Kuyruk adı çözümleme yolunda birden çok tanımlama varsa, yoldaki *ilk* tanımındaki bu özneliğin değerinden varsayılan kalıcılık alınır. Bu durumda şunlar olabilir:

- Diğer ad kuyruğu
- Yerel kuyruk
- Uzak kuyruğun yerel tanımlaması
- Kuyruk yöneticisi diğer adı
- Bir iletim kuyruğu (örneğin, *DefXmitQName* kuyruğu)

İleti konduğunda, *DefPersistence* değeri *Persistence* alanına kopyalanır. *DefPersistence* daha sonra değiştirilirse, önceden yerleştirmiş olan iletiler bundan etkilenmez.

Hem kalıcı hem de kalıcı olmayan iletiler aynı kuyruğun üzerinde var olabilir.

Bir iletiyi yanıtlarken, uygulamaların yanıt iletisi için istek iletisinin kalıcılığını kullanması gerekir.

MQGET çağrısı için döndürülen değer MQPER\_PERSISTENT ya da MQPER\_NOT\_PERSISTENT olur.

Bu, MQGET çağrısına ilişkin bir çıkış alanıdır ve MQPUT ve MQPUT1 çağrılarında ilişkin bir giriş alanı. Bu alanın ilk değeri MQPER\_PERSISTENCE\_AS\_Q\_DEF 'dir.

### **MsgId (MQBYTE24)**

Bu, bir iletiyi diğerinden ayırmak için kullanılan bayt dizilimidir. Genellikle, kuyruk yöneticisi tarafından izin verilmediği halde, iki ileti aynı ileti tanıtıcısına sahip olmalıdır. İleti tanıtıcısı, iletinin kalıcı bir özeldir ve kuyruk yöneticisinin yeniden başlatmalarında devam eder. İleti tanıtıcısı bir karakter dizisi değil, bir byte dizisi olduğu için, ileti tanıtıcısı, ileti bir kuyruk yöneticisinden diğerine akış sırasında karakter kümeleri arasında dönüştürülmez.

MQPUT ve MQPUT1 çağrıları için, uygulama tarafından MQMI\_NONE ya da MQPMO\_NEW\_MSG\_ID belirtilirse, kuyruk yöneticisi benzersiz bir ileti tanıtıcısı oluşturur.<sup>3</sup>İleti tanımlandığında ve iletiyi ileti tanımlayıcısına iletiyle birlikte gönderilen iletiyle yerleştirir. Kuyruk yöneticisi, gönderme uygulamasına ait olan ileti tanımlayıcısında bu ileti tanıtıcısını da döndürür. Uygulama, bu değeri belirli iletilere ilişkin bilgileri kaydetmek ve uygulamanın diğer kısımlarındaki sorgulara yanıt vermek için kullanabilir.

İleti bir konuya konursa, kuyruk yöneticisi yayınlanan her ileti için gerekli olduğu şekilde benzersiz ileti tanıtıcıları oluşturur. Uygulama tarafından MQPMO\_NEW\_MSG\_ID belirtilirse, kuyruk yöneticisi çıkışta döndürülebilme için benzersiz bir ileti tanıtıcısı oluşturur. Uygulama tarafından MQMI\_NONE belirtilirse, çağrıdan dönüşte MQMD ' deki *MsgId* alanının değeri değişmeden kalır.

Alıkonan yayınlara ilişkin ek bilgi için "[MQPMO seçenekleri \(MQUWT\)](#)" sayfa 497 içindeki MQPMO\_RETAIN tanımına bakın.

İleti bir dağıtım listesine konursa, kuyruk yöneticisi gereken şekilde benzersiz ileti tanıtıcıları oluşturur; ancak, MQMI\_NONE ya da MQPMO\_NEW\_MSG\_ID belirtilmişse de, MQMD ' deki *MsgId* alanının değeri

<sup>3</sup> Kuyruk yöneticisi tarafından üretilen bir *MsgId* değeri, 4 baytlık bir ürün tanıtıcısından (ASCII ya da CSQ; ASCII ya da EBCDIC ' de [ [ boş bir karakteri gösterir] ] ve onu, benzersiz bir dizginin ürüne özgü bir somutlamasını izler. IBM MQ ' da bu, kuyruk yöneticisi adının ilk 12 karakterini ve sistem saatinden türetilen bir değeri içerir. İletişim kurabilen tüm kuyruk yöneticilerinin, ileti tanıtıcılarının benzersiz olmasını sağlamak için, ilk 12 karakterde farklı olan adları olmalıdır. Benzersiz bir dizgi oluşturma yeteneği, sistem saatinin geriye doğru değiştirilmemesine de bağlıdır. Uygulama tarafından oluşturulan bir kuyruk yöneticisi tarafından oluşturulan bir ileti tanıtıcısının olasılığını ortadan kaldırmak için, uygulama ASCII ya da EBCDIC (X'41 '-X'49' ve X'C1'-X'C9') arasında A ile I arasındaki ilk karakterlere sahip tanıtıcıların oluşturulmaması gerekir. Ancak, uygulamanın bu aralıklarda başlangıç karakterleri olan tanıtıcılar oluşturmasını önlemez.

çağrıdan geri dönmeye değışmeden kalır. Uygulamanın kuyruk yöneticisi tarafından oluşturulan ileti tanıtıcılarını bilmesi gerekiyorsa, uygulamanın *MsgId* alanını içeren MQPMR kayıtlarını sağlamalıdır.

Gönderme uygulaması, MQMI\_NON; dışında ileti tanıtıcısı için bir deęer de belirtebilir; bu, kuyruk yöneticisinin benzersiz bir ileti tanıtıcısı oluşturmasını durdurur. İletiyi ileten bir uygulama, bu iletiyi özgün iletinin ileti tanıtıcısını yaymak için kullanabilir.

Kuyruk yöneticisi řu hariç olmak üzere bu alanı kullanmıyor:

- Yukarıda açıklandığı gibi, istenirse benzersiz bir deęer oluştur
- İletiyeye ilişkin alma isteğini içeren uygulamaya deęer sağlayın
- Deęeri, bu ileti hakkında oluşturduğu herhangi bir rapor iletinin *CorrelId* alanına kopyalayın (*Report* seçeneklerine baęlı olarak).

Kuyruk yöneticisi ya da bir ileti kanalı aracısı bir rapor ileti oluşturduğunda, *MsgId* alanını özgün iletinin *Report* alanı ile belirtilen şekilde ayarlar; MQRO\_NEW\_MSG\_ID ya da MQRO\_PASS\_MSG\_ID. Rapor iletileri oluşturan uygulamalar da bunu yapmalıdır.

MQGET çağırısına ilişkin *MsgId*, kuyruktan belirli bir iletiyi almak için kullanılacak beş alandan biridir. Olağan durumda MQGET çağırısı kuyruktan bir sonraki iletiyi döndürür; ancak, beş seçim ölçütünün biri ya da daha fazlası herhangi bir birleşimde belirtilerek belirli bir ileti alınabilir; bu alanlar şunlardır:

- *MsgId*
- *CorrelId*
- *GroupId*
- *MsgSeqNumber*
- *Offset*

Uygulama bu alanlardan birini ya da birkaçını gereken deęerlere ayarlar ve ardından, bu alanları seçim ölçütü olarak kullanmak için MQGMO 'daki *MatchOptions* alanındaki karşılık gelen MQMO\_\* eşleşme seçeneklerini ayarlar. Yalnızca, bu alanlarda belirtilen deęerlere sahip iletiler alma için adaydır. *MatchOptions* alanı için varsayılan deęer (uygulama tarafından deęiştirilmediyse) hem ileti tanıtıcısı, hem de ilinti tanıtıcısı ile eşleşir.

z/OS' ta, kullanabileceğiniz seçim ölçütleri, kuyruk için kullanılan izin tipi ile sınırlanır. Ek ayrıntılar için **IndexType** kuyruk özniteliğine bakın.

Olağan durumda, döndürülen ileti, kuyruğun seçim ölçütlerine uyan *ilk* iletidir. Ancak, MQGMO\_BROWSE\_NEXT belirtildiyse, döndürülen ileti seçim ölçütlerini karşılayan *sonraki* iletidir; bu iletinin taranması, imlecini yürürlükteki konumunu *izleyen* iletiyle başlar.

**Not:** Kuyruk, seçim ölçütlerini karşılayan bir ileti için sırayla taranır; bu nedenle alma süreleri, seçim ölçütü belirtilmediğinde daha yavaş olur; özellikle, uygun bir öge bulunmadan önce birçok iletinin taranması gerekir. Bunun istisnaları şunlardır:

- **Multi** an MQGET call by *CorrelId* on 64-bit Multiplatforms where the *CorrelId* index eliminates the need to perform a true sequential scan.
- **z/OS** an MQGET call by *IndexType* on z/OS.

Her iki durumda da alma başarımı iyileştirilir.

Seçim ölçütlerinin çeşitli durumlarda nasıl kullanılırla ilgili daha fazla bilgi için bkz. [Çizelge 496 sayfa 389](#).

İleti tanıtıcısı, MQMO\_MATCH\_MSG\_ID, yani, *herhangi biri* ileti tanıtıcısı eşleşmeleriyle aynı etkiye sahip olduğu için MQMI\_NONE deęerini belirtme.

MQGET çağırısının **GetMsgOpts** parametresinde MQGMO\_MSG\_UNDER\_CURSOR seçeneği belirtilirse bu alan yoksayılır.

MQGET çağırısından dönüşte, *MsgId* alanı, döndürülen iletinin ileti tanıtıcısına ayarlanır (varsa).

Aşağıdaki özel deęer kullanılabilir:

## **MQMI\_NONE**

İleti tanıtıcısı belirtilmedi.

Değer, alanın uzunluğuna ilişkin ikili sıfır değeridir.

C programlama dili için, sabit MQMI\_NONE\_DIZISI de tanımlanır; bu, MQMI\_NONE ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisidir.

Bu, MQGET, MQPUT ve MQPUT1 çağrılarında ilişkin bir giriş/çıkış alanıdır. Bu alanın uzunluğu MQ\_MSG\_ID\_LENGTH tarafından verilir. Bu alanın ilk değeri MQMI\_NONE olur.

## **CorrelId (MQBYTE24)**

CorrelId alanı, belirli bir iletiyi ya da ileti grubunu tanımlamak için kullanılabilecek ileti üstbilgisinde yer alan özelliktir.

Bu, uygulamanın bir iletiyi başka bir iletiyi ilişkilendirmek ya da iletiyi, uygulamanın gerçekleştirdiği diğer çalışmalarla ilişkilendirmek için kullanılabileceği bir bayt dizilimidir. İlinti tanıtıcısı, iletinin kalıcı bir özeldir ve kuyruk yöneticisinin yeniden başlatmalarında devam eder. İlinti tanıtıcısı bir karakter dizgisi değil, bir byte dizgisi olduğu için, ileti bir kuyruk yöneticisinden başka bir kuyruk yöneticisinden diğerine geçtiğinde ilinti tanıtıcısı karakter kümeleri arasında dönüştürülmez.

MQPUT ve MQPUT1 çağrıları için, uygulama herhangi bir değer belirtebilir. Kuyruk yöneticisi bu değeri iletiyle iletir ve iletiye ilişkin alma isteğini içeren uygulamaya iletir.

Uygulama MQPMO\_NEW\_CORREL\_ID ' yi belirtiyorsa, kuyruk yöneticisi iletiyle birlikte gönderilen benzersiz bir ilinti tanıtıcısı oluşturur ve MQPUT ya da MQPUT1 çağrısından çıkışta gönderme uygulamasına da döner.

Kuyruk yöneticisi tarafından oluşturulan bir ilinti tanıtıcısı, 3 baytlık bir ürün tanıtıcısından (ASCII ya da EBCDIC ' de AMQ ya da CSQ) oluşur; bunu, tek bir ayrılmış bayt ve benzersiz bir dizginin ürüne özgü bir somutlaması izler. IBM MQ içinde, bu ürüne özgü somutlama dizgisi kuyruk yöneticisi adının ilk 12 karakterini ve sistem saatinden türetilen bir değeri içerir. İletişim kurabilen tüm kuyruk yöneticilerinin, ileti tanıtıcılarının benzersiz olduğundan emin olmak için ilk 12 karakterde farklı adlar olması gerekir. Benzersiz bir dizgi oluşturma yeteneği, sistem saatinin geriye doğru değiştirilmemesine de bağlıdır. Uygulama tarafından oluşturulan bir kuyruk yöneticisi tarafından oluşturulan bir ileti tanıtıcısının olasılığını ortadan kaldırmak için, uygulama ASCII ya da EBCDIC (X'41 '-X'49' ve X'C1'-X'C9') arasında A ile I arasındaki ilk karakterlere sahip tanıtıcıların oluşturulamaması gerekir. Ancak, uygulamanın bu aralıklarda başlangıç karakterleri olan tanıtıcılar oluşturmasını önlemez.

Bu üretilen ilinti tanıtıcısı, saklandıysa iletiyle birlikte tutulur ve MQSUB çağrısına geçirilen MQSD ' deki SubCorrelId alanında MQCI\_NONE değerini belirten abonelere bir yayın olarak gönderildiğinde ilinti tanıtıcısı olarak kullanılır. Alıkonan yayınlarla ilgili ek ayrıntılar için [MQPMO seçenekleri başlıklı konuya](#) bakın.

Kuyruk yöneticisi ya da bir ileti kanalı aracısı bir rapor iletisi oluşturduğunda, *CorrelId* alanını, özgün iletinin *Report* alanı tarafından belirtilen şekilde ayarlar; MQRO\_COPY\_MSG\_ID\_TO\_CORREL\_ID ya da MQRO\_PASS\_COREL\_ID. Rapor iletileri oluşturan uygulamalar da bunu yapmalıdır.

MQGET çağrısına ilişkin *CorrelId* , kuyruktan alınacak belirli bir iletiyi seçmek için kullanılabilecek beş alandan biridir. Bu alana ilişkin değerlerin nasıl belirtileceğini görmek için, *MsgId* alanının açıklamasına bakın.

İlinti tanıtıcısı olarak MQCI\_NONE değerinin belirlenmesi MQMO\_MATCH\_COREL\_ID belirtmeleriyle aynı etkiye sahiptir; bu, *herhangi bir* ilinti tanıtıcısının eşleşmesine neden olur.

MQGET çağrısındaki **GetMsgOpts** parametresinde MQGMO\_MSG\_UNDER\_CURSOR seçeneği belirtilirse, bu alan yoksayılır.

MQGET çağrısından dönüşte, *CorrelId* alanı, döndürülen iletinin ilinti tanıtıcısına (varsa) ayarlanır.

Aşağıdaki özel değerler kullanılabilir:

## **MQCI\_NONE**

İlinti tanıtıcısı belirtilmedi.

Değer, alanın uzunluğuna ilişkin ikili sıfır değeridir.

C programlama dili için, sabit MQCI\_NONE\_DIZISI de tanımlanır; bu, MQCI\_NONE ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisidir.

### **MQCI\_NEW\_SESSION**

İleti, yeni bir oturumun başlangıcıdır.

Bu değer, CICS bridge tarafından yeni bir oturumun başlangıcının, yani yeni bir ileti dizisinin başlangıcının belirtildiği gibi tanınır.

C programlama dili için, sabit MQCI\_NEW\_SESSION\_ARRAY değeri de tanımlanır; bu, MQCI\_NEW\_SESSION ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisi olur.

MQGET çağrısına ilişkin bu bir giriş/çıkış alanıdır. MQPUT ve MQPUT1 çağrıları için, MQPMO\_NEW\_CORREL\_ID belirlenmediyse, bu giriş alanıdır ve MQPMO\_NEW\_CORREL\_ID belirtildiyse bir çıkış alanı olur. Bu alanın uzunluğu MQ\_CORIL\_ID\_LENGTH tarafından verilir. Bu alanın ilk değeri MQCI\_NONE olur.

### **Not:**

Bir yayının bir sıradüzeninde ilinti tanıtıcısını geçemezsiniz. Alan, kuyruk yöneticisi tarafından kullanılır.

### **BackoutCount (MQLONG)**

Bu, iletinin daha önce MQGET çağrısının bir iş biriminin bir parçası olarak geri döndürüldüğü ve daha sonra geriletildiği sayıların sayısıdır. Bu, uygulamanın ileti içeriğine dayalı işleme hatalarını saptamasına yardımcı olur. Bu sayma, MQGMO\_BROWSE\_\* seçeneklerinden herhangi birini belirten MQGET çağrılarını dışlar.

Bu sayımın doğruluğu, **HardenGetBackout** kuyruk özniteliden etkilenir; bkz. "[Kuyruklara ilişkin öznitelikler](#)" sayfa 821.

z/OSüzerinde 255 değeri, iletinin 255 ya da daha fazla kez geriletileceği anlamına gelir; döndürülen değer 255 'ten büyük bir değer değildir.

Bu, MQGET çağrısına ilişkin bir çıkış alanıdır. Bu, MQPUT ve MQPUT1 çağrıları için yoksayıdır. Bu alanın başlangıç değeri 0' dir.

### **ReplyToQ (MQCHAR48)**

Bu, ileti için alma isteğini yayınlayan uygulamanın MQMT\_REPLY ve MQMT\_REPORT iletilerini göndereceği ileti kuyruğunun adıdır. Ad, *ReplyToQMgr* ile tanımlanan kuyruk yöneticinde tanımlı olan bir kuyruğun yerel adıdır. Kuyruk yöneticisi, ileti konduğunda bunu doğrulamasına da, bu kuyruk bir model kuyruğu olmamalıdır.

MQPUT ve MQPUT1 çağrıları için, *MsgType* alanının MQMT\_REQUEST değeri varsa ya da *Report* alanı tarafından herhangi bir rapor iletisi isteniyorsa bu alan boş olmamalıdır. Ancak, belirtilen değer (ya da yerine koyulan), ileti tipi ne olursa olsun, iletiye ilişkin alma isteğini içeren uygulamaya geçirilir.

*ReplyToQMgr* alanı boşsa, yerel kuyruk yöneticisi *ReplyToQ* adını kendi kuyruk tanımlamalarında arar. Bu adı taşıyan bir uzak kuyruğun yerel tanımlaması varsa, iletilen iletteki *ReplyToQ* değeri, uzak kuyruğun tanımlamasındaki **RemoteQName** özniteliğinin değeriyle değiştirilir ve alıcı uygulama ileti için bir MQGET çağrısı yayınlarken ileti tanımlayıcısında bu değer döndürülür. Uzak kuyruğun yerel tanımlaması yoksa, *ReplyToQ* değişmez.

Ad belirlendiyse, sondaki boşlukları da içerebilir; ilk boş karakter ve izleyen karakterler boşluk olarak işlenir. Ters durumda, adın kuyruklar için adlandırma kurallarını karşıladığını doğrulayın; *ReplyToQ*, iletilen iletide değiştirilirse, iletilen ad için de geçerli olur. Yapılan tek denetim, koşul gerektiriyorsa, bir ad belirtilmesine neden olur.

Bir yanıt kuyruğu gerekli değilse, *ReplyToQ* alanını boşluk ya da (C programlama dilinde) boş değerli dizgiye ya da boş değerli bir karakter izleyen bir ya da daha çok boşluğa ayarlayın; alanı kullanıma hazırlıksız bırakmayın.

MQGET çağrısı için, kuyruk yöneticisi her zaman boşluklarla doldurulan adı, alanın uzunluğuna kadar döndürür.



Bir rapor iletisi gerektiren bir ileti teslim edilemezse ve rapor iletisi de belirtilen kuyruğa teslim edilemezse, hem özgün ileti, hem de rapor iletisi ölü-mektup (teslim edilemeyen ileti) kuyruğuna gider ( "Kuyruk yöneticisine ilişkin öznitelikler" sayfa 785 ' ta açıklanan **DeadLetterQName** özniteliğini göreceldir).

Bu, MQGET çağrısına ilişkin bir çıkış alanıdır ve MQPUT ve MQPUT1 çağrılarında ilişkin bir giriş alanı. Bu alanın uzunluğu MQ\_Q\_NAME\_LENGTH değeri tarafından verilir. Bu alanın başlangıç değeri, C ' de boş değer dizgisidir ve diğer programlama dillerindeki 48 boş karakter dizilimidir.

### **ReplyToQMgr (MQCHAR48)**

Bu ad, yanıt iletisinin ya da rapor iletisinin gönderileceği kuyruk yöneticisinin adıdır. *ReplyToQ* , bu kuyruk yöneticisinde tanımlı olan bir kuyruğun yerel adıdır.

*ReplyToQMgr* alanı boşsa, yerel kuyruk yöneticisi, kuyruk tanımlamalarında *ReplyToQ* adını arar. Bu adı taşıyan bir uzak kuyruğun yerel tanımlaması varsa, iletilen iletteki *ReplyToQMgr* değeri, uzak kuyruğun tanımlamasındaki **RemoteQMgrName** özniteliğinin değeriyle değiştirilir ve alıcı uygulama ileti için bir MQGET çağrısı yayınlarken ileti tanımlayıcısında bu değer döndürülür. Uzak kuyruğun yerel tanımlaması yoksa, iletiyle aktarılan *ReplyToQMgr* değeri yerel kuyruk yöneticisinin adıdır.

Ad belirlendiyse, sondaki boşlukları da içerebilir; ilk boş karakter ve izleyen karakterler boşluk olarak işlenir. Ters durumda, adın kuyruk yöneticilerine ilişkin adlandırma kurallarını karşıladığını ya da bu adın gönderen kuyruk yöneticisi tarafından tanındığını doğrulayın; bu değer, iletilen iletide *ReplyToQMgr* ' in yerine geçiyorsa, iletilen ad için de geçerlidir.

Bir yanıt kuyruğu gerekli değilse, *ReplyToQMgr* alanını boşluk ya da (C programlama dilinde) boş değerli dizgiye ya da boş değerli bir karakter izleyen bir ya da daha çok boşluğa ayarlayın; alanı kullanıma hazırlıksız bırakmayın.

MQGET çağrısı için, kuyruk yöneticisi her zaman boşluklarla doldurulan adı, alanın uzunluğuna kadar döndürür.

Bu, MQGET çağrısına ilişkin bir çıkış alanıdır ve MQPUT ve MQPUT1 çağrılarında ilişkin bir giriş alanı. Bu alanın uzunluğu MQ\_Q\_MGR\_NAME\_LENGTH değeri tarafından verilir. Bu alanın başlangıç değeri, C ' de boş değer dizgisidir ve diğer programlama dillerindeki 48 boş karakter dizilimidir.

### **UserIdentifier (MQCHAR12)**

Bu, iletinin **kimlik bağlamının** bir parçasıdır. İleti bağlamıyla ilgili daha fazla bilgi için bkz. "MQMD-İleti tanımlayıcı" sayfa 416 ve İleti bağlamı.

*UserIdentifier* , iletiyi oluşturan uygulamanın kullanıcı kimliğini belirtir. Kuyruk yöneticisi bu bilgileri karakter verileri olarak kabul eder, ancak biçimini tanımlamaz.

Bir ileti alındıktan sonra, sonraki bir MQOPEN ya da MQPUT1 çağrısının **ObjDesc** değiştirgesinin *AlternateUserId* alanında *UserIdentifier* komutunu kullanarak, açma işlemini gerçekleştiren uygulama yerine *UserIdentifier* kullanıcısı için yetki denetimi gerçekleştirebilirsiniz.

Kuyruk yöneticisi bir MQPUT ya da MQPUT1 çağrısı için bu bilgileri ürettiğinde:

- z/OS sistemlerinde, MQOO\_ALTERNATE\_USER\_AUTHORITY ya da MQPMO\_ALTERNATE\_USER\_AUTHORITY seçeneği belirtildiyse, kuyruk yöneticisi MQOPEN ya da MQPUT1 çağrısının **ObjDesc** değiştirgesinden *AlternateUserId* komutunu kullanır. İlgili seçenek belirtilmediyse, kuyruk yöneticisi ortamdan belirlenen bir kullanıcı kimliğini kullanır.
- Diğer ortamlarda, kuyruk yöneticisi her zaman ortamdan belirlenen bir kullanıcı kimliğini kullanır.

Kullanıcı kimliği ortamdan belirlendiğinde:

- z/OS' da, kuyruk yöneticisi aşağıdakileri kullanır:
  - MVS (toplu) için, JES JOB kartındaki ya da başlatılan görevdeki kullanıcı kimliği
  - TSO için, kullanıcı kimliği iş sunulurken işe yayıldı
  - CICS için, görevle ilişkili kullanıcı kimliği
  - IMS için, kullanıcı kimliği uygulamanın tipine bağlıdır:

- Bunun için:
  - İleti dışı BMP bölgeleri
  - İleti dışı IFP bölgeleri
  - Başarılı bir GU çağrısı yayınlamamış olan BMP ve ileti IFP bölgeleri

Kuyruk yöneticisi, bölge JES JOB kartındaki kullanıcı kimliğini ya da TSO kullanıcı kimliğini kullanır. Bunlar boş değerli ya da boşsa, program belirtimi öbeğinin (PSB) adını kullanır.

- Bunun için:
  - 'in başarılı bir GU çağrısı verdiği BMP ve ileti IFP bölgeleri
  - MPP bölgeleri

Kuyruk yöneticisi aşağıdakilerden birini kullanır:

- İletiyile ilişkili oturum açmış kullanıcı kimliği
- Mantıksal uçbirim (LTERM) adı
- Bölge JES JOB kartındaki kullanıcı kimliği
- TSO kullanıcı kimliği
- PCB adı
- IBM i' da, kuyruk yöneticisi uygulama işiyle ilişkilendirilmiş kullanıcı tanıtımının adını kullanır.
- AIX and Linux' da, kuyruk yöneticisi aşağıdakileri kullanır:
  - Uygulamanın oturum açma adı
  - Oturum açma yoksa, işlemin etkin kullanıcı kimliği
  - Uygulama bir CICS işlemiyse, işlemle ilişkili kullanıcı kimliği
- Windows sistemlerinde, kuyruk yöneticisi oturum açmış kullanıcı adının ilk 12 karakterini kullanır.

Bu alan normalde kuyruk yöneticisi tarafından üretilen bir çıkış alanıdır; ancak bir MQPUT ya da MQPUT1 çağrısı için, bu alanı giriş/çıkış alanı yapabilir ve kuyruk yöneticisinin bu bilgileri oluşturmaya izin vermek yerine UserIdentification (kullanıcı kimliği) alanını belirtebilirsiniz. PutMsgdeğiştirgesinde MQPMO\_SET\_IDENTITY\_CONTEXT ya da MQPMO\_SET\_ALL\_CONTEXT belirtin ve kuyruk yöneticisinin bir MQPUT ya da MQPUT1 çağrısı için UserIdentifier alanını oluşturmamasını istemiyorsanız, UserIdentifier alanında bir kullanıcı kimliği belirtin.

MQPUT ve MQPUT1 çağrıları için, **PutMsgOpts** değiştirgesinde MQPMO\_SET\_IDENTITY\_CONTEXT ya da MQPMO\_SET\_ALL\_CONTEXT belirtildiyse, bu bir giriş/çıkış alanıdır. Alan içinde boş bir karakteri izleyen bilgiler atılır. Kuyruk yöneticisi boş karakteri ve aşağıdaki karakterleri boşluklara dönüştürür. MQPMO\_SET\_IDENTITY\_CONTEXT ya da MQPMO\_SET\_ALL\_CONTEXT belirtilmezse, girişte bu alan yoksayılr ve yalnızca çıkış alanıdır.

Bir MQPUT ya da MQPUT1 çağrısı başarıyla tamamlandıktan sonra, bu alan bir kuyruğa yerleştirildiyse, iletiyle birlikte iletilen *UserIdentifier* ögesini içerir. Bu, alıkonan yayınlarla ilgili daha fazla ayrıntı için iletiyle birlikte tutulan *UserIdentifier* değeri olur (MQPMO\_RETAIN tanımına bakın), ancak ileti, abonelere gönderilen tüm yayınlarda *UserIdentifier* değerini geçersiz kılacak bir değer sağladığından, abonelere yayın olarak gönderildiğinde *UserIdentifier* olarak kullanılmaz. İletinin bağlamı yoksa, alan tamamen boştur.

Bu, MQGET çağrısına ilişkin bir çıkış alanıdır. Bu alanın uzunluğu MQ\_USER\_ID\_LENGTH tarafından verilir. Bu alanın ilk değeri C ' deki boş dizgi ve diğer programlama dillerindeki 12 boş karakterdir.

### **AccountingToken (MQBYTE32)**

Bu, iletinin *kimlik bağlamının* bir parçası olan muhasebe simgesidir. İleti bağlamıyla ilgili daha fazla bilgi için bkz. [“MQMD-İleti tanımlayıcı” sayfa 416](#) ; Ayrıca bkz. [İleti bağlamı](#).

AccountingToken , bir uygulamanın ileti sonucu olarak yapılan iş için uygun şekilde ücretlendirme yapmasını sağlar. Kuyruk yöneticisi bu bilgileri bir bit dizgisi olarak işler ve içeriğini denetlemez.

Kuyruk yöneticisi bu bilgileri aşağıdaki gibi oluşturur:



- Alanın ilk baytı, izleyen baytlarda var olan muhasebe bilgilerinin uzunluğuna ayarlanır; bu uzunluk, sıfır ile 30 aralığındadır ve ilk baytta ikili tamsayı olarak saklanır.
- İkinci ve sonraki baytlar (uzunluk alanı tarafından belirlendiği şekilde), ortama uygun muhasebe bilgilerine ayarlanır.

– **z/OS** z/OS ' de muhasebe bilgileri aşağıdaki şekilde ayarlanır:

- z/OS toplu iş için, JES JOB kartındaki ya da EXEC kartındaki bir JES ACCT deyimindeki muhasebe bilgileri (virgül ayırıcıları X'FF ' olarak değiştirilir). Gerekirse, bu bilgiler 31 bayta kesilir.
- TSO için, kullanıcının hesap numarası.
- CICS için, LU 6.2 iş birimi tanıtıcısı (UEPUOWDS) (26 bayt).
- IMS için, 8 karakterlik PSB adı 16 karakterlik IMS kurtarma simgesiyle bitştirilir.

– **IBM i** IBM işletim tarihinde, muhasebe bilgileri için muhasebe koduna ayarlanır.

– **Linux** **AIX** AIX and Linux' da, muhasebe bilgileri ASCII karakterlerinde sayısal kullanıcı kimliğine ayarlanır.

– **Windows** Windows üzerinde, muhasebe bilgileri sıkıştırılmış biçimde bir Windows güvenlik tanıtıcısına (SID) ayarlanır. SID, *UserIdentifier* alanında saklanan kullanıcı kimliğini benzersiz olarak tanımlar. SID *AccountingToken* alanında saklandığında, 6 baytlık Tanıtıcı Yetkisi (SID ' nin üçüncü ve sonraki baytlarında bulunur) atanır. Örneğin, Windows SID 28 bayt uzunluğdaysa, SID bilgilerinin 22 baytı *AccountingToken* alanında saklanır.

- Muhasebe alanının son baytı (bayt 32), muhasebe simgesi tipine ayarlanır (bu durumda MQACTT\_NT\_SECURITY\_ID, x '0b'):

#### **MQACTT\_CICS\_LUOW\_ID**

CICS LUOW tanıtıcısı.

– **Windows** **MQACTT\_NT\_SECURITY\_ID**

Windows güvenlik tanıtıcısı.

– **IBM i** **MQACTT\_OS400\_ACCOUNT\_TOKEN**

IBM i hesap simgesi.

– **UNIX** **MQACTT\_UNIX\_NUMERIC\_ID**

UNIX sayısal tanıtıcısı.

#### **MQACTT\_USER**

Kullanıcı tanımlı hesap simgesi.

#### **MQACTT\_UNKNOWN**

Bilinmeyen muhasebe simgesi tipi.

Muhasebe simgesi tipi, yalnızca aşağıdaki ortamlarda belirttik bir değere ayarlanır:

– **AIX** AIX

– **IBM i** IBM i

– **Linux** Linux

– **Windows** Windows

ve bu sistemlere bağlı IBM MQ MQI clients için. Diğer ortamlarda, muhasebe simgesi tipi MQACTT\_UNKNOWN değerine ayarlanır. Bu ortamlarda, alınan muhasebe simgesi tipini çıkarmak için *PutAppType* alanını kullanın.

- Diğer tüm baytlar ikili sıfır olarak ayarlanır.

MQPUT ve MQPUT1 çağrılarını için, **PutMsgOpts** değiştirilmesinde MQPMO\_SET\_IDENTITY\_CONTEXT ya da MQPMO\_SET\_ALL\_CONTEXT belirtildiyse, bu bir giriş/çıkış alanıdır. MQPMO\_SET\_IDENTITY\_CONTEXT ya

da MQPMO\_SET\_ALL\_CONTEXT belirtilmezse, girişte bu alan yoksayılr ve yalnızca çıkış alanıdır. İleti bağlamıyla ilgili daha fazla bilgi için bkz. [İleti bağlamı](#).

Bir MQPUT ya da MQPUT1 çağrısı başarıyla tamamlandıktan sonra, bu alan bir kuyruğa yerleştirildiyse, iletiyle birlikte iletilen AccountingToken ögesini içerir. Bu değer, alıkonursa iletiyle birlikte tutulan AccountingToken değeridir (alıkonan yayınlarla ilgili daha fazla ayrıntı için "[MQPMO seçenekleri \(MQUWT\)](#)" sayfa 497 içindeki MQPMO\_RETAIN tanımına bakın), ancak ileti, abonelere gönderilen tüm yayınlarda AccountingToken değerini geçersiz kılacak bir değer sağladıkları için yayın olarak gönderildiğinde AccountingToken olarak kullanılmaz. İletinin bağlamı yoksa, alan tamamen ikili sıfırdır.

Bu, MQGET çağrısına ilişkin bir çıkış alanıdır.

Bu alan, kuyruk yöneticisinin karakter kümesine dayalı olarak herhangi bir çeviriye tabi değildir; alan, karakter dizgisi olarak değil, bit dizgisi olarak işlenir.

Kuyruk yöneticisi bu alandaki bilgilerle hiçbir şey yapmaz. Bilgileri muhasebe amacıyla kullanmak istiyorsa, uygulama bilgisini yorumlamalıdır.

AccountingToken alanı için aşağıdaki özel değeri kullanabilirsiniz:

#### **MQACT\_NONE**

Hesap simgesi belirtilmedi.

Değer, alanın uzunluğu için ikili sıfırdır.

C programlama dili için, MQACT\_NONE\_ARRAY değişmezi de tanımlanır; bu değer MQACT\_NONE ile aynıdır, ancak dizgi yerine bir karakter dizisidir.

Bu alanın uzunluğu MQ\_ACCOUNTING\_TOKEN\_LENGTH tarafından verilir. Bu alanın ilk değeri MQACT\_NONE.

#### **ApplIdentityVerileri (MQCHAR32)**

Bu, iletinin **kimlik bağlamının** bir parçasıdır. İleti bağlamıyla ilgili daha fazla bilgi için bkz. "[MQMD-İleti tanımlayıcı](#)" sayfa 416 ve [İleti bağlamı](#).

*ApplIdentityData*, uygulama grubu tarafından tanımlanan bilgilerdir ve ileti ya da oluşturucusu hakkında ek bilgi sağlamak için kullanılabilir. Kuyruk yöneticisi bu bilgileri karakter verileri olarak kabul eder, ancak biçimini tanımlamaz. Kuyruk yöneticisi bu bilgileri oluşturduğunda tamamen boş olur.

MQPUT ve MQPUT1 çağrıları için, **PutMsgOpts** değiştirilmesinde MQPMO\_SET\_IDENTITY\_CONTEXT ya da MQPMO\_SET\_ALL\_CONTEXT belirtildiyse, bu bir giriş/çıkış alanıdır. Boş bir karakter varsa, boş değer ve aşağıdaki karakterler kuyruk yöneticisi tarafından boşluklara dönüştürülür. MQPMO\_SET\_IDENTITY\_CONTEXT ya da MQPMO\_SET\_ALL\_CONTEXT belirtilmezse, girişte bu alan yoksayılr ve yalnızca çıkış alanıdır. İleti bağlamıyla ilgili daha fazla bilgi için bkz. [İleti bağlamı](#).

Bir MQPUT ya da MQPUT1 çağrısı başarıyla tamamlandıktan sonra, bu alan bir kuyruğa yerleştirildiyse, iletiyle birlikte iletilen *ApplIdentityData* ögesini içerir. Bu, alıkonan yayınlarla ilgili daha fazla ayrıntı için iletiyle birlikte tutulan *ApplIdentityData* değeri olur (MQPMO\_RETAIN tanımına bakın), ancak ileti, abonelere gönderilen tüm yayınlarda *ApplIdentityData* değerini geçersiz kılacak bir değer sağladığından, abonelere yayın olarak gönderildiğinde *ApplIdentityData* olarak kullanılmaz. İletinin bağlamı yoksa, alan tamamen boştur.

Bu, MQGET çağrısına ilişkin bir çıkış alanıdır. Bu alanın uzunluğu MQ\_APPL\_IDENTITY\_DATA\_LENGTH tarafından verilir. Bu alanın ilk değeri, C içindeki boş dizgi ve diğer programlama dillerinde 32 boş karakterdir.

#### **PutApplTipi (MQLONG)**

Bu, iletiyi koyan uygulamanın tipidir ve iletinin **kaynak bağlamının** bir parçasıdır. İleti bağlamıyla ilgili daha fazla bilgi için bkz. "[MQMD-İleti tanımlayıcı](#)" sayfa 416 ve [İleti bağlamı](#).

*PutApplType* aşağıdaki standart tiplerden birine sahip olabilir. Kendi tiplerinizi de tanımlayabilirsiniz, ancak yalnızca MQAT\_USER\_LAST ile MQAT\_USER\_FIRST aralığındaki değerlerle tanımlayabilirsiniz.

**MQAT\_AIX**

AIX uygulaması (MQAT\_UNIX ile aynı deęer).

**MQAT\_AMQP**

AMQP iletiřim kuralı uygulaması

**MQAT\_BROKER**

Borsacı.

**MQAT\_CICS**

CICS hareketi.

**MQAT\_CICS\_BRIDGE**

CICS bridge.

**MQAT\_CICS\_VSE**

CICS/VSE hareketi.

**MQAT\_DOS**

PC DOS üzerinde IBM MQ MQI client uygulaması.

**MQAT\_DQM**

Daęıtılmıř kuyruk yöneticisi aracı.

**MQAT\_GUARDIAN**

Tandem Guardian uygulaması (MQAT\_NSK ile aynı deęer).

**MQAT\_IMS**

IMS uygulaması.

**MQAT\_IMS\_BRIDGE**

IMS köprüsü.

**MQAT\_JAVA**

Java.

**MQAT\_MVS**

MVS ya da TSO uygulaması (MQAT\_ZOS ile aynı deęer).

**MQAT\_NOTES\_AGENT**

Lotus Notes Agent uygulaması.

**MQAT\_OS390**

OS/390 uygulaması (MQAT\_ZOS ile aynı deęer).

**MQAT\_OS400**

IBM i uygulaması.

**MQAT\_QMGR**

Kuyruk yöneticisi.

**MQAT\_UNIX**

UNIX uygulaması.

**MQAT\_VOS**

Stratus VOS uygulaması.

**MQAT\_WINDOWS**

16 bit Windows uygulaması.

**MQAT\_WINDOWS\_NT**

32 bit Windows uygulaması.

**MQAT\_WLM**

z/OS iř yükü yöneticisi uygulaması.

**MQAT\_XCF**

-XCF.

**MQAT\_ZOS**

z/OS uygulaması.

**MQAT\_DEFAULT**

Varsayılan uygulama tipi.

Bu, uygulamanın çalıştığı altyapıya ilişkin varsayılan uygulama tipidir.

**Not:** Bu değişimin değeri ortama özgüdür. Bu nedenle, her zaman üstbilgiyi kullanarak uygulamayı derleyin, uygulamanın çalışacağı platforma uygun dosyaları ekleyin ya da kopyalayın ekleyin.

### **MQAT\_BILINMIYOR**

Diğer bağlam bilgileri var olsa da uygulama tipinin bilinmediğini belirtmek için bu değeri kullanın.

### **MQAT\_USER\_FIRST**

Kullanıcı tanımlı uygulama tipi için en düşük değer.

### **MQAT\_USER\_LAST**

Kullanıcı tanımlı uygulama tipi için en yüksek değer.

Aşağıdaki özel değer de oluşabilir:

### **MQAT\_NO\_CONTEXT**

Bu değer, bağlam olmadan bir ileti konduğunda (yani, MQPMO\_NO\_CONTEXT bağlam seçeneği belirtildiğinde) kuyruk yöneticisi tarafından ayarlanır.

Bir ileti alındığında, iletinin bağlamı olup olmadığına karar vermek için *PutApplType* sınanabilir (diğer bağlam alanlarından herhangi biri boş değilse, MQPMO\_SET\_ALL\_CONTEXT kullanan bir uygulama tarafından *PutApplType* ' un hiçbir zaman MQAT\_NO\_CONTEXT olarak ayarlanmaması önerilir).

Kuyruk yöneticisi bu bilgileri bir uygulama koymasının sonucu olarak oluşturduğunda, alan ortam tarafından belirlenen bir değere ayarlanır. IBM üzerinde MQAT\_OS400; kuyruk yöneticisi IBM üzerinde MQAT\_CICS kullanmaz.

MQPUT ve MQPUT1 çağrılar için, **PutMsgOpts** değiştirilmesinde MQPMO\_SET\_ALL\_CONTEXT belirtilirse bu bir giriş/çıkış alanıdır. MQPMO\_SET\_ALL\_CONTEXT belirtilmezse, girişte bu alan yoksayılar ve yalnızca çıkış alanıdır.

Bu, MQGET çağrısına ilişkin bir çıkış alanıdır. Bu alanın ilk değeri MQAT\_NO\_CONTEXT 'dir.

### **PutApplAdı (MQCHAR28)**

Bu, iletiyi koyan uygulamanın adıdır ve iletinin *kaynak bağlamının* bir parçasıdır. İçerik platformlar arasında farklılık gösterir ve yayınlar arasında farklılık gösterebilir.

İleti bağlamıyla ilgili daha fazla bilgi için bkz. “MQMD-İleti tanımlayıcı” sayfa 416 ve [İleti bağlamı](#).

**V 9.2.0** IBM MQ 9.1.2 ' den ek programlama dillerinde uygulama adını belirtebilirsiniz. Ek bilgi için [Desteklenen programlama dillerinde uygulama adının belirtilmesi](#) başlıklı konuya bakın.

*PutApplName* biçimi *PutApplType* değerine bağlıdır ve bir yayın düzeyinden diğerine değişebilir. Değişiklikler nadirdir, ancak ortam değişirse gerçekleşir.

Kuyruk yöneticisi bu alanı (MQPMO\_SET\_ALL\_CONTEXT dışındaki tüm seçenekler için) ayarladığında, alanı ortamın belirlediği bir değere ayarlar:

- **z/OS** z/OS' da, kuyruk yöneticisi aşağıdakileri kullanır:
  - z/OS toplu iş için, JES JOB kartındaki 8 karakterlik iş adı
  - TSO için 7 karakterlik TSO kullanıcı kimliği
  - CICS için, 8 karakterli applid, ardından 4 karakterli tranid
  - IMS için, 8 karakterlik IMS sistem tanıtıcısı ve ardından 8 karakterlik PSB adı
  - XCF için, 8 karakterlik XCF grubu adı ve ardından 16 karakterlik XCF üye adı
  - Kuyruk yöneticisi tarafından oluşturulan bir ileti için, kuyruk yöneticisi adının ilk 28 karakteri
  - CICS olmadan dağıtılmış kuyruğa alma için, kanal başlatıcısının 8 karakterlik iş adı ve ardından, teslim edilmeyen iletiler kuyruğuna yerleştirilen modülün 8 karakterlik adı ve ardından 8 karakterlik bir görev tanıtıcısı gelir.

Ad ya da adların her biri, alanın geri kalanında olduğu gibi, sağda boşluklarla doldurulur. Birden çok ad varsa, aralarında ayırıcı yoktur.

- **Windows** Windows sistemlerinde kuyruk yöneticisi aşağıdaki adları kullanır:
  - CICS uygulaması için CICS işlem adı
  - CICS dışı bir uygulama için, yürütülür dosyanın tam olarak nitelenmiş adının en sağındaki 28 karakter
- **IBM i** IBM i' de, kuyruk yöneticisi tam olarak nitelenmiş iş adını kullanır.
- **Linux** **AIX** AIX and Linux üzerinde, kuyruk yöneticisi aşağıdaki adları kullanır:
  - CICS uygulaması için CICS işlem adı
  - CICS dışı bir uygulama için, MQ işletim sisteminden işlemin adını ister. Bu, tam yol olmadan program dosyası adı olarak döndürülür. Daha sonra MQ , bu süreç adını MQMD.PutApplName alanı aşağıdaki gibidir:

#### **AIX** **AIX**

Ad 28 byte 'tan küçük ya da ona eşitse, ad boşluklarla sağa doldurulur.

Ad 28 bayttan büyükse, adın en soldaki 28 baytı eklenir.

#### **Linux** **Linux**

Ad 15 byte 'tan küçük ya da ona eşitse, ad boşluklarla sağa doldurulur.

Ad 15 bayttan büyükse, adın en solundaki 15 bayt araya eklenir ve sağa boşluklarla doldurulur.

Örneğin, /opt/mqm/samp/bin/amqspud QNAME QMNAME çalıştırırsanız, PutApplAdı 'amqspud' olur. Bu MQCHAR28 alanında 21 boşluk karakteri vardır. /opt/mqm/samp/bin içeren tam yolun PutApplAdına eklenmediğini unutmayın.

MQPUT ve MQPUT1 çağrılarını için, **PutMsgOpts** değiştirilmesinde MQPMO\_SET\_ALL\_CONTEXT belirtilirse bu bir giriş/çıkış alanıdır. Alan içinde boş bir karakteri izleyen bilgiler atılır. Boş karakter ve aşağıdaki karakterler, kuyruk yöneticisi tarafından boşluklara dönüştürülür. MQPMO\_SET\_ALL\_CONTEXT belirtilmezse, girişte bu alan yoksayılır ve yalnızca çıkış alanıdır.

### **PutDate (MQCHAR8)**

Bu, iletinin konma tarihidir ve iletinin **kaynak bağlamının** bir parçasıdır. İleti bağlamıyla ilgili daha fazla bilgi için bkz. “MQMD-İleti tanımlayıcı” sayfa 416 ve İleti bağlamı.

Bu alanın kuyruk yöneticisi tarafından oluşturulduğu tarih için kullanılan biçim:

- YYYYAAAGG

karakterlerin temsil ettiği yer:

#### **YYYY**

yıl (dört sayısal basamak)

#### **MM**

yılın ayı (01-12)

#### **GG**

ayın günü (01-31)

Greenwich Ortalama Saati (GMT), sistem saatinin GMT ' ye doğru şekilde ayarlanmasına bağlı olarak *PutDate* ve *PutTime* alanları için kullanılır.

İleti bir iş biriminin parçası olarak konduysa, tarih, iş biriminin kesinleştirildiği tarih değil, iletinin konduğu tarihtir.

MQPUT ve MQPUT1 çağrılarını için, **PutMsgOpts** değiştirilmesinde MQPMO\_SET\_ALL\_CONTEXT belirtilirse bu bir giriş/çıkış alanıdır. Alanın içeriği kuyruk yöneticisi tarafından denetlenmez; ancak, alan içindeki boş karakterin ardından gelen bilgiler atılır. Kuyruk yöneticisi boş karakteri ve aşağıdaki karakterleri boşluklara dönüştürür. MQPMO\_SET\_ALL\_CONTEXT belirtilmezse, girişte bu alan yoksayılır ve yalnızca çıkış alanıdır.

Bu, MQGET çağrısına ilişkin bir çıkış alanıdır. Bu alanın uzunluğu MQ\_PUT\_DATE\_LENGTH tarafından verilir. Bu alanın ilk değeri, C içindeki boş dizgi ve diğer programlama dillerindeki 8 boş karakterdir.

## ***PutTime (MQCHAR8)***

İletinin konma zamanıdır ve iletinin **kaynak bağlamının** bir parçasıdır. İleti bağlamıyla ilgili daha fazla bilgi için bkz. “MQMD-İleti tanımlayıcı” sayfa 416 ve [İleti bağlamı](#).

Bu alanın kuyruk yöneticisi tarafından oluşturulduğu saat için kullanılan biçim:

- HHMMSSSTH

burada karakterler (sırayla) temsil edilir:

### **SS**

saat (00-23)

### **MM**

dakika (00-59)

### **SS**

saniye (00-59; nota bakın)

### **T**

saniyenin onda biri (0-9)

### **H**

saniyenin yüzde biri (0-9)

**Not:** Sistem saati çok doğru bir zaman standardıyla eşitlenirse, nadir durumlarda *PutTime* içindeki saniyeler için 60 ya da 61 geri gönderilebilir. Bu, artık saniyeler genel zaman standardına eklendiğinde olur.

Greenwich Ortalama Saati (GMT), sistem saatinin GMT ' ye doğru şekilde ayarlanmasına bağlı olarak *PutDate* ve *PutTime* alanları için kullanılır.

İleti bir iş biriminin parçası olarak konduysa, zaman, iş biriminin kesinleştirildiği zaman değil, iletinin konduğu zamandır.

MQPUT ve MQPUT1 çağrılar için, **PutMsgOpts** değiştirilmesinde MQPMO\_SET\_ALL\_CONTEXT belirtilirse bu bir giriş/çıkış alanıdır. Kuyruk yöneticisi, alan içindeki boş karakterin ardından gelen bilgilerin atılması dışında, alanın içeriğini denetlemez. Kuyruk yöneticisi boş karakteri ve aşağıdaki karakterleri boşluklara dönüştürür. MQPMO\_SET\_ALL\_CONTEXT belirtilmezse, girişte bu alan yoksayılar ve yalnızca çıkış alanıdır.

Bu, MQGET çağrısına ilişkin bir çıkış alanıdır. Bu alanın uzunluğu MQ\_PUT\_TIME\_LENGTH tarafından verilir. Bu alanın ilk değeri, C içindeki boş dizgi ve diğer programlama dillerindeki 8 boş karakterdir.

## ***AppOriginVerileri (MQCHAR4)***

Bu, iletinin *kaynak bağlamının* bir parçasıdır. İleti bağlamıyla ilgili daha fazla bilgi için bkz. “MQMD-İleti tanımlayıcı” sayfa 416 ve [İleti bağlamı](#).

AppOriginData , uygulama grubu tarafından tanımlanan ve iletinin kökenine ilişkin ek bilgi sağlamak için kullanılacak bilgilerdir. Örneğin, kimlik verilerinin güvenilir olup olmadığını belirtmek için uygun kullanıcı yetkisiyle çalışan uygulamalar tarafından ayarlanabilir.

Kuyruk yöneticisi bu bilgileri karakter verileri olarak kabul eder, ancak biçimini tanımlamaz. Kuyruk yöneticisi bu bilgileri oluşturduğunda tamamen boş olur.

MQPUT ve MQPUT1 çağrılar için, **PutMsgOpts** değiştirilmesinde MQPMO\_SET\_ALL\_CONTEXT belirtilirse bu bir giriş/çıkış alanıdır. Alan içinde boş bir karakteri izleyen bilgiler atılır. Kuyruk yöneticisi boş karakteri ve aşağıdaki karakterleri boşluklara dönüştürür. MQPMO\_SET\_ALL\_CONTEXT belirtilmezse, girişte bu alan yoksayılar ve yalnızca çıkış alanıdır.

Bu, MQGET çağrısına ilişkin bir çıkış alanıdır. Bu alanın uzunluğu MQ\_APPL\_ORIGIN\_DATA\_LENGTH tarafından verilir. Bu alanın ilk değeri, C ' deki boş dizgi ve diğer programlama dillerindeki 4 boş karakterdir.

AppOriginData ayarlansa da ileti yayınlandığında, aldığı abonelikte boş olur.

## **GroupId (MQBYTE24)**

Bu, fiziksel iletinin ait olduğu belirli ileti grubunu ya da mantıksal iletiyi tanımlamak için kullanılan bayt dizilimidir. *GroupId* , ileti için bölümlenmeye izin veriliyorsa da kullanılır. Tüm bu durumlarda, *GroupId* boş değer olmayan bir değere sahiptir ve *MsgFlags* alanında aşağıdaki işaretlerden biri ya da birkaçı ayarlanır:

- MQMF\_MSG\_IN\_GROUP
- MQMF\_LAST\_MSG\_IN\_GROUP
- MQMF\_SEGMENT
- MQMF\_LAST\_SEGMENT
- MQMF\_SEGMENTATION\_ALLOW

Bu işaretlerin hiçbiri ayarlanmazsa, *GroupId* özel boş değer MQGI\_NONE değerine sahiptir.

Uygulama, aşağıdaki durumlarda, MQPUT ya da MQGET çağrısında bu alanı belirlemesine gerek yoktur:

- MQPUT çağrısında, MQPMO\_LOGICAL\_ORDER belirtildi.
- MQGET çağrısında, MQMO\_MATCH\_GROUP\_ID belirtilmedi.

Bunlar, rapor iletisi olmayan iletiler için bu çağrıları kullanmanın önerilen yollarıdır. Ancak, uygulama daha fazla denetim gerektiriyorsa ya da çağrı MQPUT1ise, uygulama *GroupId* ' in uygun bir değere ayarlandığını doğrulamalıdır.

İleti grupları ve bölümleri, yalnızca grup tanıtıcısı benzersizse doğru olarak işlenebilir. Bu nedenle, *uygulamaların kendi grup tanıtıcılarını oluşturmaması gerekir* ; yerine, uygulamalar aşağıdakilerden birini yapmalıdır:

- MQPMO\_LOGICAL\_ORDER belirtilirse, kuyruk yöneticisi otomatik olarak mantıksal iletinin grubundaki ya da bölümündeki ilk ileti için benzersiz bir grup tanıtıcısı oluşturur ve bu grup tanıtıcısını mantıksal iletinin grup ya da kesimlerinde kalan iletiler için kullanır; dolayısıyla, uygulamanın özel bir işlem yapması gerekmez. Bu, önerilen yordamdır.
- If MQPMO\_LOGICAL\_ORDER is not specified, the application must request the queue manager to generate the group identifier, by setting *GroupId* to MQGI\_NONE on the first MQPUT or MQPUT1 call for a message in the group or segment of the logical message. Bu çağrıdan çıkışta kuyruk yöneticisi tarafından döndürülen grup tanıtıcısı, mantıksal iletinin grup ya da kesimlerindeki geri kalan iletiler için kullanılmalıdır. Bir ileti grubunda kesimlenmiş iletiler varsa, gruptaki tüm kesimler ve iletiler için aynı grup tanıtıcısı kullanılmalıdır.

MQPMO\_LOGICAL\_ORDER değeri belirtilmediğinde, mantıksal iletilerin gruplarındaki ve bölümlerindeki iletiler herhangi bir sırayla (örneğin, ters sırada) yerleştirilebilir; ancak, grup tanıtıcısı bu iletilerin herhangi biri için verilen *ilk* MQPUT ya da MQPUT1 çağrısı tarafından ayrılmalıdır.

MQPUT ve MQPUT1 çağrılarında giriş sırasında, kuyruk yöneticisi, Kuyrukta fiziksel sıralama alanında açıklanan değeri kullanır. MQPUT ve MQPUT1 çağrılarının çıktısında, kuyruk yöneticisi bu alanı, açılan nesne bir dağıtım listesi değil, tek bir kuyruksa, ancak bir dağıtım listesi değilse, ileti ile birlikte gönderilen değere ayarlar, ancak bu açılan nesne bir dağıtım listesi ise, bu değeri değiştirmez. İkinci durumda, uygulamanın oluşturulan grup tanıtıcılarını bilmesi gerekiyorsa, uygulamanın *GroupId* alanını içeren MQPMR kayıtlarını sağlamalıdır.

MQGET çağrısına giriş sırasında, kuyruk yöneticisi Çizelge 496 sayfa 389 içinde açıklanan değeri kullanır. MQGET çağrısından çıktıda, kuyruk yöneticisi bu alanı, alınan ileti için değere ayarlar.

Aşağıdaki özel değer tanımlıdır:

### **MQGI\_NONE**

Grup tanıtıcısı belirtilmedi.

Değer, alanın uzunluğuna ilişkin ikili sıfır değeridir. Bu değer, gruplarda olmayan iletiler için, mantıksal iletilerin kesimleri değil ve kesimlere izin verilmediği iletiler için kullanılır.

C programlama dili için, MQGI\_NONE\_ARRAY değişmezi de tanımlanır; bu, MQGI\_NONE ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisidir.



Bu alanın uzunluğu MQ\_GROUP\_ID\_LENGTH tarafından verilir. Bu alanın ilk değeri MQGI\_NONE olur. *Version* , MQMD\_VERSION\_2değerinden küçükse bu alan yok sayılır.

### **MsgSeqNumarası (MQUBE)**

Bu, bir grup içindeki mantıksal iletinin sıra numarasıdır.

Sıra numaraları 1 'den başlar ve gruptaki her yeni mantıksal ileti için en çok 999 999 999 'a kadar artar. Grupta yer almayan fiziksel bir iletinin sıra numarası 1 'dir.

Uygulama, aşağıdaki durumlarda, MQPUT ya da MQGET çağrısında bu alanı ayarlamak zorunda değildir:

- MQPUT çağrısında, MQPMO\_LOGICAL\_ORDER belirtildi.
- MQGET çağrısında, MQMO\_MATCH\_MSG\_SEQ\_NUMBER belirtilmedi.

Bunlar, rapor iletisi olmayan iletiler için bu çağrıları kullanmanın önerilen yollarıdır. Ancak, uygulama daha fazla denetim gerektiriyorsa ya da çağrı MQPUT1ise, uygulama *MsgSeqNumber* ' in uygun bir değere ayarlandığını doğrulamalıdır.

MQPUT ve MQPUT1 çağrılarında giriş sırasında, kuyruk yöneticisi, Kuyrukta fiziksel sıralamaalanında açıklanan değeri kullanır. MQPUT ve MQPUT1 çağrılarının çıktısında, kuyruk yöneticisi bu alanı, iletiyle birlikte gönderilen değere ayarlar.

MQGET çağrısına giriş sırasında, kuyruk yöneticisi Çizelge 496 sayfa 389 içinde gösterilen değeri kullanır. MQGET çağrısından çıktıda, kuyruk yöneticisi bu alanı, alınan ileti için değere ayarlar.

Bu alanın ilk değeri bir alandır. *Version* , MQMD\_VERSION\_2değerinden küçükse bu alan yok sayılır.

### **Görelî Konum (MQLONG)**

Bu, veri formlarının parçası olan mantıksal iletinin başlangıcındaki fiziksel iletteki verilerin bayt cinsinden görelî konudur. Bu verilere *segment* adı verilir. Görelî konum 0 ile 999 999 999 aralığıdır. Bir mantıksal iletinin parçası olmayan fiziksel bir ileti, sıfır görelî konutuna sahiptir.

Uygulama, aşağıdaki durumlarda, MQPUT ya da MQGET çağrısında bu alanı belirlemesine gerek yoktur:

- MQPUT çağrısında, MQPMO\_LOGICAL\_ORDER belirtildi.
- MQGET çağrısında, MQMO\_MATCH\_OFFSET belirtilmedi.

Bunlar, rapor iletisi olmayan iletiler için bu çağrıları kullanmanın önerilen yollarıdır. Ancak, uygulama bu koşullara uymuyorsa ya da çağrı MQPUT1ise, uygulama *Offset* ' in uygun bir değere ayarlandığından emin olmalıdır.

MQPUT ve MQPUT1 çağrılarında giriş sırasında, kuyruk yöneticisi, Kuyrukta fiziksel sıralamaalanında açıklanan değeri kullanır. MQPUT ve MQPUT1 çağrılarının çıktısında, kuyruk yöneticisi bu alanı, iletiyle birlikte gönderilen değere ayarlar.

Bir mantıksal iletinin bir kesimine ilişkin rapor iletisi raporlaması için, kuyruk yöneticisi tarafından korunan bölüm bilgilerindeki görelî konumu güncellemek için *OriginalLength* alanı (MQOL\_UNDEFINED olarak sağlanır) kullanılır.

MQGET çağrısına giriş sırasında, kuyruk yöneticisi Çizelge 496 sayfa 389 içinde gösterilen değeri kullanır. MQGET çağrısından çıktıda, kuyruk yöneticisi bu alanı, alınan ileti için değere ayarlar.

Bu alanın ilk değeri sıfır. *Version* , MQMD\_VERSION\_2değerinden küçükse bu alan yok sayılır.

### **MsgFlags (MQUZE)**

MsgFlags , iletinin özniteliklerini belirten işaretlerdir ya da işlemeyi denetler.

MsgFlags , aşağıdaki kategorilere ayrılır:

- Bölümleme işaretleri
- Durum işaretleri



**Bölümlenme işaretleri:** Bir ileti kuyruk için çok büyük olduğunda, iletiyi kuyruğa koyma girişimi genellikle başarısız olur. Bölümlenme, kuyruk yöneticisinin ya da uygulamanın iletiyi bölümler adı verilen daha küçük parçalara ayırdığı ve kuyruğun her kesimini ayrı bir fiziksel ileti olarak kuyruğa alan bir tekniktir. İletiyi alan uygulama, kesimleri tek tek alabilir ya da kuyruk yöneticisini, kesimleri MQGET çağrısının döndürdüğü tek bir iletiyle yeniden birleştirmesini isteyebilir. İkinci olarak, MQGET çağrısında MQGMO\_COMPLETE\_MSG seçeneği belirlenerek ve iletiyi tamamlamak için yeterli büyüklükte bir arabellek sağlanarak elde edilir. (MQGMO\_COMPLE\_MSG seçeneğinin ayrıntıları için “MQGMO-Get-message seçenekleri” sayfa 364 konusuna bakın.) Gönderme kuyruğu yöneticisinde, ara kuyruk yöneticisinde ya da hedef kuyruk yöneticisinde bir ileti bölümlenebilir.

Bir iletinin bölümlenmesini denetlemek için aşağıdakilerden birini belirtebilirsiniz:

#### **MQMF\_SEGMENTATION\_ENGELLENDI**

Bu seçenek, kuyruk yöneticisi tarafından iletinin kesimlere ayrılmasını önler. Önceden bir bölüm olan bir ileti için belirtilirse, bu seçenek kesimin daha küçük kesimlere ayrılmasını önler.

Bu işaretin değeri ikili sıfırdır. Bu varsayılandır.

#### **MQMF\_SEGMENTATION\_ALLOW**

Bu seçenek, ileti, kuyruk yöneticisi tarafından kesimlere ayrılmasına olanak sağlar. Önceden bir bölüm olan bir ileti için belirtilirse, bu seçenek kesimin daha küçük kesimlere ayrılmasını sağlar. MQMF\_SEGMENTATION\_ALLOWAL, MQMF\_SEGMENT ya da MQMF\_LAST\_SEGMENT ayarlanmadan ayarlanabilir.

- z/OS üzerinde, kuyruk yöneticisi iletilerin bölümlenmesine destek vermiyor. Bir ileti kuyruk için çok büyükse, MQPUT ya da MQPUT1 çağrısı, MQRC\_MSG\_TOO\_BIG\_FOR\_Q neden koduyla başarısız olur. Ancak, MQMF\_SEGMENTATION\_ALLOWED seçeneği yine de belirtilebilir ve iletinin uzak bir kuyruk yöneticisinde bölümlenmesine olanak sağlar.

Kuyruk yöneticisi bir iletiyi kestiğinde, kuyruk yöneticisi her bir bölümle birlikte gönderilen MQMD 'nin kopyasındaki MQMF\_SEGMENT işaretini açar; ancak, MQPUT ya da MQPUT1 çağrısında uygulama tarafından sağlanan MQMD' de bu işaretlerin ayarlarını değiştirmez. Mantıksal iletiyle son bölüm için, kuyruk yöneticisi, kesimle gönderilen MQMD 'deki MQMF\_LAST\_SEGMENT işaretini de açar.

**Not:** MQMF\_SEGMENTATION\_ALLOWLU iletilerine izin verildiğinde, ancak MQPMO\_LOGICAL\_ORDER olmadan dikkatli bir şekilde dikkatli olun. İleti aşağıdaki gibi olursa:

- Bir kesim değil, ve
- Bir grupta değil,
- İletilmiyor.

Kuyruk yöneticisinin her ileti için benzersiz bir grup tanıtcısı oluşturabilmesi için, uygulamanın her MQPUT ya da MQPUT1 çağrısından önce *GroupId* alanını MQGI\_NONE olarak ilk durumuna getirmesi gerekir. Bu işlem yapılmazsa, ilgisiz iletiler aynı grup tanıtcısına sahip olabilir ve bu da daha sonra yanlış işleme yöneltebilirler. *GroupId* alanının ilk durumuna getirilmesiyle ilgili ek bilgi için, *GroupId* alanının ve MQPMO\_LOGICAL\_ORDER seçeneğine ilişkin açıklamalara bakın.

Kuyruk yöneticisi, iletileri, kuyruğa sığması için kesimler (artı gerekli tüm üstbilgi verileri) gerektiği şekilde bölümlere ayırır. Ancak, kuyruk yöneticisi tarafından oluşturulan bir kesimin büyüklüğü için daha düşük bir sınır vardır ve yalnızca bir iletiden yaratılan son bölüm bu sınırdan küçük olabilir (uygulama tarafından oluşturulan bir kesimin büyüklüğünün alt sınırı bir byte 'tır). Kuyruk yöneticisi tarafından oluşturulan kesimler eşit olmayan uzunluğa sahip olabilir. Kuyruk yöneticisi iletiyi şu şekilde işler:

- Kullanıcı tanımlı biçimler, 16 bayttan oluşan sınırlarda bölünmeye başlanır; kuyruk yöneticisi 16 bayttan küçük (son bölüm dışında) bölüm oluşturamaz.
- MQFMT\_STRING dışındaki yerleşik biçimler, veri varının doğasına uygun olan noktalarda bölünme sağlar. Ancak, kuyruk yöneticisi bir iletiyi hiçbir zaman bir IBM MQ üstbilgi yapısının ortasına bölemez. Başka bir deyişle, tek bir MQ üstbilgi yapısı içeren bir bölüm kuyruk yöneticisi tarafından daha fazla bölünemez ve sonuç olarak, o ileti için olası en düşük kesim büyüklüğü 16 byte 'tan fazla olur.

Kuyruk yöneticisi tarafından oluşturulan ikinci ya da sonraki bölüm aşağıdakilerden biriyle başlar:

- Bir MQ üstbilgi yapısı
- Uygulama iletisi verilerinin başlangıcı
- Uygulama iletisi verilerinden geçmenin bir parçası
- MQFMT\_STRING, veri (SBCS, DBCS ya da karma SBCS/DBCS) niteliği dikkate alınmadan bölünmeye başlanır. Dizgi DBCS ya da SBCS/DBCS karışık olduğunda, bu durum bir karakter takımından diğerine dönüştürülemeyen bölümlerle sonuçlanabilir. Kuyruk yöneticisi, MQFMT\_STRING iletilerini 16 byte 'tan küçük (son bölümden başka) bölümlere ayırmaz.
- Kuyruk yöneticisi, her kesimin MQMD 'deki *Format, CodedCharSetIdve Encoding* alanlarını, kesimin *başlangıcındaki* verileri doğru şekilde tanımlamak için ayarlar; biçim adı, yerleşik bir biçimin adı ya da kullanıcı tanımlı biçimin adıdır.
- Sıfırdan büyük *Offset* olan bölümlerin MQMD 'deki *Report* alanı değiştirilir. Her bir rapor tipi için, rapor seçeneği MQRO\_ \* \_WITH\_DATA ise, ancak bölüm, kullanıcı verilerinin ilk 100 baytıdan herhangi birini içeremez (bu, var olabilecek herhangi bir IBM MQ üstbilgi yapısından sonra gelen veriler), rapor seçeneği MQRO\_ \* olarak değiştirilir.

Kuyruk yöneticisi yukarıdaki kuralları izler, ancak tersi durumda iletileri tahmin edilemez olarak böler; bir iletinin bölündüğü yere ilişkin varsayımlar yapmayın.

Kuyruk yöneticisi, *kalıcı* iletiler için, bölümlenme işlemini yalnızca bir iş birimi içinde gerçekleştirebilir:

- MQPUT ya da MQPUT1 çağrısı kullanıcı tanımlı bir iş birimi içinde çalışıyorsa, bu iş birimi kullanılır. Arama, kesimlere ayırma işlemi sırasında başarısız olursa, kuyruk yöneticisi, başarısız olan aramaların sonucu olarak kuyruğa yerleştirilen tüm bölümleri kaldırır. Ancak, hata, iş biriminin başarıyla kesinleştirilmesini engellememektedir.
- Arama, kullanıcı tanımlı bir iş birimi dışında çalışıyorsa ve var olan bir kullanıcı tanımlı iş birimi yoksa, kuyruk yöneticisi yalnızca arama süresi boyunca bir iş birimi yaratır. Arama başarılı olursa, kuyruk yöneticisi iş birimini otomatik olarak kesinleştirir. Arama başarısız olursa, kuyruk yöneticisi iş birimini geri arar.
- Arama, kullanıcı tanımlı bir iş birimi dışında çalışıyorsa, ancak kullanıcı tanımlı bir iş birimi varsa, kuyruk yöneticisi bölümlenme gerçekleştiremez. İleti bölümlenmeye gerek duymuyorsa, arama hala başarılı olabilir. Ancak, ileti kesimlere ayırma gerektiriyorsa, arama başarısız olur; neden kodu MQRC\_UOW\_NOT\_KULLANILABİLİR.

*kalıcı olmayan* iletileri için, kuyruk yöneticisi bölümlenmenin gerçekleştirilmesini sağlamak için bir iş birimi kullanılabilir olmasını gerektirmez.

Kesimlere ayrılmış olabilecek iletilerde verileri dönüştürürken özel bakım gerçekleştirin:

- Alma uygulaması, MQGET çağrısına veri dönüştürürse ve MQGMO\_COMPLE\_MSG seçeneğini belirlerse, veriye dönüştürme çıkışı, dönüştürülecek çıkışa ilişkin tam iletiyi iletir ve iletinin bölümlenmiş olması, çıkışa kadar belirgin olur.
- Alma uygulaması bir defada bir parça alıyorsa, veri dönüştürme çıkışı bir defada bir kesimi dönüştürmek için çağrılır. Bu nedenle, çıkışta diğer bölümlerdeki verilerden bağımsız olarak bir kesime veri dönüştürmesi gerekir.

İletideki verilerin doğası, 16 baytlık sınırlardaki verilerin rasgele kesimlerinin çıkışa dönüştürülemeyecek bölümlerle sonuçlanabileceği durumdaysa ya da biçim, MQFMT\_STRING ve karakter kümesi DBCS ya da karışık SBCS/DBCS ise, gönderme yapan uygulamanın kesimlere ayırma engellemesi için MQMF\_SEGMENTATION\_INHIBID değerinin belirlenmesi ve kesimleri olması gerekir. Bu şekilde, gönderme uygulaması, her kesimin veri dönüştürme çıkışısının kesimini başarıyla dönüştürmesine izin vermek için yeterli bilgi içermesini güvenebilir.

- Gönderen bir ileti kanalı aracı (MCA) için gönderici dönüştürme belirtildiyse, MCA yalnızca mantıksal iletilerin kesimleri olmayan iletileri dönüştürür; MCA hiçbir zaman bölüm olan iletileri dönüştürmeyi denememez.

Bu işaret, MQPUT ve MQPUT1 çağrılarında bir giriş işaretidir ve MQGET çağrısına ilişkin bir çıkış işareti. İkinci arama sırasında, kuyruk yöneticisi, işaretin değerini MQGMO 'daki *Segmentation* alanına da yansıtıyor.

Bu işaretin başlangıç değeri MQMF\_SEGMENTATION\_INHIBID değeridir.

**Durum işaretleri:** Bunlar, fiziksel iletinin bir ileti grubuna ait olup olmadığını gösteren işaretlerdir, her ikisi de ya da ikisi de mantıksal iletinin bir bölümüdür. MQPUT ya da MQPUT1 çağrısında, aşağıdakilerden biri ya da daha fazlası belirtilebilir ya da MQGET çağrısının döndürmesini sağlar:

#### **MQMF\_MSG\_IN\_GROUP**

İleti, bir grubun üyesidir.

#### **MQMF\_LAST\_MSG\_IN\_GROUP**

İleti, bir gruptaki son mantıksal iletidir.

Bu işaret belirlendiyse, kuyruk yöneticisi iletiyle birlikte gönderilen MQMD kopyasındaki MQMF\_MSG\_IN\_GROUP ögesini açar; ancak, MQPUT ya da MQPUT1 çağrısında uygulama tarafından sağlanan MQMD ' de bu işaretlerin ayarlarını değiştirmez.

Bir grup için geçerli, tek bir mantıksal iletiden oluşur. Bu durumda, MQMF\_LAST\_MSG\_IN\_GROUP ayarlıdır, ancak *MsgSeqNumber* alanında değer olan bir değer vardır.

#### **MQMF\_SEGMENT**

İleti, mantıksal bir iletinin bir bölümüdür.

MQMF\_SEGMENT MQMF\_LAST\_SEGMENT olmadan belirtilirse, kesimde uygulama iletisi verilerinin uzunluğu (var olabilecek IBM MQ üstbilgi yapılarının uzunlukları *hariç tutma* ) en az bir olmalıdır. Uzunluk sıfırda, MQPUT ya da MQPUT1 çağrısı, MQRC\_SEGMENT\_LENGTH\_ZERO neden koduyla başarısız olur.

z/OSüzerinde, ileti bir MQIT\_GROUP\_ID dizin tipine sahip bir kuyruğa konursa, bu seçenek desteklenmez.

#### **MQMF\_LAST\_SEGMENT**

İleti, bir mantıksal iletinin son bölümüdür.

Bu işaret belirlendiyse, kuyruk yöneticisi, iletiyle birlikte gönderilen MQMD kopyasında MQMF\_SEGMENT ögesini açar; ancak, MQPUT ya da MQPUT1 çağrısında uygulama tarafından sağlanan MQMD ' de bu işaretlerin ayarlarını değiştirmez.

Mantıksal ileti yalnızca bir bölümden oluşabilir. Bu durumda, MQMF\_LAST\_SEGMENT ayarlı, ancak *Offset* alanında sıfır değeri var.

MQMF\_LAST\_SEGMENT belirtildiğinde, kesimde uygulama iletisi verilerinin uzunluğu (var olabilecek üstbilgi yapılarının uzunlukları *hariç tutma* ) sıfır olabilir.

z/OSüzerinde, ileti bir MQIT\_GROUP\_ID dizin tipine sahip bir kuyruğa konursa, bu seçenek desteklenmez.

Uygulama, ileti yerleştirilirken bu işaretlerin doğru ayarlandığından emin olmalıdır.

MQPMO\_LOGICAL\_ORDER belirtildiyse ya da kuyruk tanıtıcısı için önceki MQPUT çağrısında belirtilmişse, işaretlerin ayarları kuyruk tanıtıcısı için kuyruk yöneticisi tarafından korunan grup ve bölüm bilgileriyle tutarlı olmalıdır. Aşağıdaki koşullar, *ardışık* MQPUT çağrıları için, MQPMO\_LOGICAL\_ORDER belirtildiğinde kuyruk tanıtıcısı için geçerlidir:

- Yürürlükte bir grup ya da mantıksal ileti yoksa, tüm bu işaretler (ve bunların birleşimleri) geçerli olur.
- MQMF\_MSG\_IN\_GROUP belirtildikten sonra, MQMF\_LAST\_MSG\_IN\_GROUP belirtinceye kadar bu durumda kalmalıdır. Bu koşul yerine getirilmezse, çağrıya neden kodu MQRC\_INCOMPLETE\_GROUP neden koduyla başarısız olur.
- MQMF\_SEGMENT belirtildikten sonra, MQMF\_LAST\_SEGMENT belirtinceye kadar açık kalmalıdır. Bu koşul karşılanmazsa, çağrı neden kodu MQRC\_INCOMPLE\_MSG neden ile başarısız olur.
- MQMF\_MSG\_IN\_GROUP olmadan bir MQMF\_SEGMENT belirtildikten sonra, MQMF\_MSG\_IN\_GROUP, MQMF\_LAST\_SEGMENT belirlendikten sonra *off* (kapalı) olarak kalmalıdır. Bu koşul karşılanmazsa, çağrı neden kodu MQRC\_INCOMPLE\_MSG neden ile başarısız olur.

Kuyruklardaki fiziksel sıralama , işaretlerin geçerli birleşimlerini ve çeşitli alanlar için kullanılan değerleri gösterir.

Bu işaretler, MQPUT ve MQPUT1 çağrılarında ve MQGET çağrısına ilişkin çıkış işaretlerine ilişkin giriş işaretleridir. İkinci çağrı sırasında, kuyruk yöneticisi, işaretlerin değerlerini MQGMO 'daki *GroupStatus* ve *SegmentStatus* alanlarına da yankıyor.

Publish/Subscribe ile gruplandırılmış ya da bölümlenmiş iletileri kullanamazsınız.

**Varsayılan işaretler:** İletinin varsayılan özniteliklere sahip olduğunu belirtmek için aşağıdaki değer belirlenebilir:

### **MQMF\_NONE**

İleti işaretleri (varsayılan ileti öznitelikleri) yok.

Bu, kesimlere bölünmeyi engeller ve iletinin bir grupta yer almadığını ve mantıksal bir iletinin bir kesimi olmadığını belirtir. MQMF\_NONE, program belgelerine yardımcı olmak için tanımlıdır. Bu işaretin başka biriyle birlikte kullanılması amaçlanmaz, ancak değeri sıfır olduğu için bu tür bir kullanım saptanamaz.

*MsgFlags* alanı alt alanlara bölümlenmektedir; ayrıntılar için [“Rapor seçenekleri ve ileti işaretleri” sayfa 887](#)' e bakın.

Bu alanın ilk değeri MQMF\_NONE olur. *Version* , MQMD\_VERSION\_2değerinden küçükse bu alan yok sayılır.

### **OriginalLength (MQlong)**

Bu alan yalnızca bölüm olan rapor iletileri için anlamlıdır. Rapor iletisinin ilişkilendirdiği ileti kesiminin uzunluğunu belirler; bu, bölüm formlarının parçası olan mantıksal iletinin uzunluğunu ya da rapor iletisinde verilerin uzunluğunu belirtmez.

**Not:** When generating a report message for a message that is a segment, the queue manager and message channel agent copy into the MQMD for the report message the *GroupId*, *MsgSeqNumber*, *Offset*, and *MsgFlags*, fields from the original message. Sonuç olarak, rapor iletisi de bir bölümdür. Rapor iletileri oluşturan uygulamalar da aynı işlemi yapmalıdır ve *OriginalLength* alanını doğru şekilde ayarlamalıdır.

Aşağıdaki özel değer tanımlıdır:

### **MQOL\_TANIMSIZ**

İletinin özgün uzunluğu tanımlı değil.

*OriginalLength* , MQPUT ve MQPUT1 çağrılarında bir giriş alanıdır, ancak uygulamanın sağladığı değer, yalnızca belirli durumlarda kabul edilir:

- Gönderilmekte olan ileti bir bölümse ve aynı zamanda bir rapor iletisiyse, kuyruk yöneticisi belirtilen değeri kabul eder. Değer şu olmalıdır:
  - Kesimin son bölümü değilse sıfırdan büyük
  - Kesim son bölümse sıfırdan küçük değil
  - İletide var olan veri uzunluğundan az değil

Bu koşullar karşılanmazsa, çağrı neden kodu MQRC\_ORIGINAL\_LENGTH\_ERROR nedeniyle başarısız olur.

- Bir rapor iletisi gönderiliyorsa, ancak bir rapor iletisi değilse, kuyruk yöneticisi alanı yoksayar ve bunun yerine uygulama iletisi verilerinin uzunluğunu kullanır.
- Diğer tüm durumlarda, kuyruk yöneticisi alanı yoksayar ve MQOL\_UNDEUNDEFINED değerini kullanır.

Bu, MQGET çağrısındaki bir çıkış alanıdır.

Bu alanın ilk değeri MQOL\_UNDEUNDEFINED olur. *Version* , MQMD\_VERSION\_2değerinden küçükse bu alan yok sayılır.

### **MQMDE-İleti tanımlayıcı uzantısı**

MQMDE yapısı, bazen uygulama iletisi verilerinden önce gelen verileri tanımlar. Yapı, version-2 MQMD 'de var olan, ancak version-1 MQMD' de bulunmayan MQMD alanlarını içeriyor.

## Kullanılabilirlik

Tüm IBM MQ sistemleri ve bu sistemlere bağlı IBM MQ MQI clients .

## Biçim Adı

MQFMT\_MD\_EXTENSION

## Karakter kümesi ve kodlama

MQMDE içindeki veriler, yerel kuyruk yöneticisinin karakter kümesi ve kodlamasında olmalıdır; bunlar, C programlama dili için **CodedCharSetId** kuyruk yöneticisi özneteliği ve MQENC\_NATIVE ile verilir.

MQMDE ' nin karakter kümesini ve kodlamasını aşağıdaki *CodedCharSetId* ve *Encoding* alanlarına ayarlayın:

- MQMD (MQMDE yapısı ileti verilerinin başındadır) ya da
- MQMDE yapısından önce gelen üstbilgi yapısı (diğer tüm durumlar).

MQMDE, kuyruk yöneticisinin karakter kümesinde ve kodlamasında yoksa, MQMDE kabul edilir, ancak yerine getirilmezse, MQMDE ileti verisi olarak işlenir.

**Not:** Windows üzerinde, Micro Focus COBOL ile derlenen uygulamalar, kuyruk yöneticisinin kodlamasından farklı bir MQENC\_NATIVE değeri kullanır. MQPUT, MQPUT1 ve MQGET çağrılarındaki MQMD yapısındaki sayısal alanların Micro Focus COBOL kodlamasında olması gerekse de, MQMDE yapısındaki sayısal alanların kuyruk yöneticisinin kodlamasında olması gerekir. Bu ikincisi, C programlama dili için MQENC\_NATIVE tarafından verilir ve 546 değerine sahiptir.

## Kullanım

version-2 MQMD kullanan uygulamalar bir MQMDE yapısıyla karşılaşmaz. Ancak, version-1 MQMD ' yi kullanmaya devam eden özel uygulamalar ve uygulamalar bazı durumlarda MQMDE ile karşılaşabilir. MQMDE yapısı aşağıdaki durumlarda oluşabilir:

- MQPUT ve MQPUT1 çağrılarında belirtildi
- MQGET çağrısı tarafından döndürüldü
- İletim kuyruklarındaki iletilerde

## MQPUT ve MQPUT1 çağrılarında MQMDE belirtildi

MQPUT ve MQPUT1 çağrılarında uygulama bir version-1 MQMD sağlarsa, uygulama isteğe bağlı olarak ileti verilerine bir MQMDE öneki gönderebilir ve MQMDE ' nin var olduğunu göstermek için MQMD ' deki *Format* alanını MQFMT\_MD\_EXTENSION olarak ayarlayabilir. Uygulama bir MQMDE sağlamazsa, kuyruk yöneticisi MQMDE ' deki alanlar için varsayılan değerleri varsayar. Kuyruk yöneticisinin kullandığı varsayılan değerler, yapının ilk değerleriyle aynıdır; bkz. [Çizelge 504 sayfa 467](#).

Uygulama bir version-2 MQMD sağlarsa ve , uygulama iletilerini MQMDE ile öneklerse, yapılar aşağıdaki çizelgede gösterildiği gibi işlenir.

Çizelge 503. MQPUT üzerinde MQMDE belirtildiğinde kuyruk yöneticisi işlemi ya da MQMDE için MQPUT1			
MQMD sürümü	version-2 alanlarının değerleri	MQMDE içindeki ilgili alanların değerleri	Kuyruk yöneticisi tarafından gerçekleştirilen işlem
1	-	Geçerli	MQMDE onurlandırıldı
2	Varsayılan	Geçerli	MQMDE onurlandırıldı
2	Varsayılan değil	Geçerli	MQMDE, ileti verisi olarak kabul edilir

Çizelge 503. MQPUT üzerinde MQMDE belirtildiğinde kuyruk yöneticisi işlemi ya da MQMDE için MQPUT1 (devamı var)			
MQMD sürümü	version-2 alanlarının değerleri	MQMDE içindeki ilgili alanların değerleri	Kuyruk yöneticisi tarafından gerçekleştirilen işlem
1 ya da 2	Herhangi Biri	Geçerli değil	Arama uygun bir neden koduyla başarısız oluyor
1 ya da 2	Herhangi Biri	MQMDE yanlış karakter kümesinde ya da kodlamada ya da desteklenmeyen bir sürümde	MQMDE, ileti verisi olarak kabul edilir
<b>Not:</b> z/OS sistemlerinde, uygulama MQMDE içeren bir version-1 MQMD 'sini belirtiyorsa, kuyruk yöneticisi MQMDE ' yi yalnızca kuyrukta <i>IndexType</i> MQIT_GROUP_ID varsa doğrular.			

Özel bir durum var. Uygulama bir bölüm (yani, MQMF\_SEGMENT ya da MQMF\_LAST\_SEGMENT işareti) olan bir iletiyi koymak için version-2 MQMD kullanıyorsa ve MQMD ' deki biçim adı MQFMNT\_DEAD\_LETTER\_HEADER ise, kuyruk yöneticisi bir MQMDE yapısı oluşturur ve bunu MQDLH yapısı ile onu izleyen veriler arasında *ekler* . Kuyruk yöneticisinin iletiyle birlikte sakladığı MQMD ' de version-2 alanları varsayılan değerlerine ayarlanır.

version-2 MQMD 'de var olan, ancak version-1 MQMD' de olmayan alanlar, MQPUT ve MQPUT1' de giriş/çıkış alanlarıdır. Ancak, kuyruk yöneticisi MQPUT ve MQPUT1 çağrılarının çıkışında MQMDE ' deki eşdeğer alanlarda herhangi bir değer döndürmez; uygulama bu çıkış değerlerini gerektiriyorsa, bir version-2 MQMD kullanmalıdır.

### MQGET çağrısı tarafından döndürülen MQMDE

MQGET çağrısında, uygulama bir version-1 MQMD sağlarsa, kuyruk yöneticisi MQMDE ile döndürülen iletiyi örnekler, ancak MQMDE ' deki alanlardan birinin ya da daha fazlasının varsayılan olmayan bir değeri varsa. Kuyruk yöneticisi, MQMD 'deki *Format* alanını MQMDE' nin var olduğunu göstermek için MQFMT\_MD\_EXTENSION değerine ayarlar.

Uygulama **Buffer** değiştirgesinin başında bir MQMDE sağlarsa, MQMDE yoksayıdır. MQGET çağrısından döndüğünüzde, ileti için MQMDE ile değiştirilir (gerekliyorsa) ya da uygulama ileti verileri tarafından üzerine yazılır (MQMDE gerekli değilse).

MQGET çağrısı bir MQMDE döndürürse, MQMDE ' deki veriler genellikle kuyruk yöneticisinin karakter kümesi ve kodlamasında bulunur. Ancak, MQMDE aşağıdaki durumda başka bir karakter kümesi ve kodlamada olabilir:

- MQMDE, MQPUT ya da MQPUT1 çağrısında veri olarak kabul edildi (buna neden olabilecek durumlar için bkz. Çizelge 503 sayfa 465).
- İleti, TCP bağlantısıyla bağlanan bir uzak kuyruk yöneticisinden alındı ve alan ileti kanalı aracısı (MCA) doğru ayarlanmadı.

**Not:** Windows üzerinde, Micro Focus COBOL ile derlenen uygulamalar, kuyruk yöneticisinin kodlamasından farklı bir MQENC\_NATIVE değeri kullanır (yukarıya bakın).

### İletim kuyruklarındaki iletilerde MQMDE

İletim kuyruklarındaki iletilerin başına, içinde version-1 MQMD içeren MQXQH yapısı eklenir. MQMDE, MQXQH yapısı ve uygulama iletileri arasında konumlandırılmış olarak da var olabilir; ancak, MQMDE ' deki alanlardan birinin ya da daha fazlasının varsayılan olmayan bir değeri varsa bu durum genellikle var olur.

MQXQH yapısı ile uygulama iletileri arasında diğer MQ üstbilgi yapıları da oluşabilir. Örneğin, gitmeyen harf üstbilgisi MQDLH varsa ve ileti bir bölüm değilse, sıralama şöyle olur:

- MQXQH ( version-1 MQMD 'si içerir)

- MQMDE
- MQDLH
- uygulama iletisi verileri

## Alanlar

**Not:** Aşağıdaki tabloda, alanlar alfabetik olarak değil, kullanıma göre gruplanır. Alt konular aynı sırayı izler.

Çizelge 504. MQMDE için MQMDE içindeki alanlar		
Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
StrucId (yapı tanıtıcısı)	MQMDE_STRUC_ID	'MDE↵'
Sürüm (yapı sürümü numarası)	MQMDE_VERSION_2	2
StrucLength (MQMDE yapısının uzunluğu)	MQMDE_LENGTH_2	72
Kodlama (MQMDE ' yi izleyen verilerin sayısal kodlaması)	MQENC_NATIVE	Ortama bağlıdır
CodedCharSetId (MQMDE ' yi izleyen verilerin karakter kümesi tanıtıcısı)	MQCCSI_AYRILMAMIŞ	0
Biçim (MQMDE ' yi izleyen verilerin biçim adı)	MQFMT_NONE	Boşluklar
İşaretler (genel işaretler)	MQMDEF_NONE	0
GroupId (grup tanıtıcısı)	MQGI_NONE	Boş Değerler
MsgSeqNumber (grup içindeki mantıksal iletinin sıra numarası)	Yok	1
Görelî Konum (mantıksal iletinin başlangıcından fiziksel iletideki verilerin görelî konumu)	Yok	0
MsgFlags (ileti işaretleri)	MQMF_NONE	0
OriginalLength (özgün iletinin uzunluğu)	MQOL_UNDEFINED	-1
<p><b>Notlar:</b></p> <ol style="list-style-type: none"> <li>1. ↵ simgesi, tek bir boş karakteri temsil eder.</li> <li>2. C programlama dilinde, makro değişkeniMQMDE_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlar için başlangıç değerleri sağlamak üzere aşağıdaki şekilde kullanılabilir:</li> </ol> <pre>MQMDE MyMDE = {MQMDE_DEFAULT};</pre>		

## Dil bildirimleri

MQMDE için C bildirimi

```
typedef struct tagMQMDE MQMDE;
struct tagMQMDE {
    MQCHAR4    StrucId;           /* Structure identifier */
    MQLONG     Version;          /* Structure version number */
    MQLONG     StrucLength;      /* Length of MQMDE structure */
    MQLONG     Encoding;        /* Numeric encoding of data that follows
                                MQMDE */
    MQLONG     CodedCharSetId;  /* Character-set identifier of data that
                                follows MQMDE */
    MQCHAR8    Format;          /* Format name of data that follows
                                MQMDE */
};
```

```

MQLONG   Flags;           /* General flags */
MQBYTE24 GroupId;        /* Group identifier */
MQLONG   MsgSeqNumber;   /* Sequence number of logical message
                          within group */
MQLONG   Offset;         /* Offset of data in physical message from
                          start of logical message */
MQLONG   MsgFlags;       /* Message flags */
MQLONG   OriginalLength; /* Length of original message */
};

```

## MQMDE için COBOL bildirimi

```

** MQMDE structure
10 MQMDE.
** Structure identifier
15 MQMDE-STRUCID PIC X(4).
** Structure version number
15 MQMDE-VERSION PIC S9(9) BINARY.
** Length of MQMDE structure
15 MQMDE-STRUCLength PIC S9(9) BINARY.
** Numeric encoding of data that follows MQMDE
15 MQMDE-ENCODING PIC S9(9) BINARY.
** Character-set identifier of data that follows MQMDE
15 MQMDE-CODEDCHARSETID PIC S9(9) BINARY.
** Format name of data that follows MQMDE
15 MQMDE-FORMAT PIC X(8).
** General flags
15 MQMDE-FLAGS PIC S9(9) BINARY.
** Group identifier
15 MQMDE-GROUPID PIC X(24).
** Sequence number of logical message within group
15 MQMDE-MSGSEQNUMBER PIC S9(9) BINARY.
** Offset of data in physical message from start of logical message
15 MQMDE-OFFSET PIC S9(9) BINARY.
** Message flags
15 MQMDE-MSGFLAGS PIC S9(9) BINARY.
** Length of original message
15 MQMDE-ORIGINALLENGTH PIC S9(9) BINARY.

```

## MQMDE için PL/I bildirimi

```

dcl
1 MQMDE based,
3 StrucId char(4), /* Structure identifier */
3 Version fixed bin(31), /* Structure version number */
3 StrucLength fixed bin(31), /* Length of MQMDE structure */
3 Encoding fixed bin(31), /* Numeric encoding of data that
                          follows MQMDE */
3 CodedCharSetId fixed bin(31), /* Character-set identifier of data
                          that follows MQMDE */
3 Format char(8), /* Format name of data that follows
                  MQMDE */
3 Flags fixed bin(31), /* General flags */
3 GroupId char(24), /* Group identifier */
3 MsgSeqNumber fixed bin(31), /* Sequence number of logical message
                          within group */
3 Offset fixed bin(31), /* Offset of data in physical message
                          from start of logical message */
3 MsgFlags fixed bin(31), /* Message flags */
3 OriginalLength fixed bin(31); /* Length of original message */

```

## MQMDE için High Level Assembler (Üst Düzey Çevirici) bildirimi

MQMDE	DSECT		
MQMDE_STRUCID	DS	CL4	Structure identifier
MQMDE_VERSION	DS	F	Structure version number
MQMDE_STRUCLength	DS	F	Length of MQMDE structure
MQMDE_ENCODING	DS	F	Numeric encoding of data that follows MQMDE
MQMDE_CODEDCHARSETID	DS	F	Character-set identifier of data that follows MQMDE
MQMDE_FORMAT	DS	CL8	Format name of data that follows MQMDE
MQMDE_FLAGS	DS	F	General flags
MQMDE_GROUPID	DS	XL24	Group identifier
MQMDE_MSGSEQNUMBER	DS	F	Sequence number of logical message



* MQMDE_OFFSET	DS	F	within group Offset of data in physical message from start of logical message
* MQMDE_MSGFLAGS	DS	F	Message flags
* MQMDE_ORIGINALLENGTH	DS	F	Length of original message
* MQMDE_LENGTH	EQU	*-MQMDE	
	ORG	MQMDE	
MQMDE_AREA	DS	CL(MQMDE_LENGTH)	

## MQMDE için Visual Basic bildirimi

```

Type MQMDE
  StrucId      As String*4 'Structure identifier'
  Version      As Long    'Structure version number'
  StrucLength  As Long    'Length of MQMDE structure'
  Encoding     As Long    'Numeric encoding of data that follows'
  CodedCharSetId As Long  'Character-set identifier of data that'
  Format       As String*8 'Format name of data that follows MQMDE'
  Flags       As Long    'General flags'
  GroupId     As MQBYTE24 'Group identifier'
  MsgSeqNumber As Long    'Sequence number of logical message within'
  Offset      As Long    'Offset of data in physical message from'
  MsgFlags    As Long    'Message flags'
  OriginalLength As Long  'Length of original message'
End Type

```

### **StrucId (MQCHAR4)**

Değer şu olmalıdır:

#### **MQMDE\_STRUC\_ID**

İleti tanımlayıcı uzantısı yapısına ilişkin tanıtıcı.

C programlama dili için, aynı zamanda MQMDE\_STRUC\_ID\_ARRAY değişmezi de tanımlanır; bu, MQMDE\_STRUC\_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisi olur.

Bu alanın ilk değeri MQMDE\_STRUC\_ID ' dir.

### **Sürüm (MQUZE)**

Bu, yapı sürüm numarasıdır; değer şu olmalıdır:

#### **MQMDE\_VERSION\_2**

Version-2 ileti tanımlayıcı uzantı yapısı.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

#### **MQMDE\_CURRENT\_SÜRÜM**

İleti tanımlayıcı uzantısı yapısının yürürlükteki sürümü.

Bu alanın ilk değeri MQMDE\_VERSION\_2' dir.

### **StrucLength (MQUZE)**

Bu, MQMDE yapısının uzunluğudur; aşağıdaki değer tanımlıdır:

#### **MQMDE\_LENGTH\_2**

version-2 ileti tanımlayıcı uzantısı yapısının uzunluğu.

Bu alanın ilk değeri MQMDE\_LENGTH\_2' dir.

### **Kodlama (MQUZE)**

Bu, MQMDE yapısını izleyen verilerin sayısal kodlamasını belirtir; bu, MQMDE yapısındaki sayısal veriler için geçerli değildir.

MQPOT ya da MQPUT1 çağrısında, uygulama bu alanı veri için uygun değere ayarlamalıdır. Kuyruk yöneticisi, alanın geçerli olup olmadığını denetmez. Veri kodlamalarıyla ilgili daha fazla bilgi için “MQMD-İleti tanımlayıcı” sayfa 416 içinde açıklanan *Encoding* alanına bakın.

Bu alanın ilk değeri MQENC\_NATIVE 'dir.

### **CodedCharSetId (MQHOMER)**

Bu, MQMDE yapısını izleyen verilerin karakter kümesi tanıtıcısını belirtir; bu, MQMDE yapısındaki karakter verilerine uygulanmaz.

MQPOT ya da MQPUT1 çağrısında, uygulama bu alanı veri için uygun değere ayarlamalıdır. Kuyruk yöneticisi bu alanın geçerli olup olmadığını denetmiyor. Aşağıdaki özel değer kullanılabilir:

#### **MQCCSI\_INHERIT**

Character data in the data *takip edilen* this structure is in the same character set as this structure.

Kuyruk yöneticisi, bu değeri, iletide belirtilen yapının gerçek karakter kümesi tanıtıcısına göre değiştirir. Herhangi bir hata oluşmaması koşuluyla, MQCCSI\_INHERIT değeri MQGET çağrısıyla döndürülmez.

MQMD 'deki *PutApplType* alanının değeri MQAT\_BROKER ise, MQCCSI\_INHERIT kullanılamaz.

Bu değer aşağıdaki ortamlarda desteklenir:

-  AIX
-  IBM i
-  Linux
-  Windows

ve bu sistemlere bağlı IBM MQ istemcileri için geçerli olur.

Bu alanın ilk değeri MQCCSI\_UNDEFED değerine sahip.

### **Biçim (MQCHAR8)**

Bu, MQMDE yapısını izleyen verilerin biçim adını belirtir.

MQPOT ya da MQPUT1 çağrısında, uygulama bu alanı veri için uygun değere ayarlamalıdır. Kuyruk yöneticisi bu alanın geçerli olup olmadığını denetmiyor. Biçim adlarıyla ilgili daha fazla bilgi için “MQMD-İleti tanımlayıcı” sayfa 416 içinde açıklanan *Format* alanına bakın.

Bu alanın ilk değeri MQFMT\_NONE olur.

### **İşaretler (MQUZE)**

Aşağıdaki işaret belirlenebilir:

#### **MQMDEF\_NONE**

Bayrak yok.

Bu alanın ilk değeri MQMDEF\_NONE olur.

### **GroupId (MQBYTE24)**

“MQMD-İleti tanımlayıcı” sayfa 416 içinde açıklanan *GroupId* alanına bakın. Bu alanın ilk değeri MQGI\_NONE olur.

### **MsgSeqNumarası (MQUBE)**

“MQMD-İleti tanımlayıcı” sayfa 416 içinde açıklanan *MsgSeqNumber* alanına bakın. Bu alanın ilk değeri 1 'dir.

### **Görelî Konum (MQLONG)**

“MQMD-İleti tanımlayıcı” sayfa 416’inde açıklanan *Offset* alanına bakın. Bu alanın ilk değeri 0 'tır.

### **MsgFlags (MQUZE)**

“MQMD-İleti tanımlayıcı” sayfa 416’inde açıklanan *MsgFlags* alanına bakın. Bu alanın ilk değeri MQMF\_NONE olur.

### **OriginalLength (MQlong)**

“MQMD-İleti tanımlayıcı” sayfa 416’inde açıklanan *OriginalLength* alanına bakın. Bu alanın ilk değeri MQOL\_UNDEFINED olur.

## **MQMHBO-Arabellek seçeneklerine ileti tanıtıcısı**

MQMHBO yapısı, uygulamaların, arabelleklerin ileti tanıtıcılarından nasıl üretildiğini denetleyen seçenekleri belirtmesine olanak sağlar. Yapı, MQMHBUF çağrısındaki bir giriş değişikliğidir.

### **Karakter kümesi ve kodlama**

MQMHBO içindeki veriler, uygulamanın karakter kümesinde ve uygulamanın kodlamasında (MQENC\_NATIVE) yer almalıdır.

### **Alanlar**

**Not:** Aşağıdaki tabloda, alanlar alfabetik olarak değil, kullanıma göre gruplanır. Alt konular aynı sırayı izler.

Çizelge 505. MQMHBO 'daki Alanlar		
Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
StrucId (yapı tanıtıcısı)	MQMHBO_STRUC_ID	'MHBO'
Sürüm (yapı sürümü numarası)	MQMHBO_VERSION_1	1
Seçenekler (MQMHBUF işlemini denetleyen seçenekler)	MQMHBO_PROPERTIES_I N_MQRFH2	

**Notlar:**

- Boş değerli dizilim ya da boşluk değeri, C dizgisindeki boş dizgiyi ve diğer programlama dillerindeki boş karakterleri gösterir.
- C programlama dilinde, makro değişkeni MQMHBO\_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlar için başlangıç değerleri sağlamak üzere bunu aşağıdaki şekilde kullanın:

```
MQMHBO MyMHBO = {MQMHBO_DEFAULT};
```

### **Dil bildirimleri**

MQMHBO için C bildirimi

```
typedef struct tagMQMHBO MQMHBO;
struct tagMQMHBO {
    MQCHAR4  StrucId;          /* Structure identifier */
    MQLONG   Version;        /* Structure version number */
    MQLONG   Options;        /* Options that control the action of
                             MQMHBUF */
};
```

## MQMHB0 için COBOL bildirimi

```
** MQMHBO structure
10 MQMHBO.
**   Structure identifier
15 MQMHBO-STRUCID          PIC X(4).
**   Structure version number
15 MQMHBO-VERSION        PIC S9(9) BINARY.
**   Options that control the action of MQMHBUF
15 MQMHBO-OPTIONS        PIC S9(9) BINARY.
```

## MQMHB0 için PL/I bildirimi

```
Dcl
1 MQMHBO based,
3 StrucId      char(4),      /* Structure identifier */
3 Version      fixed bin(31), /* Structure version number */
3 Options      fixed bin(31), /* Options that control the action
                             of MQMHBUF */
```

## MQMHB0 için High Level Assembler (Üst Düzey Çevirici) bildirimi

```
MQMHB0          DSECT
MQMHBO_STRUCID  DS   CL4   Structure identifier
MQMHBO_VERSION  DS   F     Structure version number
MQMHBO_OPTIONS  DS   F     Options that control the
*                   action of MQMHBUF
MQMHBO_LENGTH   EQU  *-MQMHBO
MQMHBO_AREA     DS   CL(MQMHB0_LENGTH)
```

### **StrucId (MQCHAR4)**

İleti tanıtıcısı-arabellek seçenekleri yapısı- StrucId alanı

Bu, yapı tanıtıcısıdır. Değer şu olmalıdır:

#### **MQMHB0\_STRUCT\_ID**

Arabellek seçenekleri yapısıyla ileti tanıtıcısı için tanıtıcı.

C programlama dili için, MQMHBO\_STRUCT\_ID\_ARRAY değişmezi de tanımlıdır; bu değer MQMHBO\_STRUCT\_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisi olur.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQMHBO\_STRUCT\_ID 'dir.

### **Sürüm (MQUZE)**

İleti tanıtıcısı-arabellek seçenekleri yapısı-Sürüm alanı

Bu, yapı sürüm numarasıdır. Değer şu olmalıdır:

#### **MQMHB0\_VERSION\_1**

Arabellek seçenekleri yapısıyla ileti tanıtıcısı için sürüm numarası.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

#### **MQMHB0\_CURRENT\_VERSION**

İleti tutamacının arabellek seçenekleri yapısına ilişkin yürürlükteki sürümü.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQMHBO\_VERSION\_1' dir.

### **Seçenekler (MQUZE)**

İleti tanıtıcısı-arabellek seçenekleri yapısı-Seçenekler alanı

Bu seçenekler MQMHBUF ' un işlemini denetler.

Aşağıdaki seçeneği belirlemelisiniz:

## MQM HBO\_PROPERTIES\_IN\_MQRFH2

Özellikleri bir ileti tanıtıcısından arabelleğe dönüştürürken, bunları MQRFH2 biçimine dönüştürün.

İsteğe bağlı olarak, aşağıdaki seçeneği de belirleyebilirsiniz. birden çok seçenek belirtmek için, değerleri bir araya ekleyin (aynı değişmezi bir kereden fazla eklemeyin) ya da bit ya da işlemi kullanarak değerleri birleştirin (programlama dili, bit işlemleri destekliyse).

## MQM HBO\_DELETE\_PROPERTIES

Arabelleğe eklenen özellikler ileti tanıtıcısından silinir. Arama başarısız olursa özellikler silinmez.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQM HBO\_PROPERTIES\_IN\_MQRFH2' dir.

## MQOD-Nesne tanımlayıcı

MQOD yapısı, ada göre bir nesne belirtmek için kullanılır. Yapı, MQOPEN ve MQPUT1 çağrılarında bir giriş/ çıkış değiştirgesidir.

Aşağıdaki nesne tipleri geçerlidir:

- Kuyruk ya da dağıtım listesi
- Ad listesi
- Süreç tanımlaması
- Kuyruk yöneticisi
- Konu

## Kullanılabilirlik

Tüm IBM MQ sistemleri ve bu sistemlere bağlı IBM MQ MQI clients .

## Sürüm

Yürürlükteki MQOD sürümü: MQOD\_VERSION\_4. Birkaç ortam arasında bağlantı sağlamak istediğiniz uygulamalar, ilgili tüm ortamlarda MQOD ' un gerekli sürümünün desteklendiğinden emin olmalıdır. Yapının yalnızca daha yeni sürümlerinde bulunan alanlar, izleyen açıklamalarda bu şekilde tanımlanır.

Desteklenen programlama dilleri için sağlanan üstbilgi, COPY ve INCLUDE dosyaları, ortam tarafından desteklenen ancak *Version* alanının ilk değeri MQOD\_VERSION\_1olarak ayarlanmış MQOD ' un en son sürümünü içerir. version-1 yapısında var olmayan alanları kullanmak için uygulamanın *Version* alanını gerekli sürümün sürüm numarasına ayarlaması gerekir.

Bir dağıtım listesini açmak için *Version* , MQOD\_VERSION\_2 ya da daha büyük olmalıdır.

## Karakter kümesi ve kodlama

MQOD içindeki veriler, **CodedCharSetId** kuyruk yöneticisi özniteliği tarafından belirtilen karakter kümesinde ve MQENC\_NATIVE tarafından verilen yerel kuyruk yöneticisinin kodlamasında olmalıdır. Ancak, uygulama bir MQ MQI istemcisi olarak çalışıyorsa, yapı istemcinin karakter kümesinde ve kodlamasında olmalıdır.

## Alanlar

**Not:** Aşağıdaki tabloda, alanlar alfabetik olarak değil, kullanıma göre gruplanır. Alt konular aynı sırayı izler.

Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
<u>StrucId</u> (yapı tanıtıcısı)	MQOD_STRUC_ID	'0D--'
<u>Sürüm</u> (yapı sürümü numarası)	MQOD_VERSION_1	1
<u>ObjectType</u> (nesne tipi)	MQOT_Q	1

Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
<u>ObjectName</u> (nesne adı)	Yok	Boş değerli dizilim ya da boşluklar
<u>ObjectQMgrAdı</u> (nesne kuyruğu yöneticisi adı)	Yok	Boş değerli dizilim ya da boşluklar
<u>DynamicQName</u> (dinamik kuyruk adı)	Yok	'CSQ.*' on z/OS ; 'AMQ.*' tersi durumda
<u>AlternateUserKimliği</u> (diğer kullanıcı kimliği)	Yok	Boş değerli dizilim ya da boşluklar
<b>Not:</b> <i>Version</i> , MQOD_VERSION_2' den küçükse kalan alanlar yoksayılr.		
<u>RecsPresent</u> (var olan nesne kaydı sayısı)	Yok	0
<u>KnownDestSayı</u> (başarıyla açılan yerel kuyruk sayısı)	Yok	0
<u>UnknownDestCount</u> (başarıyla açılan uzak kuyruk sayısı)	Yok	0
<u>InvalidDestSayı</u> (açılamayan kuyrukların sayısı)	Yok	0
<u>ObjectRecGörelı Konum</u> (MQOD ' un başlangıcından ilk nesne kaydının görelı konumu)	Yok	0
<u>ResponseRecGörelı Konumu</u> (MQOD ' un başlangıcından ilk yanıt kaydının görelı konumu)	Yok	0
<u>ObjectRecPtr</u> (ilk nesne kaydının adresi)	Yok	Boş değerli gösterge ya da boş değerli byte
<u>ResponseRecPtr</u> (ilk yanıt kaydının adresi)	Yok	Boş değerli gösterge ya da boş değerli byte
<b>Not:</b> <i>Version</i> , MQOD_VERSION_3' den küçükse kalan alanlar yoksayılr.		
<u>AlternateSecurityTanıtıcısı</u> (diğer güvenlik tanıtıcısı)	MQSID_YOK	Boş Değerler
<u>ResolvedQName</u> (çözölen kuyruk adı)	Yok	Boş değerli dizilim ya da boşluklar
<u>ResolvedQMgrAd</u> (çözömlenen kuyruk yöneticisi adı)	Yok	Boş değerli dizilim ya da boşluklar
<b>Not:</b> <i>Version</i> , MQOD_VERSION_4' den küçükse kalan alanlar yoksayılr.		
<u>ObjectString</u> (uzun nesne adı)	MQCHARV_DEFAULT	MQCHARV için tanımlandığı şekilde
<u>SelectionString</u> (seçim dizgisi)	MQCHARV_DEFAULT	MQCHARV için tanımlandığı şekilde
<u>ResObjectDizesi</u> (çözömlenen uzun nesne adı)	MQCHARV_DEFAULT	MQCHARV için tanımlandığı şekilde
<u>ResolvedType</u> (çözömlenen nesne tipi)	MQOT_NONE	0

Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
<b>Notlar:</b>		
<ol style="list-style-type: none"> <li>1. – simgesi, tek bir boş karakteri temsil eder.</li> <li>2. Boş değerli dizilim ya da boşluk değeri, C dizgisindeki boş dizgiyi ve diğer programlama dillerindeki boş karakterleri gösterir.</li> <li>3. C programlama dilinde, makro değişkeniMQOD_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlar için başlangıç değerleri sağlamak üzere aşağıdaki şekilde kullanılabilir:</li> </ol>		
<pre>MQOD MyOD = {MQOD_DEFAULT};</pre>		

## Dil bildirimleri

MQOD için C bildirimi

```
typedef struct tagMQOD MQOD;
struct tagMQOD {
    MQCHAR4    StrucId;           /* Structure identifier */
    MQLONG     Version;          /* Structure version number */
    MQLONG     ObjectType;       /* Object type */
    MQCHAR48   ObjectName;       /* Object name */
    MQCHAR48   ObjectQMgrName;   /* Object queue manager name */
    MQCHAR48   DynamicQName;     /* Dynamic queue name */
    MQCHAR12   AlternateUserId;  /* Alternate user identifier */
    /* Ver:1 */
    MQLONG     RecsPresent;       /* Number of object records present */
    MQLONG     KnownDestCount;   /* Number of local queues opened
    successfully */
    MQLONG     UnknownDestCount; /* Number of remote queues opened
    successfully */
    MQLONG     InvalidDestCount; /* Number of queues that failed to
    open */
    MQLONG     ObjectRecOffset;  /* Offset of first object record from
    start of MQOD */
    MQLONG     ResponseRecOffset; /* Offset of first response record
    from start of MQOD */
    MQPTR      ObjectRecPtr;     /* Address of first object record */
    MQPTR      ResponseRecPtr;   /* Address of first response record */
    /* Ver:2 */
    MQBYTE40   AlternateSecurityId; /* Alternate security identifier */
    MQCHAR48   ResolvedQName;     /* Resolved queue name */
    MQCHAR48   ResolvedQMgrName;  /* Resolved queue manager name */
    /* Ver:3 */
    MQCHARV    ObjectString;      /* Object Long name */
    MQCHARV    SelectionString;   /* Message Selector */
    MQCHARV    ResObjectString;   /* Resolved Long object name*/
    MQLONG     ResolvedType       /* Alias queue resolved
    object type */
    /* Ver:4 */
};
```

MQOD için COBOL bildirimi

```
** MQOD structure
10  MQOD.
** Structure identifier
15  MQOD-STRUCID                PIC X(4).
** Structure version number
15  MQOD-VERSION                PIC S9(9) BINARY.
** Object type
15  MQOD-OBJECTTYPE            PIC S9(9) BINARY.
** Object name
15  MQOD-OBJECTNAME            PIC X(48).
** Object queue manager name
15  MQOD-OBJECTQGRNAME         PIC X(48).
** Dynamic queue name
15  MQOD-DYNAMICQNAME          PIC X(48).
```

```

** Alternate user identifier
15 MQOD-ALTERNATEUSERID PIC X(12).
** Number of object records present
15 MQOD-RECSPRESENT PIC S9(9) BINARY.
** Number of local queues opened successfully
15 MQOD-KNOWNDDESTCOUNT PIC S9(9) BINARY.
** Number of remote queues opened successfully
15 MQOD-UNKNOWNDDESTCOUNT PIC S9(9) BINARY.
** Number of queues that failed to open
15 MQOD-INVALIDDESTCOUNT PIC S9(9) BINARY.
** Offset of first object record from start of MQOD
15 MQOD-OBJECTRECOFFSET PIC S9(9) BINARY.
** Offset of first response record from start of MQOD
15 MQOD-RESPONSERECOFFSET PIC S9(9) BINARY.
** Address of first object record
15 MQOD-OBJECTRECPTER POINTER.
** Address of first response record
15 MQOD-RESPONSERECPTR POINTER.
** Alternate security identifier
15 MQOD-ALTERNATESECURITYID PIC X(40).
** Resolved queue name
15 MQOD-RESOLVEDQNAME PIC X(48).
** Resolved queue manager name
15 MQOD-RESOLVEDQMGRNAME PIC X(48).
** Object Long name
15 MQOD-OBJECTSTRING.
** Address of variable length string
20 MQOD-OBJECTSTRING-VSPTR POINTER.
** Offset of variable length string
20 MQOD-OBJECTSTRING-VSOFFSET PIC S9(9) BINARY.
** size of buffer
20 MQOD-OBJECTSTRING-VSBUFSIZE PIC S9(9) BINARY.
** Length of variable length string
20 MQOD-OBJECTSTRING-VSLENGTH PIC S9(9) BINARY.
** CCSID of variable length string
20 MQOD-OBJECTSTRING-VSCCSID PIC S9(9) BINARY.
** Message Selector
15 MQOD-SELECTIONSTRING.
** Address of variable length string
20 MQOD-SELECTIONSTRING-VSPTR POINTER.
** Offset of variable length string
20 MQOD-SELECTIONSTRING-VSOFFSET PIC S9(9) BINARY.
** size of buffer
20 MQOD-SELECTIONSTRING-VSBUFSIZE PIC S9(9) BINARY.
** Length of variable length string
20 MQOD-SELECTIONSTRING-VSLENGTH PIC S9(9) BINARY.
** CCSID of variable length string
20 MQOD-SELECTIONSTRING-VSCCSID PIC S9(9) BINARY.
** Resolved Long object name
15 MQOD-RESOBJECTSTRING.
** Address of variable length string
20 MQOD-RESOBJECTSTRING-VSPTR POINTER.
** Offset of variable length string
20 MQOD-RESOBJECTSTRING-VSOFFSET PIC S9(9) BINARY.
** size of buffer
20 MQOD-RESOBJECTSTRING-VSBUFSIZE PIC S9(9) BINARY.
** Length of variable length string
20 MQOD-RESOBJECTSTRING-VSLENGTH PIC S9(9) BINARY.
** CCSID of variable length string
20 MQOD-RESOBJECTSTRING-VSCCSID PIC S9(9) BINARY.
** Alias queue resolved object type
15 MQOD-RESOLVEDTYPE PIC S9(9) BINARY.

```

## MQOD için PL/I bildirimi

```

dcl
1 MQOD based,
3 StrucId char(4), /* Structure identifier */
3 Version fixed bin(31), /* Structure version number */
3 ObjectType fixed bin(31), /* Object type */
3 ObjectName char(48), /* Object name */
3 ObjectQMgrName char(48), /* Object queue manager name */
3 DynamicQName char(48), /* Dynamic queue name */
3 AlternateUserId char(12), /* Alternate user identifier */
3 RecsPresent fixed bin(31), /* Number of object records
present */
3 KnownDestCount fixed bin(31), /* Number of local queues opened
successfully */
3 UnknownDestCount fixed bin(31), /* Number of remote queues opened

```



```

3 InvalidDestCount    fixed bin(31), /* Number of queues that failed to
                    successfully */
                    open */
3 ObjectRecOffset     fixed bin(31), /* Offset of first object record
                    from start of MQOD */
3 ResponseRecOffset   fixed bin(31), /* Offset of first response record
                    from start of MQOD */
3 ObjectRecPtr        pointer, /* Address of first object record */
3 ResponseRecPtr      pointer, /* Address of first response
                    record */
3 AlternateSecurityId char(40), /* Alternate security identifier */
3 ResolvedQName       char(48), /* Resolved queue name */
3 ResolvedQMgrName    char(48), /* Resolved queue manager name */
3 ObjectString, /* Object Long name */
5 VSPtr              pointer, /* Address of variable length string */
5 VSOffset           fixed bin(31), /* Offset of variable length string */
5 VSBufSize          fixed bin(31), /* size of buffer */
5 VSLength           fixed bin(31), /* Length of variable length string */
5 VSCCSID            fixed bin(31), /* CCSID of variable length string */
3 SelectionString, /* Message Selection */
5 VSPtr              pointer, /* Address of variable length string */
5 VSOffset           fixed bin(31), /* Offset of variable length string */
5 VSBufSize          fixed bin(31), /* size of buffer */
5 VSLength           fixed bin(31), /* Length of variable length string */
5 VSCCSID            fixed bin(31), /* CCSID of variable length string */
3 ResObjectString, /* Resolved Long object name */
5 VSPtr              pointer, /* Address of variable length string */
5 VSOffset           fixed bin(31), /* Offset of variable length string */
5 VSBufSize          fixed bin(31), /* size of buffer */
5 VSLength           fixed bin(31), /* Length of variable length string */
5 VSCCSID            fixed bin(31), /* CCSID of variable length string */
3 ResolvedType        fixed bin(31); /* Alias queue resolved object type */

```

#### MQOD için High Level Assembler (Üst Düzey Çevirici) bildirimi

```

MQOD                DSECT
MQOD_STRUCID        DS    CL4  Structure identifier
MQOD_VERSION        DS    F    Structure version number
MQOD_OBJECTTYPE     DS    F    Object type
MQOD_OBJECTNAME     DS    CL48  Object name
MQOD_OBJECTQMGRNAME DS    CL48  Object queue manager name
MQOD_DYNAMICQNAME   DS    CL48  Dynamic queue name
MQOD_ALTERNATEUSERID DS    CL12  Alternate user identifier
MQOD_RECSPRESENT    DS    F    Number of object records present
MQOD_KNOWNDESTCOUNT DS    F    Number of local queues opened
*
MQOD_UNKNOWNDSTCOUNT DS    F    Number of remote queues opened
*
MQOD_INVALIDDESTCOUNT DS    F    Number of queues that failed to
*
MQOD_OBJECTRECOFFSET DS    F    Offset of first object record from
*
MQOD_RESPONSERECOFFSET DS    F    Offset of first response record
*
MQOD_OBJECTRECPTTR DS    F    Address of first object record
MQOD_RESPONSERECPTTR DS    F    Address of first response record
MQOD_ALTERNATESECURITYID DS    XL40  Alternate security identifier
MQOD_RESOLVEDQNAME   DS    CL48  Resolved queue name
MQOD_RESOLVEDQMGRNAME DS    CL48  Resolved queue manager name
MQOD_OBJECTSTRING   DS    F    Object Long name
MQOD_OBJECTSTRING_VSPTR DS    F    Address of variable length string
MQOD_OBJECTSTRING_VSOFFSET DS    F    Offset of variable length string
MQOD_OBJECTSTRING_VSBUFSIZE DS    F    size of buffer
MQOD_OBJECTSTRING_VSLLENGTH DS    F    Length of variable length string
MQOD_OBJECTSTRING_VSCCSID DS    F    CCSID of variable length string
MQOD_OBJECTSTRING_LENGTH EQU    *- MQOD_OBJECTSTRING
ORG    MQOD_OBJECTSTRING
MQOD_OBJECTSTRING_AREA DS    CL(MQOD_OBJECTSTRING_LENGTH)
*
MQOD_SELECTIONSTRING DS    F    Message Selector
MQOD_SELECTIONSTRING_VSPTR DS    F    Address of variable length string
MQOD_SELECTIONSTRING_VSOFFSET DS    F    Offset of variable length string
MQOD_SELECTIONSTRING_VSBUFSIZE DS    F    size of buffer
MQOD_SELECTIONSTRING_VSLLENGTH DS    F    Length of variable length string
MQOD_SELECTIONSTRING_VSCCSID DS    F    CCSID of variable length string
MQOD_SELECTIONSTRING_LENGTH EQU    *- MQOD_SELECTIONSTRING
ORG    MQOD_SELECTIONSTRING
MQOD_SELECTIONSTRING_AREA DS    CL(MQOD_SELECTIONSTRING_LENGTH)
*

```

```

MQOD_RESOBJECTSTRING      DS    F    Resolved Long object name
MQOD_RESOBJECTSTRING_VSPTR DS    F    Address of variable length string
MQOD_RESOBJECTSTRING_VSOFFSET DS    F    Offset of variable length string
MQOD_RESOBJECTSTRING_VSBUFSIZE DS    F    size of buffer
MQOD_RESOBJECTSTRING_VSLENGTH DS    F    Length of variable length string
MQOD_RESOBJECTSTRING_VSCCSID DS    F    CCSID of variable length string
MQOD_RESOBJECTSTRING_LENGTH EQU    *- MQOD_RESOBJECTSTRING
                           ORG    MQOD_RESOBJECTSTRING
MQOD_RESOBJECTSTRING_AREA DS    CL(MQOD_RESOBJECTSTRING_LENGTH)
MQOD_RESOLVEDTYPE         DS    F    Alias queue object resolved type
*
MQOD_LENGTH               EQU    *-MQOD
                           ORG    MQOD
MQOD_AREA                 DS    CL(MQOD_LENGTH)

```

## MQOD için Visual Basic bildirimi

```

Type MQOD
  StrucId           As String*4  'Structure identifier'
  Version           As Long      'Structure version number'
  ObjectType        As Long      'Object type'
  ObjectName        As String*48  'Object name'
  ObjectQMgrName    As String*48  'Object queue manager name'
  DynamicQName      As String*48  'Dynamic queue name'
  AlternateUserId   As String*12  'Alternate user identifier'
  RecsPresent       As Long      'Number of object records present'
  KnownDestCount    As Long      'Number of local queues opened'
                               'successfully'
  UnknownDestCount  As Long      'Number of remote queues opened'
                               'successfully'
  InvalidDestCount  As Long      'Number of queues that failed to'
                               'open'
  ObjectRecOffset   As Long      'Offset of first object record from'
                               'start of MQOD'
  ResponseRecOffset As Long      'Offset of first response record'
                               'from start of MQOD'
  ObjectRecPtr      As MQPTR      'Address of first object record'
  ResponseRecPtr    As MQPTR      'Address of first response record'
  AlternateSecurityId As MQBYTE40 'Alternate security identifier'
  ResolvedQName     As String*48  'Resolved queue name'
  ResolvedQMgrName  As String*48  'Resolved queue manager name'
End Type

```

## StrucId (MQCHAR4)

Bu, yapı tanıtıcısıdır; değer şöyle olmalıdır:

### MQOD\_STRUC\_ID

Nesne tanımlayıcı yapısına ilişkin tanıtıcı.

C programlama dili için, MQOD\_STRUC\_ID\_ARRAY değişmezi de tanımlanır; bu, MQOD\_STRUC\_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisidir.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQOD\_STRUC\_ID ' dir.

## Sürüm (MQUZE)

Bu, yapı sürüm numarasıdır; değer in aşağıdakilerden biri olması gerekir:

### MQOD\_VERSION\_1

Version-1 nesne tanımlayıcı yapısı.

### MQOD\_VERSION\_2

Version-2 nesne tanımlayıcı yapısı.

### MQOD\_VERSION\_3

Version-3 nesne tanımlayıcı yapısı.

### MQOD\_VERSION\_4

Version-4 nesne tanımlayıcı yapısı.

Tüm sürümlerin tümü IBM MQ V7.0 ortamlarında desteklenir.

Yalnızca yapının daha yeni sürümlerinde var olan alanlar, alanların açıklamalarında olduğu gibi tanımlanır. Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

#### **MQOD\_CURRENT\_VERSION**

Nesne tanımlayıcı yapısının yürürlükteki sürümü.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQOD\_VERSION\_1' dir.

#### **ObjectType (MQLONG)**

Nesne tanımlayıcıda adlandırılmakta olan nesnenin tipi. Olası değerler şunlardır:

#### **MQOT\_CLNTCONN\_CHANNEL**

İstemci bağlantı kanalı. Nesnenin adı *ObjectName* alanında bulunur.

#### **MQOT\_Q**

Sıraya girin. Nesnenin adı *ObjectName* alanında bulunur.

#### **MQOT\_NAMELIST**

-Namelist. Nesnenin adı *ObjectName* alanında bulunur

#### **MQOT\_PROCESS**

Süreç tanımlaması. Nesnenin adı *ObjectName* alanında bulunur

#### **MQOT\_Q\_MGR**

Kuyruk yöneticisi. Nesnenin adı *ObjectName* alanında bulunur

#### **MQOT\_KONU**

Konu. Tam konu adı iki farklı alandan oluşturulabilir: *ObjectName* ve *ObjectString*.

Bu iki alanın nasıl kullanıldığına ilişkin ayrıntılar için [Konu dizgilerini birleştirme](#) başlıklı konuya bakın.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQOT\_Q.

#### **ObjectName (MQCHAR48)**

Bu, *ObjectQMgrName* ile tanımlanan kuyruk yöneticisinde tanımlandığı şekilde, nesnenin yerel adıdır. Ad aşağıdaki karakterleri içerebilir:

- Büyük harf alfabetik karakterler (A-Z)
- Küçük harf alfabetik karakterler (a-z)
- Sayısal basamaklar (0-9)
- Nokta (.), eğik çizgi (/), altçizgi (\_), yüzde (%)

Ad başta ya da gömülü boşluklar içermemelidir, ancak sonda boşluklar içerebilir. Addaki önemli verilerin sonunu göstermek için boş bir karakter kullanın; bu karakteri izleyen karakterler ve boş değer boşluk olarak değerlendirilir. Belirtilen ortamlarda aşağıdaki kısıtlamalar geçerlidir:

- EBCDIC Katakana kullanan sistemlerde küçük harfli karakterler kullanılamaz.
- z/OS'ta:
  - Alt çizgiyle başlayan ya da biten adlardan kaçınınız; bunlar işlemler ve denetim panoları tarafından işlenemez.
  - Yüzde karakterinin RACFiçin özel bir anlamı vardır. Dış güvenlik yöneticisi olarak RACF kullanılırsa, adlar yüzdeyi içermemelidir. Varsa, RACF soysal tanımları kullanıldığında bu adlar güvenlik denetimlerine dahil edilmez.
- IBM üzerinde, komutlarda belirtildiğinde küçük harf, eğik çizgi ya da yüzde içeren adlar tırnak içine alınmalıdır. Bu tırnak işaretleri, yapılarda alan olarak ya da çağrılarda parametre olarak oluşan adlar için belirlenmemelidir.

Tam konu adı iki farklı alandan oluşturulabilir: *ObjectName* ve *ObjectString*. Bu iki alanın nasıl kullanıldığına ilişkin ayrıntılar için [Konu dizgilerini birleştirme](#) başlıklı konuya bakın.

Aşağıdaki noktalar, belirtilen nesne tipleri için geçerlidir:

- *ObjectName* bir model kuyruğunun adıysa, kuyruk yöneticisi model kuyruğunun özneliklerini içeren dinamik bir kuyruk yaratır ve *ObjectName* alanında yaratılan kuyruğun adını döndürür. Model kuyruğu yalnızca MQOPEN çağrısında belirtilebilir; MQPUT1 çağrısında model kuyruğu geçerli değil.
- *ObjectName* , TARGTYPE (TOPIC) ile bir diğer ad kuyruğunun adıysa, önce adı belirtilen diğer ad kuyruğunda bir güvenlik denetimi yapılır; diğer ad kuyrukları kullanıldığında bu olağandır. Güvenlik denetimi başarıyla tamamlandığında, MQOPEN çağrısı devam eder ve bir MQOT\_TOPIC; bu, denetim konusu nesnesiyle ilgili bir güvenlik denetimi yapmayı içerir.
- *ObjectName* ve *ObjectQMgrName* , yerel kuyruk yöneticisinin ait olduğu kuyruk paylaşım grubunun sahip olduğu bir paylaşılan kuyruğu tanımlarsa, yerel kuyruk yöneticisinde aynı ada sahip bir kuyruk tanımı da olmamalıdır. Böyle bir tanımlama (yerel kuyruk, diğer ad kuyruğu, uzak kuyruk ya da model kuyruğu) varsa, çağrı MQRC\_OBJECT\_NOT\_UNIQUE neden koduyla başarısız olur.
- Açılmakta olan nesne bir dağıtım listesiye (yani, *RecsPresent* var ve sıfırdan büyük), *ObjectName* boş değerli ya da boş değerli olmalıdır. Bu koşul karşılanmazsa, çağrı MQRC\_OBJECT\_NAME\_ERROR neden koduyla başarısız olur.
- *ObjectType* MQOT\_Q\_MGR ise, özel kurallar geçerli olur; bu durumda ad, alanın ilk boş karakterine ya da sonuna kadar tamamen boş olmalıdır.

Bu, *ObjectName* bir model kuyruğunun adı ve diğer tüm durumlarda yalnızca giriş alanı olduğunda MQOPEN çağrısına ilişkin bir giriş/çıkış alanıdır. Bu alanın uzunluğu MQ\_Q\_NAME\_LENGTH tarafından verilir. Bu alanın ilk değeri, C içindeki boş dizgi ve diğer programlama dillerindeki 48 boş karakterdir.

### **ObjectQMgrAdı (MQCHAR48)**

Bu ad, *ObjectName* nesnesinin tanımlandığı kuyruk yöneticisinin adıdır. Adda geçerli olan karakterler, *ObjectName* ile aynıdır (bkz. "[ObjectName \(MQCHAR48\)](#)" sayfa 479 ). İlk boş karakteri ya da alanın sonuna kadar boş olan bir ad, uygulamanın bağlı olduğu kuyruk yöneticisini belirtir (yerel kuyruk yöneticisi).

Aşağıdaki noktalar, gösterilen nesne tipleri için geçerlidir:

- *ObjectType* MQOT\_TOPIC, MQOT\_NAMELIST, MQOT\_PROCESS ya da MQOT\_Q\_MGR ise, *ObjectQMgrName* boş bırakılmalı ya da yerel kuyruk yöneticisinin adı olmalıdır.
- If *ObjectName* is the name of a model queue, the queue manager creates a dynamic queue with the attributes of the model queue, and returns in the *ObjectQMgrName* field the name of the queue manager on which the queue is created; this is the name of the local queue manager. Model kuyruğu yalnızca MQOPEN çağrısında belirtilebilir; MQPUT1 çağrısında bir model kuyruğu geçerli değildir.
- *ObjectName* bir küme kuyruğunun adıysa ve *ObjectQMgrName* boşsa, kuyruk yöneticisi tarafından döndürülen kuyruk tanıtıcısı kullanılarak gönderilen iletilerin hedefi, kuyruk yöneticisi (ya da kuruluysa küme iş yükü çıkışı) tarafından aşağıdaki gibi seçilir:
  - MQOO\_BIND\_ON\_OPEN belirtilirse, kuyruk yöneticisi, MQOP çağrısını işlerken küme kuyruğunun belirli bir eşgörünümünü seçer ve bu kuyruk tanıtıcısı kullanılarak yapılan tüm iletiler o yönetim ortamına gönderilir.
  - MQOO\_BIND\_NOT\_FIXED belirtildiyse, kuyruk yöneticisi, bu kuyruk tanıtıcısını kullanan her bir MQPUT çağrısına ilişkin hedef kuyruğun farklı bir eşgörünümünü (kümedeki farklı bir kuyruk yöneticisinde bulunuyor) seçebilir.

Uygulamanın, bir küme kuyruğunun *belirli* bir örneğine (örneğin, kümedeki belirli bir kuyruk yöneticisinde bulunan bir kuyruk örneğine) ileti göndermesi gerekiyorsa, uygulamanın *ObjectQMgrName* alanında o kuyruk yöneticisinin adını belirtmesi gerekir. Bu, yerel kuyruk yöneticisini, iletiyi belirtilen hedef kuyruk yöneticisine göndermesini sağlar.

- *ObjectName* paylaşılan bir kuyruğun adıdır. Yerel kuyruk yöneticisinin ait olduğu kuyruk paylaşım grubunun sahibi olduğu *ObjectQMgrName* , kuyruk paylaşım grubunun adı, yerel kuyruk yöneticisinin adı ya da boş; bu değerlerden hangisi belirtilirse, ileti aynı kuyruğa yerleştirilir.

Kuyruk paylaşım grupları yalnızca z/OSüzerinde desteklenir.

- *ObjectName* , bir uzak kuyruk paylaşım grubunun (yerel kuyruk yöneticisinin ait olmadığı bir kuyruk paylaşım grubu) sahip olduğu bir paylaşılan kuyruğun adıdır. *ObjectQMgrName* , kuyruk paylaşım

grubunun adı olmalıdır. Bu gruba ait olan bir kuyruk yöneticisinin adını kullanabilirsiniz; ancak, ileti kuyruk paylaşım grubuna geldiğinde, belirli bir kuyruk yöneticisi kullanılamıyorsa, bu durum iletiyi geciktirebilir.

- Açılmakta olan nesne bir dağıtım listese (yani, *RecsPresent* sıfırdan büyükse), *ObjectQMGrName* boş olmalı ya da boş dizgi olmalıdır. Bu koşul karşılanmazsa, çağrı neden kodu MQRC\_OBJECT\_Q\_MGR\_NAME\_ERROR neden koduyla başarısız olur.

Bu, *ObjectName*, bir model kuyruğunun adı ve diğer tüm durumlarda yalnızca giriş alanı olduğunda, MQOPER çağrısına ilişkin bir giriş/çıkış alanıdır. Bu alanın uzunluğu MQ\_Q\_MGR\_NAME\_LENGTH değeri tarafından verilir. Bu alanın başlangıç değeri, C ' de boş değer dizgisidir ve diğer programlama dillerindeki 48 boş karakter dizilimidir.

### **DynamicQName (MQCHAR48)**

Bu, MQOPEN çağrısıyla yaratılacak dinamik bir kuyruğun adıdır. This is of relevance only when *ObjectName* specifies the name of a model queue; in all other cases *DynamicQName* is ignored.

Adda geçerli olan karakterler, bir yıldız işaretinin de geçerli olması dışında, *ObjectName* için geçerli olan karakterlerle aynıdır. Bir model kuyruğunun adı *ObjectName* ise, boş olan (ya da ilk boş karakter karakterinden önce boşluk oluşan bir ad) geçersiz bir ad geçerli değildir.

Addaki son boş olmayan karakter bir yıldız işaretiyse (\*), kuyruk yöneticisi, yerel kuyruk yöneticisinde kuyruk için oluşturulan adın benzersiz olduğunu garanti eden bir karakter dizgisiyle yıldız işaretini değiştirir. Bu konuda yeterli sayıda karakter sayısına izin vermek için, yıldız imi yalnızca 1 ile 33 arasındaki konumlarda geçerlidir. Yıldız işaretinden sonra boşluk ya da boş karakter dışında bir karakter olmamalıdır.

Yıldız iminin ilk karakter konumunda olması geçerlidir; bu durumda ad yalnızca kuyruk yöneticisi tarafından oluşturulan karakterlerden oluşur.

z/OS' ta, otomatik olarak üretilen tam adı taşıyan bir kuyrukta güvenlik denetimi yapılmadığı için, ilk karakter konumunda yıldız işaretiyle ad kullanmayın.

Bu bir giriş alanıdır. Bu alanın uzunluğu MQ\_Q\_NAME\_LENGTH değeri tarafından verilir. Bu alanın ilk değeri ortam tarafından belirlenir:

- z/OSüzerinde değer 'CSQ.\*' dir.
- Diğer platformlarda değer 'AMQ.\*' dir.

Değer, C ' de boş karakterle biten bir dizedir ve diğer programlama dillerinde boşlukla doldurulmuş bir dizgidir.

### **AlternateUserTanıtıcısı (MQCHAR12)**

MQOPENçağrısı için MQOO\_ALTERNATE\_USER\_AUTHORITY ya da MQPUT1 çağrısı için MQPMO\_ALTERNATE\_USER\_AUTHORITY değerini belirlerseniz, bu alan, uygulamanın şu anda altında çalışmakta olduğu kullanıcı kimliğinin yerine, açık olan yetkiyi denetlemek için kullanılan bir alternatif kullanıcı kimliği içerir. Ancak, bazı denetimler yürürlükteki kullanıcı kimliğiyle (bağlam denetimleri gibi) hala gerçekleştirilmeye devam eder.

MQOO\_ALTERNATE\_USER\_AUTHORITY ya da MQPMO\_ALTERNATE\_USER\_AUTHORITY belirtildiyse ve bu alan, ilk boş karaktere ya da alanın sonuna kadar tamamen boşsa, açık olan bu nesneyi belirtilen seçeneklerle açmak için kullanıcı yetkisi gerekmiyorsa, açma işlemi başarılı olabilir.

Ne MQOO\_ALTERNATE\_USER\_AUTHORITY ne de MQPMO\_ALTERNATE\_USER\_AUTHORITY belirtilirse, bu alan yoksayılr.

Belirtilen ortamlarda aşağıdaki farklar vardır:

- z/OS'ta, açık için yetkilendirmeyi denetlemek için *AlternateUserId* ' un yalnızca ilk 8 karakteri kullanılır. Ancak, yürürlükteki kullanıcı kimliğinin bu belirli bir diğer kullanıcı kimliğini belirtme yetkisi olması gerekir; bu denetim için, diğer kullanıcı kimliğinin 12 karakteri de kullanılır. Kullanıcı kimliği, yalnızca dış güvenlik yöneticisi tarafından izin verilen karakterleri içermelidir.

Bir kuyruk için *AlternateUserId* belirtilmişse, değer konduğunda kuyruk yöneticisi tarafından bu değer kullanılabilir. MQPUT ya da MQPUT1 çağrısında belirtilen MQPMO\_\*\_CONTEXT seçenekleri, kuyruk yöneticisinin kimlik bağlamı bilgilerini üretmesine neden olursa, kuyruk yöneticisi *AlternateUserId* ögesini, yürürlükteki kullanıcı kimliğinin yerine, iletinin MQMD 'deki *UserIdentifier* alanına yerleştirir.

- Diğer ortamlarda, *AlternateUserId* yalnızca, açılmakta olan nesnede bulunan erişim denetimi denetimleri için kullanılır. If the object is a queue, *AlternateUserId* does not affect the content of the *UserIdentifier* field in the MQMD of messages sent using that queue handle.

Bu bir giriş alanıdır. Bu alanın uzunluğu MQ\_USER\_ID\_LENGTH tarafından verilir. Bu alanın ilk değeri, C içindeki boş değerli dizidir ve diğer programlama dillerindeki 12 boş karakterdir.

### **RecsPresent (MQUZE)**

Bu, uygulama tarafından sağlanan MQOR nesne kayıtlarının sayısıdır. If this number is greater than zero, it indicates that a distribution list is being opened, with *RecsPresent* being the number of destination queues in the list. Dağıtım listesi yalnızca bir hedef içerebilir.

The value of *RecsPresent* must not be less than zero, and if it is greater than zero *ObjectType* must be MQOT\_Q; the call fails with reason code MQRC\_RECS\_PRESENT\_ERROR if these conditions are not satisfied.

z/OSüzerinde bu alan sıfır olmalıdır.

Bu bir giriş alanıdır. Bu alanın ilk değeri 0 'tır. *Version* , MQOD\_VERSION\_2değerinden küçükse bu alan yok sayılır.

### **KnownDestSayı (MQUZE)**

Bu, dağıtım listesindeki yerel kuyruklara çözülen ve başarıyla açılan kuyrukların sayısıdır. Bu sayı, uzak kuyruklara çözülen kuyrukları içermez (iletiyi saklamak için yerel bir iletim kuyruğu kullanılmalı olsa da). Bu alan varsa, dağıtım listesinde olmayan tek bir kuyruk açılırken da bu alan ayarlanır.

Bu bir çıkış alanıdır. Bu alanın ilk değeri 0 'tır. *Version* , MQOD\_VERSION\_1değerinden küçükse bu alan yok sayılır.

### **UnknownDestSayı (MQUZE)**

Bu, dağıtım listesindeki uzak kuyruklara çözülen ve başarıyla açılan kuyrukların sayısıdır. Bu alan varsa, dağıtım listesinde olmayan tek bir kuyruk açılırken da bu alan ayarlanır.

Bu bir çıkış alanıdır. Bu alanın ilk değeri 0 'tır. *Version* , MQOD\_VERSION\_1değerinden küçükse bu alan yok sayılır.

### **InvalidDestSayı (MQUZE)**

Dağıtım listesindeki kuyrukların sayısı, başarıyla açılmamaktadır. Bu alan varsa, dağıtım listesinde olmayan tek bir kuyruk açılırken da bu alan ayarlanır.

**Not:** Bu alan varsa, bu alan yalnızca, MQOPEN ya da MQPUT1 çağrısındaki **CompCode** parametresi MQCC\_OK ya da MQCC\_WARNING; ise, **CompCode** parametresi MQCC\_FAILED ise bu değer belirlenmezse ayarlanır.

Bu bir çıkış alanıdır. Bu alanın ilk değeri 0 'tır. *Version* , MQOD\_VERSION\_1değerinden küçükse bu alan yok sayılır.

### **ObjectRecGörelili Konumu (MQUZE)**

Bu, MQOD yapısının başlangıcındaki ilk MQOR nesne kaydının bayt cinsinden görelili konudur. Görelili konum pozitif ya da negatif olabilir. *ObjectRecOffset* , yalnızca bir dağıtım listesi açılırken kullanılır. *RecsPresent* sıfırda, alan yoksayıdır.

Bir dağıtım listesi açılırken, dağıtım listesindeki hedef kuyrukların adlarını belirtmek için bir ya da daha çok MQOR nesne kaydı dizisi sağlanmalıdır. Bu işlem aşağıdaki iki yoldan biriyle yapılabilir:

- *ObjectRecOffset* görelili konum alanını kullanarak.

Bu durumda, uygulama bir MQOD içeren kendi yapısını, izleyen MQOR kayıtları dizisini (gerekli sayıda dizi ögesi ile) bildirmeli ve *ObjectRecOffset* değerini, MQOD 'nin başlangıcındaki dizideki ilk ögenin görelili konumuna ayarlamalıdır. Bu görelili konumun doğru olduğundan ve bir MQlong içinde konaklayabilecek bir değere sahip olduğundan emin olun (en kısıtlayıcı programlama dili COBOL 'dir; geçerli aralık, +999 999 999 ile 999 999).

İşaretçi veri tipini desteklemeyen programlama dilleri için *ObjectRecOffset* kullanın ya da gösterge veri tipini, farklı ortamlara (örneğin, COBOL programlama dili) portable olmayan bir biçimde uygulayın.

- *ObjectRecPtr* işaretçi alanını kullanarak.

Bu durumda uygulama, MQOR yapılarının dizisini, MQOD yapısından ayrı olarak bildirebilir ve *ObjectRecPtr* 'yi dizinin adresine ayarlayabilir.

İşaretçi veri tipini, farklı ortamlara (örneğin, C programlama dili) taşınabilir bir şekilde destekleyen programlama dilleri için *ObjectRecPtr* kullanın.

Seçtiğiniz teknik her neyse, *ObjectRecOffset* ve *ObjectRecPtr* 'lerden birini kullanın; Her ikisi de sıfır ya da her ikisi de sıfır değilse, çağrı neden kodu MQRC\_OBJECT\_RECORDS\_ERROR ile başarısız olur.

Bu bir giriş alanıdır. Bu alanın ilk değeri 0 'tır. *Version* , MQOD\_VERSION\_2 değerinden küçükse bu alan yok sayılır.

### **ResponseRecGörelili Konumu (MQUZE)**

Bu, MQOD yapısının başlangıcındaki ilk MQRR yanıt kaydının bayt cinsinden görelili konutdur. Görelili konum pozitif ya da negatif olabilir. *ResponseRecOffset* , yalnızca bir dağıtım listesi açılırken kullanılır. *RecsPresent* sıfırda, alan yok sayılır.

Bir dağıtım listesi açılırken, açıklamadığını (MQRR 'deCompCode alanı) ve her başarısızlığın nedenini (MQRR' deReason alanı) tanımlamak için bir ya da daha çok MQRR yanıt kaydı dizisi sağlayabilirsiniz. Veri, nesne kayıtları dizisinde yer alan kuyruk adlarıyla aynı sırayla yanıt kayıtları dizisine döndürülür. Kuyruk yöneticisi yanıt kayıtlarını yalnızca, arama sonucu karışık olduğunda (yani, bazı kuyruklar başarılı olurken başarılı bir şekilde açıldı ya da tüm başarısız oldu, ancak farklı nedenlerden dolayı) ayarlar; neden kodu çağrıdan MQRC\_MULTIPLE\_REASONS kodu bu vakayı belirtir. Aynı neden kodu tüm kuyruklar için de geçerliyse, bu neden MQOL ya da MQPUT1 çağrısının **Reason** parametresine döndürülür ve yanıt kayıtları belirlenmez. Yanıt kayıtları isteğe bağlıdır, ancak bunlar belirtilirsethem' un *RecsPresent* olması gerekir.

Yanıt kayıtları, nesne kayıtlarıyla aynı şekilde, *ResponseRecOffset* içinde bir görelili konum belirtilerek ya da *ResponseRecPtr* 'ta bir adres belirterek sağlanabilir. Bunu nasıl yapacağına ilişkin ayrıntılar için bkz. "[ObjectRecGörelili Konumu \(MQUZE\)](#)" sayfa 482. Ancak, *ResponseRecOffset* ve *ResponseRecPtr* sayısı birden fazla kullanılamaz; her ikisi de sıfır değilse, çağrı neden kodu MQRC\_RESPONSE\_RECORDS\_ERROR ile başarısız olur.

MQPUT1 çağrısı için, bu yanıt kayıtları, ileti dağıtım listesindeki kuyruklara gönderildiğinde oluşan hatalarla ilgili bilgileri ve kuyruklar açıldığında oluşan hatalarla ilgili bilgileri döndürmek için kullanılır. Bir kuyruk için koyma işlemindeki tamamlanma kodu ve neden kodu, yalnızca ikinciden gelen tamamlanma kodu MQCC\_OK ya da MQCC\_UYARI olduğunda, bu kuyruğa ilişkin açma işleminden sonra bu kuyruğa ilişkin neden kodu değiştirilir.

Bu bir giriş alanıdır. Bu alanın ilk değeri 0 'tır. *Version* , MQOD\_VERSION\_2 değerinden küçükse bu alan yok sayılır.

### **ObjectRecPtr (MQPTR)**

Bu, ilk MQOR nesne kaydının adresidir. *ObjectRecPtr* , yalnızca bir dağıtım listesi açılırken kullanılır. *RecsPresent* sıfırda, alan yok sayılır.

Nesne kayıtlarını belirtmek için *ObjectRecPtr* ya da *ObjectRecOffset* 'yi kullanabilirsiniz, ancak her ikisi de değil; *ObjectRecOffset* alanının açıklaması için "[ObjectRecGörelili Konumu \(MQUZE\)](#)" sayfa 482' e bakın. *ObjectRecPtr* kullanmayacaksa, boş değerli ya da boş byte 'a ayarlayın.

Bu bir giriş alanıdır. Bu alanın ilk değeri, göstergeleri destekleyen programlama dillerindeki boş değerli göstergedir ve tersi durumda, boş değerli byte dizilimidir. *Version* , MQOD\_VERSION\_2değerinden küçükse bu alan yok sayılır.

**Not:** Programlama dilinin gösterge veri tipini desteklemediği altyapılarda, bu alan uygun uzunluğun byte dizgisi olarak bildirilir ve ilk değer, tüm boş değerli byte dizgisi olarak kabul edilir.

### **ResponseRecPtr (MQPTR)**

Bu, ilk MQRR yanıt kaydının adresidir. *ResponseRecPtr* , yalnızca bir dağıtım listesi açılırken kullanılır. *RecsPresent* sıfırda, alan yoksayılır.

Yanıt kayıtlarını belirtmek için *ResponseRecPtr* ya da *ResponseRecOffset* seçeneğini kullanın, ancak her ikisini de belirtmeyin; ayrıntılar için bkz. “*ResponseRecGörelili Konumu (MQUZE)*” sayfa 483. *ResponseRecPtr*kullanmayacaksa, boş değerli ya da boş byte 'a ayarlayın.

Bu bir giriş alanıdır. Bu alanın ilk değeri, göstergeleri destekleyen programlama dillerindeki boş değerli göstergedir ve tersi durumda, boş değerli byte dizilimidir. *Version* , MQOD\_VERSION\_2değerinden küçükse bu alan yok sayılır.

**Not:** Programlama dilinin gösterge veri tipini desteklemediği altyapılarda, bu alan uygun uzunluğun byte dizgisi olarak bildirilir ve ilk değer, tüm boş değerli byte dizgisi olarak kabul edilir.

### **AlternateSecurityTanıtıcısı (MQBYTE40)**

Bu, uygun yetki denetimlerinin gerçekleştirilmesine olanak sağlamak için yetki hizmetine *AlternateUserId* ile geçirilen bir güvenlik tanıtıcısıdır. *AlternateSecurityId* yalnızca aşağıdaki durumlarda kullanılır:

- MQOPEN çağrısında MQOO\_ALTERNATE\_USER\_AUTHORITY belirtildi ya da
- MQPMO\_ALTERNATE\_USER\_AUTHORITY, MQPUT1 çağrısında belirtilir,

ve the *AlternateUserId* field is not entirely blank up to the first null character or the end of the field.

Windowsüzerinde, *AlternateUserId*' i benzersiz olarak tanıtan Windows güvenlik tanıtıcısını (SID) sağlamak için *AlternateSecurityId* kullanılabilir. The SID for a user can be obtained from the Windows system by use of the LookupAccountName () Windows API call.

z/OS' ta bu alan yok sayılır.

*AlternateSecurityId* alanı aşağıdaki yapıya sahiptir:

- İlk bayt, izleyen önemli verilerin uzunluğunu içeren bir ikili tamsayıdır; değer, uzunluk byte 'ının kendisini dışlar. Herhangi bir güvenlik tanıtıcısı yoksa, uzunluk sıfırdır.
- İkinci bayt, var olan güvenlik tanımlayıcının tipini gösterir; aşağıdaki değerler kullanılabilir:

#### **MQSIDT\_NT\_SECURITY\_ID**

Windows güvenlik tanıtıcısı.

#### **MQSIDT\_NONE**

Güvenlik tanıtıcısı yok.

- Üçüncü ve sonraki baytlar, ilk baytın tanımladığı uzunluğa kadar, güvenlik tanıtıcısının kendisini içerir.
- Alanda kalan bayt sayısı ikili olarak sıfır olarak ayarlanır.

Aşağıdaki özel değeri kullanabilirsiniz:

#### **MQSID\_NONE**

Güvenlik tanıtıcısı belirtilmedi.

Değer, alanın uzunluğuna ilişkin ikili sıfır değeridir.

C programlama dili için, sabit MQSID\_NONE\_DIZISI de tanımlanır; bu, MQSID\_NONE ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisi olur.

Bu bir giriş alanıdır. Bu alanın uzunluğu MQ\_SECURITY\_ID\_LENGTH tarafından verilmektedir. Bu alanın ilk değeri MQSID\_NONE olur. *Version* , MQOD\_VERSION\_3değerinden küçükse bu alan yok sayılır.



## **ResolvedQName (MQCHAR48)**

Bu ad, yerel kuyruk yöneticisinin adı çözdükten sonra hedef kuyruğun adıdır. The name returned is the name of a queue that exists on the queue manager identified by *ResolvedQMGrName*.

Boş olmayan bir değer yalnızca, nesne göz atma, giriş ya da çıkış (ya da herhangi bir birleşim) için açılmış tek bir kuyruksa döndürülür. Açılan nesne aşağıdakilerden herhangi biriye, *ResolvedQName* boşluklara ayarlanır:

- Kuyruk değil
- Bir kuyruk, ancak göz atma, giriş ya da çıkış için açılmadı
- Bir dağıtım listesi
- Bir konu nesnesine gönderme yapan bir diğer ad kuyruğu (bkz. [ResObjectDizgisi](#) ' ne bakın).
- Bir konu nesnesine çözülen bir diğer ad kuyruğu.

Bu bir çıkış alanıdır. Bu alanın uzunluğu MQ\_Q\_NAME\_LENGTH değeri tarafından verilir. Bu alanın başlangıç değeri, C ' de boş değer dizgisidir ve diğer programlama dillerindeki 48 boş karakter dizilimidir. *Version* , MQOD\_VERSION\_3değerinden küçükse bu alan yok sayılır.

## **ResolvedQMGrAdı (MQCHAR48)**

Bu ad, yerel kuyruk yöneticisinin adı çözdükten sonra hedef kuyruk yöneticisinin adıdır. Döndürülen ad, *ResolvedQName* ile tanımlanan kuyruğa sahip olan kuyruk yöneticisinin adıdır. *ResolvedQMGrName* , yerel kuyruk yöneticisinin adı olabilir.

*ResolvedQName* , yerel kuyruk yöneticisinin ait olduğu kuyruk paylaşım grubunun sahip olduğu bir paylaşılan kuyruksa, *ResolvedQMGrName* kuyruk paylaşım grubunun adıdır. If the queue is owned by some other queue sharing group, *ResolvedQName* can be the name of the queue sharing group or the name of a queue manager that is a member of the queue sharing group (the nature of the value returned is determined by the queue definitions that exist at the local queue manager).

Boş olmayan bir değer yalnızca, nesne göz atma, giriş ya da çıkış (ya da herhangi bir birleşim) için açılmış tek bir kuyruksa döndürülür. Açılan nesne aşağıdakilerden herhangi biriye, *ResolvedQMGrName* boşluklara ayarlanır:

- Kuyruk değil
- Bir kuyruk, ancak göz atma, giriş ya da çıkış için açılmadı
- **DefBind** kuyruk özneliği MQBND\_BIND\_NOT\_FIXED değerine sahip olduğunda, MQOO\_BIND\_NOT\_FIXED ile ya da MQOO\_BIND\_AS\_Q\_DEF ile birlikte bir küme kuyruğu geçerli olur.
- Bir dağıtım listesi

Bu bir çıkış alanıdır. Bu alanın uzunluğu MQ\_Q\_NAME\_LENGTH değeri tarafından verilir. Bu alanın başlangıç değeri, C ' de boş değer dizgisidir ve diğer programlama dillerindeki 48 boş karakter dizilimidir. *Version* , MQOD\_VERSION\_3değerinden küçükse bu alan yok sayılır.

## **ObjectString (MQCHARV)**

ObjectString alanı, uzun nesne adını belirtir.

Bu, kullanılacak uzun nesne adını belirtir. Bu alana yalnızca belirli *ObjectType* değerleri için başvurulur ve diğer tüm değerler için yoksayılar. Bu alanın kullanıldığını gösteren değerlerin ayrıntıları için *ObjectType* açıklamasına bakın.

*ObjectString* yanlış belirtilirse, MQCHARV yapısının nasıl kullanılacağına ilişkin açıklamaya göre ya da uzunluk üst sınırını aşarsa, çağrı MQRC\_OBJECT\_STRING\_ERROR neden koduyla başarısız olur.

Bu bir giriş alanıdır. Bu yapıdaki alanların başlangıç değerleri, MQCHARV yapısındaki değerlerle aynıdır.

Tam konu adı iki farklı alandan oluşturulabilir: *ObjectName* ve *ObjectString*. Bu iki alanın nasıl kullanıldığına ilişkin ayrıntılar için [Konu dizgilerini birleştirme](#) başlıklı konuya bakın.

## **SelectionString (MQCHARV)**

Bu dizilim, kuyruğun dışına ileti alınırken kullanılan seçim ölçütlerinin sağlanması için kullanılan dizgidir.

*SelectionString* aşağıdaki durumlarda sağlanmamalıdır:

- *ObjectType* MQOT\_Q değilse
- Açılmakta olan kuyruk, MQOO\_BROWSE ya da MQOO\_INPUT\_\* seçeneklerinden biri kullanılarak açılmamışsa

Bu durumlarda *SelectionString* sağlandıysa, çağrı neden kodu MQRC\_SELECTOR\_INVALID\_FOR\_TYPE neden koduyla başarısız olur.

*SelectionString* yanlış belirtilirse, “MQCHARV-Değişken Uzunluklu Dizgi” sayfa 292 yapısının nasıl kullanılacağı ya da uzunluk üst sınırını aştıysa, çağrı neden kodu MQRC\_SELECTION\_STRING\_ERROR ile başarısız olur. *SelectionString* uzunluğu üst sınırı MQ\_SELECTOR\_LENGTH.

*SelectionString* kullanımı, Seçiciler' de açıklanır.

## **ResObjectDizgisi (MQCHARV)**

Kuyruk yöneticisi, *ObjectName* alanında sağlanan adı çözdükten sonra, ResObjectDizgi alanı uzun nesne adıdır.

Bu alan yalnızca bir konu nesnesine gönderme yapan konular ve kuyruk diğer adları için döndürülür.

*ObjectString* 'ta uzun nesne adı sağlanırsa ve *ObjectName* ' ta hiçbir şey sağlanmıyorsa, bu alanda döndürülen değer, *ObjectString* ' de sağlanan değerle aynıdır.

Bu alan atlanırsa ( ResObjectString.VSBufSize sıfırdır), *ResObjectString* döndürülmez, ancak uzunluk ResObjectString.VSLength' te döndürülür.

Arabellek uzunluğu ( ResObjectString.VSBufSize içinde sağlanan) fulltam *ResObjectString* değerinden kısa olursa, dizgi kesilir ve sağlanan arabelleğe sığabileceği en sağdaki en sağdaki karakter döndürülür.

*ResObjectString* yanlış belirtildiyse, MQCHARV yapısının nasıl kullanılacağını ya da uzunluk üst sınırını aşılırsa, çağrı neden kodu MQRC\_RES\_OBJECT\_STRING\_ERROR ile başarısız olur.

## **ResolvedType (MQUZE)**

Açılmakta olan çözülmüş (base) nesnesinin tipi.

Olası değerler şunlardır:

### **MQOT\_Q**

Çözümlenen nesne bir kuyruktur. Bu değer, bir kuyruk doğrudan açıldığında ya da bir kuyruğu işaret eden bir diğer ad açıldığında geçerli olur.

### **MQOT\_KONUSU**

Çözümlenen nesne bir konudur. Bu değer, bir konu doğrudan açıldığında ya da bir konu nesnesini işaret eden bir diğer ad kuyruğu açıldığında geçerlidir.

### **MQOT\_NONE**



Çözümlenen tip, bir kuyruk ya da konu değil.

## **MQOR-Nesne kaydı**

Tek bir hedef kuyruğun kuyruk adını ve kuyruk yöneticisi adını belirtmek için MQOR yapısını kullanın. MQOR, MQOPEN ve MQPUT1 çağrılarında ilişkin bir giriş yapısıdır.

## **Kullanılabilirlik**

MQOR yapısı aşağıdaki altyapılarda bulunur:

-  AIX
-  IBM i

-  Linux
-  Windows

ve bu sistemlere bağı IBM MQ MQI clients için.

## Karakter kümesi ve kodlama

MQOR içindeki veriler, **CodedCharSetId** kuyruk yöneticisi özneliği tarafından belirtilen karakter kümesinde ve MQENC\_NATIVE tarafından verilen yerel kuyruk yöneticisinin kodlamasında olmalıdır. Ancak, uygulama bir MQ MQI istemcisi olarak çalışıyorsa, yapı istemcinin karakter kümesinde ve kodlamasında olmalıdır.

## Kullanım

MQOPEN çağrısında bu yapıların bir dizisini sağlayarak bir kuyruk listesi açabilirsiniz; bu listeye dağıtım listesi adı verilir. MQOPEN çağrısının döndürdüğü kuyruk tanıtıcısı kullanılarak konan her ileti, kuyruğun başarıyla açılması koşuluyla, listedeki kuyrukların her birine yerleştirilir.

## Alanlar

**Not:** Aşağıdaki tabloda, alanlar alfabetik olarak değil, kullanıma göre gruplanır. Alt konular aynı sırayı izler.

<i>Çizelge 506. MQOR için MQOR içindeki alanlar</i>		
Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
<u>ObjectName</u> (nesne adı)	Yok	Boş değerli dizilim ya da boşluklar
<u>ObjectQMgrAdı</u> (nesne kuyruğu yöneticisi adı)	Yok	Boş değerli dizilim ya da boşluklar
<p><b>Notlar:</b></p> <ol style="list-style-type: none"> <li>1. Boş değerli dizilim ya da boşluk değeri, C dizgisindeki boş dizgiyi ve diğer programlama dillerindeki boş karakterleri gösterir.</li> <li>2. C programlama dilinde, makro değişkeniMQOR_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlar için başlangıç değerleri sağlamak üzere aşağıdaki şekilde kullanılabilir:</li> </ol> <pre>MQOR MyOR = {MQOR_DEFAULT};</pre>		

## Dil bildirimleri

MQOR için C bildirimi

```
typedef struct tagMQOR MQOR;
struct tagMQOR {
    MQCHAR48 ObjectName; /* Object name */
    MQCHAR48 ObjectQMgrName; /* Object queue manager name */
};
```

MQOR için COBOL bildirimi

```
** MQOR structure
10 MQOR.
** Object name
15 MQOR-OBJECTNAME PIC X(48).
```

```
**      Object queue manager name
      15 MQOR-OBJECTQMGRNAME PIC X(48).
```

MQOR için PL/I bildirimi

```
dcl
  1 MQOR based,
  3 ObjectName      char(48), /* Object name */
  3 ObjectQMGrName char(48); /* Object queue manager name */
```

MQOR için Visual Basic bildirimi

```
Type MQOR
  ObjectName      As String*48 'Object name'
  ObjectQMGrName As String*48 'Object queue manager name'
End Type
```

### **ObjectName (MQCHAR48)**

Bu, MQOD yapısındaki *ObjectName* alanıyla aynıdır (ayrıntılar için MQOD ' ye bakın). Bunun dışında:

- Bir kuyruğun adı olmalıdır.
- Bu, bir model kuyruğunun adı olmamalıdır.

Bu her zaman bir giriş alanıdır. Bu alanın başlangıç değeri, C ' de boş değer dizgisidir ve diğer programlama dillerindeki 48 boş karakter dizilimidir.

### **ObjectQMGrAdı (MQCHAR48)**

Bu, MQOD yapısındaki *ObjectQMGrName* alanı ile aynıdır (ayrıntılar için MQOD ' ye bakın).

Bu her zaman bir giriş alanıdır. Bu alanın başlangıç değeri, C ' de boş değer dizgisidir ve diğer programlama dillerindeki 48 boş karakter dizilimidir.

### **MQPD-Özellik tanımlayıcısı**

**MQPD** yapısı, bir özelliğin özniteliklerini tanımlamak için kullanılır. Yapı, MQSETMP çağrısında bir giriş/çıkış değiştirgesi ve MQINQMP çağrısında bir çıkış değiştirgesidir.

### **Kullanılabilirlik**

**MQPD** yapısı aşağıdaki platformlarda bulunur:

-  AIX
-  IBM i
-  Linux
-  Windows
-  z/OS

ve IBM MQ MQI clients için.

### **Karakter kümesi ve kodlama**

**MQPD** içindeki veriler, uygulamanın karakter kümesinde ve uygulamanın kodlamasında olmalıdır (**MQENC\_NATIVE**).

### **Alanlar**

**Not:** Aşağıdaki tabloda, alanlar alfabetik olarak değil, kullanıma göre gruplanır. Alt konular aynı sırayı izler.

Çizelge 507. MQPD içindeki alanlar

Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
StrucId (yapı tanıtıcısı)	MQPD_STRUC_ID	'PD'
Sürüm (yapı sürümü numarası)	MQPD_VERSION_1	1
Seçenekler (seçenekler)	MQPD_NONE	0
Destek (ileti özelliği için gerekli destek)	MQPD_SUPPORT_OPTIONAL	0
Bağlam (özelliğin ait olduğu ileti bağlamı)	MQPD_NO_CONTEXT	0
CopyOptions (özelliğin ait olduğu kopyalama seçenekleri)	MQCOPY_DEFAULT	0

**Notlar:**

1. C programlama dilinde, MQPD\_DEFAULT makro değişkeni çizelgede listelenen değerleri içerir. Yapıdaki alanlar için başlangıç değerleri sağlamak üzere aşağıdaki şekilde kullanılabilir:

```
MQPD MyPD = {MQPD_DEFAULT};
```

## Dil bildirimleri

### MQPD için C bildirimi

```
typedef struct tagMQPD MQPD;
struct tagMQPD {
    MQCHAR4 StrucId; /* Structure identifier */
    MQLONG Version; /* Structure version number */
    MQLONG Options; /* Options that control the action of
MQSETMP and MQINQMP */
    MQLONG Support; /* Property support option */
    MQLONG Context; /* Property context */
    MQLONG CopyOptions; /* Property copy options */
};
```

### MQPD için COBOL bildirimi

```
** MQPD structure
10 MQPD.
** Structure identifier
15 MQPD-STRUCID PIC X(4).
** Structure version number
15 MQPD-VERSION PIC S9(9) BINARY.
** Options that control the action of MQSETMP and
** MQINQMP
15 MQPD-OPTIONS PIC S9(9) BINARY.
** Property support option
15 MQPD-SUPPORT PIC S9(9) BINARY.
** Property context
15 MQPD-CONTEXT PIC S9(9) BINARY.
** Property copy options
15 MQPD-COPYOPTIONS PIC S9(9) BINARY.
```

### MQPD için PL/I bildirimi

```
dcl
1 MQPD based,
3 StrucId char(4), /* Structure identifier */
3 Version fixed bin(31), /* Structure version number */
3 Options fixed bin(31), /* Options that control the action
```

```

of MQSETMP and MQINQMP */
3 Support      fixed bin(31), /* Property support option */
3 Context      fixed bin(31), /* Property context */
3 CopyOptions  fixed bin(31); /* Property copy options */

```

## MQPD için High Level Assembler (Üst Düzey Çevirici) bildirimi

```

MQPD          DSECT
MQPD_STRUCID  DS    CL4    Structure identifier
MQPD_VERSION  DS    F      Structure version number
MQPD_OPTIONS  DS    F      Options that control the
*              action of MQSETMP and MQINQMP
MQPD_SUPPORT  DS    F      Property support option
MQPD_CONTEXT  DS    F      Property context
MQPD_COPYOPTIONS DS    F    Property copy options
MQPD_LENGTH   EQU   *-MQPD
MQPD_AREA     DS    CL(MQPD_LENGTH)

```

### **StrucId (MQCHAR4)**

Bu, yapı tanıtıcısıdır; değer şöyle olmalıdır:

#### **MQPD\_STRUC\_ID**

Özellik tanımlayıcı yapısına ilişkin tanıtıcı.

For the C programming language, the constant **MQPD\_STRUC\_ID\_ARRAY** is also defined; this has the same value as **MQPD\_STRUC\_ID**, but is an array of characters instead of a string.

Bu her zaman bir giriş alanıdır. Bu alanın başlangıç değeri **MQPD\_STRUC\_ID**' dir.

### **Sürüm (MQUZE)**

Bu, yapı sürüm numarasıdır; değer şu olmalıdır:

#### **MQPD\_VERSION\_1**

Version-1 özellik tanımlayıcı yapısı.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

#### **MQPD\_CURRENT\_VERSION**

Özellik tanımlayıcı yapısının yürürlükteki sürümü.

Bu her zaman bir giriş alanıdır. Bu alanın başlangıç değeri **MQPD\_VERSION\_1**' dir.

### **Seçenekler (MQUZE)**

Değer şu olmalıdır:

#### **MQPD\_NONE**

Seçenek belirtilmedi

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQPD\_NONE olur.

### **Destek (MQLONG)**

Bu alanda, bu özelliği içeren iletinin kuyruğa konmasını sağlamak için kuyruk yöneticisinin ileti özelliğine ilişkin destek düzeyi ne kadar olmalıdır. Bu, yalnızca IBM MQtanımlı özellikler için geçerlidir; diğer tüm özellikler için destek isteğe bağlıdır.

The field is automatically set to the correct value when the IBM MQ-defined property is known by the queue manager. Özellik tanınmadıysa, MQPD\_SUPPORT\_OPTIONAL atanabilir. Kuyruk yöneticisi, kuyruk yöneticisinin yanlış olarak tanıdığı IBM MQtanımlı bir özelliği içeren bir ileti aldığı anda, kuyruk yöneticisi *Support* alanının değerini düzeltir.

MQCMHO\_NO\_VALIDATION seçeneğinin belirlendiği bir ileti tanıtıcısı için MQSETMP çağrısını kullanarak IBM MQtanımlı bir özellik ayarlarken, *Support* bir giriş alanı olur. Bu, bir uygulamanın, doğru değerle,

özelliğın bağılı kuyruk yöneticisi tarafından desteklenmediğı, ancak iletinin başka bir kuyruk yöneticisiyle işlenmek üzere tasarlandığı bir IBM MQ-tanımlı özelliğı koymasına olanak sağlar.

MQPD\_SUPPORT\_OPTIONAL değeri her zaman, IBM MQtanımlı özellikler olmayan özelliklere atanır.

İleti özelliklerini destekleyen bir IBM WebSphere MQ 7.0 kuyruk yöneticisi, tanınmayan bir *Support* değeri içeren bir özellik alır; özellik aşağıdaki gibi işlenir:

- MQPD\_SUPPORTUNSUP\_MASK, tanınmayan değerlerden herhangi biri varsa, MQPD\_SUPPORT\_UNREQUIR\_MASK içinde yer alıyorsa gereklidir.
- MQPD\_SUPPORT\_REQUIRED\_IF\_LOCAL, tanınmayan değerlerden herhangi biri MQPD\_ACCEPT\_UNSUP\_IF\_XMIT\_MASK içinde bulunuyorsa, MQPD\_SUPPORT\_REQUIRED\_IF\_LOCAL belirtildi.
- Aksi halde MQPD\_SUPPORT\_OPTIONAL belirtildi.

MQINQMP çağrısıyla aşağıdaki değerlerden biri ya da MQCMHO\_NO\_VALIDATION seçeneğinin belirlendiğı bir ileti tanıtıcısı için MQSETMP çağrısı kullanılırken, değerlerden biri belirtilebilir:

#### **MQPD\_SUPPORT\_OPTIONAL**

Özellik, bir kuyruk yöneticisi tarafından desteklenmese de kabul edilir. İletinin ileti özelliklerini desteklemeyen bir kuyruk yöneticisine akabilmesi için özellik atılabilir. Bu değer, IBM MQtanımlı olmayan özelliklere de atanır.

#### **MQPD\_SUPPORT\_REQUIRY**

Özelliğe ilişkin destek gereklidir. İleti, IBM MQtanımlı özelliğı desteklemeyen bir kuyruk yöneticisi tarafından reddedilir. MQPUT ya da MQPUT1 çağrısı, tamamlanma kodu MQCC\_FAILED ve neden kodu MQRC\_UNUPPORTED\_EXTY ile başarısız olur.

#### **MQPD\_SUPPORT\_REQUIRED\_IF\_LOCAL**

İleti, yerel bir kuyruk için yazılmışsa, IBM MQtanımlı özelliğı desteklemeyen bir kuyruk yöneticisi tarafından reddedilir. MQPUT ya da MQPUT1 çağrısı, tamamlanma kodu MQCC\_FAILED ve neden kodu MQRC\_UNUPPORTED\_EXTY ile başarısız olur.

İleti uzak bir kuyruk yöneticisine gönderiliyorsa, MQPUT ya da MQPUT1 çağrısı başarılı olur.

Bu, MQSEQMP çağrısındaki bir çıkış alanıdır ve MQSETMP çağrısıyla ilgili bir giriş alanı, MQCMHO\_NO\_VALIDATION seçenek kümesiyle ileti tanıtıcısı yaratıldıysa, bu alan bir giriş alanıdır. Bu alanın ilk değeri MQPD\_SUPPORT\_OPSİYONEL olarak ifade edilir.

### **Bağlam (MQUZE)**

Bu, özelliğın ait olduğı ileti bağlamını açıklar.

Kuyruk yöneticisi, kuyruk yöneticisinin yanlış olarak tanıdığı IBM MQtanımlı bir özelliğı içeren bir ileti aldığıında, kuyruk yöneticisi *Context* alanının değerini düzeltir.

Aşağıdaki seçenek belirlenebilir:

#### **MQPD\_USER\_CONTEXT**

Özellik, kullanıcı bağlamıyla ilişkilendirilir.

MQSETMP çağrısını kullanarak, kullanıcı bağlamıyla ilişkili bir özelliğı ayarlayabilmek için özel bir yetki gerekmez.

Bir IBM WebSphere MQ 7.0 kuyruk yöneticisiyle, kullanıcı bağlamıyla ilişkili bir özellik, MQOO\_SAVE\_ALL\_CONTEXT için açıklandığı şekilde saklanır. MQPMO\_PASS\_ALL\_CONTEXT ile MQPUT çağrısı, özelliğın saklanan bağlamdan yeni iletiye kopyalanmasına neden oluyor.

Önceden açıklanan seçenek gerekli değilse, aşağıdaki seçenek kullanılabilir:

#### **MQPD\_NO\_CONTEXT**

Özellik bir ileti bağlamıyla ilişkilendirilmemiş.

Tanınmayan bir değer, MQRC\_PD\_ERROR *Reason* koduyla reddedilir.

Bu, MQSETMP çağrısına ve MQINQMP çağrısından bir çıkış alanına giriş/çıkış alanıdır. Bu alanın ilk değeri MQPD\_NO\_CONTEXT ' dir.

## **CopyOptions (MQUZE)**

Bu, özelliğin hangi tip iletilere kopyalanması gerektiğini açıklar. Bu yalnızca tanınan IBM MQ tanımlı özellikler için bir çıkış alanıdır; IBM MQ uygun değeri ayarlar.

Kuyruk yöneticisi, kuyruk yöneticisinin yanlış olarak tanıdığı IBM MQ tanımlı özelliği içeren bir ileti aldığı anda, kuyruk yöneticisi *CopyOptions* alanının değerini düzeltir.

Bu seçeneklerden birini ya da birkaçını belirleyebilirsiniz. birden çok seçenek belirtmek için, değerleri bir araya ekleyin (aynı değişmezi bir kereden fazla eklemeyin) ya da bit ya da işlemi kullanarak değerleri birleştirin (programlama dili, bit işlemleri destekliyorsa).

### **MQCOPY\_FORWY**

Bu özellik iletilmekte olan bir iletiye kopyalanır.

### **MQCOPY\_YAYIN**

Bu özellik, bir ileti yayınlanırken bir abone tarafından alınan iletiye kopyalanır.

### **MQCOPY\_REPLY**

Bu özellik bir yanıt iletisine kopyalanır.

### **MQCOPY\_REPORT**

Bu özellik bir rapor iletisine kopyalanır.

### **MQCOPY\_ALL**

Bu özellik sonraki iletilerin tüm türlerine kopyalanır.

**Varsayılan seçenek:** Varsayılan kopya seçenekleri kümesini sağlamak için aşağıdaki seçenek belirlenebilir:

### **MQCOPY\_DEFAULT**

Bu özellik, iletilmekte olan bir iletiye, bir rapor iletisine ya da bir ileti yayınlanmakta olan bir abonenin aldığı iletiye kopyalanır.

Bu, MQCOPY\_FOREward, artı MQCOPY\_REPORT, artı MQCOPY\_YAYINLAN seçeneklerinin bileşimini belirtmekle eşdeğerdir.

Önceden açıklanan seçeneklerin hiçbiri gerekli değilse, aşağıdaki seçeneği kullanın:

### **MQCOPY\_NONE**

Bu değeri, başka bir kopya seçeneği belirtilmemeyi belirtmek için kullanın; bu özellik ile sonraki iletiler arasında programlı olarak herhangi bir ilişki yok. Bu, ileti tanımlayıcı özellikleri için her zaman döndürülür.

Bu, MQSETMP çağrısına ve MQINQMP çağrısından bir çıkış alanına giriş/çıkış alanıdır. Bu alanın ilk değeri MQCOPY\_default 'tır.

## **MQPMO-İleti koyma seçenekleri**

MQPMO yapısı, uygulamanın iletilerin kuyruklara nasıl yerleştirileceğini ya da konulara nasıl yayınlanacağını denetleyen seçenekleri belirtmesini sağlar. Yapı, MQPUT ve MQPUT1 çağrılarında bir giriş/çıkış değiştirgesidir.

## **Sürüm**

MQPMO ' nun yürürlükteki sürümü: MQPMO\_VERSION\_3. Belirli alanlar yalnızca MQPMO ' nun belirli sürümlerinde kullanılabilir. Uygulamaları birden çok ortam arasında bağlamanız gerekirse, MQPMO sürümünün tüm ortamlarda tutarlı olduğundan emin olmanız gerekir. Yapının yalnızca belirli sürümlerinde var olan alanlar, bu konuda ve alan açıklamalarında bu şekilde tanımlanır.

Desteklenen programlama dilleri için sağlanan üstbilgi, COPY ve INCLUDE dosyaları, ortam tarafından desteklenen en son MQPMO sürümünü içerir, ancak *Version* alanının ilk değeri MQPMO\_VERSION\_1olarak ayarlanmıştır. version-1 yapısında var olmayan alanları kullanmak için uygulamanın *Version* alanını gerekli sürümün sürüm numarasına ayarlaması gerekir.



## Karakter kümesi ve kodlama

MQPMO içindeki veriler, **CodedCharSetId** kuyruk yöneticisi özneliği tarafından belirtilen karakter kümesinde ve MQENC\_NATIVE tarafından verilen yerel kuyruk yöneticisinin kodlamasında yer almalıdır. Ancak, uygulama bir MQ MQI istemcisi olarak çalışıyorsa, yapı istemcinin karakter kümesinde ve kodlamasında olmalıdır.

## Alanlar

**Not:** Aşağıdaki tabloda, alanlar alfabetik olarak değil, kullanıma göre gruplanır. Alt konular aynı sırayı izler.

Çizelge 508. MQPMO içindeki alanlar		
Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
<u>StrucId</u> (yapı tanıtıcısı)	MQPMO_STRUC_ID	'PMO~'
<u>Sürüm</u> (yapı sürümü numarası)	MQPMO_VERSION_1	1
<u>Seçenekler</u> (MQPUT ve MQPUT1eylemine denetleyen seçenekler)	MQPMO_NONE	0
<u>Zamanaşımı</u> (ayrılmış)	Yok	-1
<u>Bağlam</u> (giriş kuyruğunun nesne tanıtıcısı)	Yok	0
<u>KnownDestSayı</u> (yerel kuyruklara başarıyla gönderilen iletilerin sayısı)	Yok	0
<u>UnknownDestCount</u> (Uzak kuyruklara başarıyla gönderilen ileti sayısı)	Yok	0
<u>InvalidDestSayı</u> (gönderilemeyen iletilerin sayısı)	Yok	0
<u>ResolvedQName</u> (hedef kuyruğun çözülmüş adı)	Yok	Boş değerli dizilim ya da boşluklar
<u>ResolvedQMgrAdı</u> (hedef kuyruk yöneticisinin çözümlenen adı)	Yok	Boş değerli dizilim ya da boşluklar
<b>Not:</b> <i>Version</i> , MQPMO_VERSION_2' den küçükse kalan alanlar yoksayılr.		
<u>RecsPresent</u> (mevcut koyma iletisi kaydı ya da yanıt kaydı sayısı)	Yok	0
<u>PutMsgRecFields</u> (hangi MQPMR alanlarının var olduğunu gösteren işaretler)	MQPMRF_NONE	0
<u>PutMsgRecOffset</u> (MQPMO ' nun başlangıcından ilk koyma iletisi kaydının görel konumu)	Yok	0
<u>ResponseRecKayma</u> (MQPMO ' nun başlangıcından ilk yanıt kaydının görel konumu)	Yok	0
<u>PutMsgRecPtr</u> (ilk koyma iletisi kaydının adresi)	Yok	Boş değerli gösterge ya da boş değerli byte
<u>ResponseRecPtr</u> (ilk yanıt kaydının adresi)	Yok	Boş değerli gösterge ya da boş değerli byte
<b>Not:</b> <i>Version</i> , MQPMO_VERSION_3' den küçükse kalan alanlar yoksayılr.		
<u>OriginalMsgTanıtıcısı</u> (özgün ileti tanıtıcısı)	MQHM_NONE	0
<u>NewMsgHandle</u> (yeni ileti tanıtıcısı)	MQHM_NONE	0

Çizelge 508. MQPMO içindeki alanlar (devamı var)

Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
İşlem (gerçekleştirilmekte olan koyma değeri tipi ve <i>OriginalMsgHandle</i> alanı tarafından belirtilen özgün ileti ile <i>NewMsgHandle</i> alanı tarafından belirtilen yeni ileti arasındaki ilişki)	MQACTP_YENI	0
PubLevel (yayın tarafından hedeflenen abonelik düzeyi)	Yok	9

**Notlar:**

1. – simgesi, tek bir boş karakteri temsil eder.
2. Boş değerli dizilim ya da boşluk değeri, C dizgisindeki boş dizgiyi ve diğer programlama dillerindeki boş karakterleri gösterir.
3. C programlama dilinde, makro değişkeniMQPMO\_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlar için başlangıç değerleri sağlamak üzere bunu aşağıdaki şekilde kullanın:

```
MQPMO MyPMO = {MQPMO_DEFAULT};
```

## Dil bildirimleri

MQPMO için C bildirimi

```
typedef struct tagMQPMO MQPMO;
struct tagMQPMO {
    MQCHAR4   StructId;           /* Structure identifier */
    MQLONG    Version;           /* Structure version number */
    MQLONG    Options;           /* Options that control the action of
    MQPUT and MQPUT1 */

    MQLONG    Timeout;           /* Reserved */
    MQHOBJS   Context;           /* Object handle of input queue */
    MQLONG    KnownDestCount;    /* Number of messages sent
    successfully to local queues */
    MQLONG    UnknownDestCount;  /* Number of messages sent
    successfully to remote queues */
    MQLONG    InvalidDestCount;  /* Number of messages that could not
    be sent */
    MQCHAR48   ResolvedQName;    /* Resolved name of destination
    queue */
    MQCHAR48   ResolvedQMgrName; /* Resolved name of destination queue
    manager */

    /* Ver:1 */
    MQLONG    RecsPresent;       /* Number of put message records or
    response records present */
    MQLONG    PutMsgRecFields;   /* Flags indicating which MQPMR fields
    are present */
    MQLONG    PutMsgRecOffset;   /* Offset of first put message record
    from start of MQPMO */
    MQLONG    ResponseRecOffset; /* Offset of first response record
    from start of MQPMO */
    MQPTR     PutMsgRecPtr;      /* Address of first put message
    record */
    MQPTR     ResponseRecPtr;    /* Address of first response record */

    /* Ver:2 */
    MQHMSG    OriginalMsgHandle; /* Original message handle */
    MQHMSG    NewMsgHandle;      /* New message handle */
    MQLONG    Action;           /* The action being performed */
    MQLONG    PubLevel;         /* Subscription level */

    /* Ver:3 */
};
```

## MQPMO için COBOL bildirimi

```
** MQPMO structure
10 MQPMO.
** Structure identifier
15 MQPMO-STRUCID PIC X(4).
** Structure version number
15 MQPMO-VERSION PIC S9(9) BINARY.
** Options that control the action of MQPUT and MQPUT1
15 MQPMO-OPTIONS PIC S9(9) BINARY.
** Reserved
15 MQPMO-TIMEOUT PIC S9(9) BINARY.
** Object handle of input queue
15 MQPMO-CONTEXT PIC S9(9) BINARY.
** Number of messages sent successfully to local queues
15 MQPMO-KNOWNDSTCOUNT PIC S9(9) BINARY.
** Number of messages sent successfully to remote queues
15 MQPMO-UNKNOWNDSTCOUNT PIC S9(9) BINARY.
** Number of messages that could not be sent
15 MQPMO-INVALIDDSTCOUNT PIC S9(9) BINARY.
** Resolved name of destination queue
15 MQPMO-RESOLVEDQNAME PIC X(48).
** Resolved name of destination queue manager
15 MQPMO-RESOLVEDQMGRNAME PIC X(48).
** Number of put message records or response records present
15 MQPMO-RECSPRESENT PIC S9(9) BINARY.
** Flags indicating which MQPMR fields are present
15 MQPMO-PUTMSGRECFIELDS PIC S9(9) BINARY.
** Offset of first put message record from start of MQPMO
15 MQPMO-PUTMSGRECOFFSET PIC S9(9) BINARY.
** Offset of first response record from start of MQPMO
15 MQPMO-RESPONSERECOFFSET PIC S9(9) BINARY.
** Address of first put message record
15 MQPMO-PUTMSGRECPtr POINTER.
** Address of first response record
15 MQPMO-RESPONSERECPtr POINTER.
** Original message handle
15 MQPMO-ORIGINALMSGHANDLE PIC S9(18) BINARY.
** New message handle
15 MQPMO-NEWMSGHANDLE PIC S9(18) BINARY.
** The action being performed
15 MQPMO-ACTION PIC S9(9) BINARY.
** Publish level
15 MQPMO-PUBLEVEL PIC S9(9) BINARY.
```

## MQPMO için PL/I bildirimi

```
dcl
1 MQPMO based,
3 StrucId char(4), /* Structure identifier */
3 Version fixed bin(31), /* Structure version number */
3 Options fixed bin(31), /* Options that control the action
of MQPUT and MQPUT1 */

3 Timeout fixed bin(31), /* Reserved */
3 Context fixed bin(31), /* Object handle of input queue */
3 KnownDestCount fixed bin(31), /* Number of messages sent
successfully to local queues */
3 UnknownDestCount fixed bin(31), /* Number of messages sent
successfully to remote queues */
3 InvalidDestCount fixed bin(31), /* Number of messages that could
not be sent */
3 ResolvedQName char(48), /* Resolved name of destination
queue */
3 ResolvedQMgrName char(48), /* Resolved name of destination
queue manager */
3 RecsPresent fixed bin(31), /* Number of put message records or
response records present */
3 PutMsgRecFields fixed bin(31), /* Flags indicating which MQPMR
fields are present */
3 PutMsgRecOffset fixed bin(31), /* Offset of first put message
record from start of MQPMO */
3 ResponseRecOffset fixed bin(31), /* Offset of first response record
from start of MQPMO */
3 PutMsgRecPtr pointer, /* Address of first put message
record */
3 ResponseRecPtr pointer, /* Address of first response
record */
3 OriginalMsgHandle fixed bin(63), /* Original message handle */
```

```

3 NewMsgHandle      fixed bin(63); /* New message handle */
3 Action            fixed bin(31); /* The action being performed */
3 PubLevel          fixed bin(31); /* Publish level */

```

### MQPMO için High Level Assembler (Üst Düzey Çevirici) bildirimi

```

MQPMO                DSECT
MQPMO_STRUCID        DS    CL4   Structure identifier
MQPMO_VERSION        DS    F     Structure version number
MQPMO_OPTIONS        DS    F     Options that control the action of
*                    MQPUT and MQPUT1
MQPMO_TIMEOUT        DS    F     Reserved
MQPMO_CONTEXT        DS    F     Object handle of input queue
MQPMO_KNOWNDESTCOUNT DS    F     Number of messages sent successfully
*                    to local queues
MQPMO_UNKNOWNDSTCOUNT DS    F     Number of messages sent successfully
*                    to remote queues
MQPMO_INVALIDDESTCOUNT DS    F     Number of messages that could not be
*                    sent
MQPMO_RESOLVEDQNAME  DS    CL48  Resolved name of destination queue
MQPMO_RESOLVEDQMGRNAME DS    CL48 Resolved name of destination queue
*                    manager
MQPMO_RECSPRESENT    DS    F     Number of put message records or
*                    response records present
MQPMO_PUTMSGRECFIELDS DS    F     Flags indicating which MQPMR
*                    fields are present
MQPMO_PUTMSGRECOFFSET DS    F     Offset of first put message record
*                    from start of MQPMO
MQPMO_RESPONSERECOFFSET DS    F     Offset of first response record
*                    from start of MQPMO
MQPMO_PUTMSGRECPtr    DS    F     Address of first put message
*                    record
MQPMO_RESPONSERECPtr DS    F     Address of first response record
MQPMO_ORIGINALMSGHANDLE DS    D     Original message handle
MQPMO_NEWMSGHANDLE   DS    D     New message handle
MQPMO_ACTION         DS    F     The action being performed
MQPMO_PUBLEVEL       DS    F     Publish level
*
MQPMO_LENGTH         EQU    *-MQPMO
                    ORG    MQPMO
MQPMO_AREA           DS    CL(MQPMO_LENGTH)

```

### MQPMO için Visual Basic bildirimi

```

Type MQPMO
  StrucId      As String*4   'Structure identifier'
  Version      As Long       'Structure version number'
  Options      As Long       'Options that control the action of'
  Timeout      As Long       'MQPUT and MQPUT1'
  Context      As Long       'Reserved'
  KnownDestCount As Long     'Object handle of input queue'
  UnknownDestCount As Long   'Number of messages sent successfully'
  InvalidDestCount As Long   'to local queues'
  ResolvedQName As String*48 'Number of messages sent successfully'
  ResolvedQMgrName As String*48 'to remote queues'
  RecsPresent   As Long     'Number of messages that could not be'
  PutMsgRecFields As Long   'sent'
  PutMsgRecOffset As Long   'Resolved name of destination queue'
  ResponseRecOffset As Long 'Resolved name of destination queue'
  PutMsgRecPtr  As MQPTR    'manager'
  ResponseRecPtr As MQPTR    'Number of put message records or'
  End Type      'response records present'
                'Flags indicating which MQPMR fields'
                'are present'
                'Offset of first put message record'
                'from start of MQPMO'
                'Offset of first response record from'
                'start of MQPMO'
                'Address of first put message record'
                'Address of first response record'

```

### StrucId (MQCHAR4)

Bu, yapı tanıtıcısıdır; değer şöyle olmalıdır:

## **MQPMO\_STRUC\_ID**

Put-message options yapısına ilişkin tanıttıcı.

C programlama dili için, MQPMO\_STRUC\_ID\_ARRAY değişmezi de tanımlanır; bu, MQPMO\_STRUC\_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisidir.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQPMO\_STRUC\_ID 'dir.

## **Sürüm (MQUZE)**

Yapı sürüm numarası.

Değer aşağıdakilerden biri olmalıdır:

### **MQPMO\_VERSION\_1**

Version-1 put-message options yapısı.

Bu sürüm, tüm ortamlarda desteklenir.

### **MQPMO\_VERSION\_2**

Version-2 put-message options structure.

Bu sürüm aşağıdaki ortamlarda desteklenir:

-  AIX
-  IBM i
-  Linux
-  Windows

ve bu sistemlere bağlı IBM MQ MQI clients için.

### **MQPMO\_VERSION\_3**

Version-3 put-message options yapısı.

Bu sürüm, tüm ortamlarda desteklenir.

Yalnızca yapının daha yeni sürümünde var olan alanlar, alanların açıklamalarında olduğu gibi tanımlanır. Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

### **MQPMO\_CURRENT\_VERSION**

Put-message options yapısının yürürlükteki sürümü.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQPMO\_VERSION\_1' dir.

## **MQPMO seçenekleri (MQUWT)**

Seçenekler alanı, MQPUT ve MQPUT1 çağrılarının çalışmasını denetler.

**Kapsam seçeneği.** MQPMO seçeneklerinden herhangi birini ya da hiçbirini belirtmebilirsiniz. Geçerli olmayanbirden çok seçenek belirtmek için, değerleri bir araya ekleyin (aynı değişmezi bir kereden fazla eklemeyin) ya da bit ya da işlemi kullanarak değerleri birleştirin (programlama dili, bit işlemleri destekliyorsa). Bileşimleri dikkat çekilir; diğer birleşimler geçerlidir.

Aşağıdaki seçenek, gönderilen yayınların kapsamını denetler:

### **MQPMO\_SCOP\_QMGR**

Bu yayın, yalnızca bu kuyruk yöneticisine abone olan abonelere gönderilir. Yayın, bu kuyruk yöneticisine abonelik yapan, PUBSCOPE konu özniteliği kullanılarak ayarlanmış herhangi bir davranışı geçersiz kılan herhangi bir uzak yayınlama/abone olma kuyruğu yöneticilerine iletilmedi.

**Not:** Ayarlanmazsa, yayınlama kapsamı PUBSCOPE konu özniteliği tarafından belirlenir.

**Yayınlama seçenekleri.** Aşağıdaki seçenekler, iletilerin bir konuya nasıl yayınlanma şeklini denetler:

## MQPMO\_SUPPRESS\_REPLYTO

Bu yayının MQMD ' nin *ReplyToQ* ve *ReplyToQMGr* alanlarında belirtilen her türlü bilgi abonelere iletilmez. Bu seçenek, *ReplyToQ* gerektiren bir rapor seçeneği ile kullanılırsa, çağrı MQRC\_MISSING\_REPLY\_TO\_Q ile başarısız olur.

## MQPMO\_TUT

Gönderilmekte olan yayın kuyruk yöneticisi tarafından alıkonabilecektir. Bu alıkoyma, bir abonenin MQSUBRQ çağrısını kullanarak, yayınlandığı zamandan sonra bu yayının bir kopyasını istemesine olanak sağlar. Ayrıca, bu yayının yapıldığı zamandan sonra (MQSO\_NEW\_PUBLICATIONS\_ONLY seçeneğini kullanarak göndermemeyi seçmedikleri sürece) aboneliğini yapan uygulamalara bir yayının gönderilmesini sağlar. Bir uygulama alıkonan bir yayını gönderdiyse, o yayının MQIsRetained ileti özelliği tarafından belirtilir.

Konu ağacının her düğümünde yalnızca bir yayın saklanabilir. Bu nedenle, başka bir uygulama tarafından yayınlanan bu konu için zaten alıkonan bir yayın varsa, bu yayının yerine bu yayınlara değiştirilir. Bu nedenle, aynı konu üzerinde birden fazla yayınlama tutma ileti bulundurmamak daha iyidir.

Alıkonan yayınlar bir abone tarafından istendiğinde, kullanılan abonelik, konuya genel arama karakteri içerebilir; bu durumda alıkonan yayınlar sayısı (konu ağacındaki çeşitli düğümlerde) ve istekte bulunan uygulamaya birkaç yayın gönderilebilir. Daha fazla ayrıntı için ["MQSUBRQ-Abonelik isteği" sayfa 782](#) çağrısının açıklamasına bakın.

Tutulan yayınların abonelik düzeyleriyle nasıl etkileşimde bulunabilmesiyle ilgili bilgi edinmek için [Yayınlar Arası Yayınlar](#) konusuna bakın.

Bu seçenek kullanılırsa ve yayın saklanamazsa, ileti yayınlanmaz ve çağrı MQRC\_PUT\_NOT\_ALIMARM ile başarısız olur.

## MQPMO\_NOT\_OWN\_SUBS

Kuyruk yöneticisine, uygulamanın yayınlarından herhangi birini sahip olduğu aboneliklere göndermek istemediğini belirtir. Bağlantı tanıtıcıları aynıysa, abonelikler aynı uygulamaya ait olarak kabul edilir.

## MQPMO\_WARN\_IF\_NO\_SUBS\_MATCHED

Hiçbir abonelik yayınlara eşleşmezse, MQCC\_UYARI ile ilgili bir tamamlanma kodu (*CompCode*) ve MQRC\_NO\_XX\_ENCODE\_CASE\_ONE subs\_matched neden kodunu döndürün.

Koyma işlemi tarafından MQRC\_NO\_XX\_ENCODE\_CASE\_ONE subs\_matched döndürülürse, yayın herhangi bir aboneliğe teslim edilemedi. Ancak, koyma işleminde MQPMO\_RETAIN seçeneği belirtilirse, ileti alıkonur ve daha sonra tanımlanmış eşleşen abonelikle teslim edilir.

Aşağıdaki koşullardan herhangi biri karşılanırsa, konuyla ilgili bir abonelik yayınlara eşleşir:

- İleti, abonelik kuyruğuna teslim edilir
- İleti abonelik kuyruğuna teslim edildi; ancak, kuyrukla ilgili bir sorun, iletinin kuyruğa konmadığı ve sonuç olarak ölü mektup kuyruğuna konması ya da atıldığı anlamına gelir.
- İletinin aboneliğe teslim edilmesini baskılayan bir yöneltme çıkışı tanımlandı

Aşağıdaki koşullardan herhangi biri karşılanırsa yayınlara ilgili bir abonelik yayınlanmaz:

- Aboneliğin, yayınlara eşleşmeyen bir seçim dizgisi var.
- Abonelik, MQSO\_PUBLICATION\_ON\_REQUEST seçeneğini belirtti.
- Put işleminde MQPMO\_NOT\_OWN\_SUBS seçeneği belirtildiğinden ve abonelik yayınlamacının kimliğiyle eşleştiğinden yayın teslim edilmiyor.

**Syncpoint seçenekleri.** Aşağıdaki seçenekler, bir iş birimi içinde MQPUT ya da MQPUT1 çağrısının katılımlarıyla ilgilidir:

## MQPMO\_SYNCPOINT

Bu istek, olağan çalışma birimi protokolleri içinde işlem yapmak. İleti, iş birimi kesinleştirilinceye kadar iş biriminin dışında görünmez. İş birimi yedeklendiyse, ileti silinir.

MQPMO\_SYNCPOINT ve MQPMO\_NO\_SYNCPOINT belirtilmediyse, koyma isteğinin iş birimi ve iş birimi protokollerine eklenmesi, kuyruk yöneticisini çalıştıran ortam tarafından belirlenir ve uygulamayı çalıştıran ortam tarafından değil. On z/OS, the put request is within a unit of work. Diğer tüm ortamlarda, put isteği bir iş birimi içinde değildir.

Bu farklılıklar nedeniyle, bağlantı noktası olarak istediğiniz bir uygulamanın bu seçeneğin varsayılan değere izin vermemesi gerekir; MQPMO\_SYNCPOINT ya da MQPMO\_NO\_SYNCPOINT seçeneğini belirttik olarak belirtin.

MQPMO\_NO\_SYNCPOINT ile MQPMO\_SYNCPOINT ögesini belirtmeyin.

### **MQPMO\_NO\_SYNCPOINT**

Bu istek, olağan çalışma birimi protokollerinin dışında işlem yapmak. İleti hemen kullanılabilir ve bir iş birimi yedeklenerek silinemez.

MQPMO\_NO\_SYNCPOINT ve MQPMO\_SYNCPOINT belirtilmediyse, koyma isteğinin iş birimi içi protokollerine eklenmesi, kuyruk yöneticisini çalıştıran ortam tarafından belirlenir ve uygulamayı çalıştıran ortam tarafından değil. On z/OS, the put request is within a unit of work. Diğer tüm ortamlarda, put isteği bir iş birimi içinde değildir.

Bu farklılıklar nedeniyle, bağlantı noktası olarak istediğiniz bir uygulamanın bu seçeneğin varsayılan değere izin vermemesi gerekir; MQPMO\_SYNCPOINT ya da MQPMO\_NO\_SYNCPOINT seçeneğini belirttik olarak belirtin.

MQPMO\_SYNCPOINT için MQPMO\_NO\_SYNCPOINT belirtmeyin.

**İleti tanıtıcısı ve ilinti tanıtıcısı seçenekleri.** Aşağıdaki seçenekler kuyruk yöneticisinin yeni bir ileti tanıtıcısı ya da ilinti tanıtıcısı oluşturmasını ister:

### **MQPMO\_NEW\_MSG\_ID**

Kuyruk yöneticisi, MQMD ' deki *MsgId* alanının içeriğini yeni bir ileti tanıtıcısıyla değiştirir. Bu ileti tanıtıcısı iletilerle birlikte gönderilir ve MQPUT ya da MQPUT1 çağrısından çıkış için uygulamaya geri gönderilir.

MQPMO\_NEW\_MSG\_ID seçeneği, ileti bir dağıtım listesine bulunduğu da belirtilebilir; ayrıntılar için, MQPMR yapısındaki *MsgId* alanının tanımına bakın.

Using this option relieves the application of the need to reset the *MsgId* field to MQMI\_NONE before each MQPUT or MQPUT1 call.

### **MQPMO\_NEW\_CORREL\_ID**

Kuyruk yöneticisi, MQMD ' deki *CorrelId* alanının içeriğini yeni bir ilinti tanıtıcısıyla değiştirir. Bu ilinti tanıtıcısı iletilerle birlikte gönderilir ve MQPUT ya da MQPUT1 çağrısından çıkış için uygulamaya geri gönderilir.

MQPMO\_NEW\_CORREL\_ID seçeneği, ileti bir dağıtım listesine bulunduğu da belirtilebilir; ayrıntılar için, MQPMR yapısındaki *CorrelId* alanının tanımına bakın.

MQPMO\_NEW\_CORREL\_ID, uygulamanın benzersiz bir ilinti tanıtıcısını gerektirdiği durumlarda kullanışlıdır.

**Grup ve bölüm seçenekleri.** Aşağıdaki seçenekler, mantıksal iletilerin gruplarındaki ve bölümlerindeki iletilerin işlenmesine ilişkin bilgi içerir. Seçeneği anlamanız için size yardımcı olacak tanımlamaları okuyun.



**Uyarı:** Yayınlama/Abone Olma ile bölümlenmiş ya da gruplanmış iletiler kullanamazsınız.

### **Fiziksel ileti**

Bir kuyruktan yerleştirilebilecek ya da bir kuyruktan kaldırılacak en küçük bilgi birimidir; genellikle, tek bir MQPUT, MQPUT1 ya da MQGET çağrısında belirtilen ya da alınan bilgilere karşılık gelir. Her fiziksel ileti, kendi ileti tanımlayıcısına (MQMD) sahiptir. Genellikle, fiziksel iletiler ileti tanıtıcısı (MQMD ' de *MsgId* alanı) için farklı değerlere göre ayırt edilir; ancak bu, kuyruk yöneticisi tarafından zorunlu kılınmaz.

## Mantıksal ileti

Mantıksal ileti, yalnızca z/OS dışı altyapılar için tek bir uygulama bilgileri birimidir. Sistem kısıtlamalarının yokluğunda, mantıksal ileti fiziksel iletiyle aynıdır. Ancak, mantıksal iletilerin son derece büyük olduğu durumlarda, sistem kısıtlamaları, bir mantıksal iletinin *segments* adı verilen iki ya da daha çok fiziksel iletiye bölünmesini önerebilir ya da gerekli kılabilir.

Kesimlere ayrılmış bir mantıksal ileti, aynı boş olmayan grup tanıtıcısına (MQMD 'de*GroupId* alanı) ve aynı ileti dizisi numarasına (MQMD 'de*MsgSeqNumber* alanı) sahip iki ya da daha çok fiziksel ileten oluşur. Kesimler, mantıksal iletime verilerin başlangıcındaki fiziksel ilettaki verilerin görel konumunu veren bölüm görel konumu (MQMD 'de*Offset* alanı) için farklı değerlere göre ayırt edilir. Her bir bölüm fiziksel bir ileti olduğu için, mantıksal ilettaki kesimler genellikle farklı ileti tanıtıcılarına sahiptir.

Kesimlere ayrılmış olmayan, ancak gönderme uygulaması tarafından bölümlemeye izin verilen bir mantıksal ileti, aynı zamanda boş olmayan bir grup tanıtıcısına sahiptir; ancak, bu durumda mantıksal ileti bir ileti grubuna ait değilse, o grup tanıtıcısına sahip tek bir fiziksel ileti vardır. Gönderme uygulamasının engellendiği mantıksal iletiler, mantıksal ileti bir ileti grubuna ait değilse, boş değerli bir grup tanıtıcısına (MQGI\_NONE) sahip olur.

## İleti grubu

İleti grubu, boş değerli olmayan grup tanıtıcısına sahip bir ya da daha çok mantıksal ileti kümesidir. Gruptaki mantıksal iletiler, 1 ile *n* arasındaki aralıktaki bir tamsayı olan ileti sıra numarasına ilişkin farklı değerlere göre ayırt edilir; burada *n*, gruptaki mantıksal iletilerin sayısıdır. Mantıksal iletilerden biri ya da daha fazlası bölümlendiyse, grupta *n* 'den fazla fiziksel ileti vardır.

## MQPMO\_LOGICAL\_ORDER

Bu seçenek, kuyruk yöneticisine uygulamanın, iletileri gruplar ve mantıksal iletiler segmentlerine nasıl yerleştirdiğini bildirir. Yalnızca MQPUT çağrısında belirtilebilir; MQPUT1 çağrısında geçerli değildir.

MQPMO\_LOGICAL\_ORDER belirtilirse, uygulamanın aşağıdaki öğeler için ardışık MQPUT çağrıları kullandığını gösterir:

1. Her mantıksal ilettaki bölümleri, 0 'dan başlayarak, aralıksız olarak, kesim görel konumunun artırılmasına göre sırgörüntüleyebilirsiniz?... ??? ...
2. Bölümleri sonraki mantıksal iletiye yerleştirmeden önce tüm bölümleri tek bir mantıksal iletiye koyun.
3. Mantıksal iletileri, her ileti grubundaki ileti sıra numarasının artma sırasına göre, 1 'den başlayarak, boşluk olmadan koyun. IBM MQ , ileti sıra numarasını otomatik olarak artırır.
4. Mantıksal iletileri sonraki ileti grubuna koymadan önce, tüm mantıksal iletileri bir ileti grubuna koyun.

MQPMO\_LOGICAL\_ORDER ile ilgili ayrıntılı bilgi için [Mantıksal ve fiziksel sıralamabaşlıklı konuya](#) bakın.

**Bağlam seçenekleri.** Aşağıdaki seçenekler ileti bağlamının işlenmesini denetler:

## MQPMO\_NO\_BAĞLAM

Hem kimlik, hem de kaynak bağlamı, bağlam olmadığını göstermek üzere ayarlanır. Bu, MQMD 'deki bağlam alanlarının şu şekilde ayarlanacak şekilde ayarlanandır:

- Karakter alanları için boşluk
- Byte alanları için boş değerler
- Sayısal alanlar için sıfırlar

## MQPMO\_DEFAULT\_CONTEXT

İleti, hem kimlik hem de kaynak için, kendisiyle ilişkilendirilmiş varsayılan bağlam bilgilerine sahip olmak içindir. Kuyruk yöneticisi, ileti tanımlayıcısındaki bağlam alanlarını aşağıdaki gibi ayarlar:

*Çizelge 509. MQMD alanları için varsayılan bağlam bilgisi değerleri*

MQMD 'de alan	Kullanılan değer
<i>UserIdentifier</i>	Olanaklıysa, ortamdan saptanır; tersi durumda boşluklara ayarlanır.



MQMD ' de alan	Kullanılan değer
<i>AccountingToken</i>	Olanaklıysa, ortamdan saptanır; tersi durumda MQACT_NONE değerine ayarlanır.
<i>AppIdentityData</i>	Boşluklara ayarlayın.
<i>PutAppType</i>	Ortamdan saptanır.
<i>PutAppName</i>	Olanaklıysa, ortamdan saptanır; tersi durumda boşluklara ayarlanır.
<i>PutDate</i>	İletinin konulduğu tarihe ayarlanır.
<i>PutTime</i>	İletinin konulduğu zamana ayarlanır.
<i>AppOriginData</i>	Boşluklara ayarlayın.

İleti bağlamına ilişkin ek bilgi için [İleti bağlamı](#) başlıklı konuya bakın.

Bağlam seçenekleri belirlenmezse, bunlar varsayılan değerler ve işlemlerdir.

#### **MQPMO\_PASS\_IDENTITY\_CONTEXT**

İleti, kendisiyle ilişkili bağlam bilgisi içermesini sağlar. Kimlik bağlamı, *Context* alanında belirtilen kuyruk tanıtıcısından alınır. Kaynak bağlamı bilgileri, kuyruk yöneticisi tarafından, aynı şekilde MQPMO\_DEFAULT\_CONTEXT ile aynı şekilde oluşturulur (değerler için önceki çizelgeye bakın). İleti bağlamına ilişkin ek bilgi için [İleti bağlamı](#) başlıklı konuya bakın.

MQPUT çağrısı için, kuyruk MQOO\_PASS\_IDENTITY\_CONTEXT seçeneğiyle açılmış olmalıdır (ya da bunu belirten bir seçenek). MQPUT1 çağrısı için, MQOO\_PAS\_IDENTITY\_CONTEXT seçeneğiyle MQOPEN çağrısına ilişkin aynı yetki denetimi gerçekleştirilir.

#### **MQPMO\_PASS\_ALL\_CONTEXT**

İleti, kendisiyle ilişkili bağlam bilgisi içermesini sağlar. Bağlam, *Context* alanında belirtilen kuyruk tanıtıcısından alınır. İleti bağlamına ilişkin ek bilgi için [Bağlam bilgilerini denetleme](#) başlıklı konuya bakın.

MQPUT çağrısı için, kuyruğun MQOO\_PASS\_ALL\_CONTEXT seçeneğiyle (ya da bunu belirten bir seçenek) açılmış olması gerekir. MQPUT1 çağrısı için, MQOO\_PASS\_ALL\_CONTEXT seçeneğiyle MQOP çağrısına ilişkin olarak aynı yetki denetimi gerçekleştirilir.

#### **MQPMO\_SET\_IDENTITY\_CONTEXT**

İleti, kendisiyle ilişkili bağlam bilgisi içermesini sağlar. Uygulama, MQMD yapısındaki kimlik bağlamını belirtir. Kaynak bağlamı bilgileri, kuyruk yöneticisi tarafından, aynı şekilde MQPMO\_DEFAULT\_CONTEXT ile aynı şekilde oluşturulur (değerler için önceki çizelgeye bakın). İleti bağlamına ilişkin ek bilgi için [İleti bağlamı](#) başlıklı konuya bakın.

MQPUT çağrısı için, kuyruk MQOO\_SET\_IDENTITY\_CONTEXT seçeneğiyle (ya da bunu belirten bir seçenekle) açılmış olmalıdır. MQPUT1 çağrısı için, MQOO\_SET\_IDENTITY\_CONTEXT seçeneğiyle MQOpen çağrısına ilişkin aynı yetki denetimi gerçekleştirilir.

#### **MQPMO\_SET\_ALL\_CONTEXT**

İleti, kendisiyle ilişkili bağlam bilgisi içermesini sağlar. Uygulama, MQMD yapısındaki kimliği, kaynağı ve kullanıcı bağlamını belirtir. İleti bağlamına ilişkin ek bilgi için [İleti bağlamı](#) başlıklı konuya bakın.

MQPUT çağrısı için, kuyruğun MQOO\_SET\_ALL\_CONTEXT seçeneğiyle açılmış olması gerekir. MQPUT1 çağrısı için, MQOO\_SET\_ALL\_CONTEXT seçeneğiyle MQOP çağrısına ilişkin olarak aynı yetki denetimi gerçekleştirilir.

MQPMO\_\*\_CONTEXT bağlam seçeneklerinden yalnızca birini belirtebilirsiniz. Hiçbirini belirtmediyseniz, MQPMO\_DEFAULT\_CONTEXT kabul edilir.

**Özellik seçenekleri.** Aşağıdaki seçenek, iletinin özellikleri ile ilgilidir:

## **MQPMO\_MD\_FOR\_OUTPUT\_ONLY**

İleti tanımlayıcı parametresi yalnızca, gönderilen iletinin ileti tanımlayıcısını döndürecek çıkış için kullanılmalıdır. The message descriptor fields associated with the *NewMsgHandle*, *OriginalMsgHandle*, or both fields, of the **MQPMO** structure must be used for input.

Geçerli bir ileti tanıtıcısı sağlanmazsa, çağrı neden kodu **MQRC\_MD\_ERROR** ile başarısız olur.

**Yanıt seçeneklerini belirleyin.** Aşağıdaki seçenekler, bir MQPUT ya da MQPUT1 çağrısına döndürülen yanıtı denetler. Bu seçeneklerden yalnızca birini belirleyebilirsiniz. MQPMO\_ASYNC\_RESPONSE ve MQPMO\_SYNC\_RESPONSE belirtilmediyse, MQPMO\_RESPONSE\_AS\_Q\_DEF ya da MQPMO\_RESPONSE\_AS\_TOPIC\_DEF varsayıldı.

## **MQPMO\_ASYNC\_RESPONSE**

MQPMO\_ASYNC\_RESPONSE seçeneği, bir MQPUT ya da MQPUT1 işleminin, kuyruk yöneticisinin çağrısı tamamlamayı beklemesi için uygulama olmadan tamamlanır. Bu seçeneğin kullanılması, özellikle istemci bağ tanımlarını kullanan uygulamalar için ileti alışverişi başarımını yükseltebilirler. Bir uygulama, önceki zamanuyumsuz çağrılar sırasında bir hata oluşup oluşmadığını belirterek, MQSTAT komutunu kullanarak düzenli aralıklarla denetleyebilir.

Bu seçenekle, MQMD ' de yalnızca aşağıdaki alanların tamamlanabileceği garanti edilir;

- ApplIdentityVerileri
- PutApplTipi
- PutApplAdı
- ApplOriginVerileri

Buna ek olarak, MQPMO\_NEW\_MSG\_ID ya da MQPMO\_NEW\_CORREL\_ID seçeneklerinin ikisi ya da her ikisi de seçenek olarak belirtilirse, döndürülen MsgId ve CorrelId işlemleri de tamamlanır. (MQPMO\_NEW\_MSG\_ID, boş bir MsgId alanı belirtilerek örtük olarak belirtilebilir).

Yalnızca belirtilen alanlar tamamlanır. Olağan durumda MQMD ya da MQPMO yapısında döndürülecek diğer bilgiler tanımsızdır.

MQPUT1 için zamanuyumsuz koyma yanıtı istenirken, MQOD yapısında döndürülen ResolvedQName ve ResolvedQMgrAdı tanımsız.

MQPUT ya da MQPUT1 için zamanuyumsuz koyma yanıtı istenirken, bir CompCode ve MQCC\_OK ve MQRC\_NONE iletisinin nedeni, iletinin bir kuyruğa başarıyla konulduğu anlamına gelmeyebilir. Zamanuyumsuz put yanıtını kullanan ve iletilerin bir kuyruğa yerleştirilmesini gerektiren bir MQI uygulaması geliştirirken, hem CompCode ' i hem de put işlemlerinden neden kodlarına bakmanız ve ayrıca zamanuyumsuz hata bilgilerini sorgulamak için MQSTAT kullanmaları gerekir.

Her bir MQPUT ya da MQPUT1 call mightnot ögesinin başarısı ya da başarısızlığı hemen döndürülse de, zamanuyumsuz bir çağrı altında oluşan ilk hata, daha sonra MQSTAT çağrısı yoluyla saptlanabilir.

Zamanuyumsuz koyma yanıtı kullanılarak eşitleme noktası altındaki kalıcı bir ileti teslim edilemezse ve hareketi kesinleştirme girişiminde bulunursanız, kesinleştirme başarısız olur ve hareket, MQCC\_FAILED tamamlama koduyla ve MQRC\_BACKED\_OUT işlemi nedeniyle yedeklenir. Uygulama, önceki bir MQPUT ya da MQPUT1 hatasının nedenini saptamak için MQSTAT çağrısını yapabilir.

## **MQPMO\_SYNC\_RESPONSE**

Bu put yanıt tipinin belirtilmesi, MQPUT ya da MQPUT1 işleminin her zaman zamanuyumlu olarak yayınlanmasını sağlar. Koyma işlemi başarılı olursa, MQMD ve MQPMO ' daki tüm alanlar tamamlanır.

Bu seçenek, kuyruğun ya da konu nesnesinde tanımlanan varsayılan değer koyma yanıtı değerinden bağımsız olarak zamanuyumlu bir yanıt verilmesini sağlar.

## **MQPMO\_RESPONSE\_AS\_Q\_DEF**

Bu değer bir MQPUT çağrısı için belirtilirse, kullanılan put yanıt tipi, uygulama tarafından ilk kez açıldığında, kuyruğunda belirtilen DEFRESPP değerinden alınır.

- Kuyruk bir küme kuyruğalıysa ve bu değer bir MQPUT çağrısı için belirtilirse, kullanılan put yanıt tipi, iletinin yerleştiği kuyruğun belirli bir örneğinin sahibi olan *hedef* kuyruk yöneticisinde tanımlanan **DEFRESP** özneliğinden alınır.

Küme kuyruğunun birden çok eşgörünümü varsa ve bu öznitelikte farklılık gösterdiklerinde, bunlardan birinin tane seçkullanım seçtiği değer tahmin edilemez. Bu nedenle, bu özneliği tüm eşgörünümelerde aynı değere ayarlamalısınız. Böyle bir durumda değilse, kuyruk yöneticisi günlüklüklerine AMQ9407 hata iletisi gönderilir. Ayrıca bkz. Diğer adlar, uzak kuyruklar ve küme kuyrukları için hedef nesne öznitelikleri nasıl çözülür?

- Kuyruk bir küme kuyruğu değilse ve bu değer bir MQPUT çağrısı için belirtilirse, hedef kuyruk yöneticisi uzak olsa bile, kullanılan put yanıt tipi, *yemel* kuyruk yöneticisinde tanımlı olan **DEFPRESP** özneliğinden alınır.

Bir istemci uygulaması IBM WebSphere MQ 7.0' dan önceki bir düzeydeki bir kuyruk yöneticisine bağlıysa, bu, MQPMO\_SYNC\_RESPONSE işlemi için belirtilmiş gibi davranır.

Bir MQPUT1 çağrısı için bu seçenek belirlenirse, DEFPRESP özneliğinin değeri sunucuya istek gönderilmeden önce bilinmez. Varsayılan olarak, MQPUT1 çağrısı MQPMO\_SYNCPOINT kullanıyorsa, MQPMO\_ASYNC\_RESPONSE için işlev görür ve MQPMO\_SYNCPOINT MQPMO\_SYNCPOINT kullanıyorsa, bu işlev MQPMO\_SYNC\_RESPONSE için işlev görür. Ancak, istemci yapılandırma dosyasında Put1DefaultAlwaysSync özelliğini ayarlayarak bu varsayılan davranışı geçersiz kılabilirsiniz; bkz. İstemci yapılandırma dosyasının STANA kısmı.

### **MQPMO\_RESPONSE\_AS\_TOPIC\_DEF**

MQPMO\_RESPONSE\_AS\_TOPIC\_DEF, konu nesneleriyle kullanılmak üzere MQPMO\_RESPONSE\_AS\_Q\_DEF ile eşanlamıdır.

**Diğer seçenekler.** Aşağıdaki seçenekler denetim yetkisi denetlemesini, kuyruk yöneticisi susturma ve kuyruk ve kuyruk yöneticisi adlarını çözümlerken ne olur:

### **MQPMO\_ALTERNATE\_USER\_AUTHORITY**

MQPMO\_ALTERNATE\_USER\_AUTHORITY, MQPUT1 çağrısının **ObjDesc** parametresindeki *AlternateUserId* alanının, kuyrukta ileti koyma yetkisinin geçerliliğini denetlemek için kullanılacak bir kullanıcı kimliği içerdiğini belirtir. The call can succeed only if *AlternateUserId* is authorized to open the queue with the specified options, regardless of whether the user identifier under which the application is running is authorized to do so. (Bu seçenek, uygulamanın çalışmakta olduğu kullanıcı kimliğine her zaman geri verilen bağlam seçenekleri için geçerli değildir.)

Bu seçenek yalnızca MQPUT1 çağrısıyla geçerli olur.

### **MQPMO\_FAIL\_IF QUIESCING**

Bu seçenek, kuyruk yöneticisi susturma durumdaysa, MQPUT ya da MQPUT1 çağrısını başarısız olarak zorlar.

z/OS' ta bu seçenek, bağlantı ( CICS ya da IMS uygulaması için) susturulmuş durumdaysa, MQPUT ya da MQPUT1 çağrısını başarısız olarak zorlar.

Çağrı dönüş kodu MQCC\_FAILED neden kodu MQRC\_Q\_MGR\_QUIESCING ya da MQRC\_CONNECTION\_QUIESCING neden ile başarısız oldu.

### **MQPMO\_RESOLVE\_LOCAL\_Q**

Use this option to fill *ResolvedQName* in the MQPMO structure with the name of the local queue to which the message is put, and *ResolvedQMGrName* with the name of the local queue manager that hosts the local queue. MQPMO\_RESOLVE\_LOCAL\_Q ile ilgili ek bilgi için MQOO\_RESOLVE\_LOCAL\_Q başlıklı konuya bakın.

Bir kuyruğa koyma yetkiniz varsa, MQPUT çağrısında bu işareti belirtebilmek için gerekli yetkiniz vardır; özel bir yetki gerekmez.

**Varsayılan seçenek.** Tanımlanan seçeneklerden hiçbirine gerek duyarsanız, aşağıdaki seçeneği kullanın:

### **MQPMO\_NONE**

Başka bir seçenek belirtilmemeyi belirtmek için bu değeri kullanın; tüm seçenekler varsayılan değerlerini kabul eder. MQPMO\_NONE, program belgelerine yardımcı olmak için tanımlıdır; bu seçeneğin diğeriyle birlikte kullanılması amaçlanmaz, ancak değeri sıfır olduğu için, bu tür bir kullanım saptanamaz.

MQPMO\_NONE bir giriş alanıdır. *Options* alanının ilk değeri MQPMO\_NONE olur.

### **Zamanaşımı (MQUZE)**

Bu ayrılmış bir alandır; değeri anlamlı değildir. Bu alanın başlangıç değeri -1' dir.

### **Bağlam (MQHOB)**

MQPMO\_PASS\_IDENTITY\_CONTEXT ya da MQPMO\_PASS\_ALL\_CONTEXT belirtilirse, bu alan, konulmakta olan iletiyle ilişkilendirilecek bağlam bilgilerinin yer aldığı giriş kuyruğu tanıtıcısını içermelidir.

MQPMO\_PASS\_IDENTITY\_CONTEXT ya da MQPMO\_PASS\_ALL\_CONTEXT belirtilmezse, bu alan yoksayılır.

Bu bir giriş alanıdır. Bu alanın başlangıç değeri 0' dir.

### **KnownDestSayı (MQUZE)**

Bu, yürürlükteki MQPUT ya da MQPUT1 çağrısının, dağıtım listesindeki yerel kuyruklar olan kuyruklara başarıyla gönderdiği ileti sayısıdır. Bu sayı, uzak kuyruklara çözülen kuyruklara gönderilen iletileri (yerel iletim kuyruğu, iletiyi saklamak için başlangıçta kullanıldığı halde) içermez. Bu alan ayrıca, dağıtım listesinde olmayan tek bir kuyruğa ileti yerleştirilirken de ayarlanır.

Bu bir çıkış alanıdır. Bu alanın ilk değeri 0 'tır. *Version* , MQPMO\_VERSION\_1değerinden küçükse bu alan ayarlanmaz.

Dağıtım listeleri desteklenmediği için, bu alan z/OS üzerinde tanımsız.

### **UnknownDestSayı (MQUZE)**

Bu, yürürlükteki MQPUT ya da MQPUT1 çağrısının uzak kuyruklara çözümleyen dağıtım listesindeki kuyruklara başarılı bir şekilde gönderdiği ileti sayısıdır. Kuyruk yöneticisinin dağıtım listesi biçiminde geçici olarak alıkonacağı iletiler, bu dağıtım listelerinin içerdiği hedef noktaların sayısı olarak sayılır. Bu alan ayrıca, dağıtım listesinde olmayan tek bir kuyruğa ileti yerleştirilirken de ayarlanır.

Bu bir çıkış alanıdır. Bu alanın ilk değeri 0 'tır. *Version* , MQPMO\_VERSION\_1değerinden küçükse bu alan ayarlanmaz.

Dağıtım listeleri desteklenmediği için, bu alan z/OS üzerinde tanımsız.

### **InvalidDestSayı (MQUZE)**

Bu, dağıtım listesindeki kuyruklara gönderilebilecek iletilerin sayısıdır. Sayım, açılmadı ve başarıyla açılan, ancak koyma işleminin başarısız olduğu kuyruklar içeriyor. Bu alan ayrıca, dağıtım listesinde olmayan tek bir kuyruğa ileti yerleştirilirken de ayarlanır.

**Not:** Bu alan, MQPUT ya da MQPUT1 çağrısındaki **CompCode** parametresi MQCC\_OK ya da MQCC\_WARNING; ise ayarlanır; **CompCode** parametresi MQCC\_FAILED ise, ancak uygulama kodunda buna güvenilmeyin.

Bu bir çıkış alanıdır. Bu alanın ilk değeri 0 'tır. *Version* , MQPMO\_VERSION\_1değerinden küçükse bu alan ayarlanmaz.

Dağıtım listeleri desteklenmediği için, bu alan z/OS üzerinde tanımsız.

### **ResolvedQName (MQCHAR48)**

Bu ad, yerel kuyruk yöneticisi tarafından ad çözme işlemi gerçekleştirildikten sonra hedef kuyruğunun adı. The name returned is the name of a queue that exists on the queue manager identified by *ResolvedQMgrName*.

Boş olmayan bir değer yalnızca, nesne tek bir kuyruksa döndürülür; nesne bir dağıtım listesi ya da bir konu ise, döndürülen değer tanımsız olur.

Bu bir çıkış alanıdır. Bu alanın uzunluğu MQ\_Q\_NAME\_LENGTH değeri tarafından verilir. Bu alanın başlangıç değeri, C ' de boş değer dizgisidir ve diğer programlama dillerindeki 48 boş karakter dizilimidir.

### **ResolvedQMgrAdı (MQCHAR48)**

Bu ad, yerel kuyruk yöneticisi tarafından ad çözme işlemi gerçekleştirildikten sonra hedef kuyruk yöneticisinin adıdır. Döndürülen ad, *ResolvedQName* ile tanımlanan kuyruğa sahip olan kuyruk yöneticisinin adıdır ve yerel kuyruk yöneticisinin adı olabilir.

*ResolvedQName* , yerel kuyruk yöneticisinin ait olduğu kuyruk paylaşım grubunun sahip olduğu bir paylaşılan kuyruksa, *ResolvedQMgrName* kuyruk paylaşım grubunun adıdır. If the queue is owned by some other queue sharing group, *ResolvedQName* can be the name of the queue sharing group or the name of a queue manager that is a member of the queue sharing group (the nature of the value returned is determined by the queue definitions that exist at the local queue manager).

Boş olmayan bir değer yalnızca, nesne tek bir kuyruksa döndürülür; nesne bir dağıtım listesi ya da bir konu ise, döndürülen değer tanımsız olur.

Bu bir çıkış alanıdır. Bu alanın uzunluğu MQ\_Q\_MGR\_NAME\_LENGTH değeri tarafından verilir. Bu alanın başlangıç değeri, C ' de boş değer dizgisidir ve diğer programlama dillerindeki 48 boş karakter dizilimidir.

### **RecsPresent (MQUZE)**

Bu, uygulama tarafından sağlanan ileti kayıtlarını ya da MQRR yanıt kayıtlarını içeren MQPMR ' nin sayısıdır. Bu sayı, yalnızca ileti bir dağıtım listesine konulmakta olduğunda sıfırdan büyük olabilir. İleti kayıtları ve yanıt kayıtları isteğe bağlıdır; uygulamanın herhangi bir kayıt sağlamaması ya da yalnızca tek bir tipte kayıt sağlamayı seçmesi gerekir. Ancak, uygulama her iki tipteki kayıtları da sağlıyorsa, her tipte *RecsPresent* kaydı sağlamalıdır.

*RecsPresent* değerinin, dağıtım listesindeki hedeflerin sayısı ile aynı olması gerekmez. Çok sayıda kayıt sağlanırsa, fazla sayıda kayıt kullanılmaz; çok az kayıt sağlanırsa, ileti kaydı olmayan hedeflere ilişkin ileti özellikleri için varsayılan değerler kullanılır (bkz. *PutMsgRecOffset*).

*RecsPresent* sıfırdan küçükse ya da sıfırdan büyükse, ancak ileti bir dağıtım listesine konmadıysa, çağrı neden kodu MQRC\_RECS\_RESENT\_ERROR ile başarısız olur.

Bu bir giriş alanıdır. Bu alanın ilk değeri 0 'tır. *Version* , MQPMO\_VERSION\_2 değerinden küçükse bu alan yok sayılır.

### **PutMsgRecFields (MQlong)**

Bu alan, uygulama tarafından sağlanan ileti kaydı kayıtlarında hangi MQPMR alanlarının bulunduğunu belirten işaretler içerir. *PutMsgRecFields* yalnızca ileti bir dağıtım listesine konduğunda kullanılır. *RecsPresent* sıfırsa ya da hem *PutMsgRecOffset* , hem de *PutMsgRecPtr* sıfırsa, alan yoksayılar.

Var olan alanlar için, kuyruk yöneticisi ilgili put iletisi kaydındaki alanlardan değerlerin her bir hedefi için kullanılır. Eksik olan alanlar için kuyruk yöneticisi, MQMD yapısından değerleri kullanır.

Koyma iletisi kayıtlarında hangi alanların bulunduğunu belirtmek için aşağıdaki işaretlerden birini ya da birkaçını kullanın:

#### **MQPMRF\_MSG\_ID**

İleti tanıtıcısı alanı var.

#### **MQPMRF\_CORREL\_ID**

İlinti tanıtıcısı alanı var.

#### **MQPMRF\_GROUP\_ID**

Grup tanıtıcısı alanı var.

#### **MQPMRF\_FEEDBACK**

Geribildirim alanı mevcut.

#### **MQPMRF\_ACCOUNTING\_TOKEN**

Muhasebe belirteci alanı var.

Bu işareti belirtirseniz, *Options* alanında MQPMO\_SET\_IDENTITY\_CONTEXT ya da MQPMO\_SET\_ALL\_CONTEXT belirtin; bu koşul karşılanmazsa, çağrı neden kodu MQRC\_PMO\_RECORD\_FLAGS\_ERROR ile başarısız olur.

Hiçbir MQPMR alanı yoksa, aşağıdaki değer belirlenebilir:

## **MQPMRF\_NONE**

Hiçbir put-message kayıt alanı mevcut değil.

Bu değer belirtilirse, *RecsPresent* sıfır ya da hem *PutMsgRecOffset* , hem de *PutMsgRecPtr* sıfır olmalıdır.

MQPMRF\_NONE, program belgelerine yardımcı olmak için tanımlıdır. Bu değışmezin başka biriyle birlikte kullanılması amaçlanmaz, ancak değeri sıfır olduđu için bu tür bir kullanım saptanamaz.

If *PutMsgRecFields* contains flags that are not valid, or put message records are provided but *PutMsgRecFields* has the value MQPMRF\_NONE, the call fails with reason code MQRC\_PMO\_RECORD\_FLAGS\_ERROR.

Bu bir giriş alanıdır. Bu alanın ilk değeri MQPMRF\_NONE olur. *Version* , MQPMO\_VERSION\_2değerinden küçükse bu alan yok sayılır.

## **PutMsgRecOffset (MQlong)**

Bu, MQPMO yapısının başlangıcındaki ilk MQPMR put ileti kaydının bayt cinsinden görelî konutdur. Görelî konum pozitif ya da negatif olabilir. *PutMsgRecOffset* yalnızca ileti bir dağıtım listesine konduğunda kullanılır. *RecsPresent* sıfırda, alan yoksayılır.

İleti bir dağıtım listesine konduğunda, her hedef için iletinin belirli özelliklerini belirtmek üzere bir ya da daha çok MQPMR içeren bir ileti kaydı dizisi sağlanabilir; bu özellikler şunlardır:

- İleti Tanıtıcısı
- İlinti tanıtıcısı
- Grup tanıtıcısı
- Feedback değeri
- Hesap simgesi

Tüm bu özellikleri belirtmenize gerek yoktur, ancak seçtiğiniz alt küme her ne olursa olsun, alanları doğru sırayla belirtin. Ek ayrıntılar için MQPMR yapısına ilişkin açıklamaya bakın.

Genellikle, dağıtım listesi açıldığında MQOD tarafından belirlenmiş nesne kayıtları olduđu için, birden çok ileti kaydı olması gerekir; her bir ileti kaydı, ilgili nesne kaydı tarafından tanımlanan kuyruğa ilişkin ileti özelliklerini sağlar. Bu durumda ileti özellikleri dikkate alınmasa da, dağıtım listesindeki dağıtım listesindeki kuyruklar, dizideki uygun konumlarda bunlar için ayrılmış ileti kayıtlarını yerleştirmeye devam etmelidir.

Koyma ileti kayıtlarının sayısı, nesne kayıtlarının sayısından farklı olabilir. Nesne kayıtlarından daha az sayıda ileti kaydı varsa, ileti kaydı kayıtları yerleştirmeyen hedeflerin ileti özellikleri, MQMD ileti tanımlayıcısındaki ilgili alanlardan alınır. Nesne kayıtlarından daha fazla ileti kaydı varsa, fazlalık kullanılmaz (bunlara erişebilmek için yine de mümkün olmalıdır). Put ileti kayıtları isteğe bağlıdır, ancak bunlar sağlanırsathem' un *RecsPresent* olması gerekir.

İleti kayıtlarını, *PutMsgRecOffset* içindeki bir görelî konum belirterek ya da *PutMsgRecPtr* içinde bir adres belirterek, MQOD 'daki nesne kayıtlarına benzer bir şekilde sağlayın. Bunu nasıl yapacağına ilişkin ayrıntılar için, "[MQOD-Nesne tanımlayıcı](#)" sayfa 473 içinde açıklanan *ObjectRecOffset* alanına bakın.

Birden çok *PutMsgRecOffset* ve *PutMsgRecPtr* kullanılabilir; her ikisi de sıfır dışında bir durumda, çağrı neden kodu MQRC\_PUT\_MSG\_RECORDS\_ERROR ile başarısız olur.

Bu bir giriş alanıdır. Bu alanın ilk değeri 0 'tır. *Version* , MQPMO\_VERSION\_2değerinden küçükse bu alan yok sayılır.

## **ResponseRecGörelî Konumu (MQUZE)**

Bu, MQPMO yapısının başlangıcındaki ilk MQRR yanıt kaydının bayt cinsinden görelî konudur. Görelî konum pozitif ya da negatif olabilir. *ResponseRecOffset* yalnızca ileti bir dağıtım listesine konduğunda kullanılır. *RecsPresent* sıfırda, alan yoksayılır.

İletiyi bir dağıtım listesine koyarken, iletinin başarıyla gönderilmediği kuyrukları (MQRR 'deCompCode alanı) ve her başarısızlığın nedenini (MQRR' deReason alanı) tanımlamak için bir ya da daha çok MQRR

yanıt kaydı dizisi sağlayabilirsiniz. Kuyruk açılmadığından ya da koyma işlemi başarısız olduğu için ileti gönderilmemiş olabilir. Kuyruk yöneticisi yanıt kayıtlarını yalnızca, arama sonucu karışık olduğunda (yani, bazı iletiler başarılı bir şekilde başarısız olurken ya da tüm başarısız, ancak farklı nedenler için başarılı bir şekilde gönderildiğinde) ayarlar; MQRD\_MULTIPLE\_REASONS çağrısından neden kodu bu vakayı gösterir. Aynı neden kodu tüm kuyruklar için geçerliyse, bu neden MQRD ya da MQRD1 çağrısının **Reason** parametresine döndürülür ve yanıt kayıtları belirlenmez.

Genellikle, dağıtım listesi açıldığında, MQRD tarafından nesne kayıtlarının olduğu kadar yanıt kaydı vardır; gerektiğinde, her yanıt kaydı, ilgili nesne kaydı tarafından tanımlanan kuyruğa konmak üzere tamamlanma koduna ve neden koduna ayarlanır. Dağıtım listesindeki, açılmayan dağıtım listesindeki kuyruklar, dizideki uygun konumlarda kendilerine ayrılan yanıt kayıtları olmalıdır; ancak, bunlar tamamlama kodu ve neden kodu, koyma işlemi yerine, açık işlemden kaynaklanan bir neden kodudur.

Yanıt kayıtlarının sayısı, nesne kayıtlarından farklı olabilir. Nesne kayıtlarından daha az sayıda yanıt kaydı varsa, uygulama, koyma işleminin başarısız olduğu tüm hedefleri ya da başarısızlıkların nedenlerini tanımlayamayabilir. Nesne kayıtlarından daha fazla yanıt kaydı varsa, fazlalık kullanılmaz (bunlara erişebilmek için yine de olanaklı olması gerekir). Yanıt kayıtları isteğe bağlıdır, ancak bunlar belirtilirsethem' un *RecsPresent* olması gerekir.

Provide the response records in a similar way to the object records in MQRD, either by specifying an offset in *ResponseRecOffset*, or by specifying an address in *ResponseRecPtr*; for details of how to do this, see the *ObjectRecOffset* field described in [“MQRD-Nesne tanımlayıcı” sayfa 473](#). Ancak, *ResponseRecOffset* ve *ResponseRecPtr* ' den en fazla birini kullanın; Her ikisi de sıfır dışında bir durumsa, çağrı neden kodu MQRD\_RESPONSE\_RECORDS\_ERROR ile başarısız olur.

MQRD1 çağrısı için bu alanın sıfır olması gerekir. Bunun nedeni, yanıt bilgilerinin (istendiyse), MQRD nesne tanımlayıcısı tarafından belirtilen yanıt kayıtlarında döndürülmesinden kaynaklanır.

Bu bir giriş alanıdır. Bu alanın ilk değeri 0 'tır. *Version*, MQRD\_VERSION\_2değerinden küçükse bu alan yok sayılır.

### ***PutMsgRecPtr (MQRD)***

Bu, ileti kaydının ilk MQRD ' nin adresidir. *PutMsgRecPtr* yalnızca ileti bir dağıtım listesine konduğunda kullanılır. *RecsPresent* sıfırda, alan yok sayılır.

Put ileti kayıtlarını belirtmek için *PutMsgRecPtr* ya da *PutMsgRecOffset* kullanılabilir, ancak her ikisi de değil; ayrıntılar için [“PutMsgRecOffset \(MQRD\)” sayfa 506](#) ' e bakın. *PutMsgRecPtr* kullanılmıyorsa, boş değerli ya da boş byte 'a ayarlayın.

Bu bir giriş alanıdır. Bu alanın ilk değeri, göstergeleri destekleyen programlama dillerindeki boş değerli göstergedir ve tersi durumda, boş değerli byte dizilimidir. *Version*, MQRD\_VERSION\_2değerinden küçükse bu alan yok sayılır.

**Not:** Programlama dilinin gösterge veri tipini desteklemediği altyapılarda, bu alan uygun uzunluğun byte dizisi olarak bildirilir ve ilk değer, tüm boş değerli byte dizisi olarak kabul edilir.

### ***ResponseRecPtr (MQRD)***

Bu, ilk MQRD yanıt kaydının adresidir. *ResponseRecPtr* yalnızca ileti bir dağıtım listesine konduğunda kullanılır. *RecsPresent* sıfırda, alan yok sayılır.

Yanıt kayıtlarını belirtmek için *ResponseRecPtr* ya da *ResponseRecOffset* seçeneğini kullanın, ancak her ikisini de belirtmeyin; ayrıntılar için bkz. [“ResponseRecGörelili Konumu \(MQRD\)” sayfa 506](#). *ResponseRecPtr* kullanılmazsa, boş değerli ya da boş değerli baytlara ayarlayın.

MQRD1 çağrısı için, bu alan boş değerli ya da boş değerli byte olmalıdır. Bunun nedeni, yanıt bilgilerinin (istendiyse), MQRD nesne tanımlayıcısı tarafından belirtilen yanıt kayıtlarında döndürülmesinden kaynaklanır.

Bu bir giriş alanıdır. Bu alanın ilk değeri, göstergeleri destekleyen programlama dillerindeki boş değerli göstergedir ve tersi durumda, boş değerli byte dizilimidir. *Version*, MQRD\_VERSION\_2değerinden küçükse bu alan yok sayılır.

**Not:** Programlama dilinin gösterge veri tipini desteklemediği altyapılarda, bu alan uygun uzunluğun byte dizgisi olarak bildirilir ve ilk değer, tüm boş değerli byte dizgisi olarak kabul edilir.

### **OriginalMsgHandle (MQHMSG)**

Bu, bir iletinin isteğe bağlı olduğu bir tanıtıcıdır. Daha önce bir kuyruktan alınmış olabilir. Bu tutamaçın kullanımı, *Action* alanının değerine tabidir; ayrıca bkz. [NewMsgHandle](#).

Özgün ileti tutamacının içeriği **MQPUT** ya da **MQPUT1** çağrısı tarafından değiştirilmez.

Bu bir giriş alanıdır. Bu alanın başlangıç değeri **MQHM\_NONE**' dir. Sürüm **MQPMO\_VERSION\_3**değerinden küçükse bu alan yok sayılır.

### **NewMsgTanıtıcısı (MQHMSG)**

Bu, işlemin işlem alanının değerine tabi olduğu isteğe bağlı bir tanıtıcıdır. Bu ileti, iletinin özelliklerini tanımlar ve belirtildiyse, *OriginalMsgHandle* değerlerinin geçersiz kılınabilmesini sağlar.

**MQPUT** ya da **MQPUT1** çağrısından geri dönüşte, tutamaçtaki içerik gerçekte yerleştiren iletiyi yansıtır.

Bu bir giriş alanıdır. Bu alanın başlangıç değeri **MQHM\_NONE**' dir. Sürüm **MQPMO\_VERSION\_3**değerinden küçükse bu alan yok sayılır.

### **İşlem (MQUZE)**

Bu, gerçekleştirilmekte olan put tipini ve OriginalMsgtanıtıcısı alanı ile belirlenen özgün ileti arasındaki ilişkiyi ve NewMsgHandle alanı tarafından belirlenen yeni iletiyi belirtir. İletinin özellikleri, kuyruk yöneticisi tarafından belirlenen işlemin değerine göre seçilir.

MQPUT ya da MQPUT1 çağrılarında MsgDesc değiştirgesini kullanarak ileti tanımlayıcısının içeriğini belirtmeyi seçebilirsiniz. Diğer bir seçenek olarak, MsgDesc parametresini sağlamamak ya da yalnızca MQPMO yapısının Options alanında MQPMO\_MD\_FOR\_OUTPUT\_ONLY içerilerek çıkış olduğunu belirtmek olanaklıdır.

MsgDesc parametresi sağlanmaz ya da yalnızca çıkış olarak belirtilirse, bu konuda açıklanan kurallara göre, yeni iletiye ilişkin ileti tanımlayıcısı, MQPMO ' nun ileti işleyici alanlarından doldurulur.

Bağlam bilgilerini denetleme içinde açıklanan bağlam ayarı ve geçirme etkinlikleri, ileti tanımlayıcısı oluşturulduktan sonra yürürlüğe girmiştir.

Yanlış bir işlem değeri belirtilirse, çağrı neden kodu MQRC\_ACTION\_ERROR ile başarısız olur.

Aşağıdaki eylemlerden herhangi biri belirtilebilir:

#### **MQACTP\_YENİ**

Yeni bir ileti alınıyor ve program tarafından bir önceki iletiyle herhangi bir ilişki belirtilmiyor. İleti tanımlayıcısı aşağıdaki gibi oluşturulur:

- MQPUT ya da MQPUT1 çağrısında bir MsgDesc değeri belirtilirse ve MQPMO\_MD\_FOR\_OUTPUT\_ONLY, MQPMO.Options, bu, ileti tanımlayıcısı değiştirilmemiş olarak kullanılır.
- If a MsgDesc is not supplied, or MQPMO\_MD\_FOR\_OUTPUT\_ONLY is in the MQPMO.Options then the queue manager generates the message descriptor using a combination of properties from OriginalMsgHandle and NewMsgHandle. Yeni ileti tanıtıcısı üzerinde belirttik olarak ayarlanan ileti tanımlayıcı alanları, özgün ileti tanıtıcısındaki önceliklerden öncelikli olarak uygulanır.

İleti verileri, MQPUT ya da MQPUT1 arabellek değiştirgesinden alınır.

#### **MQAKP\_ILERI**

Önceden alınan bir ileti iletiliyor. Özgün ileti tanıtıcısı, daha önce alınmış olan iletiyi belirtir.

Yeni ileti tanıtıcısı, özgün ileti tanıtıcısındaki özelliklerde (ileti tanımlayıcısında yer alan herhangi biri de içinde olmak üzere) yapılan değişiklikleri belirtir.



İleti tanımlayıcısı aşağıdaki gibi oluşturulur:

- MQPUT ya da MQPUT1 çağrısında bir MsgDesc değeri belirtilirse ve MQPMO\_MD\_FOR\_OUTPUT\_ONLY, MQPMO.Options, bu, ileti tanımlayıcısı değiştirilmemiş olarak kullanılır.
- If a MsgDesc is not supplied, or MQPMO\_MD\_FOR\_OUTPUT\_ONLY is in the MQPMO.Options then the queue manager generates the message descriptor using a combination of properties from OriginalMsgHandle and NewMsgHandle. Yeni ileti tanıtıcısı üzerinde belirtik olarak ayarlanan ileti tanımlayıcı alanları, özgün ileti tanıtıcısındaki önceliklerden öncelikli olarak uygulanır.
- MQPMO\_NEW\_MSG\_ID ya da MQPMO\_NEW\_CORREL\_ID, MQPMO.Options, daha sonra bunlar onurlandırılır.

İleti özellikleri aşağıdaki gibi oluşturulur:

- MQPD.CopyOptions içinde MQCOPY\_FOREward içeren özgün ileti tutamacısındaki tüm özellikler
- Yeni ileti tutamacısından tüm özellikler. Özgün ileti tanıtıcısında bir özellik olarak aynı adı taşıyan yeni ileti tanıtıcısındaki her özellik için, yeni ileti tanıtıcısından değer alınır. Bu kuralın tek kural dışı durumu, yeni ileti tanıtıcısındaki özellik özgün ileti tanıtıcısındaki bir özellik ile aynı adı taşıyan özel bir durumdur, ancak özelliğin değeri boş değerde olur. Bu durumda, özellik iletiden kaldırılır.

İletilecek ileti verileri, MQPUT ya da MQPUT1 arabellek değiştirilmesinden alınır.

### **MQACTP\_CEVAPLA**

Önceden alınan bir iletiyle bir yanıt alınıyor. Özgün ileti tanıtıcısı, daha önce alınmış olan iletiyi belirtir.

Yeni ileti tanıtıcısı, özgün ileti tanıtıcısındaki özelliklerde (ileti tanımlayıcısında yer alan herhangi biri de içinde olmak üzere) yapılan değişiklikleri belirtir.

İleti tanımlayıcısı aşağıdaki gibi oluşturulur:

- MQPUT ya da MQPUT1 çağrısında bir MsgDesc değeri belirtilirse ve MQPMO\_MD\_FOR\_OUTPUT\_ONLY, MQPMO.Options, bu, ileti tanımlayıcısı değiştirilmemiş olarak kullanılır.
- Bir MsgDesc verilmediyse ya da MQPMO.Options, ilk ileti tanımlayıcı alanları aşağıdaki gibi seçilidir:

<i>Çizelge 510. Yanıt iletisi tanıtıcısı dönüşümü</i>	
<b>MQMD ' de alan</b>	<b>Kullanılan değer</b>
Rapor	MQRO_PASST_DISCARD_ANDEXP; bittiyse ve MQRO_DISCARD_MSG ayarlanır: MQRO_DISCARD_MSG tersi durumda MQRO_NONE
MsgType	MQMT_REPLY
Son kullanma tarihi	MQRO_PASST_DISCARD_ANDEXP; bittiyse ayarlandı: Giriş iletisinden kopyalandı tersi durumda MQE_UNSIINSIZ
Geribildirim	MQFB_YOK

Çizelge 510. Yanıt iletisi tanıtıcısı dönüşümü (devamı var)	
MQMD ' de alan	Kullanılan değer
MsgId	MQPMO_NEW_MSG_ID ayarlandıysa: Yeni bir ileti tanıtıcısı oluşturulur else, MQRO_PASS_MSG_ID ayarlandıysa: Giriş iletisinden kopyalandı tersi durumda MQMI_NONE
CorrelId	MQPMO_NEW_CORREL_ID ayarlandıysa: Yeni bir ilinti tanıtıcısı oluşturulur else if MQRO_COPY_MSG_ID_TO_CORREL_ID ayarı: Copied from the MsgId field of the giriş iletisi else, MQRO_PASST_CORIL_ID ayarlıysa: Copied from the CorrelId field of the giriş iletisi tersi durumda MQCI_NONE
BackoutCount	0
ReplyToQ	Boşluklar
ReplyToQMgr	Boşluklar
GroupId	MQGI_NONE
MsgSeqNumarası	1
Görelî Konum	0
MsgFlags	MQMF_NONE
OriginalLength	MQOL_TANIMSIZ

- Daha sonra, ileti tanımlayıcısı yeni ileti tanıtıcısı tarafından değiştirilir; yeni ileti tanıtıcısında özellik olarak belirttik olarak ayarlanan ileti tanımlayıcı alanları, daha önce açıklandığı gibi ileti tanımlayıcı alanlarından önceliklidir.

İleti özellikleri aşağıdaki gibi oluşturulur:

- MQPD.CopyOptions içinde MQCOPY\_REPLAY içeren özgün ileti tutamacındaki tüm özellikler
- Yeni ileti tutamacından tüm özellikler. Özgün ileti tanıtıcısında bir özellik olarak aynı adı taşıyan yeni ileti tanıtıcısındaki her özellik için, yeni ileti tanıtıcısından değer alınır. Bu kuralın tek kural dışı durumu, yeni ileti tanıtıcısındaki özellik özgün ileti tanıtıcısındaki bir özellik ile aynı adı taşıyan özel bir durumdur, ancak özelliğin değeri boş değerde olur. Bu durumda, özellik iletiden kaldırılır.

İletilecek ileti verileri, MQPUT/MQPUT1 Arabellek değiştirilmesinden alınır.

### MQACP\_REPORT

Önceden alınan bir iletinin sonucu olarak bir rapor oluşturuluyor. Özgün ileti tanıtıcısı, raporun oluşturulmasına neden olan iletiyi belirtir.

Yeni ileti tanıtıcısı, özgün ileti tanıtıcısındaki özelliklerde (ileti tanımlayıcısında yer alan herhangi biri de içinde olmak üzere) yapılan değişiklikleri belirtir.

İleti tanımlayıcısı aşağıdaki gibi oluşturulur:

- MQPUT ya da MQPUT1 çağrısında bir MsgDesc değeri belirtilirse ve MQPMO\_MD\_FOR\_OUTPUT\_ONLY, MQPMO.Options, bu, ileti tanımlayıcısı değiştirilmemiş olarak kullanılır.

- Bir MsgDesc verilmediyse ya da MQPMO.Options , ilk ileti tanımlayıcı alanları aşağıdaki gibi seçilir:

<i>Çizelge 511. İleti tanıtıcısı dönüşümünü raporla</i>	
<b>MQMD ' de alan</b>	<b>Kullanılan değer</b>
Rapor	MQRO_PASST_DISCARD_ANDEXP; bittiyse ve MQRO_DISCARD_MSG ayarlandı: MQRO_DISCARD_MSG tersi durumda MQRO_NONE
MsgType	MQMT_REPORT
Son kullanma tarihi	MQRO_PASST_DISCARD_ANDEXP; bittiyse ayarlandı: Giriş iletisinden kopyalandı tersi durumda MQE_UNSIINSIZ
MsgId	MQPMO_NEW_MSG_ID ayarlandıysa: Yeni bir ileti tanıtıcısı oluşturulur else, MQRO_PASS_MSG_ID ayarlandıysa: Giriş iletisinden kopyalandı tersi durumda MQMI_NONE
CorrelId	MQPMO_NEW_CORREL_ID ayarlandıysa: Yeni bir ilinti tanıtıcısı oluşturulur else if MQRO_COPY_MSG_ID_TO_CORREL_ID ayarı: Copied from the MsgId field of the giriş iletisi else, MQRO_PASST_CORIL_ID ayarlıysa: Copied from the CorrelId field of the giriş iletisi tersi durumda MQCI_NONE
BackoutCount	0
ReplyToQ	Boşluklar
ReplyToQMgr	Boşluklar
OriginalLength	<i>BufferLength</i> olarak ayarla

- Daha sonra, ileti tanımlayıcısı yeni ileti tanıtıcısı tarafından değiştirilir; yeni ileti tanıtıcısında özellik olarak belirtik olarak ayarlanan ileti tanımlayıcı alanları, daha önce açıklandığı gibi ileti tanımlayıcı alanlarından önceliklidir.

İleti özellikleri aşağıdaki gibi oluşturulur:

- MQPD.CopyOptions
- Yeni ileti tutamacından tüm özellikler. Özgün ileti tanıtıcısında bir özellik olarak aynı adı taşıyan yeni ileti tanıtıcısındaki her özellik için, yeni ileti tanıtıcısından değer alınır. Bu kuralın tek kural dışı durumu, yeni ileti tanıtıcısındaki özellik özgün ileti tanıtıcısındaki bir özellikle aynı adı taşıyan özel bir durumdur, ancak özelliğin değeri boş değerde olur. Bu durumda, özellik iletiden kaldırılır.

Sonuçtaki MQMD 'deki Feedback alanı, oluşturulacak raporu temsil eder. MQFB\_NONE 'in bir Feedback değeri, MQPUT ya da MQPUT1 çağrısının, MQRC\_FEEDBACK\_ERROR neden koduyla başarısız olmasına neden olur.

Rapor iletilisinin kullanıcı verilerini seçmek için IBM MQ , sonuçtaki MQMD 'deki Rapor ve Geri Bildirim alanlarını ve MQPUT ya da MQPUT1 çağrısının Arabellek ve BufferLength deęiřtirgeleriyle iletiřim kurar.

- Geribildirim MQFB\_COA ise, MQFB\_COD ya da MQFB\_EXPIRATION ise, Rapor değeri incelenir.
- If any of the following cases is true, the full message data from Buffer for a length of BufferLength is used.
  - Geribildirim: MQFB\_EXPIRATION ve Rapor, MQRO\_EXPIRATION\_WITH\_FULL\_DATA içeriyor.
  - Feedback is MQFB\_COD ve Report, MQRO\_COD\_WITH\_FULL\_DATA içeriyor
  - Geribildirim MQFB\_COA 'dır ve Rapor, MQRO\_COA\_WITH\_FULL\_DATA içeriyor
- Ařaęıdaki durumlardan biri geęerliyse, iletinin ilk 100 baytı (ya da bu arabellek 100 'den küçükse BufferLength ) kullanılır.
  - Geribildirim: MQFB\_EXPIRATION ve Rapor, MQRO\_EXPIRATION\_WITH\_DATA içeriyor
  - Geribildirim MQFB\_COD 'dir ve Rapor, MQRO\_COD\_WITH\_DATA içeriyor.
  - Geribildirim MQFB\_COA 'dır ve Rapor, MQRO\_COA\_WITH\_DATA içeriyor
- Feedback MQFB\_EXPIRATION, MQFB\_COD ya da MQFB\_COA ise ve Rapor, o Feedback değeriyle ilgili \*\_WITH\_FULL\_DATA ya da \*\_WITH\_DATA seęeneklerini içermiyorsa, bu iletiyle hiębir kullanıcı verisi içerilmez.
- Geribildirim, yukarıda listelenen değerdan farklı bir değeri alırsa, arabellek ve BufferLength değeri normal olarak kullanılır.

Önceki listede açıklanan kullanıcı verilerinin türetme iřlemi ařaęıdaki çizelgede de gösterilmiřtir:

<i>Çizelge 512. Kullanıcı verilerinin kaynaęı</i>			
	<b>MQFB_COA</b>	<b>MQFB_COD</b>	<b>MQFB_BITIM</b>
<b>MQRO_EXPIRATION_WITH_FULL_DATA</b>	Yok	Yok	Arabellek (Bufferlength)
<b>MQRO_COD_ILE_FULL_DATA</b>	Yok	Arabellek (Bufferlength)	Yok
<b>MQRO_COA_ILE_FULL_DATA</b>	Arabellek (Bufferlength)	Yok	Yok
<b>MQRO_EXPIRATION_HIZME_VERİ</b>	Yok	Yok	Arabellek (İlk 100 bayt)
<b>MQRO_COD_WITH_DATA</b>	Yok	Arabellek (İlk 100 bayt)	Yok
<b>MQRO_CO_WITH_DATA</b>	Arabellek (İlk 100 bayt)	Yok	Yok

### **PubLevel (MQUTR)**

Bu alanın ilk değeri 9 'tır. Bu yayın tarafından hedeflenen abonelik düzeyi. Yalnızca, bu değere eřit ya da bu değerdan küçük olan en yüksek SubLevel abonelikleri bu yayını alır. Bu değeri, sıfır ile 9 aralıęında olmalıdır; sıfır, en düşük düzeydir. Ancak, bir yayın saklandıysa, PubLevel 1 düzeyinde yeniden yayınlandıęından, bu yayın artık daha yüksek düzeylerde aboneler için kullanılabilir değildir.

Bilgi için bkz. [Intercepting yayınları](#).

## MQPMR-Put-message kaydı

Bir iletiyi dağıtım listesine koyarken tek bir hedefe ilişkin çeşitli ileti özelliklerini belirtmek için MQPMR yapısını kullanın. MQPMR, MQPUT ve MQPUT1 çağrılarında ilişkin bir giriş/çıkış yapısıdır.

### Kullanılabilirlik

MQPMR yapısı aşağıdaki altyapılarda kullanılabilir:

-  AIX
-  IBM i
-  Linux
-  Windows

ve bu sistemlere bağlı IBM MQ istemcileri için.

### Karakter kümesi ve kodlama

MQPMR içindeki veriler, **CodedCharSetId** kuyruk yöneticisi özniteliği tarafından belirtilen karakter kümesinde ve MQENC\_NATIVE tarafından verilen yerel kuyruk yöneticisinin kodlamasında yer almalıdır. Ancak, uygulama bir MQ istemcisi olarak çalışıyorsa, yapı istemcinin karakter kümesinde ve kodlamasında olmalıdır.

### Kullanım

MQPUT ya da MQPUT1 çağrısında bu yapılardan oluşan bir dizi sağlayarak, dağıtım listesindeki her hedef kuyruk için farklı değerler belirtebilirsiniz. Alanlardan bazıları yalnızca giriş, diğerleri giriş/çıkış.

**Not:** Bu yapı, sabit bir düzene sahip olmadığı için alışılmadık bir yapıdır. Bu yapıdaki alanlar isteğe bağlıdır ve her alanın varlığı ya da yokluğu, MQPMO 'daki *PutMsgRecFields* alanındaki işaretlerle gösterilir. **olan alanlar aşağıdaki sırayla oluşmalıdır :**

- *MsgId*
- *CorrelId*
- *GroupId*
- *Feedback*
- *AccountingToken*

Eksik olan alanlar, kayıta yer kaplamıyor.

MQPMR sabit bir yerleşim düzenine sahip olmadığı için, üstbilgide, COPY ve INCLUDE dosyalarında desteklenen programlama dillerine ilişkin bir tanımlama sağlanmaz. Uygulama programcısı, uygulamanın gerektirdiği alanları içeren bir bildirim yaratmalı ve var olan alanları belirtmek için *PutMsgRecFields* içindeki işaretleri ayarlamalıdır.

### Alanlar

Desteklenen programlama dilleri için üstbilgi, COPYve INCLUDE dosyalarında yapı bildirimleri sağlanmadığı için bu yapı için tanımlanmış başlangıç değeri yoktur. Örnek bildirimler, tüm alanlar gerekliyse yapının nasıl bildirileceğini gösterir.

Alan adı	Alan açıklaması
<u>MsgId</u>	İleti Tanıtıcısı
<u>CorrelId</u>	İlinti tanıtıcısı

Çizelge 513. MQPMM içindeki alanlar (devamı var)

Alan adı	Alan açıklaması
GroupId	Grup tanıtıcısı
Geri bildirim	Geribildirim ya da neden kodu
AccountingToken	Muhasebe simgesi

## Dil bildirimleri

### MQPMM için C bildirim

```
typedef struct tagMQPMM MQPMM;  
struct tagMQPMM {  
    MQBYTE24  MsgId;           /* Message identifier */  
    MQBYTE24  CorrelId;        /* Correlation identifier */  
    MQBYTE24  GroupId;         /* Group identifier */  
    MQLONG    Feedback;        /* Feedback or reason code */  
    MQBYTE32  AccountingToken; /* Accounting token */  
};
```

### MQPMM için COBOL bildirim

```
** MQPMM structure  
10 MQPMM.  
** Message identifier  
15 MQPMM-MSGID PIC X(24).  
** Correlation identifier  
15 MQPMM-CORRELID PIC X(24).  
** Group identifier  
15 MQPMM-GROUPID PIC X(24).  
** Feedback or reason code  
15 MQPMM-FEEDBACK PIC S9(9) BINARY.  
** Accounting token  
15 MQPMM-ACCOUNTINGTOKEN PIC X(32).
```

### MQPMM için PL/I bildirim

```
dcl  
1 MQPMM based,  
3 MsgId char(24), /* Message identifier */  
3 CorrelId char(24), /* Correlation identifier */  
3 GroupId char(24), /* Group identifier */  
3 Feedback fixed bin(31), /* Feedback or reason code */  
3 AccountingToken char(32); /* Accounting token */
```

### MQPMM için Visual Basic bildirim

```
Type MQPMM  
MsgId As MQBYTE24 'Message identifier'  
CorrelId As MQBYTE24 'Correlation identifier'  
GroupId As MQBYTE24 'Group identifier'  
Feedback As Long 'Feedback or reason code'  
AccountingToken As MQBYTE32 'Accounting token'  
End Type
```

## MsgId (MQBYTE24)

Bu ileti, MQOPER ya da MQPUT1 çağrısında bulunan MQOR yapılarının dizesindeki ilgili ögenin belirlediği bir adla kuyruğa gönderilen ileti için kullanılacak ileti tanıtıcısıdır. Bu, tek bir kuyruk için MQMD 'deki *MsgId* alanı ile aynı şekilde işlenir.

If this field is not present in the MQPMM record, or there are fewer MQPMM records than destinations, the value in MQMD is used for those destinations that do not have an MQPMM record containing a *MsgId* field.

Bu deęer MQMI\_NONE ise, bu hedeflerin *her biri* için yeni bir ileti tanıtıcısı oluşturulur (yani, bu hedeflerin ikisi de aynı ileti tanıtıcısına sahip deęildir).

MQPMO\_NEW\_MSG\_ID belirtilirse, dağıtım listesindeki tüm hedefler için MQPMR kayıtları olup olmadığına bakılmaksızın yeni ileti tanıtıcıları oluşturulur. Bu, MQPMO\_NEW\_CORREL\_ID ' nin işleme biçiminden farklıdır (*CorrelId* alanına bakın).

Bu bir giriş/çıkış alanıdır.

### **CorrelId (MQBYTE24)**

Bu, MQOPER ya da MQPUT1 çağrısında bulunan MQOR yapılarının dizideki ilgili öęe tarafından belirtilen bir adla kuyruęa gönderilen ileti için kullanılacak ilinti tanımlayıcısıdır. Bu, tek bir kuyruk için MQMD ' deki *CorrelId* alanı ile aynı şekilde işlenir.

If this field is not present in the MQPMR record, or there are fewer MQPMR records than destinations, the value in MQMD is used for those destinations that do not have an MQPMR record containing a *CorrelId* field.

MQPMO\_NEW\_CORREL\_ID belirtilirse, bir *tek* yeni ilinti tanıtıcısı oluşturulur ve MQPMR kayıtları olup olmadığına bakılmaksızın, dağıtım listesindeki tüm hedefler için kullanılır. Bu, MQPMO\_NEW\_MSG\_ID ' nin işleme şekline farklıdır (*MsgId* alanına bakın).

Bu bir giriş/çıkış alanıdır.

### **GroupId (MQBYTE24)**

GroupId , MQOPER ya da MQPUT1 çağrısında bulunan MQOR yapılarının dizideki ilgili öęe tarafından belirtilen ada sahip kuyruęa gönderilen ileti için kullanılacak grup tanıtıcısıdır. Bu, tek bir kuyruk için MQMD ' deki *GroupId* alanı ile aynı şekilde işlenir.

If this field is not present in the MQPMR record, or there are fewer MQPMR records than destinations, the value in MQMD is used for those destinations that do not have an MQPMR record containing a *GroupId* field. Deęer, Kuyruklardaki fiziksel sıralama içinde belgelendięi şekilde işlenir, ancak aşıęıdaki farklar söz alır:

- GroupId , QMName 'den ve bir zaman damgasından yaratılır. Bu nedenle, bir GroupId benzersiz tutma kuyruęu yöneticisi adlarını da benzersiz tutmanız gerekir. Ayrıca, kuyruk yöneticileri makinesinde saatleri geri ayarlamayın.
- Yeni bir grup tanıtıcısının kullanılabileceęi durumlarda, kuyruk yöneticisi her hedef için farklı bir grup tanıtıcısı oluşturur (yani, iki hedef aynı grup tanıtıcısına sahip deęildir).
- Alandaki deęerin kullanılacağı durumlarda, çağrı neden kodu MQRC\_GROUP\_ID\_ERROR ile başarısız olur.

Bu bir giriş/çıkış alanıdır.

### **Geribildirim (MQUZE)**

Bu, MQOPED ya da MQPUT1 çağrısında bulunan MQOR yapılarına karşılık gelen öęe tarafından belirtilen adda kuyruęa gönderilen ileti için kullanılacak geri bildirim kodudur. Bu, tek bir kuyruk için MQMD ' deki *Feedback* alanı ile aynı şekilde işlenir.

Bu alan mevcut deęilse, MQMD ' deki deęer kullanılır.

Bu bir giriş alanıdır.

### **AccountingToken (MQBYTE32)**

Bu, MQOPER ya da MQPUT1 çağrısında bulunan MQOR yapılarının dizideki ilgili öęe tarafından belirtilen adla kuyruęa gönderilen ileti için kullanılacak hesap simgesindeki bir simgedir. Bu, tek bir kuyruk için MQMD ' deki *AccountingToken* alanı ile aynı şekilde işlenir. Bu alanın içerięiyle ilgili bilgi için "MQMD-İleti tanımlayıcı" sayfa 416 içindeki *AccountingToken* açıklamalarına bakın.

Bu alan mevcut deęilse, MQMD ' deki deęer kullanılır.

Bu bir giriş alanıdır.

## MQRFH-Kurallar ve biçimlendirme üstbilgisi

MQRFH yapısı, kuralların ve biçimlendirme üstbilgisinin düzenini tanımlar. Dizilim verilerini ad-değer çiftleri biçiminde göndermek için bu üstbilgiyi kullanın.

### Kullanılabilirlik

Tüm IBM MQ sistemleri ve bu sistemlere bağlı IBM MQ MQI clients .

### Biçim Adı

MQFMT\_RF\_HEADER

### Karakter kümesi ve kodlama

MQRFH yapısındaki alanlar (*NameValueString*dahil), MQRFH ' den önce gelen üstbilgi yapısındaki *CodedCharSetId* ve *Encoding* alanları tarafından verilen karakter kümesinde ve kodlamasında ya da MQRFH uygulama iletisi verilerinin başlangıcında ise MQMD yapısındaki bu alanlar tarafından bulunur.

Karakter takımı, kuyruk adlarında geçerli olan karakterler için tek baytlık karakterler içeren bir karakter olmalıdır.

### Alanlar

**Not:** Aşağıdaki tabloda, alanlar alfabetik olarak değil, kullanıma göre gruplanır. Alt konular aynı sırayı izler.

Çizelge 514. MQRFH için MQRFH ' deki Alanlar		
Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
StrucId (yapı tanıtıcısı)	MQRFH_STRUC_ID	'RFH↵'
Sürüm (yapı sürümü numarası)	MQRFH_VERSION_1	1
StrucLength (MQRFH yapısının bayt cinsinden uzunluğu)	MQRFH_STRUC_LENGTH_FIXED	32
Kodlama ( <i>NameValueString</i> ' i izleyen verilerin sayısal kodlaması)	MQENC_NATIVE	Ortama bağlıdır
CodedCharSetId ( <i>NameValueString</i> ' u izleyen verilerin karakter kümesi tanıtıcısını belirtir)	MQCCSI_AYRILMAMIŞ	0
Biçim (aşağıdaki verilerin biçim adı <i>NameValueString</i> )	MQFMT_NONE	Boşluklar
İşaretler (işaretler)	MQRFH_NONE	0
NameValueString (ad-değer çiftlerini içeren değişken uzunluklu karakter dizisi)	yok	yok
<b>Notlar:</b>		
1. ↵ simgesi, tek bir boş karakteri temsil eder.		
2. C programlama dilinde, makro değişkeniMQRFH_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlar için başlangıç değerleri sağlamak üzere aşağıdaki şekilde kullanılabilir:		
<pre>MQRFH MyRFH = {MQRFH_DEFAULT};</pre>		



## Dil bildirimleri

### MQRFH için C bildirimi

```
typedef struct tagMQRFH MQRFH;
struct tagMQRFH {
    MCHAR4   StrucId;           /* Structure identifier */
    MQLONG   Version;          /* Structure version number */
    MQLONG   StrucLength;      /* Total length of MQRFH including
                               NameValueString */
    MQLONG   Encoding;         /* Numeric encoding of data that follows
                               NameValueString */
    MQLONG   CodedCharSetId;   /* Character set identifier of data that
                               follows NameValueString */
    MCHAR8   Format;           /* Format name of data that follows
                               NameValueString */
    MQLONG   Flags;           /* Flags */
};
```

### MQRFH için COBOL bildirimi

```
** MQRFH structure
10 MQRFH.
** Structure identifier
15 MQRFH-STRUCID PIC X(4).
** Structure version number
15 MQRFH-VERSION PIC S9(9) BINARY.
** Total length of MQRFH including NAMEVALUESTRING
15 MQRFH-STRUCLENGTH PIC S9(9) BINARY.
** Numeric encoding of data that follows NAMEVALUESTRING
15 MQRFH-ENCODING PIC S9(9) BINARY.
** Character set identifier of data that follows NAMEVALUESTRING
15 MQRFH-CODEDCHARSETID PIC S9(9) BINARY.
** Format name of data that follows NAMEVALUESTRING
15 MQRFH-FORMAT PIC X(8).
** Flags
15 MQRFH-FLAGS PIC S9(9) BINARY.
```

### MQRFH için PL/I bildirimi

```
dcl
1 MQRFH based,
3 StrucId char(4), /* Structure identifier */
3 Version fixed bin(31), /* Structure version number */
3 StrucLength fixed bin(31), /* Total length of MQRFH including
                               NameValueString */
3 Encoding fixed bin(31), /* Numeric encoding of data that
                               follows NameValueString */
3 CodedCharSetId fixed bin(31), /* Character set identifier of data
                               that follows NameValueString */
3 Format char(8), /* Format name of data that follows
                               NameValueString */
3 Flags fixed bin(31); /* Flags */
```

### MQRFH için High Level Assembler bildirimi

MQRFH	DSECT	
MQRFH_STRUCID	DS CL4	Structure identifier
MQRFH_VERSION	DS F	Structure version number
MQRFH_STRUCLENGTH	DS F	Total length of MQRFH including NAMEVALUESTRING
* MQRFH_ENCODING	DS F	Numeric encoding of data that follows NAMEVALUESTRING
* MQRFH_CODEDCHARSETID	DS F	Character set identifier of data that follows NAMEVALUESTRING
* MQRFH_FORMAT	DS CL8	Format name of data that follows NAMEVALUESTRING
* MQRFH_FLAGS	DS F	Flags
* MQRFH_LENGTH	EQU *-MQRFH	
	ORG MQRFH	
MQRFH_AREA	DS CL(MQRFH_LENGTH)	

## MQRFH için Visual Basic bildirimi

```
Type MQRFH
  StrucId      As String*4 'Structure identifier'
  Version     As Long      'Structure version number'
  StrucLength  As Long      'Total length of MQRFH including'
  Encoding    As Long      'Numeric encoding of data that follows'
  CodedCharSetId As Long    'Character set identifier of data that'
  Format       As String*8  'Format name of data that follows'
  Flags       As Long      'Flags'
End Type
```

### **StrucId (MQCHAR4)**

Bu, yapı tanıtıcısıdır; değer şöyle olmalıdır:

#### **MQRFH\_STRUC\_ID**

Kurallar ve biçimlendirme üstbilgisi yapısına ilişkin tanıtıcı.

C programlama dili için, MQRFH\_STRUC\_ID\_ARRAY değişmezi de tanımlanır; bu, MQRFH\_STRUC\_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisidir.

Bu alanın ilk değeri MQRFH\_STRUC\_ID 'dir.

### **Sürüm (MQUZE)**

Bu, yapı sürüm numarasıdır; değer şu olmalıdır:

#### **MQRFH\_VERSION\_1**

Version-1 kuralları ve biçimlendirme üstbilgisi yapısı.

Bu alanın ilk değeri MQRFH\_VERSION\_1 'dir.

### **StrucLength (MQUZE)**

Bu, yapının sonundaki *NameValueString* alanı da dahil olmak üzere, MQRFH yapısının bayt cinsinden uzunluğudur. Uzunluk, *NameValueString* alanını izleyen herhangi bir kullanıcı verisi içermez.

Bazı ortamlardaki kullanıcı verilerini dönüştürmeyi önlemek için *StrucLength* , dört adet birden çok olmalıdır.

Aşağıdaki değişmez, yapının *sabit* bir kısmının uzunluğunu gösterir; yani, *NameValueString* alanı hariç olmak üzere uzunluk.

#### **MQRFH\_STRUC\_LENGTH\_FIXED**

MQRFH yapısının değişmez kısmı uzunluğu.

Bu alanın ilk değeri MQRFH\_STRUC\_LENGTH\_FIXED olur.

### **Kodlama (MQUZE)**

Bu, *NameValueString* 'i izleyen verilerin sayısal kodlamasını belirtir. MQRFH yapısındaki sayısal veriler için geçerli değildir.

MQPUT ya da MQPUT1 çağrısında, uygulama bu alanı veri için uygun değere ayarlamalıdır.

Bu alanın ilk değeri MQENC\_NATIVE 'dir.

### **CodedCharSetId (MQHOMER)**

Bu, *NameValueString* 'i izleyen verilerin karakter kümesi tanıtıcısını belirtir. MQRFH yapısındaki karakter verileri için geçerli değildir.

MQPUT ya da MQPUT1 çağrısında, uygulama bu alanı veri için uygun değere ayarlamalıdır. Aşağıdaki özel değer kullanılabilir:

### **MQCCSI\_INHERIT**

Character data in the data *takip edilen* this structure is in the same character set as this structure.

Kuyruk yöneticisi, bu değeri, iletide belirtilen yapının gerçek karakter kümesi tanıtıcısına göre değiştirir. Herhangi bir hata oluşmaması koşuluyla, MQCCSI\_INHERIT değeri MQGET çağrısıyla döndürülmez.

MQMD 'deki *PutAppType* alanının değeri MQAT\_BROKER ise, MQCCSI\_INHERIT kullanılamaz.

Bu alanın ilk değeri MQCCSI\_UNDEFED değerine sahip.

### **Biçim (MQCHAR8)**

Bu, *NameValueString*'i izleyen verilerin biçim adını belirtir.

MQPUT ya da MQPUT1 çağrısında, uygulama bu alanı veri için uygun değere ayarlamalıdır. Bu alanı kodlamaya ilişkin kurallar, MQMD 'deki *Format* alanı için yapılan kurallarla aynıdır.

Bu alanın ilk değeri MQFMT\_NONE olur.

### **İşaretler (MQUZE)**

Şu öge belirtilebilir:

#### **MQRFH\_NONE**

Bayrak yok.

Bu alanın ilk değeri MQRFH\_NONE olur.

### **NameValueDizgisi (MQCHARn)**

Bu, formdaki ad-değer çiftlerini içeren değişken uzunluklu bir karakter dizilimidir:

```
name1 value1 name2 value2 name3 value3 ...
```

Her ad ya da değer, bitişik ad ya da değerden bir ya da daha çok boşluk karakteriyle ayrılmalıdır; bu boşlukların boş olması gerekir. Bir ad ya da değer, adı ya da değeri çift tırnak işareti içine alarak ve adı ya da değeri düzeltilerek önemli boşluklar içerebilir; açık çift tırnak işareti ile eşleşen kapanış çift tırnak işareti arasındaki tüm karakterler anlamlı olarak değerlendirilir. In the following example, the name is FAMOUS\_WORDS, and the value is Hello World:

```
FAMOUS_WORDS "Hello World"
```

Bir ad ya da değer, boş karakter dışında (*NameValueString*için sınırlayıcı olarak hareket eder) dışında herhangi bir karakter içerebilir. Ancak, bir uygulamanın birlikte çalışabilirliğe yardımcı olması için adları aşağıdaki karakterlerle sınırlayabilirler:

- İlk karakter: alfabetik ya da küçük harf alfabetik (A-Z, a-z) ya da alt çizgi.
- Sonraki karakterler: büyük ya da küçük harf alfabetik, ondalık basamak (0-9), altçizgi, kısa çizgi ya da nokta.

Bir ad ya da değer bir ya da daha çok çift tırnak işareti içeriyorsa, ad ya da değer çift tırnak işareti içine alınmalıdır ve dizgi içindeki her bir çift tırnak işareti iki katına çıkarılmalıdır:

```
Famous_Words "The program displayed ""Hello World"""
```

Adlar ve değerler büyük/küçük harf duyarlıdır, yani küçük harflerin büyük harflerle aynı olması dikkate alınmamaktadır. Örneğin, FAMOUS\_WORDS ve Famous\_Words iki farklı addır.

*NameValueString* bayt cinsinden uzunluk, *StrucLength* eksi MQRFH\_STRUC\_LENGTH\_FIXED değerine eşittir. Bazı ortamlardaki kullanıcı verilerini dönüştürmeyi önlemek için bu uzunluğu dörtten

çok bir değer yapın. Dizedeki son anlamlı karakterin ardından boş değerli bir karakter yerleştirerek, *NameValueString* 'u bu uzunluktaki boşluklarla doldur ya da daha önce boş bir karakter yerleştirerek not edin. Boş değer karakteri ve bu karakteri izleyen byte 'lar, belirtilen *NameValueString* uzunluğuna kadar yoksayılr.

**Not:** Bu alanın uzunluğu değişmez olduğu için, alan, desteklenen programlama dilleri için sağlanan yapının bildirimlerinden çıkarılır.

## MQRFH2 -Kurallar ve biçimlendirme üstbilgisi 2

MQRFH2 üstbilgisi MQRFH üstbilgisine dayalıdır, ancak Unicode dizgilerinin çeviri olmadan taşınmasına izin verir ve sayısal veri tiplerini taşıyabilir. MQRFH2 yapısı, version-2 kurallarının ve biçimlendirme üstbilgisinin biçimini tanımlar. XML benzeri bir sözdizimi kullanılarak kodlanmış verileri göndermek için bu üstbilgiyi kullanırsınız. Bir ileti seri olarak iki ya da daha fazla MQRFH2 yapısı içerebilir; kullanıcı verileri isteğe bağlı olarak serideki son MQRFH2 yapısını takip edebilir.

### Kullanılabilirlik

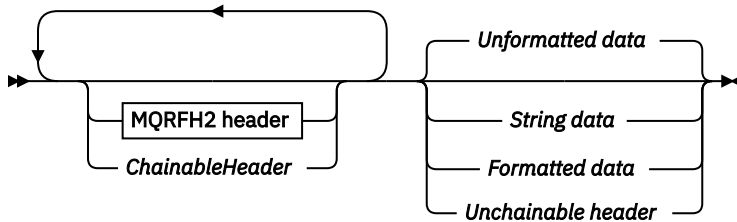
Tüm IBM MQ sistemleri ve bu sistemlere bağlı IBM MQ MQI clients .

### Biçim Adı

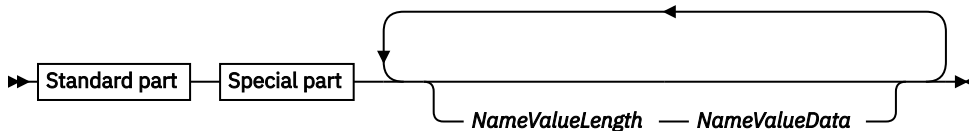
MQFMT\_RF\_HEADER\_2

### Syntax

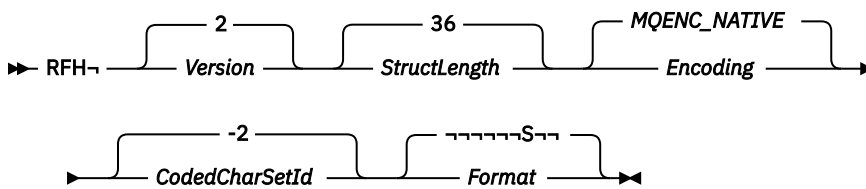
#### IBM MQ Message



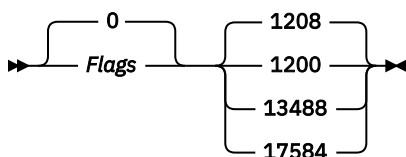
#### MQRFH2 header



#### Standard part



#### Special part



## Karakter kümesi ve kodlama

Özel kurallar, MQRFH2 yapısı için kullanılan karakter kümesi ve kodlama için geçerlidir:

- *NameValueData* dışındaki alanlar, MQRFH2' den önce gelen üstbilgi yapısındaki *CodedCharSetId* ve *Encoding* alanları tarafından verilen karakter kümesinde ve kodlamasında ya da MQRFH2 uygulama iletisi verilerinin başındaki ise MQMD yapısındaki bu alanlar tarafından bulunur.

Karakter takımı, kuyruk adlarında geçerli olan karakterler için tek baytlık karakterler içeren bir karakter olmalıdır.

MQGET çağrısında MQGMO\_CONVERT belirtildiğinde, kuyruk yöneticisi MQRFH2 alanlarını *NameValueData* dışında istenen karakter kümesine ve kodlamaya dönüştürür.

- *NameValueData* , *NameValueCCSID* alanı tarafından verilen karakter kümesinde bulunur. *NameValueCCSID* ; için yalnızca listelenen Unicode karakter kümeleri geçerlidir. Ayrıntılar için *NameValueCCSID* açıklamasına bakın.

Bazı karakter kümelerinin kodlamaya bağlı bir gösterimi vardır. *NameValueCCSID* bu karakter kümelerinden biriye, *NameValueData* , MQRFH2 içindeki diğer alanlarla aynı kodlamada olmalıdır.

MQGET çağrısında MQGMO\_CONVERT belirtildiğinde, kuyruk yöneticisi *NameValueData* ' yi istenen kodlamaya dönüştürür, ancak karakter kümesini değiştirmez.

## Alanlar

**Not:** Aşağıdaki tabloda, alanlar alfabetik olarak değil, kullanıma göre gruplanır. Alt konular aynı sırayı izler.

Alan adı	Değişmezin adı	Değişmezin değeri
<u>StrucId</u> (yapı tanıtıcısı)	MQRFH_STRUC_ID	'RFH→'
<u>Sürüm</u> (yapı sürümü numarası)	MQRFH_VERSION_2	2
<u>StrucLength</u> ( MQRFH2 yapısının bayt cinsinden uzunluğu)	MQRFH_STRUC_LENGTH_FIXED_2	36
<u>Kodlama</u> (son <i>NameValueData</i> alanını izleyen verilerin sayısal kodlaması)	MQENC_NATIVE	Ortama bağlıdır
<u>CodedCharSetId</u> (son <i>NameValueData</i> alanını izleyen verilerin karakter kümesi tanıtıcısı)	MQCCSI_INHERIT	-2
<u>Biçim</u> (son <i>NameValueData</i> alanını izleyen verilerin biçim adı)	MQFMT_NONE	Boşluklar
<u>İşaretler</u> (işaretler)	MQRFH_NONE	0
<u>NameValueCCSID</u> ( <i>NameValueData</i> alanındaki verilerin kodlanmış karakter takımı tanıtıcısı)	Yok	1208

Çizelge 515. MQRFH2 için MQRFH2 içindeki alanlar (devamı var)

Alan adı	Değişmezin adı	Değişmezin değeri
NameValueLength (NameValueData alanındaki verilerin bayt cinsinden uzunluğu)	Yok	None
NameValueVerileri (ileti özelliklerinin ad-değer çiftleri)	Yok	Yok

**Notlar:**

- ↪ simgesi, tek bir boş karakteri temsil eder.
- C programlama dilinde, makro değişkeniMQRFH2\_DEFAULT çizelgede listelenen değerleri içerir. Yapıdaki alanlar için başlangıç değerleri sağlamak üzere bunu aşağıdaki şekilde kullanın:

```
MQRFH2 MyRFH2 = {MQRFH2_DEFAULT};
```

## Dil bildirimleri

### MQRFH2 için C bildirimini

```
typedef struct tagMQRFH2 MQRFH2;
struct tagMQRFH2 {
    MQCHAR4  StrucId;          /* Structure identifier */
    MQLONG   Version;        /* Structure version number */
    MQLONG   StrucLength;    /* Total length of MQRFH2 including all
                             NameValueLength and NameValueData
                             fields */
    MQLONG   Encoding;      /* Numeric encoding of data that follows
                             last NameValueData field */
    MQLONG   CodedCharSetId; /* Character set identifier of data that
                             follows last NameValueData field */
    MQCHAR8  Format;        /* Format name of data that follows last
                             NameValueData field */
    MQLONG   Flags;        /* Flags */
    MQLONG   NameValueCCSID; /* Character set identifier of
                             NameValueData */
};
```

### MQRFH2 için COBOL bildirimini

```
** MQRFH2 structure
10 MQRFH2.
** Structure identifier
15 MQRFH2-STRUCID PIC X(4).
** Structure version number
15 MQRFH2-VERSION PIC S9(9) BINARY.
** Total length of MQRFH2 including all NAMEVALUELENGTH and
NAMEVALUEDATA fields
15 MQRFH2-STRUCLength PIC S9(9) BINARY.
** Numeric encoding of data that follows last NAMEVALUEDATA field
15 MQRFH2-ENCODING PIC S9(9) BINARY.
** Character set identifier of data that follows last NAMEVALUEDATA
field
15 MQRFH2-CODEDCHARSETID PIC S9(9) BINARY.
** Format name of data that follows last NAMEVALUEDATA field
15 MQRFH2-FORMAT PIC X(8).
** Flags
15 MQRFH2-FLAGS PIC S9(9) BINARY.
** Character set identifier of NAMEVALUEDATA
15 MQRFH2-NAMEVALUECCSID PIC S9(9) BINARY.
```

## MQRFH2 için PL/I bildirimi

```
dc1
1 MQRFH2 based,
3 StrucId      char(4),      /* Structure identifier */
3 Version      fixed bin(31), /* Structure version number */
3 StrucLength  fixed bin(31), /* Total length of MQRFH2 including
                               all NameValueLength and
                               NameValueData fields */
3 Encoding     fixed bin(31), /* Numeric encoding of data that
                               follows last NameValueData field */
3 CodedCharSetId fixed bin(31), /* Character set identifier of data
                               that follows last NameValueData
                               field */
3 Format        char(8),      /* Format name of data that follows
                               last NameValueData field */
3 Flags        fixed bin(31), /* Flags */
3 NameValueCCSID fixed bin(31); /* Character set identifier of
                               NameValueData */
```

## MQRFH2 için High Level Assembler bildirimi

```
MQRFH          DSECT
MQRFH_STRUCID  DS   CL4   Structure identifier
MQRFH_VERSION  DS   F     Structure version number
MQRFH_STRUCLNGTH DS   F     Total length of MQRFH2 including all
*              NAMEVALUELENGTH and NAMEVALUEDATA fields
MQRFH_ENCODING DS   F     Numeric encoding of data that follows
*              last NAMEVALUEDATA field
MQRFH_CODEDCHARSETID DS   F     Character set identifier of data that
*              follows last NAMEVALUEDATA field
MQRFH_FORMAT   DS   CL8   Format name of data that follows last
*              NAMEVALUEDATA field
MQRFH_FLAGS    DS   F     Flags
MQRFH_NAMEVALUECCSID DS   F     Character set identifier of
*              NAMEVALUEDATA
*
MQRFH_LENGTH   EQU   *-MQRFH
               ORG   MQRFH
MQRFH_AREA     DS   CL(MQRFH_LENGTH)
```

## MQRFH2 için Visual Basic bildirimi

```
Type MQRFH2
  StrucId      As String*4 'Structure identifier'
  Version      As Long     'Structure version number'
  StrucLength  As Long     'Total length of MQRFH2 including all'
                               'NameValueLength and NameValueData fields'
  Encoding     As Long     'Numeric encoding of data that follows'
                               'last NameValueData field'
  CodedCharSetId As Long   'Character set identifier of data that'
                               'follows last NameValueData field'
  Format        As String*8 'Format name of data that follows last'
                               'NameValueData field'
  Flags        As Long     'Flags'
  NameValueCCSID As Long   'Character set identifier of NameValueData'
End Type
```

### **StrucId (MQCHAR4)**

Bu, yapı tanıtıcısıdır; değer şöyle olmalıdır:

#### **MQRFH\_STRUC\_ID**

Kurallar ve biçimlendirme üstbilgisi yapısına ilişkin tanıtıcı.

C programlama dili için, MQRFH\_STRUC\_ID\_ARRAY değişmezi de tanımlanır; bu, MQRFH\_STRUC\_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisidir.

Bu alanın ilk değeri MQRFH\_STRUC\_ID 'dir.

### **Sürüm (MQUZE)**

Bu, yapı sürüm numarasıdır; değer şu olmalıdır:

### **MQRFH\_VERSION\_2**

Version-2 kuralları ve biçimlendirme üstbilgisi yapısı.

Bu alanın ilk değeri MQRFH\_VERSION\_2' dir.

### **StrucLength (MQUZE)**

Bu uzunluk, yapının sonundaki *NameValueLength* ve *NameValueData* alanları da içinde olmak üzere, MQRFH2 yapısının bayt cinsinden uzunluğudur. Bu, yapının sonunda birden çok sayıda *NameValueLength* ve *NameValueData* alanı çifti olmak üzere sırayla geçerlidir:

```
length1, data1, length2, data2, ...
```

*StrucLength* , yapının sonunda son *NameValueData* alanını izleyebilecek herhangi bir kullanıcı verisi içermez.

Bazı ortamlarda kullanıcı verilerini dönüştürmeye ilişkin sorunları önlemek için *StrucLength* , dörtten çok bir değer olmalıdır.

Aşağıdaki değişmez, yapının *sabit* bir kısmının uzunluğunu, yani *NameValueLength* ve *NameValueData* alanları hariç olmak üzere, uzunluğun uzunluğunu verir:

### **MQRFH\_STRUC\_LENGTH\_FIXED\_2**

MQRFH2 yapısının sabit parçası uzunluğu.

Bu alanın ilk değeri MQRFH\_STRUC\_LENGTH\_FIXED\_2' dir.

### **Kodlama (MQUZE)**

Bu, son *NameValueData* alanını izleyen verilerin sayısal kodlamasını belirtir; MQRFH2 yapısının kendisindeki sayısal veriler için geçerli değildir.

MQPUT ya da MQPUT1 çağrısında, uygulama bu alanı veri için uygun değere ayarlamalıdır.

Bu alanın ilk değeri MQENC\_NATIVE 'dir.

### **CodedCharSetId (MQHOMER)**

Bu, son *NameValueData* alanını izleyen verilerin karakter kümesi tanıtcısını belirtir; MQRFH2 yapısının kendisindeki karakter verileri için geçerli değildir.

MQPUT ya da MQPUT1 çağrısında, uygulama bu alanı veri için uygun değere ayarlamalıdır. Aşağıdaki özel değer kullanılabilir:

### **MQCCSI\_INHERIT**

Character data in the data *takip edilen* this structure is in the same character set as this structure.

Kuyruk yöneticisi, bu değeri, iletide belirtilen yapının gerçek karakter kümesi tanıtcısına göre değiştirir. Herhangi bir hata oluşmaması koşuluyla, MQCCSI\_INHERIT değeri MQGET çağrısıyla döndürülmez.

MQMD ' deki *PutApplType* alanının değeri MQAT\_BROKER ise, MQCCSI\_INHERIT kullanılamaz.

Bu alanın ilk değeri, MQCCSI\_INHERIT olarak kabul edilir.

### **Biçim (MQCHAR8)**

Bu, son *NameValueData* alanını izleyen verilerin biçim adını belirtir.

MQPUT ya da MQPUT1 çağrısında, uygulama bu alanı veri için uygun değere ayarlamalıdır. Bu alanı kodlamaya ilişkin kurallar, MQMD ' deki *Format* alanı için yapılan kurallarla aynıdır.

Bu alanın ilk değeri MQFMT\_NONE olur.

### **İşaretler (MQUZE)**



Bu alanın başlangıç değeri MQRFH\_NONE' dir. MQRFH\_NONE belirtilmeli.

### **MQRFH\_NONE**

Bayrak yok.

### **MQRFH\_INTERNAL**

MQRFH2 üstbilgisi, dahili olarak ayarlanmış özellikleri içerir.

MQRFH\_INTERNAL is for queue manager use.

En üstteki 16 bit ( MQRFH\_FLAGS\_RESTRICTED\_MASK), kuyruk yöneticisi kümelerini işaretlere ayrılmıştır. Bir kullanıcının ayarlayabileceği işaretler, alt 16 bit olarak tanımlanır.

### **NameValueCCSID (MQUZE)**

Bu, *NameValueData* alanındaki verilerin kodlanmış karakter takımı tanıtıcısını belirtir. Bu, MQRFH2 yapısındaki diğer dizelerin karakter takımından farklıdır ve yapının sonundaki son *NameValueData* alanını izleyen verilerden (varsa) farklı karakter takımından farklı olabilir.

*NameValueCCSID* aşağıdaki değerlerden birine sahip olmalıdır:

CCSID	Anlamı
1200	UTF-16, en son Unicode sürümü destekleniyor
13488	UTF-16, Unicode sürüm 2.0 altkümesi
17584	UTF-16, Unicode sürüm 3.0 altkümesi (Euro simgesini içerir)
1208	UTF-8, en son Unicode sürümü destekleniyor

UTF-16 karakter takımları için, *NameValueData* kodlamasının kodlaması (byte sırası), MQRFH2 yapısındaki diğer alanların kodlamasıyla aynı olmalıdır.

Characters beyond the Unicode Basic Multilingual Plane (those above U+FFFF), represented in UTF-16 by surrogate code points (X'D800' through X'DFFF'), or four bytes in UTF-8, are not supported.

**Not:** *NameValueCCSID* yukarıda listelenen değerlerden birine sahip değilse ve MQRFH2 yapısı MQGET çağrısına dönüştürülmesini gerektiriyorsa, çağrı neden kodu MQRC\_SOURCE\_CCSIDID\_ERROR ile tamamlanır ve ileti dönüştürülmez olarak döndürülür.

Bu alanın ilk değeri 1208 'dir.

### **NameValueUzunluğu (MQLONG)**

İlgili *NameValueData* alanının uzunluğu

Bu, *NameValueData* alanındaki verilerin bayt cinsinden uzunluğunu belirtir. *NameValueLength* , dörtten birden çok olmalıdır.

**Not:** *NameValueLength* ve *NameValueData* alanları isteğe bağlıdır, ancak varsa çift olarak oluşmalıdır ve bitişik olmalıdır. Alan çifti, gerektiği kadar yinelenebilir; örneğin:

```
length1 data1 length2 data2 length3 data3
```

Bu alanlar isteğe bağlı olduğu için, desteklenen çeşitli programlama dilleri için sağlanan yapının bildirimlerinden çıkarılır.

### **NameValueVerileri (MQCHARn)**

*NameValueVerileri* , ileti özelliklerinin ad-değer çiftlerini içeren bir klasörü içeren değişken uzunluklu bir alandır. Klasör, XML benzeri bir sözdizimi kullanılarak kodlanmış verileri içeren değişken uzunluklu bir karakter dizisidir. Karakter dizisinin bayt cinsinden uzunluğu, *NameValueVeri* alanından önce gelen *NameValueLength* alanı tarafından verilir. Uzunluk, dört çarpı olmalıdır.

*NameValueLength* ve *NameValueData* alanları isteğe bağlıdır, ancak varsa bunlar bir çift olarak oluşmalı ve bitişik olmalıdır. Alan çifti gerektiği kadar yinelenebilir, örneğin:

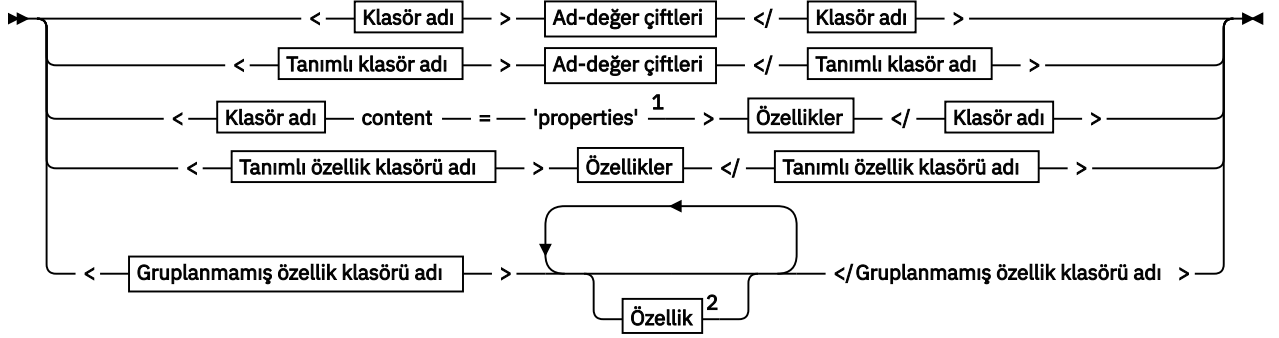
```
length1 data1 length2 data2 length3 data3
```

*NameValueVerileri* , MQGET çağrısında belirtilen karakter kümesine dönüştürülmez. İleti, MQGMO\_CONVERT seçeneğiyle *NameValueVerileri* geçerli olarak alınsa bile özgün karakter kümesinde kalır. Ancak *NameValueVerileri* , MQGET çağrısında belirtilen kodlamaya dönüştürülür.

#### Notlar:

- Bu alanlar isteğe bağlı olduğundan, desteklenen çeşitli programlama dilleri için sağlanan yapının bildirimlerinden çıkarılır.
- Sözdizimi şemasında "tanımlı" ve "ayrılmış" terimleri kullanılır. "Tanımlı" , adın IBM MQ tarafından kullanıldığı anlamına gelir. "Ayrılmış" , adın ileride IBM MQ tarafından kullanılmak üzere ayrıldığı anlamına gelir.

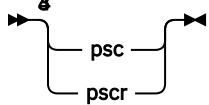
#### *NameValueData* sözdizimi



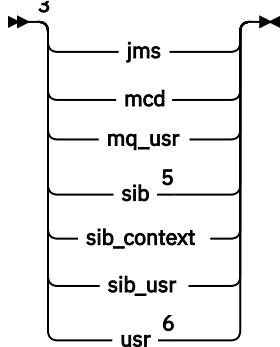
#### Klasör adı



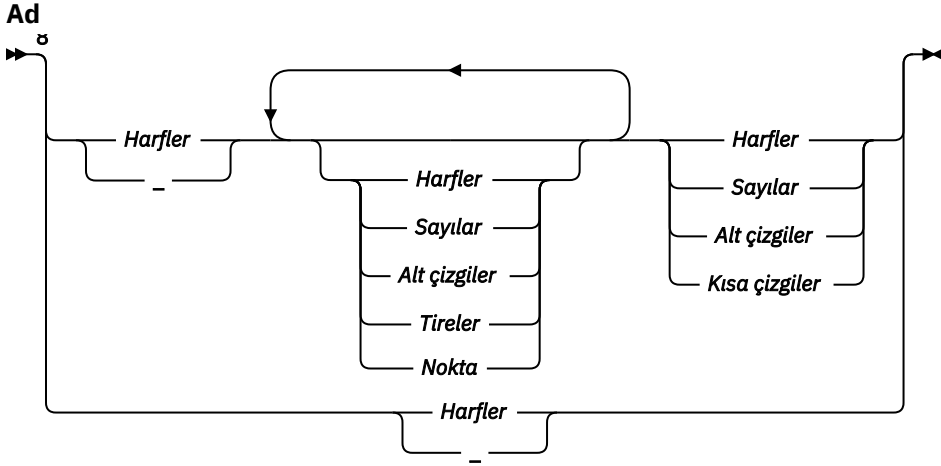
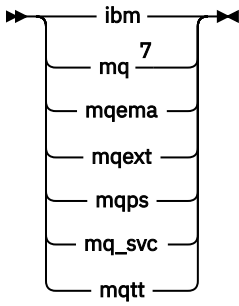
#### Tanımlı klasör adı



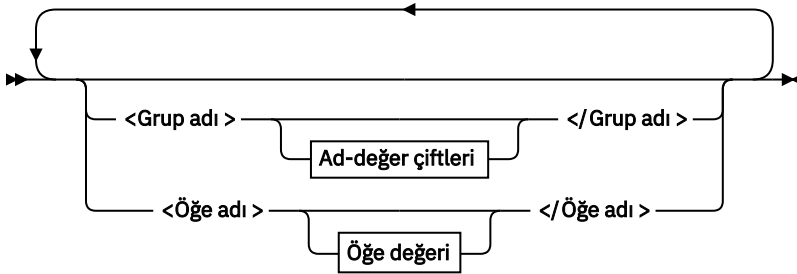
#### Tanımlı özellik klasörü adı



#### Gruplanmamış özellik klasörü adı



### Ad-değer çiftleri



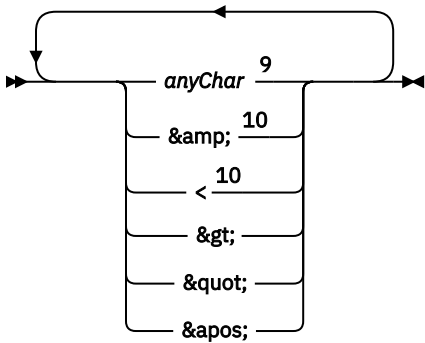
### Grup adı



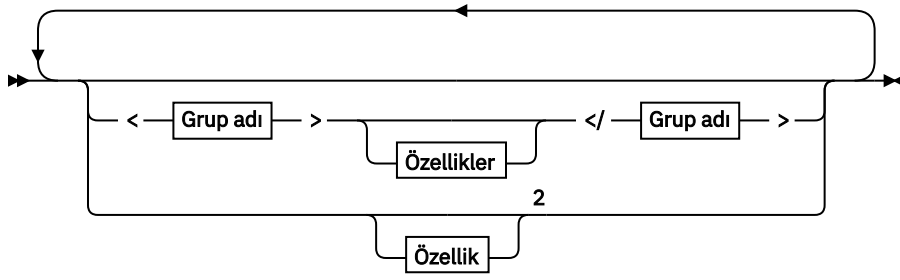
### Öğe adı



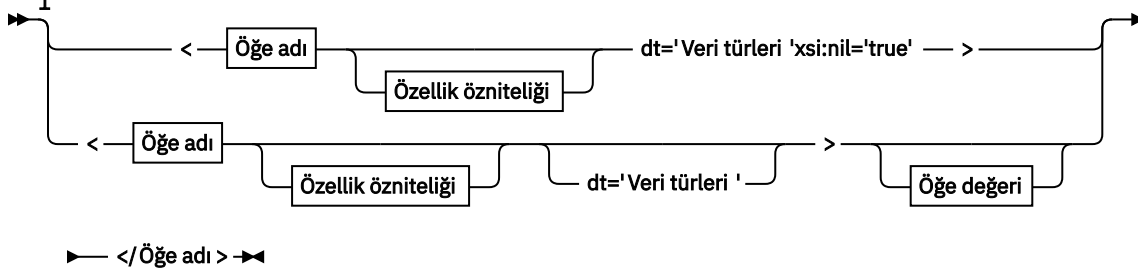
### Öğe değeri



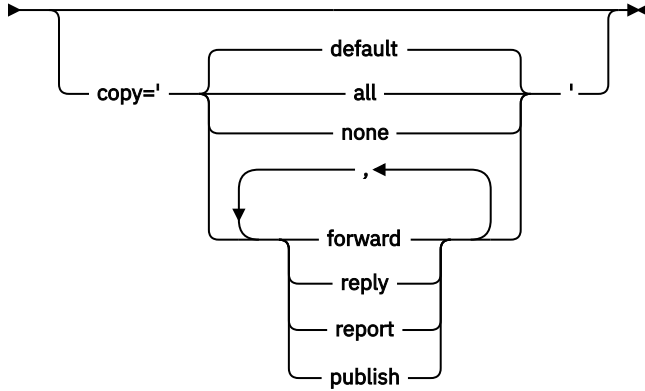
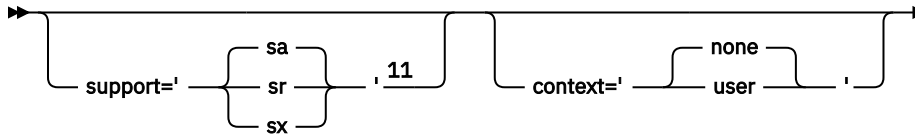
### Özellikler



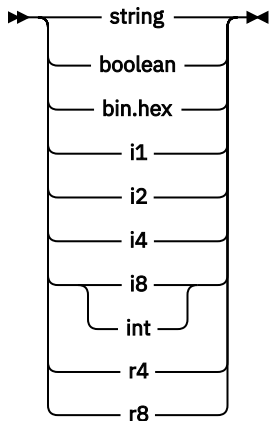
### Özellik



### Özellik özniteliği



### Veri türleri



Notlar:

<sup>1</sup> Çift tırnak işaretleri ya da tek tırnak işaretleri geçerlidir.

<sup>2</sup> Geçersiz bir özellik adı kullanmayın; bkz. “Özellik adı geçersiz” sayfa 539. Ayrılmış bir özellik adını yalnızca tanımlı amacı için kullanın; bkz. “Tanımlı özellik adları” sayfa 539.

<sup>3</sup> Ad küçük harfli olmalıdır.

<sup>4</sup> Yalnızca bir psc ve pscı klasörü desteklenir.

<sup>5</sup> WebSphere Application Server Service Integration Bus , sonraki MQRFH2 üstbilgilerinde sib, sib\_contextve sib\_usr klasörlerini yoksayar ve yalnızca ilk MQRFH2 üstbilgisindeki özellikler önemlidir.

<sup>6</sup> Bir MQRFH2 içinde birden çok usı klasörü bulunmamalıdır. usı klasöründeki özellikler bir kereden fazla oluşmamalıdır.

<sup>7</sup> Yalnızca ilk mq klasöründeki özellikler önemlidir. Klasör UTF-8 ise, yalnızca tek bayt UTF-8 karakterleri desteklenir. Tek beyaz alan karakteri Unicode U+0020' dur.

<sup>8</sup> Geçerli karakterler W3C XML belirtiminde tanımlanır ve temelde Unicode kategorilerinden oluşur L1, Lu, Lo, Lt, Nl, Mc, Mn, Lm, ve Nd. Bkz. [Unicode karakter kategorileri](#).

<sup>9</sup> Tüm karakterler anlamlıdır. Baştaki ve sondaki boşluklar öge değerinin bir parçasıdır.

<sup>10</sup> Geçersiz bir karakter kullanmayın; bkz. “Geçersiz karakterler” sayfa 539. Bu geçersiz karakterler yerine bir çıkış sırası kullanın.

<sup>11</sup> Destek özelliği özneliği yalnızca mq klasöründe geçerlidir.

## Klasör adı

*NameValueVerileri* tek bir klasör içerir. Birden çok klasör oluşturmak için birden çok *NameValueVeri* alanı oluşturun. Bir ileti içindeki tek bir MQRFH2 üstbilgisinde birden çok *NameValueData* alanı oluşturabilirsiniz. Alternatif olarak, her biri birden çok *NameValueVeri* alanı içeren birden çok zincirli MQRFH2 üstbilgisi oluşturabilirsiniz.

MQRFH2 üstbilgilerinin sırası ve *NameValueVeri* alanlarının sırası, bir klasörün mantıksal içeriği için fark yaratmaz. Aynı klasör bir iletide birden çok kez bulunursa, klasör bir bütün olarak ayrıştırılır. Aynı özellik aynı klasörün birden çok örneğindeyse, bir liste olarak ayrıştırılır.

MQRFH2 ögesinin doğru ayrıştırması, bir klasörün iletide fiziksel olarak saklanabileceği diğer yöntemlerden etkilenmez.

Dört klasör bu kuralı izlemiyor. Yalnızca mq, sib, sib\_contextve sib\_usr klasörünün ilk örneği ayrıştırılır.

Aynı özellik zincirleme MQRFH2 üstbilgilerinin birleşik içeriğinde bir kereden fazla ortaya çıkarsa, özelliğin yalnızca ilk eşgörünümü ayrıştırılır. MQSETMP gibi bir API çağrısı kullanılarak bir özellik ayarlanırsa ve doğrudan bir uygulama tarafından MQRFH2 ' a eklenirse, API çağrısı önceliklidir.

Klasör adı, ad-değer çiftlerini ya da gruplarını içeren bir klasörün adıdır. Gruplar ve ad-değer çiftleri, klasör ağacında aynı düzeyde karıştırılabilir. Bkz. [Şekil 1 sayfa 529](#). Bir grup adı ile öge adını birleştirmeyin; bkz. [Şekil 2 sayfa 529](#).

```
<group1><nvp1>value</nvp1></group1><group2><nvp2>value</nvp2></group2>
<group3><nvp1>value</nvp1></group3><nvp3>value</nvp3>
```

Şekil 1. Grupların ve ad-değer çiftlerinin doğru kullanımları

```
<group1><nvp1>value</nvp1>value</group1>
```

Şekil 2. Grupların ve ad-değer çiftlerinin yanlış kullanımı

Geçersiz ya da ayrılmış bir klasör adı kullanmayın; bkz. “Yol adı geçersiz” sayfa 539 ve “Ayrılmış klasör ya da özellik klasörü adı” sayfa 538. Tanımlı bir klasör adını yalnızca tanımlı amacı için kullanın; bkz. “Tanımlı klasör adı” sayfa 530.

Klasör adı etiketine ' content=properties ' özniteliğini eklerseniz, klasör bir özellik klasörü olur; bkz. [Şekil 3 sayfa 530](#).

```
<myFolder></myfolder>  
<myPropertyFolder contents=' properties '></myPropertyFolder>
```

*Şekil 3. Bir klasör ve özellik klasörü örneği*

Klasör adları büyük ve küçük harfe duyarlıdır. Klasör adları ve özellik klasörü adları aynı ad alanını paylaşır. Farklı isimleri olmalı. Örneğin, [Şekil 4 sayfa 530](#) içindeki Folder1 , [Şekil 5 sayfa 530](#) içindeki Folder2 ile farklı bir ad olmalıdır.

```
<Folder1><NVP1>value</NVP1></Folder1>
```

*Şekil 4. Folder1 ad alanı*

```
<Folder2 content=' properties '><Property1>value</Property1></Folder2>
```

*Şekil 5. Folder2 ad alanı*

Farklı klasörlerdeki gruplar, özellikler ve ad-değer çiftleri farklı ad alanlarına sahiptir. [Şekil 5 sayfa 530](#) içindeki Property1 , [Şekil 6 sayfa 530](#) içindeki Property1 ile farklı bir özelliktir.

```
<Folder3 content=' properties '><Property1>value</Property1></Folder3>
```

*Şekil 6. Folder3 ad alanı*

Özellik klasörleri, iki önemli açıdan özellik dışı klasörlerden farklıdır:

1. Özellik klasörleri özellikleri içerir ve özellik olmayan klasörler ad-değer çiftlerini içerir. Klasörler, sözdizimsel olarak biraz farklıdır.
2. İleti özelliklerine erişmek için MQI ya da JMS ileti özellikleri gibi tanımlı arabirimleri kullanın. Arabirimler, MQRFH2 içindeki özellik klasörlerinin doğru biçimlendirildiğinden emin olur. Doğru biçimlendirilmiş bir özellik klasörü, farklı platformlardaki kuyruk yöneticileri ile farklı yayınlarda birlikte çalışabilir.

İleti özelliği MQI, MQRFH2'ı okumak ve yazmak için sağlam bir yoldur ve bir MQRFH2 ' nin doğru olarak ayrıştırılmasının zorluklarını önler.

## Tanımlı klasör adı

Tanımlı klasör adı, IBM MQ'ya da başka bir ürün tarafından kullanılmak üzere ayrılmış bir klasörün adıdır. Aynı ada sahip bir klasör oluşturmayın ve klasörlere kendi ad-değer çiftlerinizi eklemeyin. Tanımlı klasörler: psc ve psc1.

psc ve psc1 , kuyruğa alınan yayınlama/abone olma tarafından kullanılır.

MQMF\_SEGMENT ya da MQMF\_SEGMENTATION\_ALLOWED ile yerleştirilen bölümlenmiş bir ileti, tanımlı klasör adına sahip bir MQRFH2 içeremez. MQPUT , 2443, MQRC\_SEGMENTATION\_NOT\_ALLOWEDneden koduyla başarısız olur.

## Tanımlı özellik klasörü adı

Tanımlı özellik klasörü adı, IBM MQya da başka bir ürün tarafından kullanılan bir özellik klasörünün adıdır. Klasörlerin ve içeriklerinin adları için bkz. [Özellik klasörleri](#). Tanımlı özellik klasörü adları, IBM MQtarafından ayrılmış tüm klasör adlarının bir altkümesidir. Bkz. [“Ayrılmış klasör ya da özellik klasörü adı”](#) sayfa 538.

Tanımlı bir özellik klasöründe saklanan her öge bir özelliktir. Tanımlı bir özellik klasöründe saklanan bir ögenin content='properties' özniteliği olmamalıdır.

Özellikleri yalnızca tanımlı özellik klasörlerine us1, mq\_us1ve sib\_us1ekleyebilirsiniz. mq ve sibgibi diğer özellik klasörlerinde IBM MQ , tanımadığı özellikleri yoksayar ya da atar.

Tanımlanan her özellik klasörünün tanımı, uygulama programları tarafından kullanılacak IBM MQ ' in tanımladığı özellikleri listeler. Bazı özelliklere bir JMS özelliği ayarlanarak ya da alınırken dolaylı olarak erişilir ve bazılarında MQSETMP ve MQINQMP MQI çağruları kullanılarak doğrudan erişilir.

Tanımlanan özellik klasörleri, IBM MQ ' in ayırdığı, ancak hangi uygulamalara erişemeyen diğer özellikleri de içerir. Ayrılmış özelliklerin adları listelenmez. us1, mq\_us1ve sib\_us1 özellik klasörlerinde ayrılmış bir özellik yoktur. Ancak, özellik adları geçersiz olan özellikler oluşturmayın; bkz. [“Özellik adı geçersiz”](#) sayfa 539.

## Özellik klasörleri

### jms

jms , MQMDiçinde tam olarak ifade edilemeyen JMS üstbilgi alanlarını ve JMSX özelliklerini içerir. jms klasörü her zaman bir JMS MQRFH2içinde bulunur.

Çizelge 516. JMS özellik adı, eşanlamlı, veri tipi ve klasör			
Özellik eşanlamlısı	Özellik adı	Veri tipi	Klasör
JMSDestination (JMS Hedefi)	jms.Dst	string	<jms><Dst>destination</Dst></jms>
JMSExpiration	jms.Exp	i8	<jms><Exp>expiration</Exp></jms>
JMSCorrelation	jms.Cid	string	<jms><Cid>correlationId</Cid></jms>
JMSDelivery	jms.Dlv	i4	<jms><Dlv>delivery</Dlv></jms>
JMSPriority (JMSPriority)	jms.Pri	i4	<jms><Pri>priority</Pri></jms>
JMSReplyTo	jms.Rto	string	<jms><Rto>replyToURI</Rto></jms>
JMSTimestamp	jms.Tms	i8	<jms><Tms>timestamp</Tms></jms>
JMSXGroup ID	jms.Gid	string	<jms><Gid>groupId</Gid></jms>
JMSXGroup Seq	jms.Seq	i4	<jms><Seq>messageSequenceNo</Seq></jms>

jms klasörüne kendi özelliklerinizi eklemeyin.

## **mcd**

mcd , iletinin biçimini açıklayan özellikleri içerir. Örneğin, ileti hizmeti etki alanı Msd özelliği bir JMS iletisini JMSTextMessage, JMSByteMessage, JMSStreamMessage, JMSMapMessage, JMSObjectMessageya da boş değer olarak tanımlar.

mcd klasörü, MQRFH2içeren bir JMS iletisinde her zaman bulunur.

Bu ileti, her bana yılında IBM Integration Bus' dan gönderilen bir MQRFH2 iletisini içeren iletide bulunur. Bir iletinin etki alanını, biçimini, tipini ve ileti kümesini açıklar.

<i>Çizelge 517. mcd özellik adı, eşanlamlı, veri tipi ve klasör</i>			
<b>Özellik eşanlamlısı</b>	<b>Özellik adı</b>	<b>Veri tipi</b>	<b>Klasör</b>
	mcd.Msd	string	<mcd><Msd>messageDomain</Msd></mcd>
	mcd.Set	string	<mcd><Set>messageDomain</Set></mcd>
	mcd.Type	string	<mcd><Type>messageDomain</Type></mcd>
	mcd.Fmt	string	<mcd><Fmt>messageDomain</Fmt></mcd>

mcd klasörüne kendi özelliklerinizi eklemeyin.

## **mq\_usr**

mq\_usr , JMS kullanıcı tanımlı özellikleri olarak gösterilmeyen uygulama tanımlı özellikleri içerir. JMS gereksinimlerini karşılamayan özellikler bu klasöre yerleştirilebilir.

mq\_usr klasöründe özellikler oluşturabilirsiniz. mq\_usr içinde oluşturduğunuz özellikler, content='properties' özniteliğiyle yeni klasörlerde oluşturduğunuz özellikler gibidir.

## **sib**

sib , WebSphere Application Server hizmet bütünleştirme veriyolu (WAS/SIB) sistem iletisi özelliklerini içerir. sib özellikleri, desteklenen tiplerde olmadığından IBM MQ JMS uygulamalarına JMS özellikleri olarak gösterilmez. Örneğin, bazı sib özellikleri bayt dizileri olduğundan JMS özellikleri olarak gösterilemez. Bazı sib özellikleri WAS/SIB uygulamalarına JMS\_IBM\_\* özellikleri olarak gösterilir; bunlar ileriye doğru ve ters yöneltme yolları özelliklerini içerir.

sib klasörüne kendi özelliklerinizi eklemeyin.

## **sib\_context**

sib\_context , WAS/SIB kullanıcı uygulamalarına ya da JMS özellikleri olarak gösterilmeyen WAS/SIB sistem iletisi özelliklerini içerir. sib\_context , web hizmetleri için kullanılan güvenlik ve hareket özelliklerini içerir.

sib\_context klasörüne kendi özelliklerinizi eklemeyin.

## **sib\_usr**

sib\_usr , desteklenen tipte olmadığından JMS kullanıcı özellikleri olarak gösterilmeyen WAS/SIB kullanıcı iletisi özelliklerini içerir. sib\_usr , SIMessage arabiriminde WAS/SIB uygulamalarına gösterilir. Bkz. [Developing Service Integration](#).

sib\_usr özelliğinin tipi bin . hexolmalı ve değer doğru biçimde olmalıdır. Bir IBM MQ uygulaması bin . hex tipinde bir öğeyi klasöre yanlış biçimde yazarsa, uygulama bir IOExceptionalır. Özelliğin veri tipi bin . hex değilse, uygulama bir ClassCastExceptionalır.



Bu klasörü kullanarak JMS kullanıcı özelliklerini WAS/SIB için kullanılabilir kılmayı denemeyin; bunun yerine usr klasörünü kullanın.

sib\_usr klasöründe özellikler oluşturabilirsiniz.

## usr

usr , iletiyle ilişkili uygulama tanımlı JMS özelliklerini içerir. usr klasörü, yalnızca bir uygulama uygulama tanımlı bir özellik ayarladıysa bulunur.

usr varsayılan özellik klasörüdür. Bir özellik klasör adı olmadan ayarlanırsa, usr klasörüne yerleştirilir.

Çizelge 518. usr özellik adı, eşanlamlı, veri tipi ve klasör			
Özellik eşanlamlısı	Özellik adı	Veri tipi	Klasör
	usr.contentType	string	<usr><contentType>text/xml;charset=utf-8/</contentType></usr>
	usr.endpointURL	string	<usr><endpointURL>URI</endpointURL></usr>
	usr.targetService	string	<usr><targetService>serviceName</targetService></usr>
	usr.soapAction	string	<usr><soapAction>name</soapAction></usr>
	usr.transportVersion	string	<usr><transportVersion>version</transportVersion></usr>

usr klasöründe özellikler oluşturabilirsiniz.

MQMF\_SEGMENT ya da MQMF\_SEGMENTATION\_ALLOWED içeren bölümlenmiş bir ileti, tanımlı bir özellik klasörü adına sahip bir MQRFH2 içeremez. MQPUT , 2443, MQRC\_SEGMENTATION\_NOT\_ALLOWEDneden koduyla başarısız olur.

## Gruplanmamış özellik klasörü adı

### ibm

ibm , yalnızca IBM MQtarafından kullanılan özellikleri içerir.

Çizelge 519. ibm özellik adı, eşanlamlı, veri tipi ve klasör			
Özellik eşanlamlısı	Özellik adı	Veri tipi	Klasör
	ibm.rfp	string	<ibm><rfp>fingerprint</rfp></ibm>

ibm klasörüne kendi özelliklerinizi eklemeyin.

### mq

mq , yalnızca IBM MQtarafından kullanılan özellikleri içerir.

mq klasöründeki özellikler için aşağıdaki kısıtlamalar geçerlidir:

- Yalnızca iletideki ilk önemli mq klasöründeki özellikler MQtarafından işlendi; iletideki başka bir mq klasöründeki özellikler yoksayılr.
- Klasörde yalnızca tek baytlık UTF-8 karakterlerine izin verilir. Klasördeki çok baytlı bir karakter, ayrıştırmanın başarısız olmasına ve iletinin reddedilmesine neden olabilir.

- Klasörde çıkış dizgilerini kullanmayın. Bir çıkış dizgisi, ögenin gerçek değeri olarak kabul edilir.
- Klasör içinde yalnızca Unicode karakteri U+0020 beyaz alan olarak kabul edilir. Diğer tüm karakterler önemli olarak değerlendirilir ve klasör ayrıştırmasının başarısız olmasına ve iletinin reddedilmesine neden olabilir.

mq klasörünün ayrıştırılması başarısız olursa ya da klasör bu kısıtlamalara uymazsa, ileti 2527, MQRC\_RFH\_RESTRICTED\_FORMAT\_ERRneden koduyla reddedilir.

mq klasörüne kendi özelliklerinizi eklemeyin.

### mqema

mqema , yalnızca WebSphere Application Servertarafından kullanılan özellikleri içerir. Klasör mqextile değiştirildi.

mqema klasörüne kendi özelliklerinizi eklemeyin.

### mqext

mqext aşağıdaki özellik tiplerini içerir:

- Yalnızca WebSphere Application Servertarafından kullanılan özellikler.
- İletilerin gecikmeli teslimiyle ilgili özellikler.

Uygulama, IBM tarafından tanımlanan özelliklerden en az birini ya da kullanılan teslim gecikmesini ayarladiysa klasör vardır.

Çizelge 520. mqext özellik adı, eşanlamlı, veri tipi ve klasör			
Özellik eşanlamlısı	Özellik adı	Veri tipi	Klasör
JMSArmCorrelator	mqext.Arm	string	<mqext><Arm>armCorrelator</Arm></mqext>
JMSRMCorrelator	mqext.Wrm	string	<mqext><Wrm>wrmCorrelator</Wrm></mqext>
JMSDeliveryTime	mqext.Dlt	i8	<mqext><Dlt>DeliveryTime</Dlt></mqext>
JMSDeliveryDelay	mqext.Dly	i8	<mqext><Dly>DeliveryTime</Dly></mqext>

mqext klasörüne kendi özelliklerinizi eklemeyin.

### mqps

mqps , yalnızca IBM MQ yayınlama/abone olma tarafından kullanılan özellikleri içerir. Klasör, yalnızca uygulama, tümleşik yayınlama/abone olma özelliklerinden en az birini ayarladiysa bulunur.

Çizelge 521. mqps özellik adı, eşanlamlı, veri tipi ve klasör			
Özellik eşanlamlısı	Özellik adı	Veri tipi	Klasör
MQTopicString	mqps.Top	string	<mqps><Top>topicString</Top></mqps>
MQSubUserVerileri	mqps.Sud	string	<mqps><Sud>subscriberUserData...</Sud></mqps>
MQIsRetained	mqps.Ret	boolean	<mqps><Ret>isRetained</Ret></mqps>
MQPubOptions	mqps.Pub	i8	<mqps><Pub>publicationOptions</Pub></mqps>
MQPubLevel	mqps.Pbl	i8	<mqps><Pbl>publicationLevel</Pbl></mqps>

Çizelge 521. mqps özellik adı, eşanlamlı, veri tipi ve klasör (devamı var)			
Özellik eşanlamlısı	Özellik adı	Veri tipi	Klasör
MQPubTime	mqpse.Pts	string	<mqps><Pts>publicationTime</Pts></mqps>
MQPubSeqNum	mqpse.Seq	i8	<mqps><Seq>publicationSequenceNumber</Seq></mqps>
MQPubStrInData	mqpse.Sid	string	<mqps><Sid>publicationData</Sid></mqps>
MQPubFormat	mqpse.Pfmt	i8	<mqps><Pfmt>messageFormat</Pfmt></mqps>

mqps klasörüne kendi özelliklerinizi eklemeyin.

### mq\_svc

mq\_svc , SupportPac MA93tarafından kullanılan özellikleri içerir.

mq\_svc klasörüne kendi özelliklerinizi eklemeyin.

### mqt

mqt , MQ Telemetry tarafından kullanılan özellikleri içerir

Çizelge 522. mqt özellik adı, eşanlamlı, veri tipi ve klasör			
Özellik eşanlamlısı	Özellik adı	Veri tipi	Klasör
	mqt.clientId	string	<mqt><clientId>topicString</clientId></mqt>
	mqt.qos	i4	<mqt><qos>qualityOfService</qos></mqt>
	mqt.msgid	string	<mqt><msgid>messageIdentifier</msgid></mqt>

mqt klasörüne kendi özelliklerinizi eklemeyin.

MQMF\_SEGMENT ya da MQMF\_SEGMENTATION\_ALLOWED ile yerleştirilen bölümlenmiş bir ileti, gruplanmamış özellik klasörü adına sahip bir MQRFH2 içeremez. MQPUT , 2443, MQRC\_SEGMENTATION\_NOT\_ALLOWEDneden koduyla başarısız olur.

## Ad-değer çiftleri

Sözdizimi şemasında, "Ad-değer çiftleri" olağan bir klasörün içeriğini açıklar. Sıradan bir klasör grupları ve öğeleri içerir. Öğeler bir ad-değer çiftidir. Bir grup öğeleri ve diğer grupları içerir.

Ağaçlar açısından, öğeler yaprak düğümlerdir ve gruplar iç düğümlerdir. Bir iç düğüm ve kök düğüm olan klasör, iç düğümler ve yaprak düğümlerin bir karışımını içerebilir. Bir düğüm aynı anda hem bir iç düğüm hem de bir yaprak düğüm olamaz; bkz. [Şekil 2 sayfa 529](#).

## Özellikler

Sözdizimi şemasında "Özellikler" seçeneği, bir özellik klasörünün içeriğini açıklar. Bir özellik klasörü grupları ve özellikleri içerir. Özellik, isteğe bağlı veri tipi özniteliği olan bir ad-değer çiftidir. Bir grup özellikleri ve diğer grupları içerir.

Ağaçlar açısından, özellikler yaprak düğümlerdir ve gruplar iç düğümlerdir. Bir iç düğüm ve kök düğüm olan özellik klasörü, iç düğümler ve yaprak düğümlerin bir karışımını içerebilir. Bir düğüm aynı anda hem bir iç düğüm hem de bir yaprak düğüm olamaz; bkz. [Şekil 2 sayfa 529](#).

## Özellik

İleti özelliği, bir özellik klasöründeki ad-değer çiftidir. İsteğe bağlı olarak bir veri tipi özniteliği ve bir özellik özniteliği içerebilir; örneğin, aşağıdaki koda bakın. Veri tipi özniteliği atlanırsa, özellik tipi string olur.

```
<pf><p1 dt='i8'>value</p1></pf>
```

Bir ileti özelliğinin adı, XML benzeri <> sözdizimiyle tam yol adıdır ve nokta ile değiştirilir. Örneğin, myPropertyFolder1.myGroup1.myGroup2.myProperty1 aşağıdaki gibi bir *NameValueData* dizisiyle eşlenir. Dizgi, daha kolay okunmak için biçimlendirilir.

```
<myPropertyFolder1>
  <myGroup1>
    <myGroup2>
      <myProperty1>value</myProperty1>
    </myGroup2>
  </myGroup1>
</myPropertyFolder1>
```

Bir özellik klasörü birden çok özellik içerebilir. Örneğin, [Şekil 7 sayfa 536](#) içindeki özellikler, [Şekil 8 sayfa 536](#) içindeki özellik klasörüyle eşlenir.

```
myPropertyFolder1.myProperty4
myPropertyFolder1.myGroup1.myGroup2.myProperty1
myPropertyFolder1.myGroup1.myGroup2.myProperty2
myPropertyFolder1.myGroup1.myProperty3
```

*Şekil 7. Aynı kök ada sahip birden çok özellik*

```
<myPropertyFolder1>
  <myProperty4>value</myProperty4>
  <myGroup1>
    <myGroup2>
      <myProperty1>value</myProperty1>
      <myProperty2>value</myProperty2>
    </myGroup2>
    <myProperty3>value</myProperty3>
  </myGroup1>
</myPropertyFolder1>
```

*Şekil 8. Birden çok özellik adı eşlemesi*

## Ad

Bir ad *Harf* ya da *Altçizgi* ile başlamalıdır. *İki nokta* içermemelidir, *Nokta* ile bitmemelidir ve yalnızca *Harfler*, *Sayılar*, *Alt çizgiler*, *Kısa çizgiler* ve *Nokta* içermelidir. Geçerli karakterler W3C XML belirtiminde tanımlanır ve temelde Unicode kategorilerinden oluşur L1, Lu, Lo, Lt, Nl, Mc, Mn, Lm, ve Nd. Bkz. [Unicode karakter kategorileri](#).

Bir özelliğin ya da ad-değer çiftinin tam yolu, “[Yol adı geçersiz](#)” sayfa 539’ünde açıklanan kuralı bozmamalıdır. Yollar 4095 byte ile sınırlıdır, Unicode uyumluluk karakterleri içermemelidir ve XML dizisiyle başlamamalıdır.

## Grup adı

Bir grup adı, bir adla aynı sözdizimine sahip. Grup adları isteğe bağlıdır. Özellikler ve ad-değer çiftleri bir klasörün köküne yerleştirilebilir. Özellikleri ve ad-değer çiftlerini düzenlemeye yardımcı olacaksa grupları kullanın.

## Öğe adı

Bir öğe adı, adla aynı sözdizimine sahip.

## Öğe değeri

Bir öğe değeri, `<Element name>` etiketi ile `</Element name>` etiketi arasındaki tüm beyaz alanı içerir. Bir değerde `<` ve `&` karakterlerini kullanmayın. `<` ve `&` ile değiştirin.

## Özellik öznitelikleri

Özellik öznitelikleri, özellik tanımlayıcı alanlarını eşler. Eşlemeler aşağıdaki gibidir:

### Destek

#### **sa (varsayılan)**

MQPD\_SUPPORT\_OPTIONAL

#### **sr**

MQPD\_SUPPORT\_REQUIRED

#### **sx**

MQPD\_SUPPORT\_REQUIRED\_IF\_LOCAL

### Bağlam

#### **none (varsayılan)**

MQPD\_NO\_CONTEXT

#### **kullanıcı**

MQPD\_USER\_CONTEXT

### CopyOptions

#### **ileri**

MQPD\_COPY\_FORWARD

#### **Yanıt**

MQPD\_COPY\_REPLY

#### **rapor**

MQPD\_COPY\_REPORT

#### **yayınlamak**

MQPD\_COPY\_PUBLISH

#### **tümü**

MQPD\_COPY\_ALL

Diğer seçeneklerle birlikte tümünü kullanmayın.

#### **varsayılan**

MQPD\_COPY\_DEFAULT

Diğer seçeneklerle birlikte default kullanmayın. default , forward + report + publish ile aynıdır.

#### **yok**

MQPD\_COPY\_NONE

Diğer seçeneklerle birlikte none kullanmayın.

Destek özellik öznitelikleri yalnızca mq klasöründeki özellikler için geçerlidir.

Bağlam ve CopyOptions özellik öznitelikleri tüm özellik klasörleri için geçerlidir.

## Veri türleri

MQRFH2 veri tipleri, aşağıdaki gibi ileti özelliği tipleriyle eşlenir:

Çizelge 523. Veri tipi eşlemeleri

MQRFH2 veri tipi	İleti özelliği tipi
bin.hex	MQBYTE []
boolean	MQBOOL
i1	MQINT8
i2	MQINT16
i4	MQINT32
i8	MQINT64
r4	MQFLOAT32
r8	MQFLOAT64
string	MQCHAR []

Veri tipi olmayan herhangi bir öğenin stringtipinde olduğu varsayılır.

`xsi:nil='true'` öge özniteliğiyle boş değer gösterilir. Boş olmayan değerler için `xsi:nil='false'` özniteliğini kullanmayın. Örneğin, aşağıdaki özelliğin boş değeri vardır:

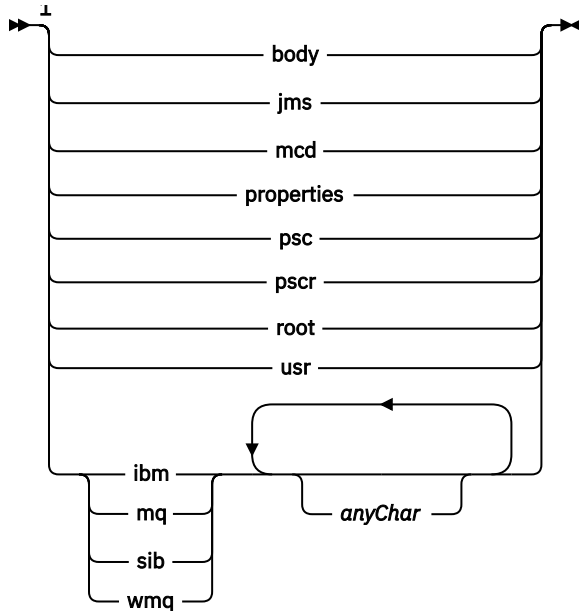
```
<NullProperty xsi:nil='true'></NullProperty>
```

Bir byte ya da karakter dizisi özelliğinin değeri boş olabilir. Boş bir değer, sıfır uzunluklu öge değeri olan bir MQRFH2 ögesiyle gösterilir. Örneğin, aşağıdaki özelliğin boş bir değeri vardır:

```
<EmptyProperty></EmptyProperty>
```

## Ayrılmış klasör ya da özellik klasörü adı

Bir klasörün ya da özellik klasörünün adını, aşağıdaki dizgilerden biriyle başlamayacak şekilde kısıtlayın. Önekler, IBM tarafından yaratılan klasör ya da özellik adları için ayrılır.

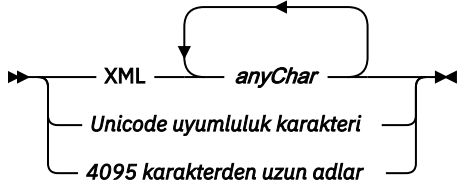


Notlar:

<sup>1</sup> Ayrılmış bir klasör ya da özellik adı, küçük ve büyük harflerin karışımını içerir.

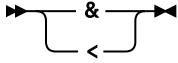
## Yol adı geçersiz

Bir ad-değer çiftinin ya da bir özelliğin tam yolunu, aşağıdaki dizgilerden herhangi birini içermeyecek şekilde kısıtlayın.



## Geçersiz karakterler

"&" ve "<"hazır bilgileri yerine her zaman &amp ; ve < > çıkış dizilerini kullanın.

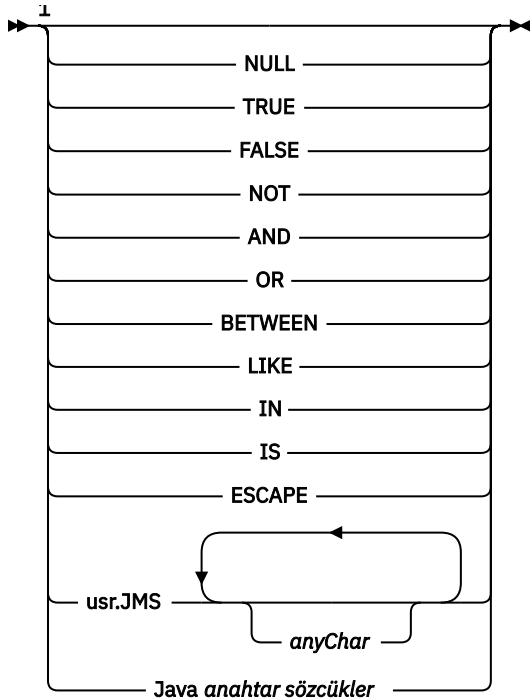


## Tanımlı özellik adları

Tanımlı özellik adları, IBM MQya da diğer ürünler tarafından tanımlanan ve IBM MQ ve kullanıcı uygulamaları tarafından kullanılan özelliklerin adlarıdır. Tanımlı özellikler yalnızca tanımlı özellik klasörlerinde var. Tanımlanan özellik adları, özellik klasörlerinin tanımında açıklanır; bkz. [Özellik klasörleri](#).

## Özellik adı geçersiz

Aşağıdaki kuralla eşleşen özellik adları oluşturmayın. Kural, yalnızca özellik ögesi adı için değil, bir özelliği adlandıran tam özellik yolu için de geçerlidir.



Notlar:

<sup>1</sup> Geçersiz bir özellik adı, herhangi bir büyük ve küçük harf birleşimi içerebilir.

## Geçersiz öznitelikler

Özellik klasörleri ve özellikleri yalnızca desteklenen “Özellik öznitelikleri” sayfa 537 ve “Veri türleri” sayfa 537 içerebilir.

Özellik klasörlerinde ya da özelliklerinde bulunan, desteklenmeyen XML benzeri öznitelikler (örneğin, tırnak içine alınmış dizgi değerleri olan adlar) kaldırılabilir.

Özellik olmayan klasörlerde ya da özellik olmayan öğelerde bulunan ve MQRFH2 üstbilgilerinde kalan XML benzeri öznitelikler.

## MQRMH-Başvuru iletisi üstbilgisi

MQRMH yapısı, başvuru iletisi üstbilgisinin biçimini tanımlar. Bu üstbilgi, çok büyük miktarlarda veri (*toplu veri* olarak adlandırılır) göndermek için kullanıcı tarafından yazılan ileti kanalı çıkışlarıyla birlikte kullanılır. bir kuyruk yöneticisinden diğerine. Normal ileti sistemiyle karşılaştırıldığında, toplu veriler bir kuyrukta saklanmaz; bunun yerine, kuyrukta yalnızca toplu verilere ilişkin bir *başvuru* saklanır. Bu, IBM MQ kaynaklarının çok az sayıda çok büyük ileti tarafından tüketilebilme olasılığını azaltır.

## Kullanılabilirlik

MQRMH yapısı aşağıdaki platformlarda kullanılabilir:

-  AIX
-  IBM i
-  Linux
-  Windows

ve bu sistemlere bağlı IBM MQ istemcileri için.

## Biçim Adı

MQFMT\_REF\_MSG\_HEADER

## Karakter kümesi ve kodlama

MQRMH ' deki karakter verileri ve görelî konum alanları tarafından adreslenen dizgiler, yerel kuyruk yöneticisinin karakter kümesinde olmalıdır; bu, **CodedCharSetId** kuyruk yöneticisi özneliği tarafından verilir. MQRMH ' deki sayısal veriler yerli makine kodlamasında olmalıdır; bu, C programlama dili için MQENC\_NATIVE değeriyle verilir.

MQRMH ' nin karakter takımını ve kodlamasını aşağıdaki *CodedCharSetId* ve *Encoding* alanlarına ayarlayın:

- MQMD (MQRMH yapısı ileti verilerinin başındadır) ya da
- MQRMH yapısından önce gelen üstbilgi yapısı (diğer tüm durumlar).

## Kullanım

Bir uygulama, MQRMH ' den oluşan bir ileti koyar, ancak toplu verileri atlar. Bir ileti kanalı aracısı (MCA) iletiyi iletim kuyruğundan okuduğunda, başvuru iletisi üstbilgisini işlemek için kullanıcı tarafından sağlanan bir ileti çıkışı çağırılır. Çıkış, MCA iletiyi kanal üzerinden sonraki kuyruk yöneticisine göndermeden önce, MQRMH yapısıyla tanıtilan toplu verilerin başvuru iletisinin sonuna eklenebilir.

Alma sonunda, başvuru iletilerini bekleyen bir ileti çıkışı var olmalıdır. Bir başvuru iletisi alındığında, çıkışın nesneyi iletideki MQRMH ' yi izleyen toplu verilerden yaratması ve daha sonra, başvuru iletisini toplu veriler olmadan aktarması gerekir. Başvuru iletisi, daha sonra bir kuyruktan başvuru iletisini (toplu veri olmadan) okuyan bir uygulama tarafından alınabilir.



Normalde, MQRMH yapısı iletideki tek yapıdır. Ancak, ileti bir iletim kuyruğuysa, MQRMH yapısından önce bir ya da daha çok ek üstbilgi gelir.

Dağıtım listesine bir başvuru iletisi de gönderilebilir. Bu durumda, ileti bir iletim kuyruğunda olduğunda MQDH yapısı ve ilgili kayıtları MQRMH yapısından önce gelir.

**Not:** Bir başvuru iletisini bölümlenmiş ileti olarak göndermeyin; ileti çıkışı iletiyi doğru olarak işleyemez.

## Veri dönüştürme

Veri dönüştürme amacıyla, MQRMH yapısının dönüştürülmesi, kaynak ortam verilerinin, kaynak nesne adının, hedef ortam verilerinin ve hedef nesne adının dönüştürülmesini içerir. Yapının başlangıcını izleyen *StrucLength* bayt içindeki diğer tüm baytlar atılır ya da veri dönüştürmeden sonra tanımlanmamış değerlere sahip olur. Aşağıdaki tüm deyimlerin doğru olması koşuluyla toplu veriler dönüştürülür:

- Veri dönüştürme işlemi gerçekleştirildiğinde iletide toplu veri bulunur.
- MQRMH ' deki *Format* alanında MQFMT\_NONE dışında bir değer var.
- Belirtilen biçim adına sahip, kullanıcı tarafından yazılan bir veri dönüştürme çıkışı var.

Ancak, ileti kuyruktayken genellikle toplu verilerin iletide bulunmadığını ve sonuç olarak toplu verilerin MQGMO\_CONVERT seçeneğiyle dönüştürüldüğünü unutmayın.

## Alanlar

**Not:** Aşağıdaki tabloda, alanlar alfabetik olarak değil, kullanıma göre gruplanır. Alt konular aynı sırayı izler.

Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
<u>StrucId</u> (yapı tanıtıcısı)	MQRMH_STRUC_ID	'RMH→'
<u>Sürüm</u> (yapı sürümü numarası)	MQRMH_VERSION_1	1
<u>StrucLength</u> (toplu veriler değil, sabit alanların sonundaki dizgiler de içinde olmak üzere toplam MQRMH uzunluğu)	Yok	0
<u>Kodlama</u> (toplu verilerin sayısal kodlaması)	MQENC_NATIVE	Ortama bağlıdır
<u>CodedCharSetId</u> (toplu verilerin karakter kümesi tanıtıcısı)	MQCCSI_AYRILMAMIŞ	0
<u>Biçim</u> (toplu verilerin biçim adı)	MQFMT_NONE	Boşluklar
<u>İşaretler</u> (başvuru iletisi işaretleri)	MQRMHF_NOT_LAST	0
<u>ObjectType</u> (nesne tipi)	Yok	Boşluklar
<u>ObjectInstanceId</u> (nesne eşgörünümü tanıtıcısı)	MQII_NONE	Boş Değerler
<u>SrcEnvLength</u> (kaynak ortam verilerinin uzunluğu)	Yok	0
<u>SrcEnvOffset</u> (kaynak ortam verilerinin görel konumu)	Yok	0
<u>SrcNameLength</u> (kaynak nesne adının uzunluğu)	Yok	0
<u>SrcNameGörel Konumu</u> (kaynak nesne adının görel konumu)	Yok	0
<u>DestEnvLength</u> (hedef ortam verilerinin uzunluğu)	Yok	0
<u>DestEnvGörel Konum</u> (Hedef Ortam Verilerinin Görel Konumu)	Yok	0

Çizelge 524. MQRMH için MQRMH 'deki Alanlar (devamı var)

Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
<u>DestNameLength</u> (hedef nesne adının uzunluğu)	Yok	0
<u>DestNameGörelili Konumu</u> (hedef nesne adının görelili konumu)	Yok	0
<u>DataLogicalLength</u> (Toplu veri uzunluğu)	Yok	0
<u>DataLogicalOffset</u> (Toplu verilerin düşük görelili konumu)	Yok	0
<u>DataLogicalOffset2</u> (toplu verilerin yüksek görelili konumu)	Yok	0

**Notlar:**

1. - simgesi, tek bir boş karakteri temsil eder.
2. C programlama dilinde, makro değişkeniMQRMH\_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlar için başlangıç değerleri sağlamak üzere bunu aşağıdaki şekilde kullanın:

```
MQRMH MyRMH = {MQRMH_DEFAULT};
```

## Dil bildirimleri

### MQRMH için C bildirimini

```
typedef struct tagMQRMH MQRMH;
struct tagMQRMH {
    MQCHAR4    StrucId;           /* Structure identifier */
    MQLONG     Version;          /* Structure version number */
    MQLONG     StrucLength;      /* Total length of MQRMH, including
                                strings at end of fixed fields, but
                                not the bulk data */
    MQLONG     Encoding;         /* Numeric encoding of bulk data */
    MQLONG     CodedCharSetId;   /* Character set identifier of bulk
                                data */
    MQCHAR8    Format;           /* Format name of bulk data */
    MQLONG     Flags;            /* Reference message flags */
    MQCHAR8    ObjectType;       /* Object type */
    MQBYTE24   ObjectInstanceId; /* Object instance identifier */
    MQLONG     SrcEnvLength;     /* Length of source environment data */
    MQLONG     SrcEnvOffset;     /* Offset of source environment data */
    MQLONG     SrcNameLength;    /* Length of source object name */
    MQLONG     SrcNameOffset;    /* Offset of source object name */
    MQLONG     DestEnvLength;    /* Length of destination environment
                                data */
    MQLONG     DestEnvOffset;    /* Offset of destination environment
                                data */
    MQLONG     DestNameLength;   /* Length of destination object name */
    MQLONG     DestNameOffset;   /* Offset of destination object name */
    MQLONG     DataLogicalLength; /* Length of bulk data */
    MQLONG     DataLogicalOffset; /* Low offset of bulk data */
    MQLONG     DataLogicalOffset2; /* High offset of bulk data */
};
```

### MQRMH için COBOL bildirimini

```
** MQRMH structure
   10 MQRMH.
**   Structure identifier
   15 MQRMH-STRUCID          PIC X(4).
**   Structure version number
   15 MQRMH-VERSION        PIC S9(9) BINARY.
**   Total length of MQRMH, including strings at end of fixed fields,
```

```

**      but not the bulk data
15 MQRMH-STRUCLength PIC S9(9) BINARY.
**      Numeric encoding of bulk
15 MQRMH-ENCODING PIC S9(9) BINARY.
**      Character set identifier of bulk data
15 MQRMH-CODEDCHARSETID PIC S9(9) BINARY.
**      Format name of bulk data
15 MQRMH-FORMAT PIC X(8).
**      Reference message flags
15 MQRMH-FLAGS PIC S9(9) BINARY.
**      Object type
15 MQRMH-OBJECTTYPE PIC X(8).
**      Object instance identifier
15 MQRMH-OBJECTINSTANCEID PIC X(24).
**      Length of source environment data
15 MQRMH-SRCENVLENGTH PIC S9(9) BINARY.
**      Offset of source environment data
15 MQRMH-SRCENVOFFSET PIC S9(9) BINARY.
**      Length of source object name
15 MQRMH-SRCNAMELENGTH PIC S9(9) BINARY.
**      Offset of source object name
15 MQRMH-SRCNAMEOFFSET PIC S9(9) BINARY.
**      Length of destination environment data
15 MQRMH-DESTENVLENGTH PIC S9(9) BINARY.
**      Offset of destination environment data
15 MQRMH-DESTENVOFFSET PIC S9(9) BINARY.
**      Length of destination object name
15 MQRMH-DESTNAMELENGTH PIC S9(9) BINARY.
**      Offset of destination object name
15 MQRMH-DESTNAMEOFFSET PIC S9(9) BINARY.
**      Length of bulk data
15 MQRMH-DATALOGICALENGTH PIC S9(9) BINARY.
**      Low offset of bulk data
15 MQRMH-DATALOGICALOFFSET PIC S9(9) BINARY.
**      High offset of bulk data
15 MQRMH-DATALOGICALOFFSET2 PIC S9(9) BINARY.

```

#### MQRMH için PL/I bildirimi

```

dcl
  1 MQRMH based,
  3 StrucId          char(4),          /* Structure identifier */
  3 Version          fixed bin(31), /* Structure version number */
  3 StrucLength      fixed bin(31), /* Total length of MQRMH,
                                     including strings at end of
                                     fixed fields, but not the bulk
                                     data */
  3 Encoding         fixed bin(31), /* Numeric encoding of bulk
                                     data */
  3 CodedCharSetId  fixed bin(31), /* Character set identifier of
                                     bulk data */
  3 Format            char(8),          /* Format name of bulk data */
  3 Flags            fixed bin(31), /* Reference message flags */
  3 ObjectType       char(8),          /* Object type */
  3 ObjectInstanceId char(24),         /* Object instance identifier */
  3 SrcEnvLength     fixed bin(31), /* Length of source environment
                                     data */
  3 SrcEnvOffset     fixed bin(31), /* Offset of source environment
                                     data */
  3 SrcNameLength    fixed bin(31), /* Length of source object name */
  3 SrcNameOffset    fixed bin(31), /* Offset of source object name */
  3 DestEnvLength    fixed bin(31), /* Length of destination
                                     environment data */
  3 DestEnvOffset    fixed bin(31), /* Offset of destination
                                     environment data */
  3 DestNameLength   fixed bin(31), /* Length of destination object
                                     name */
  3 DestNameOffset   fixed bin(31), /* Offset of destination object
                                     name */
  3 DataLogicalLength fixed bin(31), /* Length of bulk data */
  3 DataLogicalOffset fixed bin(31), /* Low offset of bulk data */
  3 DataLogicalOffset2 fixed bin(31); /* High offset of bulk data */

```

#### MQRMH için High Level Assembler (Üst Düzey Çevirici) bildirimi

```

MQRMH          DSECT
MQRMH_STRUCID  DS   CL4  Structure identifier

```

MQRMH_VERSION	DS	F	Structure version number
MQRMH_STRUCLNGTH	DS	F	Total length of MQRMH, including strings at end of fixed fields, but not the bulk data
*			
MQRMH_ENCODING	DS	F	Numeric encoding of bulk data
MQRMH_CODEDCHARSETID	DS	F	Character set identifier of bulk data
*			
MQRMH_FORMAT	DS	CL8	Format name of bulk data
MQRMH_FLAGS	DS	F	Reference message flags
MQRMH_OBJECTTYPE	DS	CL8	Object type
MQRMH_OBJECTINSTANCEID	DS	XL24	Object instance identifier
MQRMH_SRCENVLENGTH	DS	F	Length of source environment data
MQRMH_SRCENVOFFSET	DS	F	Offset of source environment data
MQRMH_SRCNAMELENGTH	DS	F	Length of source object name
MQRMH_SRCNAMEOFFSET	DS	F	Offset of source object name
MQRMH_DESTENVLENGTH	DS	F	Length of destination environment data
*			
MQRMH_DESTENVOFFSET	DS	F	Offset of destination environment data
*			
MQRMH_DESTNAMELENGTH	DS	F	Length of destination object name
MQRMH_DESTNAMEOFFSET	DS	F	Offset of destination object name
MQRMH_DATALOGICALENGTH	DS	F	Length of bulk data
MQRMH_DATALOGICALOFFSET	DS	F	Low offset of bulk data
MQRMH_DATALOGICALOFFSET2	DS	F	High offset of bulk data
*			
MQRMH_LENGTH	EQU	*-MQRMH	
	ORG	MQRMH	
MQRMH_AREA	DS	CL(MQRMH_LENGTH)	

## MQRMH için Visual Basic bildirimi

```

Type MQRMH
  StrucId          As String*4 'Structure identifier'
  Version          As Long     'Structure version number'
  StrucLength     As Long     'Total length of MQRMH, including'
                                     'strings at end of fixed fields, but'
                                     'not the bulk data'

  Encoding        As Long     'Numeric encoding of bulk data'
  CodedCharSetId As Long     'Character set identifier of bulk data'
  Format          As String*8 'Format name of bulk data'
  Flags          As Long     'Reference message flags'
  ObjectType     As String*8 'Object type'
  ObjectInstanceId As MQBYTE24 'Object instance identifier'
  SrcEnvLength   As Long     'Length of source environment data'
  SrcEnvOffset   As Long     'Offset of source environment data'
  SrcNameLength  As Long     'Length of source object name'
  SrcNameOffset  As Long     'Offset of source object name'
  DestEnvLength  As Long     'Length of destination environment'
                                     'data'
  DestEnvOffset  As Long     'Offset of destination environment'
                                     'data'
  DestNameLength As Long     'Length of destination object name'
  DestNameOffset As Long     'Offset of destination object name'
  DataLogicalLength As Long 'Length of bulk data'
  DataLogicalOffset As Long 'Low offset of bulk data'
  DataLogicalOffset2 As Long 'High offset of bulk data'
End Type

```

### **StrucId (MQCHAR4)**

Bu, yapı tanıttıcısıdır; değer şöyle olmalıdır:

#### **MQRMH\_STRUC\_ID**

Başvuru iletisi üstbilgi yapısına ilişkin tanıttıcı.

C programlama dili için, MQRMH\_STRUC\_ID\_ARRAY değişmezi de tanımlanır; bu, MQRMH\_STRUC\_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisidir.

Bu alanın ilk değeri MQRMH\_STRUC\_ID 'dir.

### **Sürüm (MQUZE)**

Yapı sürüm numarası. Değer şu olmalıdır:

## **MQRMH\_VERSION\_1**

Version-1 başvuru iletisi üstbilgi yapısı.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

## **MQRMH\_CURRENT\_VERSION**

Başvuru iletisi üstbilgi yapısının yürürlükteki sürümü.

Bu alanın ilk değeri MQRMH\_VERSION\_1' dir.

## **StrucLength (MQUZE)**

Sabit alanların sonundaki dizgiler de dahil olmak üzere MQRMH ' nin toplam uzunluğu, ancak toplu veriler değil.

Bu alanın ilk değeri sıfır.

## **Kodlama (MQUZE)**

Bu, toplu verilerin sayısal kodlamasını belirtir; MQRMH yapısındaki sayısal veriler için geçerli değildir.

MQPUT ya da MQPUT1 çağrısında, uygulama bu alanı veri için uygun değere ayarlamalıdır.

Bu alanın ilk değeri MQENC\_NATIVE 'dir.

## **CodedCharSetId (MQHOMER)**

Bu, toplu verilerin karakter kümesi tanıtıcısını belirtir; MQRMH yapısındaki karakter verileri için geçerli değildir.

MQPUT ya da MQPUT1 çağrısında, uygulama bu alanı veri için uygun değere ayarlamalıdır. Aşağıdaki özel değer kullanılabilir:

## **MQCCSI\_INHERIT**

Bu yapıyı izleyen verilerdeki karakter verileri, bu yapıyla aynı karakter kümesinde yer alıyor.

Kuyruk yöneticisi, bu değeri, iletide belirtilen yapının gerçek karakter kümesi tanıtıcısına göre değiştirir. Herhangi bir hata oluşmaması koşuluyla, MQCCSI\_INHERIT değeri MQGET çağrısıyla döndürülmez.

MQMD ' deki PutApp1Type alanının değeri MQAT\_BROKER ise, MQCCSI\_INHERIT kullanmayın.

Bu değer aşağıdaki ortamlarda desteklenir:

-  AIX
-  IBM i
-  Linux
-  Windows

ve bu sistemlere bağlı IBM MQ istemcileri için geçerli olur.

Bu alanın ilk değeri MQCCSI\_UNDEFED değerine sahip.

## **Biçim (MQCHAR8)**

Bu, toplu verilerin biçim adını belirtir.

MQPUT ya da MQPUT1 çağrısında, uygulama bu alanı veri için uygun değere ayarlamalıdır. Bu alanı kodlamaya ilişkin kurallar, MQMD ' deki *Format* alanı için yapılan kurallarla aynıdır.

Bu alanın ilk değeri MQFMT\_NONE olur.

## **İşaretler (MQUZE)**

Bunlar, başvuru ileti işaretleridir. Aşağıdaki işaretler tanımlanır:

## **MQRMHF\_SON**

Bu işaret, başvuru iletisinin başvuruda bulunulan nesnenin son bölümünü temsil ettiğini ya da içerdiğini gösterir.

## **MQRMHF\_NOT\_SON**

Başvuru iletisi nesnenin son bölümünü içermiyor ya da göstermiyor. MQRMHF\_NOT\_SON yardımcı program belgeleri. Bu seçeneğin başka biriyle birlikte kullanılması amaçlanmaz, ancak değeri sıfır olduğu için bu tür bir kullanım saptanamaz.

Bu alanın başlangıç değeri, MQRMHF\_NOT\_SON olarak geçerli olur.

## **ObjectType (MQCHAR8)**

Bu ad, ileti çıkışının desteklediği başvuru iletisi tiplerini tanımak için kullanabileceği bir addir. Ad, *Format* alanıyla aynı kurallara uygun olmalıdır, bkz. "[Biçim \(MQCHAR8\)](#)" sayfa 545.

Bu alanın ilk değeri 8 boşluktur.

## **ObjectInstanceTanıtıcısı (MQBYTE24)**

Bir nesnenin belirli bir örneğini tanımlamak için bu alanı kullanın. Gerekli değilse, aşağıdaki değere ayarlayın:

### **MQOL\_NONE**

Nesne eşgörünümü tanıtıcısı belirtilmedi. Değer, alanın uzunluğuna ilişkin ikili sıfır değeridir.

C programlama dili için, MQOII\_NONE\_ARRAY değişmezi de tanımlanır; bu, MQOLI\_NONE ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisi olur.

Bu alanın uzunluğu MQ\_OBJECT\_INSTANCE\_ID\_LENGTH tarafından verilmektedir. Bu alanın ilk değeri MQOLI\_NONE olur.

## **SrcEnvUzunluğu (MQUZE)**

Kaynak ortam verisinin uzunluğu. Bu alan sıfırsa, kaynak ortam verileri yoktur ve *SrcEnvOffset* yoksayılr.

Bu alanın ilk değeri 0 'tır.

## **SrcEnvGörelî Konumu (MQUZE)**

Bu alan, MQRMH yapısının başlangıcından kaynak ortam verilerinin kayma değerini belirtir. Kaynak ortam verileri, bu veriler yaratan kişi tarafından biliniyorsa, başvuru iletisinin yaratıcısı tarafından belirtilebilir. Örneğin, Windows üzerinde kaynak ortam verileri, toplu verileri içeren nesnenin dizin yolu olabilir. Ancak, yaratan kaynak ortam verilerini bilmiyorsa, kullanıcı tarafından sağlanan ileti çıkışı, gereken ortam bilgilerini saptamalıdır.

Kaynak ortam verilerinin uzunluğu *SrcEnvLength* tarafından verilir; Bu uzunluk sıfırsa, kaynak ortam verileri yoktur ve *SrcEnvOffset* yoksayılr. Bu durumda kaynak ortam verileri, yapının başlangıcından itibaren *StrucLength* bayt içinde tamamen bulunmalıdır.

Uygulamaların, yapıdaki son sabit alandan hemen sonra ortam verilerinin başladığını ya da *SrcNameOffset*, *DestEnvOffset* ve *DestNameOffset* alanları tarafından adreslenen verilerin herhangi biriyle bitişik olduğunu varsaymamalıdır.

Bu alanın ilk değeri 0 'tır.

## **SrcNameUzunluğu (MQUZE)**

Kaynak nesne adının uzunluğu. Bu alan sıfırsa, kaynak nesne adı yoktur ve *SrcNameOffset* yoksayılr.

Bu alanın ilk değeri 0 'tır.

## **SrcNameGörelî Konumu (MQUZE)**

Bu alan, MQRMH yapısının başlangıcından kaynak nesne adının kayma değerini belirtir. Kaynak nesne adı, bu veriler oluşturucuda biliniyorsa, başvuru iletilisinin yaratıcısı tarafından belirtilebilir. Ancak, yaratan kaynak nesne adını bilmiyorsa, kullanıcı tarafından sağlanan ileti çıkışı, erişilecek nesneyi tanımlamalıdır.

Kaynak nesne adının uzunluğu *SrcNameLength* tarafından verilir; Bu uzunluk sıfırsa, kaynak nesne adı yoktur ve *SrcNameOffset* yoksayıdır. If present, the source object name must reside completely within *StrucLength* bytes from the start of the structure.

Uygulamalar, kaynak nesne adının *SrcEnvOffset*, *DestEnvOffset* ve *DestNameOffset* alanları tarafından adreslenen herhangi bir veriyle bitişik olduğunu varsaymamalıdır.

Bu alanın ilk değeri 0 'tır.

### ***DestEnvUzunluğu (MQUZE)***

Bu, hedef ortam verilerinin uzunluğudur. Bu alan sıfırsa, hedef ortam verileri yoktur ve *DestEnvOffset* yoksayıdır.

### ***DestEnvGörelî Konumu (MQUZE)***

Bu alan, MQRMH yapısının başlangıcından hedef ortam verilerinin kayma değerini belirtir. Hedef ortam verileri, bu veriler yaratan kişi tarafından biliniyorsa, başvuru iletilisinin yaratıcısı tarafından belirtilebilir. Örneğin, Windows üzerinde hedef ortam verileri, toplu verilerin saklanabileceği nesne dizin yolu olabilir. Ancak, yaratan hedef ortam verilerini bilmiyorsa, gereken ortam bilgilerini belirlemek için kullanıcı tarafından sağlanan ileti çıkışının sorumluluğundadır.

Hedef ortam verilerinin uzunluğu *DestEnvLength* tarafından verilir; Bu uzunluk sıfırsa, hedef ortam verileri yoktur ve *DestEnvOffset* yoksayıdır. Bu durumda, hedef ortam verilerinin, yapının başlangıcından itibaren *StrucLength* bayt içinde tamamen bulunması gerekir.

Uygulamalar, hedef ortam verilerinin *SrcEnvOffset*, *SrcNameOffset* ve *DestNameOffset* alanları tarafından adreslenen herhangi bir veriyle bitişik olduğunu varsaymamalıdır.

Bu alanın ilk değeri 0 'tır.

### ***DestNameUzunluğu (MQUZE)***

Hedef nesne adının uzunluğu. Bu alan sıfırsa, hedef nesne adı yoktur ve *DestNameOffset* yoksayıdır.

### ***DestNameGörelî Konumu (MQUZE)***

Bu alan, MQRMH yapısının başlangıcından hedef nesne adının kayma değerini belirtir. Hedef nesne adı, bu veriler oluşturucuda biliniyorsa, başvuru iletilisinin yaratıcısı tarafından belirtilebilir. Ancak, yaratan hedef nesne adını bilmiyorsa, yaratılacak ya da değiştirilecek nesneyi tanımlamak için kullanıcı tarafından sağlanan ileti çıkışının sorumluluğundadır.

Hedef nesne adının uzunluğu *DestNameLength* tarafından verilir; Bu uzunluk sıfırsa, hedef nesne adı yoktur ve *DestNameOffset* yoksayıdır. Varsa, hedef nesne adının, yapının başlangıcından itibaren *StrucLength* bayt içinde tamamen bulunması gerekir.

Uygulamalar, hedef nesne adının *SrcEnvOffset*, *SrcNameOffset* ve *DestEnvOffset* alanları tarafından adreslenen herhangi bir veriyle bitişik olduğunu varsaymamalıdır.

Bu alanın ilk değeri 0 'tır.

### ***DataLogicalLength (MQUZE)***

*DataLogicalLength* alanı, MQRMH yapısının gönderme yaptığı toplu verilerin uzunluğunu belirtir.

Toplu veriler iletide gerçekten varsa, veriler, MQRMH yapısının başlangıcındaki *StrucLength* byte 'tan itibaren bir görelî konum olarak başlar. Tüm ileti eksi *StrucLength* uzunluğunun uzunluğu, toplu veri uzunluğunun uzunluğunu gösterir.

İletide veri varsa, *DataLogicalLength* , ilgili verilerin miktarını belirtir. Olağan durumda, *DataLogicalLength* için, iletide bulunan veri uzunluğunun aynı değere sahip olması gerekir.

MQRMH yapısı nesnedeki kalan verileri (belirtilen mantıksal görel konumdan başlayarak) gösteriyorsa, toplu verilerin iletide gerçekte var olmamasını sağlamak için *DataLogicalLength* için sıfır değerini kullanabilirsiniz.

Veri yoksa, MQRMH ' nin sonu iletinin sonuna denk gelir.

Bu alanın ilk değeri 0 'tır.

### **DataLogicalGörel Konumu (MQUZE)**

Bu alan, toplu veri formlarının parçası olan nesnenin başlangıcından elde edilen toplu verilerin alt görel konumunu belirtir. Nesnenin başlangıcından elde edilen toplu verilerin görel konumu, *mantıksal görel konum* olarak adlandırılır. Bu, MQRMH yapısının başlangıcından elde edilen toplu verilerin fiziksel görel konumu değildir; görel konum *StrucLength* tarafından verilir.

Büyük nesnelerin başvuru iletileri kullanılarak gönderilmesine izin vermek için, mantıksal görel konum iki alana ayrılır ve gerçek mantıksal görel konum, bu iki alanın toplamını verir:

- *DataLogicalOffset* , mantıksal görel konum 1 000 000 000 olarak bölüldüğünde elde edilen kalanı temsil eder. Böylece, 0 ile 999 999 999 aralığında bir değer vardır.
- *DataLogicalOffset2* , mantıksal görel konum 1 000 000 000 olarak bölüldüğünde elde edilen sonucu temsil eder. Bu, mantıksal kayma içinde var olan 1 000 000 000 'in tam katının sayısıdır. Birden çok sayı, 0 ile 999 999 999 arasında bir aralıktaki sayıdır.

Bu alanın ilk değeri 0 'tır.

### **DataLogicalOffset2 (MQUZE)**

Bu alan, toplu veri formlarının parçası olan nesnenin başlangıcından elde edilen toplu verilerin yüksek görel konum değerini belirtir. Bu değer, 0 ile 999 999 999 arasındaki bir aralıktaki bir değerdir. Ayrıntılar için bkz. *DataLogicalOffset* .

Bu alanın ilk değeri 0 'tır.

### **MQRR-Yanıt kaydı**

Hedef bir dağıtım listesi olduğunda, tek bir hedef kuyruğa ilişkin açma ya da koyma işleminin sonucu olan tamamlanma kodunu ve neden kodunu almak için MQRR yapısını kullanın. MQRR, MQOPEN, MQPUT ve MQPUT1 çağrılarına ilişkin bir çıkış yapısıdır.

### **Kullanılabilirlik**

MQRR yapısı aşağıdaki altyapılarda kullanılabilir:

-  AIX
-  IBM i
-  Linux
-  Windows

ve bu sistemlere bağlı IBM MQ istemcileri için.

### **Karakter kümesi ve kodlama**

MQRR içindeki veriler, **CodedCharSetId** kuyruk yöneticisi özniteliği ve MQENC\_NATIVE tarafından verilen yerel kuyruk yöneticisinin kodlaması tarafından belirtilen karakter kümesinde olmalıdır. Ancak, uygulama bir MQ MQI istemcisi olarak çalışıyorsa, yapı istemcinin karakter kümesinde ve kodlamasında olmalıdır.



## Kullanım

MQOPEN ve MQPUT çağrılarında ya da MQPUT1 çağrısında bu yapıların bir dizisini sağlayarak, bir dağıtım listesindeki tüm kuyrukların tamamlanma kodlarını ve neden kodlarını, çağrı sonucu karışık olduğunda (yani, çağrı listedeki bazı kuyruklar için başarılı olduğunda ancak diğerleri için başarısız olduğunda) saptayabilirsiniz. Çağrıdaki neden kodu MQRC\_MULTIPLE\_REASON, kuyruk yöneticisi tarafından yanıt kayıtlarının (uygulama tarafından sağlandıysa) ayarlandığını gösteriyor.

## Alanlar

**Not:** Aşağıdaki tabloda, alanlar alfabetik olarak değil, kullanıma göre gruplanır. Alt konular aynı sırayı izler.

Çizelge 525. MQRR 'deki Alanlar		
Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
CompCode (kuyruk için tamamlama kodu)	MQCC_OK	0
Neden (kuyruğa ilişkin neden kodu)	MQRC_NONE	0

**Notlar:**

1. C programlama dilinde, makro değişkeniMQRR\_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlar için başlangıç değerleri sağlamak üzere bunu aşağıdaki şekilde kullanın:

```
MQRR MyRR = {MQRR_DEFAULT};
```

## Dil bildirimleri

### MQRR için C bildirimi

```
typedef struct tagMQRR MQRR;  
struct tagMQRR {  
    MQLONG CompCode; /* Completion code for queue */  
    MQLONG Reason; /* Reason code for queue */  
};
```

### MQRR için COBOL bildirimi

```
** MQRR structure  
10 MQRR.  
** Completion code for queue  
15 MQRR-COMPCODE PIC S9(9) BINARY.  
** Reason code for queue  
15 MQRR-REASON PIC S9(9) BINARY.
```

### MQRR için PL/I bildirimi

```
dcl  
1 MQRR based,  
3 CompCode fixed bin(31), /* Completion code for queue */  
3 Reason fixed bin(31); /* Reason code for queue */
```

### MQRR için Visual Basic bildirimi

```
Type MQRR  
CompCode As Long 'Completion code for queue'  
Reason As Long 'Reason code for queue'  
End Type
```

## CompCode (MQUZN)

Bu, MQOPED ya da MQPUT1 çağrısında bulunan MQOR yapılarının dizideki ilgili öge tarafından belirtilen ada ilişkin açma ya da koyma işleminin sonucundaki tamamlanma kodudur.

Bu her zaman bir çıkış alanıdır. Bu alanın ilk değeri MQCC\_OK olur.

## Neden (MQUZE)

Bu, MQOPED ya da MQPUT1 çağrısında bulunan MQOR yapılarının dizideki ilgili öge tarafından belirtilen ada ilişkin açma ya da koyma işleminin sonucundaki açık ya da koyma işleminin neden kodudur.

Bu her zaman bir çıkış alanıdır. Bu alanın ilk değeri MQRC\_NONE olur.

## MQSCO-SSL/TLS yapılandırma seçenekleri

MQCD yapısındaki TLS alanlarıyla birlikte MQSCO yapısı, IBM MQ MQI client olarak çalışan bir uygulamanın, kanal protokolü TCP/IP olduğunda istemci bağlantısı için TLS kullanımını denetleyen yapılandırma seçeneklerini belirtmesine olanak sağlar. Yapı, MQCONNX çağrısındaki bir giriş değiştirgesidir.

## Kullanılabilirlik

MQSCO yapısı aşağıdaki istemcilerde kullanılabilir:

-  AIX
-  IBM i
-  Linux
-  Windows

İstemci kanalına ilişkin kanal protokolü TCP/IP değilse, MQSCO yapısı yoksayılr.

## Karakter kümesi ve kodlama

MQSCO içindeki veriler, **CodedCharSetId** kuyruk yöneticisi özniteliği tarafından belirtilen karakter kümesinde ve MQENC\_NATIVE tarafından verilen yerel kuyruk yöneticisinin kodlamasında olmalıdır.

## Alanlar

**Not:** Aşağıdaki tabloda, alanlar alfabetik olarak değil, kullanıma göre gruplanır. Alt konular aynı sırayı izler.

Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
StrucId (yapı tanıtıcısı)	MQSCO_STRUC_ID	'SCO-'
Sürüm (yapı sürümü numarası)	MQSCO_CURRENT_VERSION	1
KeyRepository (anahtar havuzunun yeri)	Yok	Boş değerli dizilim ya da boşluklar
CryptoHardware (şifreleme donanımına ilişkin ayrıntılar)	Yok	Boş değerli dizilim ya da boşluklar
AuthInfoRecCount (var olan MQAIR kayıtlarının sayısı)	Yok	0

Çizelge 526. MQSCO içindeki alanlar (devamı var)		
Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
AuthInfoRecOffset (MQSCO ' nun başlangıcından ilk MQAIR kaydının görelî konumu)	Yok	0
AuthInfoRecPtr (ilk MQAIR kaydının adresi)	Yok	Boş değeri gösterge ya da boş değeri byte
<b>Not:</b> <i>Version</i> , MQSCO_VERSION_2değerinden küçükse, aşağıdaki iki alan yoksayılır.		
KeyResetCount (TLS gizli anahtar sıfırlama sayısı)	MQSCO_RESET_COUNT_DEFAULT	0
“FipsRequired (MQLONG)” sayfa 555 ( IBM MQÇinde FIPS sertifikalı şifreleme algoritmalarını kullanın)	MQSSL_FIPS_NO	0
<b>Not:</b> <i>Version</i> , MQSCO_VERSION_3değerinden küçükse, aşağıdaki iki alan yoksayılır.		
EncryptionPolicySuiteB (yalnızca Suite B şifreleme algoritmalarını kullanın)	MQ_SUITE_B_NONE, MQ_SUITE_B_NOT_AVAILABLE, MQ_SUITE_B_NOT_AVAILABLE, MQ_SUITE_B_NOT_AVAILABLE	1, 0, 0, 0
<b>Not:</b> <i>Version</i> , MQSCO_VERSION_4değerinden küçükse, aşağıdaki iki alan yoksayılır.		
CertificateValİlkesi (sertifika doğrulama ilkesi)	MQ_CERT_VAL_POLICY_DEFAULT	0
<b>Not:</b> <i>Version</i> , MQSCO_VERSION_5değerinden küçükse, aşağıdaki iki alan yoksayılır.		
CertificateLabel (kullanılmakta olan sertifika etiketinin ayrıntıları)	Yok	Boş değeri dizilim ya da boşluklar

#### Notlar:

1. – simgesi, tek bir boş karakteri temsil eder.
2. C programlama dilinde, makro değışkeniMQSCO\_DEFAULT , çizelgede listelenen değeri içerir. Yapıdaki alanlar için başlangıç değeri sağlamak üzere bunu aşağıdaki şekilde kullanın:

```
MQSCO MySCO = {MQSCO_DEFAULT};
```

## Dil bildirimleri

### MQSCO için C bildirim

```
typedef struct tagMQSCO MQSCO;
struct tagMQSCO {
    MQCHAR4    StrucId;                /* Structure identifier */
    MQLONG     Version;                /* Structure version number */
    MQCHAR256  KeyRepository;          /* Location of TLS key */
    MQCHAR256  CryptoHardware;         /* repository */
    MQCHAR256  ConfigurationString;    /* Cryptographic hardware */
    MQLONG     AuthInfoRecCount;       /* configuration string */
    /* Number of MQAIR records */
};
```

```

MQLONG      AuthInfoRecOffset;          /* present */
                                                /* Offset of first MQAIR */
                                                /* record from start of */
                                                /* MQSCO structure */
PMQAIR      AuthInfoRecPtr;            /* Address of first MQAIR */
                                                /* record */
/* Ver:1 */
MQLONG      KeyResetCount;             /* Number of unencrypted */
                                                /* bytes sent/received */
                                                /* before secret key is */
                                                /* reset */
MQLONG      FipsRequired;              /* Using FIPS-certified */
/* Ver:2 */
                                                /* algorithms */
MQLONG      EncryptionPolicySuiteB[4]; /* Use only Suite B */
/* Ver:3 */
MQLONG      CertificateValPolicy;       /* cryptographic algorithms */
                                                /* Certificate validation */
                                                /* policy */
/* Ver:4 */
MQCHAR64    CertificateLabel;          /* Certificate label */
/* Ver:5 */

};

```

### MQSCO için COBOL bildirimi

```

** MQSCO structure
10 MQSCO.
** Structure identifier
15 MQSCO-STRUCID PIC X(4).
** Structure version number
15 MQSCO-VERSION PIC S9(9) BINARY.
** Location of TLS key repository
15 MQSCO-KEYREPOSITORY PIC X(256).
** Cryptographic hardware configuration string
15 MQSCO-CRYPTOHARDWARE PIC X(256).
** Number of MQAIR records present
15 MQSCO-AUTHINFORECCOUNT PIC S9(9) BINARY.
** Offset of first MQAIR record from start of MQSCO structure
15 MQSCO-AUTHINFORECOFFSET PIC S9(9) BINARY.
** Address of first MQAIR record
15 MQSCO-AUTHINFORECPTTR POINTER.
** Version 1 **
** Number of unencrypted bytes sent/received before secret key is
** reset
15 MQSCO-KEYRESETCOUNT PIC S9(9) BINARY.
** Using FIPS-certified algorithms
15 MQSCO-FIPSREQUIRED PIC S9(9) BINARY.
** Version 2 **
** Use only Suite B cryptographic algorithms
15 MQSCO-ENCRYPTIONPOLICYSUITEB PIC S9(9) BINARY OCCURS 4.
** Version 3 **
** Certificate validation policy setting
15 MQSCO-CERTIFICATEVALPOLICY PIC S9(9) BINARY.
** Version 4 **
** SSL/TLS certificate label
15 MQSCO-CERTIFICATELABEL PIC X(64).
** Version 5 **

```

### MQSCO için PL/I bildirimi

```

dcl
1 MQSCO based,
3 StrucId char(4), /* Structure identifier */
3 Version fixed bin(31), /* Structure version number */
3 KeyRepository char(256), /* Location of TLS key
repository */
3 CryptoHardware char(256), /* Cryptographic hardware
configuration string */
3 AuthInfoRecCount fixed bin(31), /* Number of MQAIR records
present */
3 AuthInfoRecOffset fixed bin(31), /* Offset of first MQAIR record
from start of MQSCO structure */
3 AuthInfoRecPtr pointer, /* Address of first MQAIR record */
3 KeyResetCount fixed bin(31), /* Key reset count */
/* Version 1 */

```

```

3 FipsRequired          fixed bin(31), /* FIPS required */
/* Version 2 */
3 EncryptionPolicySuiteB (4) fixed bin(31), /* Suite B encryption policy */
/* Version 3 */
3 CertificateValPolicy  fixed bin(31), /* Certificate validation policy */
/* Version 4 */
3 CertificateLabel      char(64),      /* SSL/TLS certificate label */
/* Version 5 */

```

## MQSCO için Visual Basic bildirimi

```

Type MQSCO
StrucId      As String*4  'Structure identifier'
Version      As Long      'Structure version number'
KeyRepository As String*256 'Location of TLS key repository'
CryptoHardware As String*256 'Cryptographic hardware configuration'
'AuthInfoRecCount As Long      'Number of MQAIR records present'
AuthInfoRecOffset As Long      'Offset of first MQAIR record from'
'AuthInfoRecPtr As MQPTR     'Address of first MQAIR record'
KeyResetCount As Long      'Number of unencrypted bytes sent/received before secret key
is reset'
'Version 1'
FipsRequired As Long      'Mandatory FIPS CipherSpecs?'
'Version 2'
End Type

```

## İlgili başvurular

“MQCNO-Bağlantı seçenekleri” sayfa 314

MQCNO yapısı, uygulamanın kuyruk yöneticisine yönelik bağlantıyla ilgili seçenekleri belirtmesine olanak sağlar. Yapı, MQCONNX çağrısındaki bir giriş/çıkış değiştirgesidir.

## StrucId (MQCHAR4)

Bu, yapı tanıtıcısıdır; değer şöyle olmalıdır:

### MQSCO\_STRUC\_ID

TLS yapılandırma seçenekleri yapısına ilişkin tanıtıcı.

C programlama dili için, sabit MQSCO\_STRUC\_ID\_ARRAY değişmezi de tanımlanır; bu, MQSCO\_STRUC\_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisi olur.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQSCO\_STRUC\_ID ' dir.

## Sürüm (MQUZE)

Bu, yapı sürüm numarasıdır; değer şu olmalıdır:

### MQSCO\_VERSION\_1

Version-1 TLS yapılandırma seçenekleri yapısı.

### MQSCO\_VERSION\_2

Version-2 TLS yapılandırma seçenekleri yapısı.

### MQSCO\_VERSION\_3

Version-3 TLS yapılandırma seçenekleri yapısı.

### MQSCO\_VERSION\_4

Version-4 TLS yapılandırma seçenekleri yapısı.

### MQSCO\_VERSION\_5

Version-5 TLS yapılandırma seçenekleri yapısı.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

### MQSCO\_CURRENT\_VERSION

TLS yapılandırma seçenekleri yapısının geçerli sürümü.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQSCO\_VERSION\_1' dir.

**KeyRepository (MQCHAR256)**

Bu alan yalnızca AIX, Linux, and Windows sistemlerinde çalışan IBM MQ MQI clients için anlamlıdır. Anahtarların ve sertifikaların saklandığı anahtar veri tabanı dosyasının yerini belirtir. Anahtar veri tabanı dosyasının dosya adı zzz.kdbbiçiminde olmalıdır; burada zzz kullanıcı tarafından seçilebilir. *KeyRepository* alanı, dosya adı gövdesiyle birlikte bu dosyanın yolunu içerir (dosya adı, son .kdbdahil değil, ancak dosya adı dahil değildir). .kdb dosya soneki otomatik olarak eklenir.

Her anahtar veri tabanı dosyasının ilişkili bir *parola saklama dosyası kütüğü* vardır. Bu, anahtar veri tabanına programlı erişim izni vermek için kullanılan kodlanmış parolaları içerir. Parola şifreleme dosyası, aynı dizinde yer almalıdır ve anahtar veritabanı ile aynı dosya sapına sahip olmalıdır ve .sthsonekiyle sona ermelidir.

Örneğin, *KeyRepository* alanının değeri /xxx/yyy/keyise, anahtar veritabanı dosyası /xxx/yyy/key.kdbolmalı ve parola şifreleme dosyası /xxx/yyy/key.stholmalıdır; burada xxx ve yyy dizin adlarını gösterir.

Değer, alanın uzunluğundan kısaysa, değeri boş değerli bir karakterle sonlandırın ya da alanın uzunluğuna göre boşluklarla doldurur. Değer denetlenmez; anahtar havuzuna erişilirken bir hata varsa, çağrı neden kodu MQRC\_KEY\_REPOSITORY\_ERROR ile başarısız olur.

Bir IBM MQ MQI client' den TLS bağlantısı çalıştırmak için, *KeyRepository* seçeneğini geçerli bir anahtar veritabanı dosyası adına ayarlayın.

Bu bir giriş alanıdır. Bu alanın uzunluğu MQ\_SSL\_KEY\_REPOSITORY\_LENGTH tarafından verilir. Bu alanın ilk değeri, C içindeki boş değerli dizidir ve diğer programlama dillerindeki boşluk karakteridir.

**CryptoHardware (MQCHAR256)**

Bu alan, istemci sistemine bağlı olan şifreleme donanımları için yapılandırma ayrıntılarını verir.

Alanı aşağıdaki biçimde bir dizgiye ayarlayın ya da boş bırakın ya da boş bırakın:

```
GSK_PKCS11=the PKCS #11 driver path and file name;the PKCS #11
token label;the PKCS #11 token password;symmetric cipher setting;
```

PKCS #11 arabirimine uyan şifreleme donanımını kullanmak için, örneğin, IBM 4960 ya da IBM 4764, PKCS #11 sürücü yolu, PKCS #11 simgesi etiketi ve PKCS #11 simgeli parola dizilimleri belirtilmelidir, her biri noktalı virgülle sonlandırıldı.

PKCS #11 sürücüsü yolu, PKCS #11 kartı için destek sağlayan paylaşılan kitaplık için mutlak bir yoldur. PKCS #11 sürücüsü dosya adı, paylaşılan kitaplığın adıdır. PKCS #11 yolu ve dosya adı için gereken değer örneği:

```
/usr/lib/pkcs11/PKCS11_API.so
```

PKCStokenbelirtici etiketi, donanımınızı yapılandırıdığınız etiketle eşleşmelidir.

Herhangi bir şifreleme donanımı yapılandırması gerekmiyorsa, alanı boş ya da boş olarak ayarlayın.

Değer, alanın uzunluğundan kısaysa, değeri boş değerli bir karakterle sonlandırın ya da alanın uzunluğuna göre boşluklarla doldurur. Değer geçerli değilse ya da şifreleme donanımını yapılandırmak için kullanılan bir hataya yol açarsa, çağrı, MQRC\_CRYPTO\_HARDWARE\_ERROR neden kodunda başarısız olur.

Bu bir giriş alanıdır. Bu alanın uzunluğu MQ\_SSL\_CRYPTO\_HARDWARE\_LENGTH tarafından verilir. Bu alanın ilk değeri, C içindeki boş değerli dizidir ve diğer programlama dillerindeki boşluk karakteridir.

**AuthInfoRecCount (MQUBE)**

Bu, *AuthInfoRecPtr* ya da *AuthInfoRecOffset* alanları tarafından adreslenen kimlik doğrulama bilgileri (MQAIR) kayıtlarından biri. Daha fazla bilgi için bkz. "[MQAIR-Kimlik doğrulama bilgileri kaydı](#)" sayfa 269. Değer sıfır ya da daha büyük olmalıdır. Değer geçerli değilse, çağrı neden kodu MQRC\_AUTH\_INFO\_REC\_COUNT\_ERROR neden ile başarısız olur.

Bu bir giriş alanıdır. Bu alanın ilk değeri 0 'tır.

### ***AuthInfoRecOffset (MQUZE)***

Bu, MQSCO yapısının başlangıcındaki ilk kimlik doğrulama bilgileri kaydının bayt cinsinden görelî konutdur. Görelî konum pozitif ya da negatif olabilir. *AuthInfoRecCount* sıfırda, alan yoksayılr.

MQAIR kayıtlarını belirtmek için *AuthInfoRecOffset* ya da *AuthInfoRecPtr* değerini kullanabilirsiniz, ancak her ikisini birden belirleyemez; ayrıntılar için *AuthInfoRecPtr* alanının tanımına bakın.

Bu bir giriş alanıdır. Bu alanın ilk değeri 0 'tır.

### ***AuthInfoRecPtr (PMQAIR)***

Bu, ilk kimlik doğrulama bilgileri kaydının adresidir. *AuthInfoRecCount* sıfırda, alan yoksayılr.

MQAIR kayıtları dizisini aşağıdaki iki yoldan biriyle sağlayabilirsiniz:

- *AuthInfoRecPtr* işaretçi alanını kullanarak

Bu durumda uygulama, MQSCO yapısından ayrı olan bir MQAIR kaydı dizisini bildirebilir ve *AuthInfoRecPtr* ' ı dizinin adresine ayarlayabilir.

Farklı ortamlara (örneğin, C programlama dili gibi) taşınabilir bir şekilde işaretçi veri tipini destekleyen programlama dilleri için *AuthInfoRecPtr* ' i kullanmayı düşünün.

- By using the offset field *AuthInfoRecOffset*

Bu durumda, uygulama MQAIR kayıtları dizisini izleyen bir MQSCO içeren bir bileşik yapı bildirmeli ve *AuthInfoRecOffset* değerini, MQSCO yapısının başlangıcındaki dizideki ilk kaydın kayma değerine ayarlamalıdır. Bu değerin doğru olduğundan ve bir MQlong (en kısıtlayıcı programlama dili COBOL) içinde konaklayabilecek bir değere sahip olduğundan emin olun (geçerli aralık -999 999 999 ile +999 999 999 aralığında).

İşaretçi veri tipini desteklemeyen programlama dilleri için *AuthInfoRecOffset* komutunu kullanmayı ya da işaretçi veri tipini farklı ortamlara (örneğin, COBOL programlama dili gibi) portatif olmayan bir şekilde uygulamayı düşünün.

Hangi teknik seçerseniz, *AuthInfoRecPtr* ve *AuthInfoRecOffset* ' den yalnızca biri kullanılabilir; her ikisi de sıfırda, arama neden kodu MQRC\_AUTH\_INFO\_REC\_ERROR ile başarısız olur.

Bu bir giriş alanıdır. Bu alanın ilk değeri, göstergeleri destekleyen programlama dillerindeki boş değeri gösterir ve tersi durumda, boş değeri byte dizilimidir.

**Not:** Programlama dilinin işaretçi veri tipini desteklemediği altyapılarda, bu alan uygun uzunluğun byte dizisi olarak bildirilir.

### ***KeyResetCount (MQLONG)***

Bu, gizli anahtar yeniden anlaşılmadan önce bir TLS etkileşimi içinde gönderilen ve alınan şifrelenmemiş baytların toplam sayısını gösterir.

Bayt sayısı, MCA tarafından gönderilen denetim bilgilerini içerir.

1 bayt-32 KB aralığında bir TLS gizli anahtarı sıfırlama sayısı belirtirseniz, TLS kanalları 32 KB ' lik gizli anahtar sıfırlama sayısını kullanır. Bu, küçük TLS gizli anahtar ilk duruma getirme değerleri için oluşacağı aşırı anahtar sıfırlamalarının işlem maliyetinden kaçınmak içindir.

Bu bir giriş alanıdır. Değer, 0 ile 999 999 999 aralığındaki bir sayıdır ve varsayılan değer olarak 0 değerini içerir. Gizli anahtarların hiçbir zaman yeniden anlaşılmadığını belirtmek için 0 değerini kullanın.

### ***FipsRequired (MQLONG)***

IBM MQ , kullanılan şifreleme modüllerinin donanım ürünü tarafından sağlanacak şekilde şifreleme donanımıyla yapılandırılabilir; bunlar, kullanılmakta olan şifreleme donanım ürününe bağlı olarak belirli bir düzeyde FIPS onaylı olabilir. Şifreleme IBM MQ tarafından sağlanan yazılımda sağlanırsa yalnızca FIPS onaylı algoritmaların kullanılacağını belirtmek için bu alanı kullanın.

**Not:** AIX, Linux, and Windows işletim sistemlerinde IBM MQ , "IBM Crypto for C" şifreleme modülü aracılığıyla FIPS 140-2 uyumluluğu sağlar. Bu modüle ilişkin sertifika Geçmiş durumuna taşındı. Müşteriler, [IBM Crypto for C certificate](#) belgesini görüntüleyip NIST tarafından sağlanan tüm tavsiyeleri dikkate almalıdır. Yeni bir FIPS 140-3 modülü şu anda devam ediyor ve durumu [İşlem listesindeki NIST CMVP modüllerinde](#) aranarak görüntülenebilir.

IBM MQ kurulduğunda, FIPS onaylı bazı modüller sağlayan bir TLS şifreleme uygulaması da kurulur.

Değerler şunlar olabilir:

#### **MQSSL\_FIPS\_NO**

Bu varsayılan değerdir. Bu değere ayarlandığında:

- Belirli bir platformda desteklenen herhangi bir CipherSpec kullanılabilir.
- Şifreleme donanımı kullanılmadan çalıştırılırsa, CipherSpecs , IBM MQ platformlarında FIPS 140-2 sertifikalı şifreleme kullanılarak çalıştırılır.

FIPS onaylı CipherSpecs çizelgesinin bir listesi için [CipherSpecs](#) özelliğinin etkinleştirilmesi başlıklı konuda açıklanan çizelgeye bakın.

#### **MQSSL\_FIPS\_YES**

Bu değere ayarlandığında, şifrelemeyi gerçekleştirmek için şifreleme donanımı kullanmıyorsan,

- Bu istemci bağlantısına uygulanan CipherSpec içinde yalnızca FIPS onaylı şifreleme algoritmaları kullanılabilir.
- Belirli Şifre Belirtilimleri kullanılıyorsa, gelen ve giden TLS kanal bağlantıları yalnızca başarılı olur.

Ek bilgi için [CipherSpecs](#) özelliğinin etkinleştirilmesi başlıklı konuya bakın.

**Not:** Olanaklıysa, FIPS-only CipherSpecs yapılandırılırsa, MQI istemcisi MQRC\_SSL\_INITIALIZATION\_ERROR ile FIPS olmayan CipherSpec belirten bağlantıları reddeder. IBM MQ , bu tür bağlantıların tümünü reddetmeyi garanti etmez ve IBM MQ yapılandırmanızın FIPS uyumlu olup olmadığını belirlemek sizin sorumluluğunuzdadır.

### ***EncryptionPolicySuiteB(MQlong)***

Bu alan, Suite B uyumlu şifrelemesinin kullanılıp kullanılmadığını ve hangi güç düzeyinin çalıştırılacağını belirtir. Değer bir ya da daha fazla olabilir:

- MQ\_SUITE\_B\_NONE  
Suite B uyumlu şifreleme kullanılmıyor.
- MQ\_SUITE\_B\_128\_BIT  
B Suite 128 bit güvenlik düzeyi güvenliği kullanılır.
- MQ\_SUITE\_B\_192\_BIT  
B Grubu 192 bit güvenlik düzeyi güvenliği kullanılır.

**Not:** Bu alandaki diğer herhangi bir değerle MQ\_SUITE\_B\_NONE değerinin kullanılması geçersizdir.

### ***CertificateValİlkesi (MQUBE)***

Bu alan, kullanılacak sertifika doğrulama ilkesinin tipini belirtir. Alan, aşağıdaki değerlerden birine ayarlanabilir:

#### **MQ\_CERT\_VAL\_POLICY\_ANY**

Güvenli yuva kitaplığı tarafından desteklenen sertifika geçerlilik denetimi ilkelerinin her birini uygulayın. Sertifika zincirini geçerli olarak gören ilkelerden biri varsa, sertifika zincirini kabul edin.

#### **MQ\_CERT\_VAL\_POLICY\_RFC5280**

Yalnızca RFC5280 uyumlu sertifika geçerlilik denetimi ilkesini uygulayın. Bu ayar ANY ayarından daha katı geçerlilik denetimi sağlar, ancak bazı eski dijital sertifikaları reddeder.

Bu alanın ilk değeri MQ\_CERT\_VAL\_POLICY\_ANY olarak geçerli olur.



## **CertificateLabel (MQCHAR64)**

Bu alan, kullanılmakta olan sertifika etiketinin ayrıntılarını verir.

IBM MQ , *CertificateLabel* alanı için varsayılan değeri boşluk olarak başlatır.

Bu, yürütüm sırasında varsayılan değer olarak yorumlanır ve geriye doğru uyumludur.

For example, specifying a MQSCO version less than 5.0, or using the default value of blanks for the *CertificateLabel* field, uses the preexisting default value of `ibmwebsphereuser_id`.

## **MQSD-Abonelik tanımlayıcısı**

MQSD yapısı, yapılmakta olan abonelik ile ilgili ayrıntıları belirtmek için kullanılır. Yapı, MQSUB çağrısında bir giriş/çıkış değiştirgesidir. Daha fazla bilgi için bkz. [MQSUB kullanım notları](#).

### **Kullanılabilirlik**

MQSD yapısı aşağıdaki altyapılarda bulunur:

-  AIX
-  IBM i
-  Linux
-  Windows
-  z/OS

ve bu sistemlere bağlı IBM MQ MQI clients için.

### **Sürüm**

MQSD ' nin yürürlükteki sürümü: MQSD\_VERSION\_1.

### **Karakter kümesi ve kodlama**

MQSD ' deki veriler, **CodedCharSetId** kuyruk yöneticisi özniteliği tarafından belirtilen karakter kümesinde ve MQENC\_NATIVE tarafından verilen yerel kuyruk yöneticisinin kodlamasında yer almalıdır. Ancak, uygulama bir MQ MQI istemcisi olarak çalışıyorsa, yapı istemcinin karakter kümesinde ve kodlamasında olmalıdır.

### **Yönetilen abonelikler**

Bir uygulamanın belirli bir kuyruğu aboneliğiyle eşleşen yayınlar için hedef olarak kullanması gerekmiyorsa, yönetilen abonelik özelliğini kullanabilir. Bir uygulama yönetilen abonelik kullanmayı seçerse, kuyruk yöneticisi, MQSUB çağrısından çıkış olarak bir nesne tanıtcısı sağlayarak aboneyi yayınlanan iletilerin gönderildiği hedef hakkında bilgilendirir. Daha fazla bilgi için bkz. [Hobj \(MQHOBJ\)-giriş/çıkış](#).

Abonelik kaldırıldığında, kuyruk yöneticisi aşağıdaki durumlarda yönetilen hedeften alınmayan iletileri temizlemeyi de taahhüt eder:

- Abonelik kaldırıldığında-MQCO\_REMOVE\_SUB ile MQCLOSE kullanımı ile yönetilen Hobj kapatılır.
- Kalıcı olmayan bir abonelik (MQSO\_NON\_DURABLE) kullanılarak bir uygulamayla bağlantı kesildiğinde örtük olarak (MQSO\_NON\_DURABLE)
- Süresi dolduğu ve yönetilen Hobj kapatıldığı için bir abonelik kaldırıldığında süre bitimine göre.

Sürekli olmayan aboneliklerle yönetilen abonelikleri kullanmanız gerekir; böylece bu temizleme işlemi gerçekleştirilebilir ve böylece, kapalı kalıcı olmayan aboneliklere ilişkin iletiler kuyruk yöneticinizde yer kaplamaz. Sürekli abonelikler, yönetilen hedefleri de kullanabilir.

## Alanlar

**Not:** Aşağıdaki tabloda, alanlar alfabetik olarak değil, kullanıma göre gruplanır. Alt konular aynı sırayı izler.

Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
<u>StrucId</u> (yapı tanıtıcısı)	MQSD_STRUC_ID	'SD--'
<u>Sürüm</u> (yapı sürümü numarası)	MQSD_VERSION_1	1
<u>Seçenekler</u> (seçenekler)	MQSO_NON_DURABLE	0
<u>ObjectName</u> (nesne adı)	Yok	Boş değerli dizilim ya da boşluklar
<u>AlternateUserKimliği</u> (diğer kullanıcı kimliği)	Yok	Boş değerli dizilim ya da boşluklar
<u>AlternateSecurityTanıtıcısı</u> (diğer güvenlik tanıtıcısı)	MQSID_YOK	Boş Değerler
<u>SubExpiry</u> (abonelik süre bitimi)	MQEI_UNLIMITED	-1
<u>ObjectString</u> (nesne dizgisi)	Yok	MQCHARV için tanımlanan adlar ve değerler
<u>SubName</u> (abonelik adı)	Yok	MQCHARV için tanımlanan adlar ve değerler
<u>SubUserVerileri</u> (abonelik kullanıcı verileri)	Yok	MQCHARV için tanımlanan adlar ve değerler
<u>SubCorrelTanıtıcısı</u> (abonelik ilinti tanıtıcısı)	MQCI_NONE	Boş Değerler
<u>PubPriority</u> (yayın önceliği)	MQPRI_PRIORITY_AS_Q_DEF	-3
<u>PubAccountingBelirteci</u> (yayın muhasebesi simgesi)	MQACT_NONE	Boş Değerler
<u>PubAppIdentityData</u> (yayın uygulaması kimlik verileri)	Yok	Boş değerli dizilim ya da boşluklar
<u>SelectionString</u> (seçim ölçütlerini sağlayan dizgi)	Yok	MQCHARV için tanımlanan adlar ve değerler
<u>SubLevel</u> (abonelik düzeyi)	Yok	1
<u>ResObjectDizesi</u> (uzun nesne adı)	Yok	MQCHARV için tanımlanan adlar ve değerler

Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
<b>Notlar:</b>		
<ol style="list-style-type: none"> <li>1. – simgesi, tek bir boş karakteri temsil eder.</li> <li>2. Boş değerli dizilim ya da boşluk değeri, C dizgisindeki boş dizgiyi ve diğer programlama dillerindeki boş karakterleri gösterir.</li> <li>3. C programlama dilinde, makro değişkeniMQSD_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlar için başlangıç değerleri sağlamak üzere aşağıdaki şekilde kullanılabilir:</li> </ol>		
<pre>MQSD MySD = {MQSD_DEFAULT};</pre>		

## Dil bildirimleri

### MQSD için C bildirim

```
typedef struct tagMQSD MQSD;
struct tagMQSD {
    MQCHAR4    StrucId;           /* Structure identifier */
    MQLONG     Version;          /* Structure version number */
    MQLONG     Options;          /* Options associated with subscribing */
    MQCHAR48   ObjectName;       /* Object name */
    MQCHAR12   AlternateUserId;  /* Alternate user identifier */
    MQBYTE40   AlternateSecurityId; /* Alternate security identifier */
    MQLONG     SubExpiry;        /* Expiry of Subscription */
    MQCHARV    ObjectString;     /* Object Long name */
    MQCHARV    SubName;          /* Subscription name */
    MQCHARV    SubUserData;      /* Subscription User data */
    MQBYTE24   SubCorrelId;      /* Correlation Id related to this subscription */
    MQLONG     PubPriority;       /* Priority set in publications */
    MQBYTE32   PubAccountingToken; /* Accounting Token set in publications */
    MQCHAR32   PubApplIdentityData; /* Appl Identity Data set in publications */
    MQCHARV    SelectionString;  /* Message selector structure */
    MQLONG     SubLevel;         /* Subscription level */
    MQCHARV    ResObjectString;  /* Resolved Long object name*/
    /* Ver:1 */
};
```

### MQSD için COBOL bildirim

```
** Address of variable length string
20 MQSD-OBJECTSTRING-VSPTR          POINTER.
** Offset of variable length string
20 MQSD-OBJECTSTRING-VSOFFSET       PIC S9(9) BINARY.
** size of buffer
20 MQSD-OBJECTSTRING-VSBUFSIZE      PIC S9(9) BINARY.
** Length of variable length string
20 MQSD-OBJECTSTRING-VSLENGTH       PIC S9(9) BINARY.
** CCSID of variable length string
20 MQSD-OBJECTSTRING-VSCCSID        PIC S9(9) BINARY.
** Subscription name
15 MQSD-SUBNAME.
** Address of variable length string
20 MQSD-SUBNAME-VSPTR              POINTER.
** Offset of variable length string
20 MQSD-SUBNAME-VSOFFSET           PIC S9(9) BINARY.
** size of buffer
20 MQSD-SUBNAME-VSBUFSIZE          PIC S9(9) BINARY.
** Length of variable length string
20 MQSD-SUBNAME-VSLENGTH           PIC S9(9) BINARY.
** CCSID of variable length string
20 MQSD-SUBNAME-VSCCSID            PIC S9(9) BINARY.
** Subscription User data
15 MQSD-SUBUSERDATA.
** Address of variable length string
20 MQSD-SUBUSERDATA-VSPTR          POINTER.
** Offset of variable length string
20 MQSD-SUBUSERDATA-VSOFFSET       PIC S9(9) BINARY.
```

```

** size of buffer
20 MQSD-SUBUSERDATA-VSBUFSIZE PIC S9(9) BINARY.
** Length of variable length string
20 MQSD-SUBUSERDATA-VSLENGTH PIC S9(9) BINARY.
** CCSID of variable length string
20 MQSD-SUBUSERDATA-VSCCSID PIC S9(9) BINARY.
** Correlation Id related to this subscription
15 MQSD-SUBCORRELID PIC X(24).
** Priority set in publications
15 MQSD-PUBPRIORITY PIC S9(9) BINARY.
** Accounting Token set in publications
15 MQSD-PUBACCOUNTINGTOKEN PIC X(32).
** Appl Identity Data set in publications
15 MQSD-PUBAPPLIDENTITYDATA PIC X(32).
** Message Selector
15 MQSD-SELECTIONSTRING.
** Address of variable length string
20 MQSD-SELECTIONSTRING-VSPTR POINTER.
** Offset of variable length string
20 MQSD-SELECTIONSTRING-VSOFFSET PIC S9(9) BINARY.
** size of buffer
20 MQSD-SELECTIONSTRING-VSBUFSIZE PIC S9(9) BINARY.
** Length of variable length string
20 MQSD-SELECTIONSTRING-VSLENGTH PIC S9(9) BINARY.
** CCSID of variable length string
20 MQSD-SELECTIONSTRING-VSCCSID PIC S9(9) BINARY.
** Selection criteria
20 MQSD-SELECTIONSTRING-SUBLEVEL PIC S9(9) BINARY.
** Long object name
20 MQSD-SELECTIONSTRING-RESOBJSTRING PIC S9(9) BINARY.

```

## MQSD için PL/I bildirimi

```

dcl
1 MQSD based,
3 StructId char(4), /* Structure identifier */
3 Version fixed bin(31), /* Structure version number */
3 Options fixed bin(31), /* Options associated with subscribing */
3 ObjectName char(48), /* Object name */
3 AlternateUserId char(12), /* Alternate user identifier */
3 AlternateSecurityId char(40), /* Alternate security identifier */
3 SubExpiry fixed bin(31), /* Expiry of Subscription */
3 ObjectString, /* Object Long name */
5 VSPtr pointer, /* Address of variable length string */
5 VSOffset fixed bin(31), /* Offset of variable length string */
5 VSBufSize fixed bin(31), /* size of buffer */
5 VSLength fixed bin(31), /* Length of variable length string */
5 VSCCSID fixed bin(31); /* CCSID of variable length string */
3 SubName, /* Subscription name */
5 VSPtr pointer, /* Address of variable length string */
5 VSOffset fixed bin(31), /* Offset of variable length string */
5 VSBufSize fixed bin(31), /* size of buffer */
5 VSLength fixed bin(31), /* Length of variable length string */
5 VSCCSID fixed bin(31); /* CCSID of variable length string */
3 SubUserData, /* Subscription User data */
5 VSPtr pointer, /* Address of variable length string */
5 VSOffset fixed bin(31), /* Offset of variable length string */
5 VSBufSize fixed bin(31), /* size of buffer */
5 VSLength fixed bin(31), /* Length of variable length string */
5 VSCCSID fixed bin(31), /* CCSID of variable length string */
3 SubCorrelId char(24), /* Correlation Id related to this subscription */
3 PubPriority fixed bin(31), /* Priority set in publications */
3 PubAccountingToken char(32), /* Accounting Token set in publications */
3 PubApplIdentityData char(32), /* Appl Identity Data set in publications */
3 SelectionString, /* Message Selection */
5 VSPtr pointer, /* Address of variable length string */
5 VSOffset fixed bin(31), /* Offset of variable length string */
5 VSBufSize fixed bin(31), /* size of buffer */
5 VSLength fixed bin(31), /* Length of variable length string */
5 VSCCSID fixed bin(31), /* CCSID of variable length string */
3 SubLevel fixed bin(31), /* Subscription level */
3 ResObjectString, /* Resolved Long object name */
5 VSPtr pointer, /* Address of variable length string */
5 VSOffset fixed bin(31), /* Offset of variable length string */
5 VSBufSize fixed bin(31), /* size of buffer */
5 VSLength fixed bin(31), /* Length of variable length string */
5 VSCCSID fixed bin(31); /* CCSID of variable length string */

```

## MQSD için High Level Assembler (Üst Düzey Çevirici) bildirimi

```
MQSD          DSECT
MQSD_STRUCID  DS CL4 Structure identifier
MQSD_VERSION  DS F Structure version number
MQSD-OPTIONS  DS F Options associated with subscribing
MQSD_OBJECTNAME DS CL48 Object name
MQSD_ALTERNATEUSERID DS CL12 Alternate user identifier
MQSD_ALTERNATESECURITYID DS CL40 Alternate security identifier
MQSD_SUBEXPIRY DS F Expiry of Subscription
MQSD_OBJECTSTRING DS 0F Object Long name
MQSD_OBJECTSTRING_VSPTR DS F Address of variable length string
MQSD_OBJECTSTRING_VSOFFSET DS F Offset of variable length string
MQSD_OBJECTSTRING_VSBUFSIZE DS F size of buffer
MQSD_OBJECTSTRING_VSLENGTH DS F Length of variable length string
MQSD_OBJECTSTRING_VSCCSID DS F CCSID of variable length string
MQSD_OBJECTSTRING_LENGTH EQU *-MQSD_OBJECTSTRING
ORG MQSD_OBJECTSTRING
MQSD_OBJECTSTRING_AREA DS CL(MQSD_OBJECTSTRING_LENGTH)
*
MQSD_SUBNAME DS 0F Subscription name
MQSD_SUBNAME_VSPTR DS F Address of variable length string
MQSD_SUBNAME_VSOFFSET DS F Offset of variable length string
MQSD_SUBNAME_VSBUFSIZE DS F size of buffer
MQSD_SUBNAME_VSLENGTH DS F Length of variable length string
MQSD_SUBNAME_VSCCSID DS F CCSID of variable length string
MQSD_SUBNAME_LENGTH EQU *-MQSD_SUBNAME
ORG MQSD_SUBNAME
MQSD_SUBNAME_AREA DS CL(MQSD_SUBNAME_LENGTH)
*
MQSD_SUBUSERDATA DS 0F Subscription User data
MQSD_SUBUSERDATA_VSPTR DS F Address of variable length string
MQSD_SUBUSERDATA_VSOFFSET DS F Offset of variable length string
MQSD_SUBUSERDATA_VSBUFSIZE DS F size of buffer
MQSD_SUBUSERDATA_VSLENGTH DS F Length of variable length string
MQSD_SUBUSERDATA_VSCCSID DS F CCSID of variable length string
MQSD_SUBUSERDATA_LENGTH EQU *-MQSD_SUBUSERDATA
ORG MQSD_SUBUSERDATA
MQSD_SUBUSERDATA_AREA DS CL(MQSD_SUBUSERDATA_LENGTH)
*
MQSD_SUBCORRELID DS CL24 Correlation Id related to this subscription
MQSD_PUBPRIORITY DS F Priority set in publications
MQSD_PUBACCOUNTINGTOKEN DS CL32 Accounting Token set in publications
MQSD_PUBAPPLIDENTITYDATA DS CL32 Appl Identity Data set in publications
*
MQSD_SELECTIONSTRING DS F Message Selector
MQSD_SELECTIONSTRING_VSPTR DS F Address of variable length string
MQSD_SELECTIONSTRING_VSOFFSET DS F Offset of variable length string
MQSD_SELECTIONSTRING_VSBUFSIZE DS F size of buffer
MQSD_SELECTIONSTRING_VSLENGTH DS F Length of variable length string
MQSD_SELECTIONSTRING_VSCCSID DS F CCSID of variable length string
MQSD_SELECTIONSTRING_LENGTH EQU *- MQSD_SELECTIONSTRING
ORG MQSD_SELECTIONSTRING
MQSD_SELECTIONSTRING_AREA DS CL(MQSD_SELECTIONSTRING_LENGTH)
*
MQSD-SUBLEVEL DS F Subscription level
*
MQSD_RESOBJECTSTRING DS F Resolved Long object name
MQSD_RESOBJECTSTRING_VSPTR DS F Address of variable length string
MQSD_RESOBJECTSTRING_VSOFFSET DS F Offset of variable length string
MQSD_RESOBJECTSTRING_VSBUFSIZE DS F size of buffer
MQSD_RESOBJECTSTRING_VSLENGTH DS F Length of variable length string
MQSD_RESOBJECTSTRING_VSCCSID DS F CCSID of variable length string
MQSD_RESOBJECTSTRING_LENGTH EQU *- MQSD_RESOBJECTSTRING
ORG MQSD_RESOBJECTSTRING
MQSD_RESOBJECTSTRING_AREA DS CL(MQSD_RESOBJECTSTRING_LENGTH)
*
MQSD_LENGTH EQU *-MQSD
ORG MQSD
MQSD_AREA DS CL(MQSD_LENGTH)
```

### **StrucId (MQCHAR4)**

Bu, yapı tanıttıcısıdır; değer şöyle olmalıdır:

#### **MQSD\_STRUC\_ID**

Abonelik Tanımlayıcısı yapısına ilişkin tanıttıcı.

C programlama dili için, MQSD\_STRUC\_ID\_ARRAY değişmezi de tanımlanır; bu, MQSD\_STRUC\_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisidir.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQSD\_STRUC\_ID 'dir.

### **Sürüm (MQUZE)**

Bu, yapı sürüm numarasıdır; değer şu olmalıdır:

#### **MQSD\_VERSION\_1**

Version-1 Abonelik Tanımlayıcısı yapısı.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

#### **MQSD\_CURRENT\_VERSION**

Abonelik Tanımlayıcısı yapısının yürürlükteki sürümü.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQSD\_VERSION\_1' dir.

### **Seçenekler (MQUZE)**

Bu, MQSUB çağrısının işleminin denetlenmesini sağlar.

Aşağıdaki seçeneklerden en az birini belirlemelisiniz:

- MQSO ALTER
- MQSO RESUME
- MQSO CREATE

birden çok seçenek belirtmek için, değerleri bir araya ekleyin (aynı değişmezi bir kereden fazla eklemeyin) ya da bit ya da işlemi kullanarak değerleri birleştirin (programlama dili, bit işlemleri destekliyse).

Geçerli olmayan birleşimler bu konuda belirtilir; diğer birleşimler geçerli olur.

**Erişim ya da yaratma seçenekleri:** Erişim ve yaratma seçenekleri, bir aboneliğin oluşturulup oluşturulmadığını ya da var olan bir aboneliğin döndürülüp değiştirilmediğini ya da değiştirilip değiştirilmediğini denetler. Bu seçeneklerden en az birini belirtmeniz gerekir.

<i>Çizelge 527. Geçerli erişim ve yaratma seçenekleri birleşimleri</i>	
<b>Seçenek bileşimi</b>	<b>Notlar</b>
MQSO_CREATE	Var olmayan bir abonelik yaratır. Abonelik varsa bu birleşim başarısız olur.
MQSO_RESUME	Var olan bir aboneliğe devam eder. Abonelik yoksa bu birleşim başarısız olur.
MQSO_CREATE + MQSO_RESUME	Bir abonelik yaratır (varsa) ve varsa eşleşen bir aboneliği sürdürür. Bu birleşim, birkaç kez çalıştırılan bir uygulamada kullanılırsa kullanışlıdır.
MQSO ALTER (nota bakın)	Var olan bir aboneliğe devam eder ve MQSD ' de belirtilen alanlarla eşleşecek şekilde değiştirin. Abonelik yoksa bu birleşim başarısız olur.
MQSO_CREATE + MQSO ALTER (nota bakın)	Bir abonelik yaratır ve varsa, eşleşen bir abonelik yaratır (varsa), herhangi bir alanı MQSD ' de belirtilen ile eşleşecek şekilde değiştirin. Bu birleşim, devam etmeden önce aboneliğinin belirli bir durumda olduğundan emin olmak isteyen bir uygulamada kullanıldığında yararlı bir birleşim olur.

**Not:**

MQSO ALTER belirtiminin belirtilmesine ilişkin seçenekler MQSO\_RESUME değerini de belirtebilir, ancak bu birleşimin yalnızca MQSO ALTER belirtimini tek başına belirten bir başka etkisi yoktur. MQSO ALTER, MQSO\_RESUME ögesini belirtir; bir aboneliği değiştirmesi için MQSUB çağrılması, aboneliğin de sürdürüleceğini belirtir. Bunun tersi doğru değil, ancak bir aboneliğin sürdürülmesi, bunun değiştirileceği anlamına da gelmez.

## MQSO\_CREATE

Belirtilen konu için yeni bir abonelik yaratın. Aynı *SubName* ' u kullanan bir abonelik varsa, çağrı MQRC\_AL\_ALREADY\_EXISTS ile başarısız olur. MQSO\_CREATE seçeneği MQSO\_RESUME ile birleştirilerek bu hata önlenir. *SubName* her zaman gerekli değildir. Daha fazla ayrıntı için, bu alanın açıklamasına bakın.

MQSO\_RESUME ile MQSO\_CREATE birleştirilmesi, belirtilen *SubName* için önceden var olan bir aboneliğin tanıtıcısını döndürür; ancak, varolan bir abonelik yoksa, MQSD ' de sağlanan tüm alanlar kullanılarak yeni bir abonelik yaratılır.

MQSO\_CREATE işlemi de aynı etkiyi MQSO ALTER ile birlikte birleştirebilir.

## MQSO\_RESUME

Return a handle to a pre-existing subscription which matches that specified by *SubName*. Eşleşen abonelikler özniteliklerinde hiçbir değişiklik yapılmadı ve bu öznitelikler MQSD yapısındaki çıkışa döndürülebilir. Yalnızca şu MQSD alanları kullanılır: StrucId, Version, Options, AlternateUserId ve AlternateSecurityId, SubName.

Bir abonelik tam abonelik adıyla eşleşmiyorsa, çağrıya neden kodu MQRC\_NO\_XX\_ENCODE\_CASE\_ONE subscription ile başarısız olur. MQSO\_CREATE seçeneği MQSO\_RESUME ile birleştirilerek bu hata önlenir.

Aboneliğin kullanıcı kimliği, aboneliği yaratan kullanıcı kimliğidir ya da daha sonra farklı bir kullanıcı kimliği tarafından değiştirilmişse, en son başarılı değişikliklerin kullanıcı kimliğidir. Bir AlternateUser tanıtıcısı kullanılırsa ve bu kullanıcı için diğer kullanıcı kimliklerinin kullanılmasına izin verilirse, diğer kullanıcı kimliği, aboneliğin yapıldığı kullanıcı kimliği yerine aboneliği yaratan kullanıcı kimliği olarak kaydedilir.

MQSO\_ANY\_USERID seçeneği olmayan bir eşleşen abonelik varsa ve aboneliğin kullanıcı kimliği, abonelik ile ilgili bir tanıtıcı isteyen uygulamanın kullanıcı kimliği farklıysa, MQRC\_IDENTITY\_MISMATCH neden koduyla başarısız olur.

Eşleşen bir abonelik varsa ve şu anda kullanımda olan bir abonelik varsa, çağrı MQRC\_XX\_ENCODE\_CASE\_ONE subscription\_in\_use ile başarısız olur.

SubName (Alt Ad) adlı abonelik, devam etmek ya da bir uygulamadaki değişiklik yapmak için geçerli bir abonelik değilse, çağrı MQRC\_INVALID\_SUBSCRIPTION ile başarısız olur.

MQSO\_RESUME, MQSO ALTER tarafından örtük olarak, bu seçeneği bu seçenekle birleştirmenize gerek kalmadığını belirtmektedir. Ancak, iki seçeneğin birleştirilmesi bir hataya neden olmaz.

## MQSO ALTER

Return a handle to a pre-existing subscription with the full subscription name matching that specified by the name in *SubName*. Bu öznitelik için değiştirilmeye izin verilmediği sürece, MQSD ' de belirtilen abonelikten farklı olan aboneliğin herhangi bir özneliği, abonelikte değiştirilir. Ayrıntılar, her özneliğin açıklamasında belirtilir ve aşağıdaki tabloda özetlenir. Değiştirilemeyecek bir özneliği değiştirmeye ya da MQSO\_IMMUTABLE seçeneğini ayarlayan bir aboneliği değiştirmeye çalışırsanız, çağrı aşağıdaki çizelgede gösterilen neden koduyla başarısız olur.

Tam abonelik adıyla eşleşen bir abonelik yoksa, çağrıya neden kodu MQRC\_NO\_XX\_ENCODE\_CASE\_ONE subscription ile başarısız olur. Bu başarısızlığı önlemek için, MQSO\_CREATE seçeneğini MQSO ALTER ile birleştirerek bu başarısızlığı önleyebilirsiniz.

MQSO\_CREATE ile MQSO\_CREATE birleştirilmesi, belirtilen *SubName* için önceden var olan bir aboneliğin tanıtıcısını döndürür; ancak, varolan bir abonelik yoksa, MQSD ' de sağlanan tüm alanlar kullanılarak yeni bir abonelik yaratılır.

Aboneliğin kullanıcı kimliği, aboneliği yaratan kullanıcı kimliğidir ya da daha sonra farklı bir kullanıcı kimliği tarafından değiştirilmişse, en son, başarılı değişikliklerin kullanıcı kimliğidir. Bir AlternateUserTanıtıcısı kullanılıyorsa ve o kullanıcı için diğer kullanıcı kimliklerinin kullanılmasına izin verilirse, diğer kullanıcı kimliği, aboneliğin yapıldığı kullanıcı kimliği yerine aboneliği yaratan kullanıcı kimliği olarak kaydedilir.

MQSO\_ANY\_USERID seçeneği olmadan yaratılmış bir abonelik varsa ve aboneliğin kullanıcı kimliği, abonelik ile ilgili bir tanıtıcı isteyen uygulamanın kullanıcı kimliği farklıysa, MQRC\_IDENTITY\_MISMATCH neden koduyla başarısız olur.

Eşleşen bir abonelik varsa ve şu anda kullanımda olan bir abonelik varsa, çağrı MQRC\_XX\_ENCODE\_CASE\_ONE subscription\_in\_use ile başarısız olur.

SubName (Alt Ad) adlı abonelik, devam etmek ya da bir uygulamadaki değişiklik yapmak için geçerli bir abonelik değilse, çağrı MQRC\_INVALID\_SUBSCRIPTION ile başarısız olur.

Aşağıdaki çizelge, MQSO ALTER 'ın MQSD ve MQSUB' da öznelik değerlerini değiştirebilme yeteneğini göstermektedir.

<i>Çizelge 528. MQSD ve MQSUB ' da değiştirilebilen öznelikler</i>			
Veri tipi tanımlayıcısı ya da işlev çağırısı	Alan adı	Bu öznelik MQSO ALTER kullanılarak değiştirilebilir mi?	Neden kodu
MQSD	Dayanıklılık seçenekleri	Hayır	MQRC_DURABILITY_NOT_ALTERABLE
MQSD	Hedef Seçenekleri	Evet	Yok
MQSD	Kayıt seçenekleri	Evet (bkz. not "1" sayfa 564)	MQRC_GROUPING_NOT_ALTERABLE, MQSO_GROUP_SUB değiştirmeyi denesiyor
MQSD	Yayın seçenekleri	Evet (bkz. not "2" sayfa 564)	Yok
MQSD	Joker seçenekleri	Hayır	MQRC_TOPIC_NOT_ALTERABLE
MQSD	Diğer seçenekler	Hayır (bkz. not "3" sayfa 564)	Yok
MQSD	ObjectName	Hayır	MQRC_TOPIC_NOT_ALTERABLE
MQSD	AlternateUserTanıtıcısı	Hayır (bkz. not "4" sayfa 565)	Yok
MQSD	AlternateSecurityTanıtıcısı	Hayır (bkz. not "4" sayfa 565)	Yok
MQSD	SubExpiry	Evet	Yok
MQSD	ObjectString	Hayır	MQRC_TOPIC_NOT_ALTERABLE
MQSD	SubName	Hayır (bkz. not "5" sayfa 565)	Yok
MQSD	SubUserVerileri	Evet	Yok
MQSD	SubCorrelTanıtıcısı	Evet (bkz. not "6" sayfa 565)	Gruplanmış bir abonelikte MQRC_GROUPING_NOT_ALTERABLE
MQSD	PubPriority	Evet	Yok
MQSD	PubAccountingSimgesi	Evet	Yok
MQSD	PubApplIdentityData	Evet	Yok
MQSD	SubLevel	Hayır	MQRC_XX_ENCODE_CASE_ONE sublevel_not_alterable
MQSUB	Hobj	Evet (bkz. not "6" sayfa 565)	Gruplanmış bir abonelikte MQRC_GROUPING_NOT_ALTERABLE

#### Notlar:

1. MQSO\_GROUP\_SUB değiştirilemiyor.
2. MQSO\_NEW\_PUBLICATIONS\_ONLY, aboneliğin bir parçası olmadığı için değiştirilemiyor
3. Bu seçenekler aboneliğin bir parçası değil



4. Bu öznitelik, aboneliğin bir parçası değil
5. Bu öznitelik, değiştirilmekte olan aboneliğin kimliğidir.
6. Gruplanmış bir alt (MQSO\_GROUP\_SUB) bir kısmının dışında değiştirilebilir

**Dayanıklılık seçenekleri:** Aşağıdaki seçenekler aboneliğin ne kadar dayanıklı olduğunu denetler. Bu seçeneklerden yalnızca birini belirleyebilirsiniz. Varolan bir aboneliği MQSO ALTER seçeneğini kullanarak değiştiriyorsanız, aboneliğin dayanıklılığını değiştiremezsiniz. MQSO\_RESUME komutunu kullanarak bir MQSUB çağrısından geri dönerek, uygun dayanıklılık seçeneği ayarlanır.

#### **MQSO\_DAYANIKLI**

Bu konuya ilişkin aboneliğin, MQCO\_REMOVE\_SUB seçeneği ile MQCLOSE kullanılarak belirtik olarak kaldırılincaya kadar devam etmesi için istekte bulunmayı isteyin. Bu abonelik belirtik olarak kaldırılmamışsa, kuyruk yöneticisine yapılan uygulamalar kapatıldıktan sonra da bu abonelik kalır.

Kalıcı aboneliklere izin vermemek olarak tanımlanmış bir konuya kalıcı abonelik isteniyorsa, çağrı MQRC\_DURABILITY\_NOT\_ALLOWALIZE ile başarısız olur.

#### **MQSO\_NON\_DAYANIKLI**

Önceden belirtik olarak kaldırılmamışsa, kuyruk yöneticisine uygulama bağlantısı kapatıldığında, bu konuya ilişkin aboneliğin kaldırıldığını isteyin. MQSO\_NON\_DAYANIKLI, MQSO\_DAYANIKLI seçeneğinin tersidir ve program belgelerine yardımcı olmak için tanımlanır. İkisi de belirtilmediyse, bu varsayılan değerdir.

**Hedef seçenekleri:** Aşağıdaki seçenek, abone olunan bir konuya ilişkin yayınların gönderileceği hedefi denetler. Var olan bir aboneliği MQSO ALTER seçeneğini kullanarak değiştiriyorsa, aboneliğe ilişkin yayınlar için kullanılan hedef değiştirilebilir. MQSO\_RESUME komutunu kullanarak bir MQSUB çağrısından dönüşte bu seçenek ayarlıdır (uygun).

#### **MQSO\_MANAGED**

Yayınların gönderileceği hedefin kuyruk yöneticisi tarafından yönetildiğinden emin olun.

*Hobj* içinde döndürülen nesne tanıtıcısı, bir kuyruk yöneticisi tarafından yönetilen kuyruğu gösterir ve sonraki MQGET, MQCB, MQINQ ya da MQCLOSE çağrılarıyla kullanılmak üzere.

MQSO\_Managed belirtilmediğinde **Hobj** değiştirilmesinde önceki bir MQSUB çağrısından döndürülen bir nesne tanıtıcısı sağlanamaz.

#### **MQSO\_NO\_MULTICAST**

Yayınların gönderileceği hedefin, çoklu yayın grubu adresi olarak değil, hedef olarak gönderileceğini isteyin. Bu seçenek yalnızca MQSO\_MANAGED seçeneği ile birleştirildiğinde geçerlidir. **Hobj** parametresinde bir kuyruk tanıtıcısı belirtildiğinde, bu abonelik için çoklu yayın kullanılamaz ve bu seçenek geçerli değildir.

Konu yalnızca çoklu yayın aboneliklerine izin vermek için tanımlandıysa, MCAST (ONLY) ayarını kullanarak çağrı başarısız olur; bu durumda çağrı, MQRC\_MULTICAST\_REQUIREMOND neden kodunda başarısız olur.

**Kapsam Seçeneği:** Aşağıdaki seçenek, yapılmakta olan aboneliğin kapsamını denetler. MQSO ALTER seçeneğini kullanarak var olan bir aboneliğin değiştirilmesi durumunda, bu abonelik kapsamı seçeneği değiştirilemez. MQSO-RESUME komutunu kullanarak bir MQSUB çağrısından dönerek, uygun kapsam seçeneği belirlenmiş olur.

#### **MQSO\_SCOPE\_QMGR**

Bu abonelik yalnızca yerel kuyruk yöneticisinde yapılır. Ağdaki diğer kuyruk yöneticilerine herhangi bir yetkili abonelik aboneliği dağıtılmadı. Bu aboneye yalnızca, bu kuyruk yöneticisinde yayınlanan yayınlar gönderilir. Bu, SUBSCOPE konu özniteliğini kullanarak herhangi bir davranış kümesini geçersiz kılar.

**Not:** Ayarlanmazsa, abonelik kapsamı SUBSCOPE konu özniteliği tarafından belirlenir.

**Kayıt seçenekleri:** Aşağıdaki seçenekler, bu abonelik için kuyruk yöneticisine yapılan kayda ilişkin ayrıntıları denetler. Var olan bir aboneliğin MQSO ALTER seçeneğini kullanarak değiştirilmesi durumunda,

bu kayıt seçenekleri değiştirilebilir. MQSO\_RESUME komutunu kullanarak MQSUB çağrısından dönüşte uygun kayıt seçenekleri ayarlanır.

## **MQSO\_GROUP\_SUB**

Bu abonelik, aynı kuyruğu kullanarak aynı SubLevel ' in diğer abonelikleriyle gruplanacaktır ve aynı ilinti tanıtıcısını belirterek, herhangi bir yayınların, kullanılmakta olan konu dizgilerinin çakışması nedeniyle, abonelikler grubuna birden fazla yayın iletisi sağlanmasına neden olacak şekilde gruplandırılması, yalnızca bir iletinin kuyruğa gönderilmesine neden olur. Bu seçenek kullanılmıyorsa, her bir benzersiz abonelik ( SubNameile tanımlanır), bu yayının bir kopyası ile sağlanır. Bu, yayının birden fazla kopyasının, bir dizi abonelikte paylaşılan kuyruğa yerleştirilebilir.

Yayının bir kopyasıyla yalnızca gruptaki en önemli abonelik sağlanır. En önemli abonelik, bir genel arama karakterinin bulunduğu noktaya kadar Tam konu adını temel alır. Grup içinde genel arama karakterlerinin bir karışımı kullanılırsa, yalnızca genel arama karakterinin konumu önemlidir. Aynı kuyruğu paylaşan bir abonelikler grubunda farklı joker şemaları birleştirmemeniz önerilir.

Yeni gruplanmış bir abonelik yaratırken, yine de benzersiz bir SubNameolmalıdır; ancak grupta var olan bir aboneliğin tam konu adıyla eşleşirse, çağrı MQRC\_DUPLICATE\_GROUP\_SUB ile başarısız olur.

Gruptaki en önemli abonelik, aynı zamanda MQSO\_NOT\_OWN\_PUBS belirtirse ve bu, aynı uygulamadan bir yayınıysa, kuyruğa yayın teslim edilmez.

Bu seçenekle yapılan bir aboneliği değiştirirken, gruplamayı, MQSUB çağrısındaki Hobj 'ı (kuyruk ve kuyruk yöneticisi adını temsil eden) ve SubCorrel(Alt Korel) tanıtıcısını gösteren alanlar değiştirilemez. Bu değişikliklerin değiştirilmeye çalışılması, MQRC\_GROUPING\_NOT\_ALTERNATEABLE çağrısının başarısız olmasına neden olur.

Bu seçenek, MQCI\_NONE değerine ayarlanmamış bir SubCorrelTanıtıcısıyla MQSO\_SET\_COREL\_ID ile birleştirilmelidir ve MQSO\_YANED ile birleştirilemez.

## **MQSO\_ANY\_USERID**

MQSO\_ANY\_USERID belirtildiğinde, abonenin kimliği tek bir kullanıcı kimliğiyle sınırlı değildir. Bu, herhangi bir kullanıcının uygun yetkiye sahip olduğunda aboneliği değiştirmesine ya da sürdürmesine olanak sağlar. Aboneliğin herhangi bir zamanda yalnızca tek bir kullanıcı tarafından olması gerekir. Şu anda başka bir uygulama tarafından kullanılmakta olan bir aboneliğin kullanımını sürdürme girişimi, çağrının MQRC\_XX\_ENCODE\_CASE\_ONE subscription\_in\_use ile başarısız olmasına neden olur.

Bu seçeneği var olan bir aboneliğe eklemek için, MQSOB çağrısının (MQSO\_ALTER kullanılarak), özgün aboneliğin kendisiyle aynı kullanıcı kimliğiyle gelmesi gerekir.

Bir MQSUB çağrısı, MQSO\_ANY\_USERID ayarına sahip var olan bir aboneliğe başvuruyorsa ve kullanıcı kimliği özgün abonelikten farklıysa, çağrı yalnızca yeni kullanıcı kimliğinin konuya abone olma yetkisi varsa başarılı olur. Başarıyla tamamlandığında, bu aboneye gelecek yayınlar, yayın iletisinde ayarlanan yeni kullanıcı kimliği ile aboneler kuyruğuna konabilir.

Hem MQSO\_ANY\_USERID, hem de MQSO\_FIXED\_USERID değerini belirtmeyin. İkisi de belirlenmezse, varsayılan değer MQSO\_FIXED\_USERID 'dir.

## **MQSO\_FIXED\_USERID**

MQSO\_FIXED\_USERID değeri belirtildiğinde, abonelik değiştirilebilmesi için yalnızca son kullanıcı kimliği tarafından değiştirilebilir ya da sürdürülür. Abonelik değiştirilmediyse, aboneliği yaratan kullanıcı kimliğidir.

Bir MQSUB komutu MQSO\_ANY\_USERID kümesiyle var olan bir aboneliğe başvuruyorsa ve MQSO\_FIXED\_USERID seçeneğini kullanmak için MQSO\_ALTER aboneliğini değiştirirse, aboneliğin kullanıcı kimliği şu anda bu yeni kullanıcı kimlide düzeltilir. Bu çağrı, yalnızca yeni kullanıcı kimliğinin konuya abone olma yetkisi varsa başarılı olur.

Abonelik sahibi olarak kaydedilen kullanıcı kimliği dışındaki bir kullanıcı kimliği MQSO\_FIXED\_USERID aboneliğini sürdürmeyi ya da değiştirmeyi denerse, çağrı MQRC\_IDENTITY\_MISMATCH ile başarısız olur. Bir aboneliğin sahibi olan kullanıcı kimliği, DISPLAY SBSTATUS komutu kullanılarak görüntülenebilir.

Hem MQSO\_ANY\_USERID, hem de MQSO\_FIXED\_USERID deęerini belirtmeyin. İki de belirlenmezse, varsayılan deęer MQSO\_FIXED\_USERID 'dir.

**Yayın seenekleri:** Ařaęıdaki seenekler, yayınların bu aboneye gnderilmesine yol ama řeklini denetler. Var olan bir abonelięin MQSO\_ALTER seeneęini kullanarak deęiřtirilmesi durumunda, bu yayın seenekleri deęiřtirilebilir.

#### **MQSO\_NOT\_OWN\_PUBS**

Aracıya uygulamanın kendi yayınlarından herhangi birini grmek istemedięini belirtir. Baęlantı tanıtıcıları aynıysa, yayınların aynı uygulamadan kaynaklandığı dřnlenir. MQSO\_RESUME komutunu kullanarak bir MQSUB aęrısından dnřte bu seenek ayarlıdır (uygun).

#### **MQSO\_NEW\_PUBLICATIONS\_ONLY**

Bu abonelik oluřturulduęunda, yalnızca yeni yayınlar gnderildięinde, yrrlkte tutulan yayınların gnderilmemesine neden olmaz. Bu seenek yalnızca, MQSO\_CREATE belirtildięinde geerlidir. Abonelikte yapılacak her trl deęiřiklik, yayınların akıřını deęiřtirmez ve bu nedenle bir konuda tutulan yayınlar, aboneye yeni yayınlar olarak gnderilecektir.

Bu seenek MQSO\_CREATE olmadan belirtilirse, aęrı MQRC\_OPTIONS\_ERROR ile bařarısız olur. MQSO\_RESUME komutunu kullanarak bir MQSUB aęrısından dnřte, bu seenek bu seenek kullanılarak yaratıldıysa bile bu seenek belirlenmez.

Bu seenek kullanılmıyorsa, nceden saklanan iletiler, saęlanan hedef kuyruęa gnderilir. Bu iřlem bir hata nedeniyle bařarısız olursa, MQRC\_RETAINED\_MSG\_Q\_ERROR ya da MQRC\_RETAINED\_NOT\_TESLIM edilemedi, abonelięin yaratılması bařarısız olur.

#### **MQSO\_PUBLICATIONS\_ON\_REQUEST**

Bu seeneęin belirlenmesi, abonenin gerektięinde zellikle bilgi isteyeceęini belirtir. Kuyruk yneticisi aboneye istenmeyen iletiler gndermez. nceki bir MQSUB aęrısından Hsub tanıtıcısı kullanılarak bir MQSUBRQ aęrısı yapıldığında, alıkonan yayın (ya da konu iinde bir genel arama karakteri belirtilirse, byk olasılıkla birden ok yayın) aboneye gnderilir. Bu seeneęi kullanarak MQSUB aęrısının sonucu olarak hibir yayın gnderilmez. MQSO\_RESUME komutunu kullanarak bir MQSUB aęrısından dnřte bu seenek ayarlıdır (uygun).

Bu seenek, 1 'den byk bir SubLevel ile birlikte geerli deęildir.

**İleriye okuma seenekleri:** Ařaęıdaki seenekler, kalıcı olmayan iletilerin, uygulama isteęinde bulunan uygulamaya ncesinde bir uygulamaya gnderilip gnderilmeyeceęini denetler.

#### **MQSO\_READ\_AHEAD\_AS\_Q\_DEF**

MQSUB aęrısı ynetilen bir tanıtıcı kullanıyorsa, uygulamanın uygulamaya istekte bulunmadan nce uygulamaya ileti gnderilip gnderilmeyeceęini belirlemek iin, abone olunan konu ile iliřkili model kuyruęunun varsayılan okuma deęeri znitelięi kullanılır.

Bu varsayılan deęerdir.

#### **MQSO\_NO\_READ\_AHEAD**

MQSUB aęrısı ynetilen bir tanıtıcı kullanıyorsa, uygulama bu tanıtıcıyı uygulamadan nce uygulamaya gnderilmez.

#### **MQSO\_READ\_AHEAD**

MQSUB aęrısı ynetilen bir tanıtıcı kullanıyorsa, uygulama bunları istemeden nce uygulamaya ileti gnderilebilir.

#### **Not:**

ndeki okuma seenekleri iin ařaęıdaki notlar geerlidir:

1. Bu seeneklerden yalnızca biri belirlenebilir. Hem MQSO\_READ\_AHEAD, hem de MQSO\_NO\_READ\_AHEAD belirtilirse, neden kodu MQRC\_OPTIONS\_ERROR dndrlr. Bu seenekler yalnızca MQSO\_MANAGED belirtilirse geerlidir.
2. nceden aılmıř bir kuyruk geirildięinde, bunlar MQSUB iin geerli deęildir. İstendięinde okuma seeneęi etkinleřtirilmemiř olabilir. İlk MQGET aęrısında kullanılan MQGET seenekleri, nnn etkinleřtirilmesini engelleyebilir. Ayrıca, istemcinin nden okuma desteklenmedięi bir kuyruk

yöneticisine bağlanırken ileriye doğru okuma geçersiz kılınır. Uygulama bir IBM MQ istemcisi olarak çalışmazsa, bu seçenekler yoksayılar.

**Genel arama karakteri seçenekleri:** Aşağıdaki seçenekler, genel arama karakterlerinin, MQSD ' nin ObjectString alanında sağlanan dizgide yorumlanır. Bu seçeneklerden yalnızca birini belirleyebilirsiniz. Var olan bir aboneliğin MQSO ALTER seçeneğini kullanarak değiştirilmesi durumunda, bu genel arama karakterlerinin değiştirilmesi değiştirilemez. MQSO\_RESUME komutunu kullanarak bir MQSUB çağrısından geri dönerek, uygun genel arama karakteri seçeneği ayarlanır.

### MQSO\_WILDCARD\_CHAR

Genel arama karakterleri yalnızca konu dizgisindeki karakterler üzerinde çalışır.

İzleyen çizelgede, MQSO\_WILDCARD\_CHAR tarafından tanımlanan davranış gösterilir.

Özel Karakter	Davranış
Eğik Çizgi (/)	Önemi yok, sadece başka bir karakter.
Yıldız işareti (*)	Joker karakter, sıfır ya da daha fazla karakter
Soru işareti (?)	Genel arama karakteri, 1 karakter
Yüzde işareti (%)	Karakter (*), (?) ya da (%) karakterlerinin bir dizilimde kullanılmasını ve özel karakter olarak yorumlanmamasını (örneğin, (% *), (%?) ya da (%%) olarak yorumlamak için çıkış karakteri.

Örneğin, aşağıdaki konuda yayınlayın:

```
/level0/level1/level2/level3/level4
```

aboneleri aşağıdaki konuları kullanarak eşleştirir:

```
*  
/*  
/ level0/level1/level2/level3/*  
/ level0/level1/*/level3/level4  
/ level0/level1/le?e12/level3/level4
```

**Not:** Bu genel arama karakterleri kullanımı, yayınlama/abone olma için MQRFH1 biçimlendirilmiş iletileri kullanılırken tam olarak IBM MQ V6 ve WebSphere MB V6 ' de sağlanan anlamlara erir. Bu, yeni yazılan uygulamalar için kullanılmaması ve yalnızca daha önce o sürüme karşı çalışan uygulamalar için kullanılması ve MQSO\_WILDCARD\_TOPIC içinde açıklandığı gibi, varsayılan genel arama karakteri davranışını kullanmak üzere değiştirilmemiş uygulamalar için kullanılması önerilir.

### MQSO\_WILDCARD\_KONUSU

Genel arama karakterleri yalnızca konu dizgisindeki konu öğelerinde işlem görmektedir. Bu, hiçbir seçilmezse, varsayılan davranıştır.

MQSO\_WILDCARD\_TOPIC için gerekli olan davranış aşağıdaki çizelgede gösterilmiştir:

Özel Karakter	Davranış
(/)	Konu düzeyi ayırıcı
Sayı işareti (#)	Joker karakter: birden çok konu düzeyi
Artı işareti (+)	Joker karakter: tek konu düzeyi

## Notlar:

(+) ve (#) karakterleri, bir konu düzeyi içinde başka karakterler (kendileri de dahil olmak üzere) ile karıştırılmışsa, genel arama karakteri olarak kabul edilmez. Aşağıdaki dizgide, (#) ve (+) karakterleri sıradan karakterler olarak ele alınır.

```
level0/level1/#+/level3/level#
```

Örneğin, aşağıdaki konuda yayınlayın:

```
/level0/level1/level2/level3/level4
```

aboneleri aşağıdaki konuları kullanarak eşleştirir:

```
#  
/#  
/ level0/level1/level2/level3/#  
/ level0/level1+/level3/level4
```

**Diğer seçenekler:** Aşağıdaki seçenekler, abonelik yerine API çağrısının yayını kontrol eder. MQSO\_RESUME komutunu kullanarak bir MQSUB çağrısından dönüşte, bu seçenekler değişmeden kalır. Daha ayrıntılı bilgi için bkz. [“AlternateUserTanıtıcısı \(MQCHAR12\)” sayfa 570](#).

## MQSO\_ALTERNATE\_USER\_AUTHORITY

AlternateUserTanıtıcısı alanı, bu MQSUB çağrısını doğrulamak için kullanılacak bir kullanıcı kimliği içerir. The call can succeed only if this AlternateUserId is authorized to open the object with the specified access options, regardless of whether the user identifier under which the application is running is authorized to do so.

## MQSO\_SET\_COREL\_ID

Abonelik, *SubCorrelId* alanında sağlanan ilinti tanıtıcısını kullanmandır. Bu seçenek belirlenmezse, abonelik sırasında kuyruk yöneticisi tarafından otomatik olarak bir ilinti tanıtıcısı yaratılır ve *SubCorrelId* alanında uygulamaya döndürülür. Daha fazla bilgi için bkz. [“SubCorrelTanıtıcısı \(MQBYTE24\)” sayfa 573](#).

Bu seçenek, MQSO\_MANAGED ile birleştirilemez.

## MQSO\_SET\_IDENTITY\_CONTEXT

Abonelik, *PubAccountingToken* ve *PubApplIdentityData* alanlarında sağlanan muhasebe belirteci ve uygulama kimliği verilerini kullanmandır.

Bu seçenek belirlenirse, aynı yetki denetimi, MQOO\_SET\_IDENTITY\_CONTEXT ile hedef kuyruğa bir MQSO\_set\_identity\_context kullanılarak erişildiği gibi yürütülür. Bu durumda, hedef kuyrukta bir yetki denetimi yapılmadığı durumlarda, MQSO\_MANAGED seçeneğinin de kullanıldığı durumlar dışında.

Bu seçenek belirlenmezse, bu aboneye gönderilen yayınların varsayılan bağlam bilgileri aşağıdaki gibi olur:

Çizelge 531. Bu aboneye gönderilen yayınlara ilişkin varsayılan bağlam bilgileri	
MQMD ' de alan	Kullanılan değer
<i>UserIdentifier</i>	Aboneliğin yapıldığı sırada, abonelik ile ilişkili kullanıcı kimliği.
<i>AccountingToken</i>	Olanaklıysa, ortamdan saptanır; değilse MQACT_NONE olarak ayarlayın.
<i>ApplIdentityData</i>	Boşluklara ayarla

Bu seçenek yalnızca MQSO\_CREATE ve MQSO ALTER ile geçerlidir. MQSO\_RESUME ile kullanılırsa, *PubAccountingToken* ve *PubApplIdentityData* alanları yoksayılar, bu nedenle bu seçeneğin herhangi bir etkisi yoktur.

Bir abonelik, önceden aboneliğin sağladığı kimlik bağlamı bilgilerinin bulunduğu bu seçenek kullanılmadan değiştirilirse, değiştirilen abonelik için varsayılan bağlam bilgileri oluşturulur.

Farklı kullanıcı kimlikleri için MQSO\_ANY\_USERID seçeneği ile farklı kullanıcı kimliklerinin kullanılmasına izin veren bir abonelik farklı bir kullanıcı kimliği tarafından sürdürülürse, artık abonelik sahibi olan yeni kullanıcı kimliği için varsayılan kimlik bağlamı oluşturulur ve sonraki yayınlar yeni kimlik bağlamını içeren teslim edilir.

## **MQSO\_FAIL\_IF QUIESCING**

Kuyruk yöneticisi susturma durumundaysa, MQSUB çağrısı başarısız olur. z/OS üzerinde, CICS ya da IMS uygulaması için bu seçenek, bağlantı susturma durumundaysa MQSUB çağrısını da zorlar.

## **ObjectName (MQCHAR48)**

Bu, yerel kuyruk yöneticisinde tanımlandığı şekilde, konu nesnesinin adıdır.

Ad aşağıdaki karakterleri içerebilir:

- Büyük harf alfabetik karakterler (A-Z)
- Küçük harf alfabetik karakterler (a-z)
- Sayısal basamaklar (0-9)
- Nokta (.), eğik çizgi (/), altçizgi (\_), yüzde (%)

Ad başta ya da gömülü boşluklar içermemelidir, ancak sonda boşluklar içerebilir. Addaki önemli verilerin sonunu göstermek için boş bir karakter kullanın; bu karakteri izleyen karakterler ve boş değer boşluk olarak değerlendirilir. Belirtilen ortamlarda aşağıdaki kısıtlamalar geçerlidir:

- EBCDIC Katakana kullanan sistemlerde küçük harfli karakterler kullanılamaz.
- z/OS'ta:
  - Alt çizgiyle başlayan ya da biten adlardan kaçınınız; bunlar işlemler ve denetim panoları tarafından işlenemez.
  - Yüzde karakterinin RACFiçin özel bir anlamı vardır. Dış güvenlik yöneticisi olarak RACF kullanılırsa, adlar yüzdeyi içermemelidir. Varsa, RACF sosyal tanımları kullanıldığında bu adlar güvenlik denetimlerine dahil edilmez.
- IBM üzerinde, komutlarda belirtildiğinde küçük harf, eğik çizgi ya da yüzde içeren adlar tırnak içine alınmalıdır. Bu tırnak işaretleri, yapılarda alan olarak ya da çağrılarda parametre olarak oluşan adlar için belirlenmemelidir.

*ObjectName* , tam konu adını oluşturmak için kullanılır.

Tam konu adı iki farklı alandan oluşturulabilir: *ObjectName* ve *ObjectString*. Bu iki alanın nasıl kullanıldığına ilişkin ayrıntılar için [Konu dizgilerini birleştirme](#) başlıklı konuya bakın.

*ObjectName* alanıyla tanıtılan nesne bulunamazsa, *ObjectString* içinde belirtilmiş bir dizgi olsa bile, çağrı MQRC\_UNKNOWN\_OBJECT\_NAME neden koduyla başarısız olur.

MQSO\_RESUME seçeneği kullanılarak bir MQSUB çağrısından dönüş yapılıyorsa, bu alan değişmez.

Bu alanın uzunluğu MQ\_TOPIC\_NAME\_LENGTH tarafından verilir. Bu alanın ilk değeri, C içindeki boş dizgi ve diğer programlama dillerindeki 48 boş karakterdir.

MQSO ALTER seçeneği kullanılarak var olan bir abonelik değiştiriliyorsa, abone olunan konu nesnesinin adı değiştirilemez. Bu alan ve *ObjectString* alanı atlanabilir. Sağlanırsa, aynı tam konu adına çözümleri gerekir. Çağrılmazsa, çağrı MQRC\_TOPIC\_NOT ALTERABLE ile başarısız olur.

## **AlternateUserTanıtıcısı (MQCHAR12)**

MQSO\_ALTERNATE\_USER\_AUTHORITY değerini belirlerseniz, bu alan, uygulamanın şu anda çalışmakta olduğu kullanıcı kimliği yerine, abonelik yetkisini ve hedef kuyruğa (MQSUB çağrısının **Hobj** parametresinde belirlenen) çıkış için gereken değeri denetlemek için kullanılan bir diğer kullanıcı kimliği içerir.

Başarılı olursa, bu alanda belirtilen kullanıcı kimliği, uygulamanın şu anda altında çalışmakta olduğu kullanıcı kimliğinin yerine, abonelik sahibi kullanıcı kimliği olarak kaydedilir.

MQSO\_ALTERNATE\_USER\_AUTHORITY belirtildiyse ve bu alan, ilk boş karakter ya da alanın sonuna kadar tamamen boşsa, abonelik yalnızca, belirtilen seçeneklerle ya da çıkış için hedef kuyrukla bu konuya abone olmak için hiçbir kullanıcı yetkisinin gerekmediği durumlarda başarılı olabilir.

MQSO\_ALTERNATE\_USER\_AUTHORITY belirtilmediyse, bu alan yoksayılr.

Belirtilen ortamlarda aşağıdaki farklar vardır:

- z/OS' ta, aboneliğe ilişkin yetkiyi denetlemek için yalnızca AlternateUserTanıtıcısı 'nın ilk 8 karakteri kullanılır. Ancak, yürürlükteki kullanıcı kimliğinin bu belirli bir diğer kullanıcı kimliğini belirtme yetkisi olması gerekir; bu denetim için, diğer kullanıcı kimliğinin 12 karakteri de kullanılır. Kullanıcı kimliği, yalnızca dış güvenlik yöneticisi tarafından izin verilen karakterleri içermelidir.

MQSO\_RESUME komutunu kullanarak bir MQSUB çağrısından dönüşte bu alan değiştirilmez.

Bu bir giriş alanıdır. Bu alanın uzunluğu MQ\_USER\_ID\_LENGTH tarafından verilir. Bu alanın ilk değeri, C içindeki boş değerli dizidir ve diğer programlama dillerindeki 12 boş karakterdir.

### ***AlternateSecurityTanıtıcısı (MQBYTE40)***

Bu, uygun yetki denetimlerinin gerçekleştirilmesine izin vermek için yetki hizmetine AlternateUserTanıtıcısı ile geçirilen bir güvenlik tanıtıcısıdır.

AlternateSecuritytanıtıcısı yalnızca MQSO\_ALTERNATE\_USER\_AUTHORITY belirtildiyse kullanılır ve AlternateUserId alanı, ilk boş karakter ya da alanın sonuna kadar tam olarak boş bırakılmaz.

MQSO\_RESUME komutunu kullanarak bir MQSUB çağrısından dönüşte bu alan değiştirilmez.

Daha fazla bilgi için, MQOD veri tipindeki [“AlternateSecurityTanıtıcısı \(MQBYTE40\)”](#) sayfa 484 tanımlamasına bakın.

### ***SubExpiry (MQUZY)***

Bu süre, aboneliğin sona ermesinden sonraki bir saniyenin onda biri olarak ifade edilir. Bu aralık iletildikten sonra, bu abonelikte eşleşen yayın yok. Aboneliğin sona ereceği zaman, yayınlar artık kuyruğa gönderilmez. ne var ki, zaten orada olan yayınlar hiçbir şekilde etkilenmedi. *SubExpiry* ' in yayın süre bitimi üzerinde etkisi yoktur.

Şu özel değer tanındı:

#### **MQE\_UNSIINSIZ**

Aboneliğin süre bitimi sınırsız.

Var olan bir aboneliği MQSO\_ALTER seçeneğini kullanarak değiştirebiliyorsa, aboneliğin süre bitimi değiştirilebilir.

MQSO\_RESUME seçeneğini kullanarak bir MQSUB çağrısından geri dönüş için, bu alan aboneliğin özgün süre bitimi olarak ayarlanır ve kalan süre bitimi olmaz.

### ***ObjectString (MQCHARV)***

Bu, kullanılacak uzun nesne adıdır.

*ObjectString* , Tam konu adını oluşturmak için kullanılır.

Tam konu adı iki farklı alandan oluşturulabilir: *ObjectName* ve *ObjectString*. Bu iki alanın nasıl kullanıldığına ilişkin ayrıntılar için [Konu dizgilerini birleştirme](#) başlıklı konuya bakın.

*ObjectString* uzunluk üst sınırı 10240 'tır.

*ObjectString* doğru belirtilmezse, MQCHARV yapısının nasıl kullanılacağına ilişkin açıklamaya göre ya da uzunluk üst sınırını aşarsa, çağrı MQRC\_OBJECT\_STRING\_ERROR neden koduyla başarısız olur.

Bu bir giriş alanıdır. Bu yapıdaki alanların başlangıç değerleri, MQCHARV yapısındaki değerlerle aynıdır.

*ObjectString* içinde genel arama karakterleri varsa, bu genel arama karakterlerinin yorumlanması, MQSD ' nin Seçenekler alanında belirtilen genel arama karakteri seçenekleri kullanılarak denetlenebilir.

MQSO\_RESUME seçeneği kullanılarak bir MQSUB çağrısından dönüş yapılıyorsa, bu alan değişmez. Bir arabellek sağlanırsa, *ResObjectString* alanında kullanılan tam konu adı döndürülür.

MQSO\_ALTER seçeneği kullanılarak var olan bir abonelik değiştiriliyorsa, abone olunan konu nesnesinin uzun adı değiştirilemez. Bu alan ve *ObjectName* alanı atlanabilir. Sağlanırsa, bunların aynı tam konu adına çözülmesi gerekir ya da çağrı MQRC\_TOPIC\_NOT\_ALTERABLE ile başarısız olur.

### ***SubName* (MQCHARV)**

Bu, abonelik adını belirtir. Bu alan yalnızca *Options* MQSO\_DURABLE seçeneğini belirtiyorsa, ancak sağlanırsa, kuyruk yöneticisi tarafından MQSO\_NON\_DURABLE için de kullanılır.

Belirtilirse, *SubName* kuyruk yöneticisi içinde benzersiz olmalıdır; bu, aboneliği tanımlamak için kullanılan yöntemdir.

*SubName* uzunluk üst sınırı 10240 'tır.

Bu alan iki amaca hizmet eder. Bir MQSO\_DURABLE aboneliği için, abonelik yaratıldıktan sonra (MQCO\_KEEP\_SUB seçeneğini kullanarak) aboneliği kapattıysanız ya da kuyruk yöneticisiyle bağlantınızı kesildiyse, aboneliği devam ettirebilmeniz için bu alanı kullanırsınız. Bu, MQSO\_RESUME seçeneğiyle MQSUB çağrısı kullanılarak yapılır. Ayrıca, DISPLAY SBSTATUS içindeki SUBID alanındaki aboneliklerin yönetim görünümünde de görüntülenir.

*SubName* yanlış belirtilirse, MQCHARV yapısının nasıl kullanılacağına ilişkin açıklamaya göre, gerekli olduğunda (yani *SubName*) dışarıda bırakılır. *VSLength* sıfır) ya da uzunluk üst sınırını aşarsa, çağrı başarısız olur; neden kodu MQRC\_SUB\_NAME\_ERROR.

Bu bir giriş alanıdır. Bu yapıdaki alanların başlangıç değerleri, MQCHARV yapısındaki değerlerle aynıdır.

MQSO\_ALTER seçeneği kullanılarak var olan bir abonelik değiştiriliyorsa, abonelik adı, gönderme yapılan aboneliği bulmak için kullanılan tanıttıcı alan olduğundan değiştirilemez. MQSO\_RESUME seçeneğiyle bir MQSUB çağrısının çıkışında değiştirilmez.

### ***SubUserVerileri* (MQCHARV)**

Bu, abonelik kullanıcı verilerini belirtir. Bu alandaki aboneliğe ilişkin sağlanan veriler, bu aboneliğe gönderilen her yayının MQSubUserVeri iletisi özelliği olarak içerilir.

*SubUserData* uzunluğu üst sınırı 10240 'tır.

*SubUserData* yanlış belirtildiyse, MQCHARV yapısının nasıl kullanılacağı ya da uzunluk üst sınırını aştıysa, çağrı neden kodu MQRC\_SUB\_USER\_DATA\_ERROR koduyla başarısız olur.

Bu bir giriş alanıdır. Bu yapıdaki alanların ilk değerleri, MQCHARV yapısındaki alanlarla aynıdır.

Var olan bir aboneliği MQSO\_ALTER seçeneğini kullanarak değiştirebiliyorsa, abonelik kullanıcısı verileri değiştirilebilir.

Bu değişken uzunluk alanı, bir arabellek sağlanırsa ve *VSBuflen* içinde pozitif bir arabellek uzunluğu varsa, MQSU\_RESUME seçeneği kullanılarak bir MQSUB çağrısından çıkışta döndürülür. Çağrıda arabellek sağlanmazsa, MQCHARV ' ın *VSLength* alanında yalnızca abonelik kullanıcı tarihi uzunluğu döndürülür.



Sağlanan arabellek, alanı döndürmek için gereken alandan küçükse, sağlanan arabellekte yalnızca *VSBufLen* byte 'ları döndürülür.

### **SubCorrelTanıtıcısı (MQBYTE24)**

Bu alan, bu abonelikte eşleşen tüm yayınlara ortak bir ilinti tanıtıcısı içerir.



**Uyarı:** Bir ilinti tanıtıcısı yalnızca, bir sıradüzeninde değil, yayınlama/abone olma kümesindeki kuyruk yöneticileri arasında geçirebilir.

Bu abonelikte eşleşmesi için gönderilen tüm yayınlar, ileti tanımlayıcısında bu ilinti tanıtıcısını içerir. Birden çok abonelik, yayınlarını aynı kuyruktan alacaksa, MQGET by ilinti tanıtıcısı kullanılarak, yalnızca belirli bir abonelik için yayınların elde edilmesi sağlanır. Bu ilinti tanıtıcısı kuyruk yöneticisi ya da kullanıcı tarafından yaratılabilir.

MQSO\_SET\_COREL\_ID seçeneği belirtilmediyse, ilinti tanıtıcısı kuyruk yöneticisi tarafından oluşturulur ve bu alan, bu abonelik için yayınlanan her iletide ayarlanacak ilinti tanıtıcısını içeren bir çıkış alanıdır. Üretilen ilinti tanıtıcısı, 4 baytlık bir ürün tanıtıcısından (AMQX ya da CSQM ' de ASCII ya da EBCDIC) oluşur ve ardından benzersiz bir dizginin ürüne özgü bir somutlamasını izler.

MQSO\_SET\_COREL\_ID seçeneği belirtilirse, kullanıcı tarafından ilinti tanıtıcısı oluşturulur ve bu alan, bu abonelik için her yayınında ayarlanacak ilinti tanıtıcısını içeren bir giriş alanıdır. Bu durumda, alan MQCI\_NONE içeriyorsa, bu abonelik için yayınlanan her iletide belirlenen ilinti tanıtıcısı, iletinin özgün tanıtıcısıyla yaratılan ilinti tanıtıcısıdır.

MQSO\_GROUP\_SUB seçeneği belirtilirse ve belirtilen ilinti tanıtıcısı, aynı kuyruğu ve çakışan bir konu dizesini kullanan var olan gruplanmış bir abonelikte aynıysa, yayının bir kopyasıyla yalnızca gruptaki en önemli abonelik sağlanır.

Bu alanın uzunluğu MQ\_CORIL\_ID\_LENGTH tarafından verilir. Bu alanın ilk değeri MQCI\_NONE olur.

Var olan bir aboneliği MQSO\_ALTER seçeneğini kullanarak değiştiriyorsanız ve bu alan bir giriş alanıysa, abonelik ilintilendirme tanıtıcısı değiştirilebilir; abonelik gruplanmış bir abonelik olmadıkça, MQSO\_GROUP\_SUB seçeneği kullanılarak yaratılmıştır; bu durumda, abonelik ilintilendirme tanıtıcısı değiştirilemez.

MQSO\_RESUME komutunu kullanarak bir MQSUB çağrısından dönüşte, bu alan, abonelik için yürürlükteki ilinti tanıtıcısı olarak ayarlanır.

### **PubPriority (MQHOMER)**

Bu, bu abonelikte eşleşen tüm yayın iletilerinin İleti Tanımlayıcısının (MQMD) *Priority* alanında yer alacak değerdir. MQMD ' deki *Priority* alanıyla ilgili daha fazla bilgi için bkz. [“Öncelik \(MQUBE\)” sayfa 445.](#)

Değer sıfırdan büyük ya da sıfıra eşit olmalıdır; sıfır, en düşük önceliğe sahip olmalıdır. Aşağıdaki özel değerler de kullanılabilir:

#### **MQPRI\_PRIORITY\_AS\_Q\_DEF**

Bir abonelik kuyruğu, MQSUB çağrısındaki *Hobj* alanında sağlandığında ve yönetilen bir tanıtıcı değilse, bu iletinin önceliği bu kuyruğun **DefPriority** özniteliğinden alınır. If the queue is a cluster queue or there is more than one definition in the queue-name resolution path then the priority is determined when the publication message is put to the queue as described for [“Öncelik \(MQUBE\)” sayfa 445.](#)

MQSUB çağrısı yönetilen bir tanıtıcı kullanıyorsa, ileti için öncelik, abone olunan konu ile ilişkilendirilmiş model kuyruğunun **DefPriority** özniteliğinden alınır.

#### **MQPRI\_PRIORITY\_AS\_PUDID**

İletiyeye ilişkin öncelik, özgün yayının önceliğidir. Bu, alanın ilk değeridir.

Var olan bir aboneliğin MQSO\_ALTER seçeneğini kullanarak değiştirilmesi durumunda, ilerideki yayın iletilerinin *Priority* değeri değiştirilebilir.

MQSO\_RESUME komutunu kullanarak bir MQSUB çağrısından dönüşte bu alan, abonelik için kullanılmakta olan geçerli önceliğe ayarlanır.

### **PubAccountingSimgesi (MQBYTE32)**

Bu, bu abonelikte eşleşen tüm yayın iletilerinin İleti Tanımlayıcısının (MQMD) *AccountingToken* alanında yer alacak değerdir. *AccountingToken* , iletinin tanıtıcı bağlamının bir parçasıdır. İleti bağlamına ilişkin ek bilgi için İleti bağlamı başlıklı konuya bakın. MQMD ' deki *AccountingToken* alanıyla ilgili daha fazla bilgi için bkz. "AccountingToken (MQBYTE32)" sayfa 452

*PubAccountingToken* alanı için aşağıdaki özel değeri kullanabilirsiniz:

#### **MQACT\_NONE**

Hesap belirteci belirtilmedi.

Değer, alanın uzunluğuna ilişkin ikili sıfır değeridir.

C programlama dili için, sabit MQACT\_NONE\_DIZISI de tanımlanır; bu, MQACT\_NONE ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisi olur.

MQSO\_SET\_IDENTITY\_CONTEXT seçeneği belirtilmediyse, muhasebe simgesi kuyruk yöneticisi tarafından varsayılan bağlam bilgisi olarak oluşturulur ve bu alan, bu abonelik için yayınlanan her iletide ayarlanacak *AccountingToken* ögesini içeren bir çıkış alanıdır.

MQSO\_SET\_IDENTITY\_CONTEXT seçeneği belirtilirse, hesap simgesi kullanıcı tarafından oluşturulur ve bu alan, bu abonelik için her yayında ayarlanacak *AccountingToken* ' u içeren bir giriş alanıdır.

Bu alanın uzunluğu MQ\_ACCOUNTING\_TOKEN\_LENGTH tarafından verilir. Bu alanın ilk değeri MQACT\_NONE olur.

Var olan bir aboneliği MQSO\_ALTER seçeneğini kullanarak değiştiriyorsa, gelecekteki yayın iletilerindeki *AccountingToken* değeri değiştirilebilir.

MQSO\_RESUME komutunu kullanarak bir MQSUB çağrısından dönüşte bu alan, abonelik için kullanılmakta olan *AccountingToken* değerine ayarlanır.

### **PubApplIdentityData (MQCHAR32)**

Bu değer, bu abonelikte eşleşen tüm yayın iletilerinin İleti Tanımlayıcısının (MQMD) *ApplIdentityData* alanında bulunan değerdir. *ApplIdentityData* , iletinin tanıtıcı bağlamının bir parçasıdır. İleti bağlamına ilişkin ek bilgi için İleti bağlamı başlıklı konuya bakın. MQMD ' deki *ApplIdentityData* alanıyla ilgili daha fazla bilgi için bkz. "ApplIdentityVerileri (MQCHAR32)" sayfa 454

MQSO\_SET\_IDENTITY\_CONTEXT seçeneği belirtilmediyse, varsayılan bağlam bilgileri olarak, bu abonelik için yayınlanan her iletide belirtilen *ApplIdentityData* değeri boşluklara yol gösterir.

If the option MQSO\_SET\_IDENTITY\_CONTEXT is specified, the *PubApplIdentityData* is being generated by the user and this field is an input field which contains the *ApplIdentityData* to be set in each publication for this subscription.

Bu alanın uzunluğu MQ\_APPL\_IDENTITY\_DATA\_LENGTH tarafından verilir. Bu alanın ilk değeri, C ' deki boş değerli dizedir ve diğer programlama dillerindeki 32 boş karakterdir.

Var olan bir aboneliğin MQSO\_ALTER seçeneğini kullanarak değiştirilmesi durumunda, ilerideki yayın iletilerinin *ApplIdentityData* değeri değiştirilebilir.

MQSO\_RESUME komutunu kullanarak bir MQSUB çağrısından dönüşte bu alan, abonelik için kullanılmakta olan *ApplIdentityData* değerine ayarlanır.

### **SelectionString (MQCHARV)**

Bu, bir konudaki iletiler için abone olurken kullanılan seçim ölçütlerini sağlamak için kullanılan dizgidir.

Bu deęişken uzunluk alanı, bir arabellek saęlandıysa ve VSBufSize içinde artı bir arabellek uzunluęu da varsa, MQSU\_RESUME seeneęi kullanılarak bir MQSUB aęrısından ıkıřa dndrlecektir. aęrıda arabellek saęlanmazsa, MQCHARV ' in VSLecgth alanında yalnızca seim dizgisinin uzunluęu dndrlr. Saęlanan arabellek, alanı dndrmek iin gereken alandan kkse, saęlanan arabellete yalnızca VSBufSize byte deęeri dndrlr.

*SelectionString* yanlış belirtilirse, “MQCHARV-Deęişken Uzunluklu Dizgi” sayfa 292 yapısının nasıl kullanılacağı ya da uzunluk st sınırını ařtıysa, aęrı neden kodu MQRC\_SELECTION\_STRING\_ERROR ile bařarısız olur.

SelectionString kullanımı, [Seicileriinde](#) aıklanmaktadır.

### **SubLevel (MQUZE)**

Bu, abonelik ile iřlişli dzeydir. Bu abonelik yalnızca, yayınlama sırasında kullanılan PubLevel deęerinden kk ya da bu deęere eřit olan en yksek SubLevel deęerine sahip abonelikler kmesinde yer alıyorsa, bu abonelik iin yalnızca bu abonelikte teslim edilir. Ancak, bir yayın saklandıysa, PubLevel 1 dzeyinde yeniden yayımlandıęından, bu yayın artık daha yksek dzeylerde aboneler iin kullanılabilir deęildir.

Deęer, sıfır ile 9 aralıęında olmalıdır. Sıfır en dřk dzeydir.

Bu alanın ilk deęeri 1 'dir.

Daha fazla bilgi iin bkz. [Yayınlar Arası Yayınlar](#).

Varolan bir abonelięi MQSO ALTER seeneęini kullanarak deęiřtiriyorsa, SubLevel deęeri deęiřtirilemez.

MQSO\_PUBLICATIONS\_ON\_REQUEST seeneęiyle bir SubLevel deęerinin 1 'den byk bir deęere sahip olmasına izin verilmez.

MQSO\_RESUME komutunu kullanarak bir MQSUB aęrısından dnřte, bu alan abonelik iin kullanılmakta olan yrrlkteki dzeye ayarlanır.

### **ResObjectDizgisi (MQCHARV)**

Kuyruk yneticisi, *ObjectName* ' ta saęlanan adı zdkten sonra uzun nesne adıdır.

*ObjectString* ' ta uzun nesne adı saęlanırsa ve *ObjectName* ' ta hibir řey saęlanmıyorsa, bu alanda dndrlen deęer, *ObjectString* ' de saęlanan deęerle aynıdır.

Bu alan atılırsa ( *ResObjectString.VSBufSize* sıfırdır), *ResObjectString* dndrlmez, ancak uzunluk *ResObjectString.VSLength* ' te dndrlr. Uzunluk tam *ResObjectDizgisinden* kısaysa, kısaltılır ve saęlanan uzunluęa sıęabileceęi en saędaki en saędaki karakter dndrlr.

*ResObjectString* yanlış belirtildiyse, [MQCHARV](#) yapısının nasıl kullanılacağını ya da uzunluk st sınırını ařtıyorsa, aęrı neden kodu MQRC\_RES\_OBJECT\_STRING\_ERROR ile bařarısız olur.

## **MQSMPO-İleti zellięi seeneklerini ayarla**

**MQSMPO** yapısı, uygulamaların, ileti zelliklerinin nasıl ayarlanacağını denetleyen seenekleri belirtmesine olanak saęlar. Yapı, **MQSETMP** aęrısındaki bir giriř parametresidir.

### **Kullanılabilirlik**

Tm IBM MQ sistemleri ve IBM MQ istemcileri.

### **Karakter kmesi ve kodlama**

**MQSMPO** iindeki veriler, uygulamanın karakter kmesinde ve uygulamanın kodlamasında olmalıdır ( **MQENC\_NATIVE** ).

### **Alanlar**

**Not:** Ařaęıdaki tabloda, alanlar alfabetik olarak deęil, kullanıma gre gruplanır. Alt konular aynı sırayı izler.

Çizelge 532. MQSMPO içindeki alanlar

Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
StrucId (yapı tanıtıcısı)	MQSMPO_STRUC_ID	'SMPO'
Sürüm (yapı sürümü numarası)	MQSMPO_VERSION_1	1
Seçenekler (seçenekler)	MQSMPO_NONE	0
ValueEncoding (özellik değeri kodlaması)	MQENC_NATIVE	Ortama bağlıdır
ValueCCSID (özellik değeri karakter takımı)	MQCCSI_APPL	-3

**Notlar:**

- Boş değerli dizilim ya da boşluk değeri, C dizgisindeki boş dizgiyi ve diğer programlama dillerindeki boş karakterleri gösterir.
- C programlama dilinde, makro değişkeniMQSMPO\_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlar için başlangıç değerleri sağlamak üzere aşağıdaki şekilde kullanılabilir:

```
MQSMPO MySMPO = {MQSMPO_DEFAULT};
```

## Dil bildirimleri

### MQSMPO için C bildirimini

```
typedef struct tagMQSMPO MQSMPO;
struct tagMQSMPO {
    MQCHAR4    StrucId;           /* Structure identifier */
    MQLONG     Version;          /* Structure version number */
    MQLONG     Options;          /* Options that control the action of MQSETMP */
    MQLONG     ValueEncoding;    /* Encoding of Value */
    MQLONG     ValueCCSID;       /* Character set identifier of Value */
};
```

### MQSMPO için COBOL bildirimini

```
** MQSMPO structure
10 MQSMPO.
** Structure identifier
15 MQSMPO-STRUCID PIC X(4).
** Structure version number
15 MQSMPO-VERSION PIC S9(9) BINARY.
** Options that control the action of MQSETMP
15 MQSMPO-OPTIONS PIC S9(9) BINARY.
** Encoding of VALUE
15 MQSMPO-VALUEENCODING PIC S9(9) BINARY.
** Character set identifier of VALUE
15 MQSMPO-VALUECCSID PIC S9(9) BINARY.
```

### MQSMPO için PL/I bildirimini

```
dcl
1 MQSMPO based,
3 StrucId char(4), /* Structure identifier */
3 Version fixed bin(31), /* Structure version number */
3 Options fixed bin(31), /* Options that control the action of MQSETMP */
3 ValueEncoding fixed bin(31), /* Encoding of Value */
3 ValueCCSID fixed bin(31), /* Character set identifier of Value */
```

## MQSMPO için High Level Assembler (Üst Düzey Çevirici) bildirimi

MQSMPO	DSECT	
MQSMPO_STRUCID	DS	CL4 Structure identifier
MQSMPO_VERSION	DS	F Structure version number
MQSMPO_OPTIONS	DS	F Options that control the action of
*		MQSETMP
MQSMPO_VALUEENCODING	DS	F Encoding of VALUE
MQSMPO_VALUECCSID	DS	F Character set identifier of VALUE
MQSMPO_LENGTH	EQU	*-MQSMPO
MQSMPO_AREA	DS	CL(MQSMPO_LENGTH)

### **StrucId (MQCHAR4)**

Bu, yapı tanıtıcısıdır; değer şöyle olmalıdır:

#### **MQSMPO\_STRUC\_ID**

İleti özelliği seçenekleri yapısının belirlenmesine ilişkin tanıtıcı.

For the C programming language, the constant **MQSMPO\_STRUC\_ID\_ARRAY** is also defined; this has the same value as **MQSMPO\_STRUC\_ID**, but is an array of characters instead of a string.

Bu her zaman bir giriş alanıdır. Bu alanın başlangıç değeri **MQSMPO\_STRUC\_ID**' dir.

### **Sürüm (MQUZE)**

Bu, yapı sürüm numarasıdır; değer şu olmalıdır:

#### **MQSMPO\_VERSION\_1**

Version-1 , ileti özelliği seçenekleri yapısını ayarlar.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

#### **MQSMPO\_CURRENT\_VERSION**

İleti özelliği seçenekleri yapısının yürürlükteki sürümü.

Bu her zaman bir giriş alanıdır. Bu alanın başlangıç değeri **MQSMPO\_VERSION\_1**' dir.

### **Seçenekler (MQUZE)**

#### **Yer seçenekleri**

Aşağıdaki seçenekler, özellik imlecine göre özelliğin görelî konularıyla ilişkilidir:

#### **MQSMPO\_SET\_FIRST**

Belirtilen adla eşleşen ilk özelliğin değerini ayarlar ya da yoksa, eşleşen bir sıradüzene sahip diğer tüm özelliklerden sonra yeni bir özellik ekler.

#### **MQSMPPPO\_SET\_PROP\_UNDER\_CURSOR**

Özellik imlecine işaret edilen özelliğin değerini ayarlar. Özellik imlecinin gösterdiği özellik, en son MQIMPO\_INQ\_FIRST ya da MQIMPO\_INQ\_INQ\_Next seçeneğini kullanarak sorgulanandır.

Bir MQGET çağrısında ileti tanıtıcısı yeniden kullanıldığında ya da MQGMO ya da MQPUT çağrısındaki MQGMO ya da MQPMO yapısının *MsgHandle* alanında ileti tanıtıcısı belirtildiğinde, özellik geçici çizelgesi ilk durumuna getirilir.

Bu seçenek, özellik geçici çizelgesi henüz oluşturulmadığında ya da özellik imlecine ait özellik işaretçisi silindiğinde kullanılırsa, çağrı tamamlanma kodu MQCC\_FAILED ve neden kodu MQRC\_PROPERTY\_NOT\_AVAM neden koduyla başarısız olur.

### **MQSMPPPO\_SET\_PROP\_BEFORE\_CURSOR**

Özellik imlecine işaret eden özelliğin önüne yeni bir özellik ayarlar. Özellik imlecinin gösterdiği özellik, en son MQIMPO\_INQ\_FIRST ya da MQIMPO\_INQ\_INQ\_Next seçeneğini kullanarak sorgulanandır.

Bir MQGET çağrısında ileti tanıtıcısı yeniden kullanıldığında ya da MQGMO ya da MQPUT çağrısındaki MQGMO ya da MQPMO yapısının *MsgHandle* alanında ileti tanıtıcısı belirtildiğinde, özellik geçici çizelgesi ilk durumuna getirilir.

Bu seçenek, özellik geçici çizelgesi henüz oluşturulmadığında ya da özellik imlecine ait özellik işaretçisi silindiğinde kullanılırsa, çağrı tamamlanma kodu MQCC\_FAILED ve neden kodu MQRC\_PROPERTY\_NOT\_AVAM neden koduyla başarısız olur.

### **MQSMPPPO\_SET\_PROP\_AFTER\_CURSOR**

Özellik imlecine işaret eden özellik sonrasında yeni bir özellik ayarlar. Özellik imlecinin gösterdiği özellik, en son MQIMPO\_INQ\_FIRST ya da MQIMPO\_INQ\_INQ\_Next seçeneğini kullanarak sorgulanandır.

Bir MQGET çağrısında ileti tanıtıcısı yeniden kullanıldığında ya da MQGMO ya da MQPUT çağrısındaki MQGMO ya da MQPMO yapısının *MsgHandle* alanında ileti tanıtıcısı belirtildiğinde, özellik geçici çizelgesi ilk durumuna getirilir.

Bu seçenek, özellik geçici çizelgesi henüz oluşturulmadığında ya da özellik imlecine ait özellik işaretçisi silindiğinde kullanılırsa, çağrı tamamlanma kodu MQCC\_FAILED ve neden kodu MQRC\_PROPERTY\_NOT\_AVAM neden koduyla başarısız olur.

### **MQSMPO\_APPEND\_ÖZELLİĞİ**

Eşleşen bir sıradüzene sahip diğer tüm özelliklerden sonra yeni bir özelliğin eklenmesine neden olur. Belirtilen adla eşleşen en az bir özellik varsa, bu özellik listesinin sonundan sonra sonuna yeni bir özellik eklenir.

Bu seçenek, aynı adı taşıyan özellikler listesinin yaratılmasına olanak sağlar.

Tanımlanan seçeneklerden hiçbirine gerek duyarsanız, aşağıdaki seçeneği kullanın:

### **MQSMPO\_NONE**

Seçenek belirtilmedi.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQSMPO\_SET\_FIRST ' dir.

### **ValueEncoding (MQUZE)**

Değer sayıysa, ayarlanacak özellik değerinin kodlaması.

Bu her zaman bir giriş alanıdır. Bu alanın başlangıç değeri **MQENC\_NATIVE**' dir.

### **ValueCCSID (MQHOT)**

Değer bir karakter dizilimiye, ayarlanacak özellik değerinin karakter kümesi.

Bu her zaman bir giriş alanıdır. Bu alanın başlangıç değeri **MQCCSI\_APPL**' dir.


### **MQSRO-Abonelik isteği seçenekleri**

MQSRO yapısı, uygulamanın abonelik isteğinin nasıl yapılacağını denetleyen seçenekleri belirtmesini sağlar. Yapı, MQSUBRQ çağrısında bir giriş/çıkış değiştirgesidir.

### **Kullanılabilirlik**

MQSRO yapısı aşağıdaki altyapılarda bulunur:

-  AIX

-  IBM i
-  Linux
-  Windows
-  z/OS

ve bu sistemlere baęlı IBM MQ MQI clients için.

## Sürüm

MQSRO ' nun yürürlükteki sürümü: MQSRO\_VERSION\_1.

## Karakter kümesi ve kodlama

MQSRO ' daki veriler, **CodedCharSetId** kuyruk yöneticisi öznitelięi tarafından belirtilen karakter kümesinde ve MQENC\_NATIVE tarafından verilen yerel kuyruk yöneticisinin kodlamasında yer almalıdır. Ancak, uygulama bir MQ MQI istemcisi olarak çalışıyorsa, yapı istemcinin karakter kümesinde ve kodlamasında olmalıdır.

## Alanlar

**Not:** Aşağıdaki tabloda, alanlar alfabetik olarak deęil, kullanıma göre gruplanır. Alt konular aynı sırayı izler.

Alan adı ve açıklaması	Deęişmezin adı	Deęişmezin ilk deęeri (varsa)
StrucId (yapı tanıtıcısı)	MQSRO_STRUC_ID	' SRO- '
Sürüm (yapı sürümü numarası)	MQSRO_VERSION_1	1
Seçenekler (seçenekler)	MQSRO_NONE	0
NumPubs (yayın sayısı)	Yok	0

**Notlar:**

1. - simgesi, tek bir boş karakteri temsil eder.
2. C programlama dilinde, makro deęişkeniMQSRO\_DEFAULT, çizelgede listelenen deęerleri içerir. Yapıdaki alanlar için başlangıç deęerleri sağlamak üzere aşağıdaki şekilde kullanılabilir:

```
MQSRO MySRO = {MQSRO_DEFAULT};
```

## Dil bildirimleri

MQSRO için C bildirimi

```
typedef struct tagMQSRO MQSRO;
struct tagMQSRO {
    MQCHAR4    StrucId;           /* Structure identifier */
    MQLONG     Version;         /* Structure version number */
    MQLONG     Options;         /* Options that control the action of MQSUBRQ */
    MQLONG     NumPubs;         /* Number of publications sent */
    /* Ver:1 */
};
```

MQSRO için COBOL bildirimi

```
** MQSRO structure
10 MQSRO.
```

```

** Structure identifier
15 MQSRO-STRUCID          PIC X(4).
** Structure version number
15 MQSRO-VERSION          PIC S9(9) BINARY.
** Options that control the action of MQSUBRQ
15 MQSRO-OPTIONS          PIC S9(9) BINARY.
** Number of publications sent
15 MQSRO-NUMPUBS          PIC S9(9) BINARY.

```

### MQSRO için PL/I bildirimi

```

dcl
1 MQSRO based,
3 StrucId          char(4),          /* Structure identifier */
3 Version          fixed bin(31),    /* Structure version number */
3 Options          fixed bin(31),    /* Options that control the action of MQSUBRQ */
3 NumPubs          fixed bin(31);    /* Number of publications sent */

```

### MQSRO için High Level Assembler bildirimi

```

MQSRO          DSECT
MQSRO_STRUCID  DS    CL4    Structure identifier
MQSRO_VERSION  DS    F      Structure version number
MQSRO_OPTIONS  DS    F      Options that control the action of MQSUBRQ
MQSRO_NUMPUBS  DS    F      Number of publications sent
*
MQSRO_LENGTH   EQU    *-MQSRO
               ORG    MQSRO
MQSRO_AREA     DS    CL(MQSRO_LENGTH)

```

### **StrucId (MQCHAR4)**

Bu, yapı tanıtıcısıdır; değer şöyle olmalıdır:

#### **MQSRO\_STRUC\_ID**

Abonelik İsteği Seçenekleri yapısına ilişkin tanıtıcı.

C programlama dili için, değişmez MQSRO\_STRUC\_ID\_ARRAY değişmezi de tanımlanır; bu, MQSRO\_STRUC\_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisi olur.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQSRO\_STRUC\_ID ' dir.

### **Sürüm (MQUZE)**

Bu, yapı sürüm numarasıdır; değer şu olmalıdır:

#### **MQSRO\_VERSION\_1**

Version-1 Abonelik İsteği Seçenekleri yapısı.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

#### **MQSRO\_CURRENT\_VERSION**

Abonelik İsteği Seçenekleri yapısının geçerli sürümü.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQSRO\_VERSION\_1' dir.

### **Seçenekler (MQUZE)**

Aşağıdaki seçeneklerden biri belirtilmeli. Tek bir seçenek belirtilebilir.

#### **MQSRO\_FAIL\_IF QUIESCING**

Kuyruk yöneticisi susturma durumdaysa, MQSUBRQ çağrısı başarısız olur. z/OSüzerinde, CICS ya da IMS uygulaması için bu seçenek, bağlantının susturulmuş durumda olması durumunda MQSUBRQ çağrısını da zorlar.

**Varsayılan seçenek:** Önceden açıklanan seçenek gerekmiyorsa, aşağıdaki seçeneğin kullanılması gerekir:



## MQSRO\_YOK

Başka bir seçenek belirtilmemeyi belirtmek için bu değeri kullanın; tüm seçenekler varsayılan değerlerini kabul eder.

MQSRO\_NONE, program belgelerine yardımcı olur. Bu seçeneğin başka biriyle birlikte kullanılması amaçlanmasa da, değeri sıfır olduğu için bu kullanım saptanamaz.

## NumPubs (MQUZE)

Bu bir çıkış alanıdır ve bu çağrının sonucu olarak abonelik kuyruğuna gönderilen yayın sayısını belirtmek için uygulamaya geri döndürülür. bu çağrı sayısının bir sonucu olarak bu sayıda yayın gönderilmesine rağmen, bu kadar çok mesajın, özellikle de kalıcı olmayan mesajlar olması halinde, uygulamanın elde edilmesi için bu kadar çok mesaja ulaşabileceğine dair bir garanti yoktur.

Konu joker karakterle içerildiye birden çok yayın olabilir. If no wildcards were present in the topic string when the subscription represented by *Hsub* was created, then at most one publication is sent as a result of this call.

## MQSTS-Durum raporlama yapısı

MQSTS yapısı, MQSTAT komutunun bir çıkış değiştirgesidir. Durum bilgilerini almak için MQSTAT komutu kullanılır. Bu bilgiler bir MQSTS yapısında döndürülür.

## Karakter kümesi ve kodlama

MQSTS içindeki karakter verileri yerel kuyruk yöneticisinin karakter kümesinde; bu, *CodedCharSetId* kuyruk yöneticisi özneteliği tarafından verilir. MQSTS ' deki sayısal veriler yerli makine kodlamasında; bu, *Kodlamatarafından* verilir.

## Alanlar

**Not:** Aşağıdaki tabloda, alanlar alfabetik olarak değil, kullanıma göre gruplanır. Alt konular aynı sırayı izler.

Çizelge 533. MQSTS ' deki Alanlar		
Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
<u>StrucId</u> (yapı tanıtıcısı)	MQSTS_STRUC_ID	'STAT-'
<u>Sürüm</u> (yapı sürümü numarası)	MQSTS_VERSION_1	1
<u>CompCode</u> (ilk hatanın tamamlanma kodu)	MQCC_OK	0
<u>Neden</u> (ilk hatanın neden kodu)	MQRC_NONE	0
<u>PutSuccessCount</u> (başarılı zamanuyumsuz koyma çağrısı sayısı)	Yok	0
<u>PutWarningCount</u> (uyarıları olan zamanuyumsuz koyma çağrılarının sayısı)	Yok	0
<u>PutFailureCount</u> (başarısız zamanuyumsuz koyma çağrısı sayısı)	Yok	0
<u>ObjectType</u> (arızalı nesnenin tipi)	MQOT_Q	1
<u>ObjectName</u> (başarısız olan nesnenin adı)	Yok	Boş değerli dizilim ya da boşluklar
<u>ObjectQMgrAd</u> (arızalı nesnenin sahibi olan kuyruk yöneticisinin adı)	Yok	Boş değerli dizilim ya da boşluklar

Çizelge 533. MQSTS 'deki Alanlar (devamı var)

Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
<u>ResolvedObjectName</u> (hedef kuyruğun çözülmüş adı)	Yok	Boş değerli dizilim ya da boşluklar
<u>ResolvedQMgrAdı</u> (hedef kuyruk yöneticisinin çözümlenen adı)	Yok	Boş değerli dizilim ya da boşluklar
<b>Not:</b> Sürüm MQSTS_VERSION_2' den küçükse kalan alanlar yoksayılır.		
<u>ObjectString</u> (başarısız olan nesnenin uzun nesne adı)	MQCHARV_DEFAULT	{NULL,0,0,0,-3}
<u>SubName</u> (başarısız olan aboneliğin abonelik adı)	MQCHARV_DEFAULT	{NULL,0,0,0,-3}
<u>OpenOptions</u> (hatayla ilişkili açma seçenekleri)	Yok	0
<u>SubOptions</u> (hatayla ilişkili abonelik seçenekleri)	Yok	0
<b>Notlar:</b> 1. ~ simgesi, tek bir boş karakteri temsil eder. 2. Boş değerli dizilim ya da boşluk değeri, C dizgisindeki boş dizgiyi ve diğer programlama dillerindeki boş karakterleri gösterir. 3. C programlama dilinde, MQSTS_DEFAULT makro değişkeni çizelgede listelenen değerleri içerir. Yapıdaki alanlar için başlangıç değerleri sağlamak üzere aşağıdaki şekilde kullanılabilir:  <pre>MQSTS MySTS = {MQSTS_DEFAULT};</pre>		

## Dil bildirimleri

### MQSTS için C bildirimi

```
typedef struct tagMQSTS MQSTS;  
struct tagMQSTS {  
    MQCHAR4    StrucId;           /* Structure identifier */  
    MQLONG     Version;          /* Structure version number */  
    MQLONG     CompCode;         /* Completion Code of first error */  
    MQLONG     Reason;          /* Reason Code of first error */  
    MQLONG     PutSuccessCount;  /* Number of Async calls succeeded */  
    MQLONG     PutWarningCount; /* Number of Async calls had warnings */  
    MQLONG     PutFailureCount; /* Number of Async calls had failures */  
    MQLONG     ObjectType;       /* Failing object type */  
    MQCHAR48   ObjectName;       /* Failing object name */  
    MQCHAR48   ObjectQMgrName;   /* Failing object queue manager name */  
    MQCHAR48   ResolvedObjectName; /* Resolved name of destination queue */  
    MQCHAR48   ResolvedQMgrName; /* Resolved name of destination qmgr */  
    /* Ver:1 */  
    MQCHARV    ObjectString;     /* Failing object long name */  
    MQCHARV    SubName;         /* Failing subscription name */  
    MQLONG     OpenOptions;     /* Failing open options */  
    MQLONG     SubOptions;      /* Failing subscription options */  
    /* Ver:2 */  
};
```

### MQSTS için COBOL bildirimi

```
** MQSTS structure  
   10 MQSTS.  
** Structure identifier  
   15 MQSTS-STRUCID PIC X(4).  
** Structure version number
```

```

15 MQSTS-VERSION PIC S9(9) BINARY.
** Completion Code of first error
15 MQSTS-COMPCODE PIC S9(9) BINARY.
** Reason Code of first error
15 MQSTS-REASON PIC S9(9) BINARY.
** Number of Async put calls succeeded
15 MQSTS-PUTSUCCESSCOUNT PIC S9(9) BINARY.
** Number of Async put calls had warnings
15 MQSTS-PUTWARNINGCOUNT PIC S9(9) BINARY.
** Number of Async put calls had failures
15 MQSTS-PUTFAILURECOUNT PIC S9(9) BINARY.
** Failing object type
15 MQSTS-OBJECTTYPE PIC S9(9) BINARY.
** Failing object name
15 MQSTS-OBJECTNAME PIC X(48).
** Failing object queue manager
15 MQSTS-OBJECTQMGRNAME PIC X(48).
** Resolved name of destination queue
15 MQSTS-RESOLVEDOBJECTNAME PIC X(48).
** Resolved name of destination qmgr
15 MQSTS-RESOLVEDQMGRNAME PIC X(48).
** Ver:1 **
** Failing object long name
15 MQSTS-OBJECTSTRING.
** Address of variable length string
20 MQSTS-OBJECTSTRING-VSPTR POINTER.
** Offset of variable length string
20 MQSTS-OBJECTSTRING-VSOFFSET PIC S9(9) BINARY.
** Size of buffer
20 MQSTS-OBJECTSTRING-VSBUFSIZE PIC S9(9) BINARY.
** Length of variable length string
20 MQSTS-OBJECTSTRING-VSLENGTH PIC S9(9) BINARY.
** CCSID of variable length string
20 MQSTS-OBJECTSTRING-VSCCSID PIC S9(9) BINARY.
** Failing subscription name
15 MQSTS-SUBNAME.
** Address of variable length string
20 MQSTS-SUBNAME-VSPTR POINTER.
** Offset of variable length string
20 MQSTS-SUBNAME-VSOFFSET PIC S9(9) BINARY.
** Size of buffer
20 MQSTS-SUBNAME-VSBUFSIZE PIC S9(9) BINARY.
** Length of variable length string
20 MQSTS-SUBNAME-VSLENGTH PIC S9(9) BINARY.
** CCSID of variable length string
20 MQSTS-SUBNAME-VSCCSID PIC S9(9) BINARY.
** Failing open options
15 MQSTS-OPENOPTIONS PIC S9(9) BINARY.
** Failing subscription options
15 MQSTS-SUBOPTIONS PIC S9(9) BINARY.
** Ver:2 **

```

## MQSTS için PL/I bildirimi

```

dcl
1 MQSTS based,
3 StructId          char(4),          /* Structure identifier */
3 Version           fixed bin(31),   /* Structure version number */
3 CompCode          fixed bin(31),   /* Completion code */
3 Reason            fixed bin(31),   /* Reason code */
3 PutSuccessCount   fixed bin(31),   /* Put success count */
3 PutWarningCount   fixed bin(31),   /* Put warning count */
3 PutFailureCount   fixed bin(31),   /* Put failure count */
3 ObjectType        fixed bin(31),   /* Object type */
3 ObjectName        char(48),        /* Object name */
3 ObjectQmgrName    char(48),        /* Object queue manager */
3 ResolvedObjectName char(48),       /* Resolved Object name */
3 ResolvedQmgrName  char(48);       /* Resolved Object queue manager */
/* Ver:1 */
3 ObjectString,    /* Failing object long name */
5 VSPtr pointer,  /* Address of variable length string */
5 VSOffset fixed bin(31), /* Offset of variable length string */
5 VSBufSize fixed bin(31), /* Size of buffer */
5 VSLength fixed bin(31), /* Length of variable length string */
5 VSCCSID fixed bin(31); /* CCSID of variable length string */
3 SubName,        /* Failing subscription name */
5 VSPtr pointer,  /* Address of variable length string */
5 VSOffset fixed bin(31), /* Offset of variable length string */
5 VSBufSize fixed bin(31), /* Size of buffer */

```

```

    5 VSLength fixed bin(31), /* Length of variable length string */
    5 VSCCSID fixed bin(31); /* CCSID of variable length string */
    3 OpenOptions fixed bin(31), /* Failing open options */
    3 SubOptions fixed bin(31); /* Failing subscription options */
/* Ver:2 */

```

## MQSTS için High Level Assembler (Üst Düzey Çevirici) bildirimi

MQSTS	DSECT	
MQSTS_STRUCID	DS	CL4 Structure identifier
MQSTS_VERSION	DS	F Structure version number
MQSTS_COMPCODE	DS	F Completion code
MQSTS_REASON	DS	F Reason code
MQSTS_PUTSUCESSCOUNT	DS	F Success count
MQSTS_PUTWARNINGCOUNT	DS	F Warning count
MQSTS_PUTFAILURECOUNT	DS	F Failure count
MQSTS_OBJTYPE	DS	F Object type
MQSTS_OBJNAME	DS	CL48 Object name
MQSTS_OBJQMGR	DS	CL48 Object queue manager
MQSTS_ROBJNAME	DS	CL48 Resolved object name
MQSTS_ROBJQMGR	DS	CL48 Resolved object queue manager
MQSTS_OBJECTSTRING	DS	0F Force fullword alignment
MQSTS_OBJECTSTRING_VSPTR	DS	A Address of variable length string
MQSTS_OBJECTSTRING_VSOFFSET	DS	F Offset of variable length string
MQSTS_OBJECTSTRING_VSBUFFSIZE	DS	F Size of buffer
MQSTS_OBJECTSTRING_VSLENGTH	DS	F Length of variable length string
MQSTS_OBJECTSTRING_VSCCSID	DS	F CCSID of variable length string
MQSTS_OBJECTSTRING_LENGTH	EQU	*-MQSTS_OBJECTSTRING
	ORG	MQSTS_OBJECTSTRING
MQSTS_OBJECTSTRING_AREA	DS	CL(MQSTS_OBJECTSTRING_LENGTH)
*		
MQSTS_SUBNAME	DS	0F Force fullword alignment
MQSTS_SUBNAME_VSPTR	DS	A Address of variable length string
MQSTS_SUBNAME_VSOFFSET	DS	F Offset of variable length string
MQSTS_SUBNAME_VSBUFFSIZE	DS	F Size of buffer
MQSTS_SUBNAME_VSLENGTH	DS	F Length of variable length string
MQSTS_SUBNAME_VSCCSID	DS	F CCSID of variable length string
MQSTS_SUBNAME_LENGTH	EQU	*-MQSTS_SUBNAME
	ORG	MQSTS_SUBNAME
MQSTS_SUBNAME_AREA	DS	CL(MQSTS_SUBNAME_LENGTH)
*		
MQSTS_OPENOPTIONS	DS	F Failing open options
MQSTS_SUBOPTIONS	DS	F Failing subscription option
MQSTS_LENGTH	EQU	*-MQSTS
	ORG	MQSTS
MQSTS_AREA	DS	CL(MQSTS_LENGTH)

### İlgili başvurular

“MQSTAT-Durum bilgilerini al” sayfa 771

Durum bilgilerini almak için MQSTAT çağrısını kullanın. Döndürülen durum bilgisi tipi, çağrıda belirtilen Tip değeri tarafından belirlenir.

### StrucId (MQCHAR4)

Durum raporlama yapısına ilişkin tanıttıcı, MQSTS.

StrucId yapı tanıttıcısıdır. Değer şu olmalıdır:

#### MQSTS\_STRUC\_ID

Durum raporlama yapısına ilişkin tanıttıcı.

For the C programming language, the constant MQSTS\_STRUC\_ID\_ARRAY is also defined; this has the same value as MQSTS\_STRUC\_ID, but is an array of characters instead of a string.

StrucId her zaman bir giriş alanıdır. İlk değer MQSTS\_STRUC\_IDdeğeridir.

### Sürüm (MQUZE)

Yapı sürüm numarası.

Değer aşağıdakilerden biri olmalıdır:

#### MQSTS\_VERSION\_1

Sürüm 1 durumu raporlama yapısı.

## **MQSTS\_VERSION\_2**

Sürüm 2 durum raporlama yapısı.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

## **MQSTS\_CURRENT\_VERSION**

Durum raporlama yapısının geçerli sürümü. Geçerli sürüm: MQSTS\_VERSION\_2.

Version her zaman bir giriş alanıdır. İlk değer MQSTS\_VERSION\_1değeridir.

## **CompCode (MQUZN)**

Raporlanmakta olan işlemin tamamlanma kodu.

CompCode 'ın yorumu, MQSTAT **Type** parametresinin değerine bağlıdır.

## **MQSTAT\_TYPE\_ASYNC\_ERROR**

Bu, ObjectName' ta belirtilen nesneye ilişkin önceki bir zamanuyumsuz koyma işleminin sonucundaki tamamlanma kodudur.

## **MQSTAT\_TYPE\_RECONNECTION**

Bağlantı yeniden bağlanmaya devam ederse ya da yeniden bağlanmadıysa, bağlantının yeniden bağlanmasına neden olan tamamlanma kodudur.

Bağlantı şu anda bağlıysa, değer MQCC\_OKolur.

## **MQSTAT\_TYPE\_RECONNECTION\_ERROR**

Bağlantı yeniden bağlanmadıysa, yeniden bağlantının başarısız olmasına neden olan tamamlama kodudur.

Bağlantı şu anda bağlıysa ya da yeniden bağlanıyorsa, değer MQCC\_OKolur.

CompCode her zaman bir çıkış alanıdır. İlk değer MQCC\_OKdeğeridir.

## **Neden (MQUZE)**

Raporlanmakta olan işlemin neden kodu.

Reason 'ın yorumu, MQSTAT **Type** parametresinin değerine bağlıdır.

## **MQSTAT\_TYPE\_ASYNC\_ERROR**

This is the reason code resulting from a previous asynchronous put operation on the object specified in ObjectName.

## **MQSTAT\_TYPE\_RECONNECTION**

Bağlantı yeniden bağlanmaya devam ederse ya da yeniden bağlanmadıysa, yeniden bağlanmanın yeniden bağlanmasına neden olan neden kodudur.

Bağlantı şu anda bağlıysa, değer MQRC\_NONEolur.

## **MQSTAT\_TYPE\_RECONNECTION\_ERROR**

Bağlantı yeniden bağlanmayı başaramadıysa, yeniden bağlantının başarısız olmasına neden olan neden kodudur.

Bağlantı şu anda bağlıysa ya da yeniden bağlanıyorsa, değer MQRC\_NONEolur.

Reason , bir çıkış alanıdır. İlk değer MQRC\_NONEdeğeridir.

## **PutSuccessCount (MQLONG)**

Başarılı olan zamanuyumsuz koyma işlemlerinin sayısı.

PutSuccessCount değeri, MQSTAT **Type** parametresinin değerine bağlıdır.

### **MQSTAT\_TYPE\_ASYNC\_ERROR**

MQCC\_OK ile tamamlanan MQSTS yapısındaki nesneye zamanuysuz koyma işlemlerinin sayısı.

### **MQSTAT\_TYPE\_RECONNECTION**

Sıfır.

### **MQSTAT\_TYPE\_RECONNECTION\_ERROR**

Sıfır.

PutSuccessCount , bir çıkış alanıdır. Başlangıç değeri sıfır.

### **PutWarningSayı (MQUZE)**

Bir uyarıyla sona erdiren zamanuysuz koyma işlemlerinin sayısı.

PutWarningCount değeri, MQSTAT **Type** parametresinin değerine bağlıdır.

### **MQSTAT\_TYPE\_ASYNC\_ERROR**

MQCC\_WARNING ile tamamlanan MQSTS yapısındaki nesneye zamanuysuz koyma işlemlerinin sayısı.

### **MQSTAT\_TYPE\_RECONNECTION**

Sıfır.

### **MQSTAT\_TYPE\_RECONNECTION\_ERROR**

Sıfır.

PutWarningCount , bir çıkış alanıdır. Başlangıç değeri sıfır.

### **PutFailureCount (MQLONG)**

Başarısız olan zamanuysuz koyma işlemlerinin sayısı.

PutFailureCount değeri, MQSTAT **Type** parametresinin değerine bağlıdır.

### **MQSTAT\_TYPE\_ASYNC\_ERROR**

MQCC\_FAILED ile tamamlanan MQSTS yapısındaki nesneye zamanuysuz koyma işlemlerinin sayısı.

### **MQSTAT\_TYPE\_RECONNECTION**

Sıfır.

### **MQSTAT\_TYPE\_RECONNECTION\_ERROR**

Sıfır.

PutFailureCount , bir çıkış alanıdır. Başlangıç değeri sıfır.

### **ObjectType (MQUZE)**

Raporlanmakta olan *ObjectName* ' ta adı belirtilen nesnenin tipi.

Olası ObjectType değerleri, "[MQOT\\_ \\* \(Nesne Tipleri ve Genişletilmiş Nesne Tipleri\)](#)" sayfa 164 içinde listelenir.

ObjectType , bir çıkış alanıdır. İlk değer MQOT\_Q değeridir.

### **ObjectName (MQCHAR48)**

Raporlanmakta olan nesnenin adı.

ObjectName ' in yorumu, MQSTAT **Type** parametresinin değerine bağlıdır.

### **MQSTAT\_TYPE\_ASYNC\_ERROR**

This is the name of the queue or topic used in the put operation, the failure of which is reported in the *CompCode* and *Reason* fields in the MQSTS structure.

## **MQSTAT\_TYPE\_RECONNECTION**

Bağlantı yeniden bağlanıyorsa, bu, bağlantıyla ilişkilendirilmiş kuyruk yöneticisinin adıdır.

## **MQSTAT\_TYPE\_RECONNECTION\_ERROR**

Başlantı yeniden başlatılamazsa, yeniden başlatılamamasına neden olan nesne adlarıdır. Başarısızlığın nedeni, MQSTS yapısındaki *CompCode* ve *Reason* alanlarında raporlanır.

*ObjectName* , bir çıkış alanıdır. Başlangıç değeri, C ' deki boş değerli dizedir ve diğer programlama dillerinde 48 boş karakter.

### ***ObjectQMGrAdı (MQCHAR48)***

Raporlanmakta olan kuyruk yöneticisinin adı.

*ObjectQMGrName* ' ın yorumu, MQSTAT **Type** parametresinin değerine bağlıdır.

## **MQSTAT\_TYPE\_ASYNC\_ERROR**

Bu ad, *ObjectName* nesnesinin tanımlandığı kuyruk yöneticisinin adıdır. İlk boş karakteri ya da alanın sonuna kadar boş olan bir ad, uygulamanın bağlı olduğu kuyruk yöneticisini belirtir (yerel kuyruk yöneticisi).

V 9.2.0

## **MQSTAT\_TYPE\_RECONNECTION**

Multi

***ObjectQMGrName*** alanı, yeniden bağlantı istenmekte olan bir kuyruk yöneticisinin adını içerir ya da kuyruk yöneticisi belirtilmediyse boş değer alır. Olanaklıysa, istemci o adı içeren bir kuyruk yöneticisine yeniden bağlanmayı dener.

z/OS

Boş.

## **MQSTAT\_TYPE\_RECONNECTION\_ERROR**

Başlantı yeniden başlatılamazsa, yeniden başlatılamamasına neden olan nesne adlarıdır. Başarısızlığın nedeni, MQSTS yapısındaki *CompCode* ve *Reason* alanlarında raporlanır.

*ObjectQMGrName* , bir çıkış alanıdır. Bu değer, C içindeki boş değerli dizgidir ve diğer programlama dillerinde 48 boş karakter içerir.

### ***ResolvedObjectAdı (MQCHAR48)***

Yerel kuyruk yöneticisi adı çözümden sonra, *ObjectName* içinde belirtilen nesnenin adı.

*ResolvedObjectName* ' ın yorumu, MQSTAT **Type** parametresinin değerine bağlıdır.

## **MQSTAT\_TYPE\_ASYNC\_ERROR**

*ResolvedObjectName* , yerel kuyruk yöneticisi adı çözümden sonra *ObjectName* içinde belirtilen nesnenin adıdır. The name returned is the name of an object that exists on the queue manager identified by *ResolvedQMGrName*.

## **MQSTAT\_TYPE\_RECONNECTION**

Boş.

## **MQSTAT\_TYPE\_RECONNECTION\_ERROR**

Boş.

*ResolvedObjectName* , bir çıkış alanıdır. Başlangıç değeri, C ' deki boş değerli dizedir ve diğer programlama dillerinde 48 boş karakter.

### ***ResolvedQMGrAdı (MQCHAR48)***

Yerel kuyruk yöneticisinin adı çözümden sonra hedef kuyruk yöneticisinin adı.

*ResolvedQMGrName* ' ın yorumu, MQSTAT **Type** parametresinin değerine bağlıdır.

### **MQSTAT\_TYPE\_ASYNC\_ERROR**

ResolvedQMGrName , yerel kuyruk yöneticisinin adı çözümledikten sonra hedef kuyruk yöneticisinin adıdır. Döndürülen ad, *ResolvedObjectName* ile tanımlanan nesnenin sahibi olan kuyruk yöneticisinin adıdır. *ResolvedQMGrName* , yerel kuyruk yöneticisinin adı olabilir.

### **MQSTAT\_TYPE\_RECONNECTION**

Boş.

### **MQSTAT\_TYPE\_RECONNECTION\_ERROR**

Boş.

ResolvedQMGrName her zaman bir çıkış alanıdır. Başlangıç değeri, C ' deki boş değerli dizedir ve diğer programlama dillerinde 48 boş karakter.

### **ObjectString (MQCHARV)**

Raporlanmakta olan başarısız nesnenin uzun nesne adı. Yalnızca MQSTS ya da daha sonraki bir sürüm olan sürüm 2 'de bulunur.

ObjectString ' ın yorumu, MQSTAT **Type** parametresinin değerine bağlıdır.

### **MQSTAT\_TYPE\_ASYNC\_ERROR**

This is the long object name of the queue or topic used in the MQPUT operation, which failed.

### **MQSTAT\_TYPE\_RECONNECTION**

Sıfır uzunluklu dizgi

### **MQSTAT\_TYPE\_RECONNECTION\_ERROR**

Bu, yeniden bağlanmanın başarısız olmasına neden olan nesnenin uzun nesne adıdır.

ObjectString , bir çıkış alanıdır. Başlangıç değeri sıfır uzunluklu bir dizilimdir.

### **SubName (MQCHARV)**

Başarısız olan aboneliğin adı. Yalnızca MQSTS ya da daha sonraki bir sürüm olan sürüm 2 'de bulunur.

SubName ' ın yorumu, MQSTAT **Type** parametresinin değerine bağlıdır.

### **MQSTAT\_TYPE\_ASYNC\_ERROR**

Sıfır uzunluklu dizgi.

### **MQSTAT\_TYPE\_RECONNECTION**

Sıfır uzunluklu dizgi.

### **MQSTAT\_TYPE\_RECONNECTION\_ERROR**

Yeniden bağlantı başarısız olmasına neden olan aboneliğin adı. Herhangi bir abonelik adı yoksa ya da başarısızlık bir abonelik ile ilgili değilse, bu sıfır uzunluklu bir dizgidir.

SubName , bir çıkış alanıdır. Başlangıç değeri sıfır uzunluklu bir dizilimdir.

### **OpenOptions (MQUZE)**

The OpenOptions used to open the object being reported upon. Yalnızca MQSTS ya da daha sonraki bir sürüm olan sürüm 2 'de bulunur.

OpenOptions değeri, MQSTAT **Type** parametresinin değerine bağlıdır.

### **MQSTAT\_TYPE\_ASYNC\_ERROR**

Sıfır.

### **MQSTAT\_TYPE\_RECONNECTION**

Sıfır.



## **MQSTAT\_TYPE\_RECONNECTION\_ERROR**

Hata ortaya çıktığında OpenOptions kullanılır. Başarısızlığın nedeni, MQSTS yapısındaki *CompCode* ve *Reason* alanlarında raporlanır.

OpenOptions , bir çıkış alanıdır. Başlangıç değeri sıfır.

### **SubOptions (MQUZE)**

The SubOptions used to open the failing subscription. Yalnızca MQSTS ya da daha sonraki bir sürüm olan sürüm 2 'de bulunur.

SubOptions ' in yorumu, MQSTAT **Type** parametresinin değerine bağlıdır.

## **MQSTAT\_TYPE\_ASYNC\_ERROR**

Sıfır.

## **MQSTAT\_TYPE\_RECONNECTION**

Sıfır.

## **MQSTAT\_TYPE\_RECONNECTION\_ERROR**

Hata ortaya çıktığında SubOptions kullanılır. Hata bir konuya abone olma ile ilgili değilse, döndürülen değer sıfır olur.

SubOptions , bir çıkış alanıdır. Başlangıç değeri sıfır.

## **MQTM-Tetikleyici iletisi**

MQTM yapısı, kuyruk yöneticisi tarafından bir kuyruk için bir tetikleyici olayı oluştuğunda tetikleyici uygulamasına gönderilen tetikleyici iletideki verileri tanımlar. Bu yapı, IBM MQ çerçeve arabirimlerinden biri olan IBM MQ Trigger Monitor Interface (TMI) olanağının bir parçasıdır.

### **Biçim Adı**

MQFMT\_TRIGGER.

### **Karakter kümesi ve kodlama**



MQTM 'deki karakter verileri, MQTM' yi oluşturan kuyruk yöneticisinin karakter kümesinde yer aldı. MQTM 'deki sayısal veriler, MQTM' yi oluşturan kuyruk yöneticisinin makine kodlamasında yer aldı.

MQTM ' nin karakter kümesi ve kodlaması *CodedCharSetId* ve *Encoding* alanları tarafından aşağıdaki alanlarda verilir:

- MQMD (MQTM yapısı ileti verilerinin başındadır) ya da
- MQTM yapısından önce gelen üstbilgi yapısı (diğer tüm durumlar).

### **Kullanım**

Tetikleyici izleme uygulamasının, tetikleyici iletisindeki bilgilerin bir kısmını ya da tümünü tetikleyici izleme uygulamasının başlattığı uygulamaya aktarması gerekebilir. Başlatılan uygulamanın gereksinim duyabileceği bilgiler arasında *QName*, *TriggerDatave UserDatayer* alır. Tetikleyici izleyici uygulaması MQTM yapısını doğrudan başlatılan uygulamaya geçirebilir ya da ortamın izin verdiğine ve başlatılan uygulama için uygun olduğuna bağlı olarak bir MQTMC2 yapısı geçirebilir. MQTMC2 hakkında bilgi için bkz. “MQTMC2 -Tetikleyici iletisi 2 (karakter biçimi)” sayfa 596.

-  z/OS üzerinde, CKTI hareketi kullanılarak başlatılan bir MQAT\_CICS uygulaması için, tetikleyici iletisi yapısının tamamı MQTM başlatılmış hareket tarafından kullanılabilir; bilgiler EXEC CICS RETRIEVE komutu kullanılarak alınabilir.
-  IBM i sistemlerinde, IBM MQ ile sağlanan tetikleyici izleyicisi uygulaması, başlatılan uygulamaya bir MQTMC2 yapısı geçirir.

Tetikleyicilerin kullanılmasıyla ilgili bilgi için [Tetikleyici kullanarak IBM MQ uygulamalarının başlatılması](#) başlıklı konuya bakın.

## Alanlar

**Not:** Aşağıdaki tabloda, alanlar alfabetik olarak değil, kullanıma göre gruplanır. Alt konular aynı sırayı izler.

<i>Çizelge 534. MQTM for MQTM içindeki alanlar</i>		
Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
<u>StrucId</u> (yapı tanıtıcısı)	MQTM_STRUC_ID	'TM↵↵'
<u>Sürüm</u> (yapı sürümü numarası)	MQTM_VERSION_1	1
<u>QName</u> (tetiklenen kuyruğun adı)	Yok	Boş değerli dizilim ya da boşluklar
<u>ProcessName</u> (süreç nesnesinin adı)	Yok	Boş değerli dizilim ya da boşluklar
<u>TriggerData</u> (tetikleyici verileri)	Yok	Boş değerli dizilim ya da boşluklar
<u>ApplType</u> (uygulama tipi)	Yok	0
<u>ApplId</u> (uygulama tanıtıcısı)	Yok	Boş değerli dizilim ya da boşluklar
<u>EnvData</u> (ortam verileri)	Yok	Boş değerli dizilim ya da boşluklar
<u>UserData</u> (kullanıcı verileri)	Yok	Boş değerli dizilim ya da boşluklar

**Notlar:**

- ↵ simgesi, tek bir boş karakteri temsil eder.
- Boş değerli dizilim ya da boşluk değeri, C dizgisindeki boş dizgiyi ve diğer programlama dillerindeki boş karakterleri gösterir.
- C programlama dilinde, makro değişkeniMQTM\_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlar için başlangıç değerleri sağlamak üzere bunu aşağıdaki şekilde kullanın:

```
MQTM MyTM = {MQTM_DEFAULT};
```

## Dil bildirimleri

MQTM için C bildirimi

```
typedef struct tagMQTM MQTM;  
struct tagMQTM {  
    MQCHAR4    StrucId;        /* Structure identifier */  
    MQLONG     Version;        /* Structure version number */  
    MQCHAR48   QName;         /* Name of triggered queue */  
    MQCHAR48   ProcessName;   /* Name of process object */  
    MQCHAR64   TriggerData;   /* Trigger data */  
    MQLONG     ApplType;      /* Application type */  
    MQCHAR256  ApplId;        /* Application identifier */  
    MQCHAR128  EnvData;       /* Environment data */  
    MQCHAR128  UserData;      /* User data */  
};
```

## MQTM için COBOL bildirim

```
** MQTM structure
10 MQTM.
** Structure identifier
15 MQTM-STRUCID PIC X(4).
** Structure version number
15 MQTM-VERSION PIC S9(9) BINARY.
** Name of triggered queue
15 MQTM-QNAME PIC X(48).
** Name of process object
15 MQTM-PROCESSNAME PIC X(48).
** Trigger data
15 MQTM-TRIGGERDATA PIC X(64).
** Application type
15 MQTM-APPLTYPE PIC S9(9) BINARY.
** Application identifier
15 MQTM-APPLID PIC X(256).
** Environment data
15 MQTM-ENVDATA PIC X(128).
** User data
15 MQTM-USERDATA PIC X(128).
```

## MQTM için PL/I bildirim

```
dcl
1 MQTM based,
3 StrucId char(4), /* Structure identifier */
3 Version fixed bin(31), /* Structure version number */
3 QName char(48), /* Name of triggered queue */
3 ProcessName char(48), /* Name of process object */
3 TriggerData char(64), /* Trigger data */
3 ApplType fixed bin(31), /* Application type */
3 ApplId char(256), /* Application identifier */
3 EnvData char(128), /* Environment data */
3 UserData char(128); /* User data */
```

## MQTM için High Level Assembler (Üst Düzey Çevirici) bildirim

```
MQTM          DSECT
MQTM_STRUCID  DS CL4   Structure identifier
MQTM_VERSION  DS F     Structure version number
MQTM_QNAME    DS CL48  Name of triggered queue
MQTM_PROCESSNAME DS CL48 Name of process object
MQTM_TRIGGERDATA DS CL64 Trigger data
MQTM_APPLTYPE DS F     Application type
MQTM_APPLID   DS CL256 Application identifier
MQTM_ENVDATA  DS CL128 Environment data
MQTM_USERDATA DS CL128 User data
*
MQTM_LENGTH   EQU *-MQTM
              ORG MQTM
MQTM_AREA     DS CL(MQTM_LENGTH)
```

## MQTM için Visual Basic bildirim

```
Type MQTM
StrucId As String*4 'Structure identifier'
Version As Long 'Structure version number'
QName As String*48 'Name of triggered queue'
ProcessName As String*48 'Name of process object'
TriggerData As String*64 'Trigger data'
ApplType As Long 'Application type'
ApplId As String*256 'Application identifier'
EnvData As String*128 'Environment data'
UserData As String*128 'User data'
End Type
```

## Tetikleyici iletisine ilişkin MQMD

Çizelge 535. Kuyruk yöneticisi tarafından oluşturulan bir tetikleyici iletisine ilişkin MQMD 'deki alanlara ilişkin ayarlar

MQMD 'deki Alan	Kullanılan değer
<i>StrucId</i>	MQMD_STRUC_ID
<i>Version</i>	MQMD_VERSION_1
<i>Report</i>	MQRO_NONE
<i>MsgType</i>	MQMT_DATAGRAM
<i>Expiry</i>	MQEI_UNLIMITED
<i>Feedback</i>	MQFB_NONE
<i>Encoding</i>	MQENC_NATIVE
<i>CodedCharSetId</i>	Kuyruk yöneticisinin <b>CodedCharSetId</b> özneliği
<i>Format</i>	MQFMT_TRIGGER
<i>Priority</i>	Başlatma kuyruğunun <b>DefPriority</b> özneliği
<i>Persistence</i>	MQPER_NOT_KALICI
<i>MsgId</i>	Benzersiz bir değer
<i>CorrelId</i>	MQCI_NONE
<i>BackoutCount</i>	0
<i>ReplyToQ</i>	Boşluklar
<i>ReplyToQMGr</i>	Kuyruk yöneticisinin adı
<i>UserIdentifier</i>	Boşluklar
<i>AccountingToken</i>	MQACT_NONE
<i>ApplIdentityData</i>	Boşluklar
<i>PutApplType</i>	MQAT_QMGR ya da ileti kanalı aracısı için uygun olduğu şekilde
<i>PutApplName</i>	Kuyruk yöneticisi adının ilk 28 baytı
<i>PutDate</i>	Tetikleyici iletisinin gönderildiği tarih
<i>PutTime</i>	Tetikleyici iletisinin gönderildiği zaman
<i>ApplOriginData</i>	Boşluklar

Aşağıdakiler dışında benzer değerleri ayarlamak için bir tetikleyici iletisi oluşturan bir uygulama önerilir:

- *Priority* alanı MQPRI\_PRIORITY\_AS\_Q\_DEF olarak ayarlanabilir (kuyruk yöneticisi, ileti bulunduğu başlatma kuyruğu için varsayılan önceliğe çevirir).
- *ReplyToQMGr* alanı boş olarak ayarlanabilir (kuyruk yöneticisi, yerleştirdiği iletide bu değeri yerel kuyruk yöneticisinin adına çevirir).
- Bağlam alanlarını uygulama için uygun şekilde ayarlayın.

### **StrucId (MQCHAR4)**

Bu, yapı tanıtıcısıdır. Değer şu olmalıdır:

#### **MQTM\_STRUC\_ID**

Tetikleyici ileti yapısına ilişkin tanıtıcı.

C programlama dili için, sabit MQTM\_STRUC\_ID\_ARRAY değişmezi de tanımlanır; bu, MQTM\_STRUC\_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisi olur.

Bu alanın ilk değeri MQTM\_STRUC\_ID 'dir.

### ***MQTM\_VERSION\_1 (MQUZE)***

Bu, yapı sürüm numarasıdır. Değer şu olmalıdır:

#### **MQTM\_VERSION\_1**

Tetikleyici ileti yapısına ilişkin sürüm numarası.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

#### **MQTM\_CURRENT\_VERSION**

Tetikleyici ileti yapısının yürürlükteki sürümü.

Bu alanın ilk değeri MQTM\_VERSION\_1' dir.

### ***QName (MQCHAR48)***

Tetikleme olayının oluşturduğu kuyruğun adı ve uygulama tarafından başlatılan uygulama tarafından tetikleme izleme uygulaması tarafından kullanılır. Kuyruk yöneticisi bu alanı, tetiklenen kuyruğun **QName** özneliğinin değeriyle kullanıma hazırlar; bu özneliğe ilişkin ayrıntılar için [“Kuyruklara ilişkin öznelikler” sayfa 821](#) ' e bakın.

Alanın tanımlanan uzunluğundan daha kısa olan adlar boşluklarla doldurulur ve boş bir karakter karakteriyle sona erdirilmez.

Bu alanın uzunluğu MQ\_Q\_NAME\_LENGTH değeri tarafından verilir. Bu alanın başlangıç değeri, C ' de boş değer dizgisidir ve diğer programlama dillerindeki 48 boş karakter dizilimidir.

### ***ProcessName (MQCHAR48)***

Bu, tetiklenen kuyruk için belirtilen kuyruk yöneticisi işlemi nesnesinin adıdır ve tetikleme iletisini alan tetikleyici izleme programı tarafından kullanılabilir. Kuyruk yöneticisi, bu alanı, **QName** alanı tarafından tanımlanan kuyruğun **ProcessName** özneliğinin değeriyle birlikte başlatır; bu özneliğin ayrıntıları için [“Kuyruklara ilişkin öznelikler” sayfa 821](#) ' e bakın.

Alanın tanımlanan uzunluğundan daha kısa olan adlar her zaman boşluklarla sağa doldurulur; boş bir karakter karakterinden önce sona erdirilmez.

Bu alanın uzunluğu MQ\_PROCESS\_NAME\_LENGTH tarafından verilir. Bu alanın başlangıç değeri, C ' de boş değer dizgisidir ve diğer programlama dillerindeki 48 boş karakter dizilimidir.

### ***TriggerData (MQCHAR64)***

Bu, tetikleme iletisini alan, tetikleme izleme uygulaması tarafından kullanılacak serbest biçimli verilerdir. Kuyruk yöneticisi, bu alanı, **QName** alanı tarafından tanımlanan kuyruğun **TriggerData** özneliğinin değeriyle birlikte başlatır; bu özneliğin ayrıntıları için [“Kuyruklara ilişkin öznelikler” sayfa 821](#) ' e bakın. Bu verilerin içeriği kuyruk yöneticisi için bir önem göstermiyor.

z/OS üzerinde, bir CICS uygulaması için CKTI işlemi kullanılarak başlatılan bu bilgiler kullanılmaz.

Bu alanın uzunluğu MQ\_TRIGGER\_DATA\_LENGTH tarafından verilir. Bu alanın başlangıç değeri, C içindeki boş değerli dizedir ve diğer programlama dillerindeki 64 boş karakterdir.

### ***ApplType (MQUBE)***

Bu, başlatılacak programın niteliyi tanıtır ve tetikleme iletisini alan tetikleme izleme programı uygulaması tarafından kullanılır. Kuyruk yöneticisi, bu alanı, **ProcessName** alanı tarafından tanımlanan süreç nesnesinin **ApplType** özneliğinin değeriyle birlikte başlatır; bu özneliğe ilişkin ayrıntılar için [“Süreç tanımlamalarına ilişkin öznelikler” sayfa 856](#) ' e bakın. Bu verilerin içeriği kuyruk yöneticisi için bir önem göstermiyor.

*AppType* , aşağıdaki standart değerlerden birine sahip olabilir. Kullanıcı tanımlı tipler de kullanılabilir, ancak MQAT\_USER\_FIRST ile MQAT\_USER\_SON ile değerler arasında sınırlandırılmalıdır:

**MQAT\_AIX**

AIX uygulaması (MQAT\_UNIX ile aynı değer).

**MQAT\_BATCH**

Toplu iş uygulaması

**MQAT\_BROKER**

Aracı uygulaması

**MQAT\_CICS**

CICS işlemi.

**MQAT\_CICS\_BRIDGE**

CICS bridge uygulaması.

**MQAT\_CICS\_VSE**

CICS/VSE işlemi.

**MQAT\_DOS**

PC DOS üzerindeIBM MQ MQI client uygulaması.

**MQAT\_IMS**

IMS uygulaması.

**MQAT\_IMS\_BRIDGE**

IMS köprü uygulaması.

**MQAT\_JAVA**

Java uygulaması.

**MQAT\_MVS**

MVS ya da TSO uygulaması (MQAT\_ZOS ile aynı değer).

**MQAT\_NOTES\_AGENT**

Lotus Notes Agent uygulaması.

**MQAT\_OS390**

OS/390 uygulaması (MQAT\_ZOS ile aynı değer).

**MQAT\_OS400**

IBM i uygulaması.

**MQAT\_RRS\_BATCH**

RRS toplu iş uygulaması.

**MQAT\_UNIX**

UNIX uygulaması.

**MQAT\_BILINMIYOR**

Bilinmeyen tipte uygulama.

**MQAT\_USER**

Kullanıcı tanımlı uygulama tipi.

**MQAT\_VOS**

Stratus VOS uygulaması.

**MQAT\_WINDOWS**

16 bit Windows uygulaması.

**MQAT\_WINDOWS\_NT**

32 bit Windows uygulaması.

**MQAT\_WLM**

z/OS iş yükü yöneticisi uygulaması.

**MQAT\_XCF**

XCF.

## **MQAT\_ZOS**

z/OS uygulaması.

## **MQAT\_USER\_FIRST**

Kullanıcı tanımlı uygulama tipi için en düşük değer.

## **MQAT\_USER\_SON**

Kullanıcı tanımlı uygulama tipi için en yüksek değer.

Bu alanın başlangıç değeri 0' dir.

## **ApplId (MQCHAR256)**

Bu, başlatılacak uygulamayı tanıtan ve tetikleme iletisini alan tetikleme izleme programı uygulaması tarafından kullanılan bir karakter dizilimidir. Kuyruk yöneticisi, bu alanı, *ProcessName* alanı tarafından tanımlanan süreç nesnesinin **ApplId** özniteliğinin değeriyle birlikte başlatır; bu özniteliğe ilişkin ayrıntılar için “Süreç tanımlamalarına ilişkin öznitelikler” sayfa 856 ' e bakın. Bu verilerin içeriği kuyruk yöneticisi için bir önem göstermiyor.

The meaning of *ApplId* is determined by the trigger-monitor application. The trigger monitor provided by IBM MQ requires *ApplId* to be the name of an executable program. Aşağıdaki notlarda belirtilen ortamlar için geçerlidir:

- z/OS üzerinde, *ApplId* şudur:
  - CICS tetikleyicisi kullanılarak başlatılan uygulamalar için bir CICS işlem tanıtıcısı. CKTI
  - IMS tetikleyici izleyicisi CSQQTRMN kullanılarak başlatılan uygulamalar için bir IMS işlem tanıtıcısı
- Windows sistemlerinde program adının başına bir sürücü ve izin yolu eklenerek önek olarak eklenebilirsiniz.
- IBM üzerinde, program adının başına bir kitaplık adı ve/karakteri eklenerek eklenmelidir.
- AIX and Linux üzerinde, program adının başına bir izin yolu önek olarak eklenebilir.

Bu alanın uzunluğu MQ\_PROCESS\_APPL\_ID\_LENGTH tarafından verilir. Bu alanın ilk değeri, C ' deki boş değerli dizedir ve diğer programlama dillerindeki 256 boş karakterdir.

## **EnvData (MQCHAR128)**

Bu, başlatılacak uygulamaya ilişkin ortamla ilgili bilgileri içeren ve tetikleme iletisini alan tetikleyici izleme uygulaması tarafından kullanılan bir karakter dizilimidir. Kuyruk yöneticisi, bu alanı, *ProcessName* alanı tarafından tanımlanan süreç nesnesinin **EnvData** özniteliğinin değeriyle birlikte başlatır; bu özniteliğe ilişkin ayrıntılar için “Süreç tanımlamalarına ilişkin öznitelikler” sayfa 856 ' e bakın. Bu verilerin içeriği kuyruk yöneticisi için bir önem göstermiyor.

z/OS üzerinde, bir CICS uygulaması CKTI işlemi kullanılarak başlatılmış ya da CSQQTRMN işlemi kullanılarak başlatılacak bir IMS uygulaması için bu bilgiler kullanılmaz.

Bu alanın uzunluğu MQ\_PROCESS\_ENV\_DATA\_LENGTH tarafından verilir. Bu alanın ilk değeri, C ' de boş değer dizgisidir ve diğer programlama dillerinde 128 boşluk karakteridir.

## **UserData (MQCHAR128)**

Bu, başlatılacak uygulamayla ilgili kullanıcı bilgilerini içeren ve tetikleme iletisini alan tetikleyici izleme uygulaması tarafından kullanılan bir karakter dizilimidir. Kuyruk yöneticisi, bu alanı, *ProcessName* alanı tarafından tanımlanan süreç nesnesinin **UserData** özniteliğinin değeriyle birlikte başlatır; bu özniteliğe ilişkin ayrıntılar için “Süreç tanımlamalarına ilişkin öznitelikler” sayfa 856 ' e bakın. Bu verilerin içeriği kuyruk yöneticisi için bir önem göstermiyor.

For Microsoft Windows, the character string must not contain double quotation marks if the process definition is going to be passed to **runmqtrm**.

Bu alanın uzunluğu MQ\_PROCESS\_USER\_DATA\_LENGTH tarafından verilmektedir. Bu alanın ilk değeri, C ' de boş değer dizgisidir ve diğer programlama dillerinde 128 boşluk karakteridir.

## MQTM2 -Tetikleyici ileti 2 (karakter biçimi)

Tetikleyici uygulaması başlatma kuyruğundan bir tetikleyici iletisi (MQTM) aldığımda, tetikleyici izleme programının tetikleyici iletisindeki bilgilerin bir kısmını ya da tümünü tetikleyici izleyicinin başlatıldığı uygulamaya geçirmesi gerekebilir.

Başlatılan uygulamanın gereksinim duyabileceği bilgiler arasında *QName*, *TriggerData*ve *UserData*yer alır. Tetikleyici izleyici uygulaması MQTM yapısını doğrudan başlatılan uygulamaya geçirebilir ya da ortamın izin verdiğine ve başlatılan uygulama için uygun olduğuna bağlı olarak bir MQTM2 yapısı geçirebilir.

Bu yapı, IBM MQ çerçeve arabirimlerinden biri olan IBM MQ Trigger Monitor Interface (TMI) olanağının bir parçasıdır.

### Karakter kümesi ve kodlama

MQTM2 içindeki karakter verileri yerel kuyruk yöneticisinin karakter kümesinde; bu, **CodedCharSetId** kuyruk yöneticisi özniteliği tarafından verilir.

### Kullanım

MQTM2 yapısı, MQTM yapısının biçimine çok benzer. Fark, MQTM 'deki karakter olmayan alanların MQTM2 ' de aynı uzunlukta karakter alanlarına değiştirilmesi ve kuyruk yöneticisi adının yapının sonuna eklenmeleridir.

- **z/OS** z/OSüzerinde, CSQQTRMN uygulaması kullanılarak başlatılan bir MQAT\_IMS uygulaması için, başlatılan uygulama için bir MQTM2 yapısı kullanılır kılmıştır.
- **IBM i** IBM sistemlerinde, IBM MQ ile sağlanan tetikleyici izleyicisi uygulaması, başlatılan uygulamaya bir MQTM2 yapısı geçirir.

### Alanlar

**Not:** Aşağıdaki tabloda, alanlar alfabetik olarak değil, kullanıma göre gruplanır. Alt konular aynı sırayı izler.

Çizelge 536. MQTM2 içindeki alanlar		
Alan adı ve açıklaması	Değişmez adı	Değişmez ilk değeri (varsa)
<u>StrucId</u> (yapı tanıtıcısı)	MQTM2_STRUC_ID	' TMC2 '
<u>Sürüm</u> (yapı sürümü numarası)	MQTM2_VERSION_2	' 2222 '
<u>QName</u> (tetiklenen kuyruğun adı)	Yok	Boş değerli dizilim ya da boşluklar
<u>ProcessName</u> (süreç nesnesinin adı)	Yok	Boş değerli dizilim ya da boşluklar
<u>TriggerData</u> (tetikleyici verileri)	Yok	Boş değerli dizilim ya da boşluklar
<u>ApplType</u> (uygulama tipi)	Yok	Boşluklar
<u>AppId</u> (uygulama tanıtıcısı)	Yok	Boş değerli dizilim ya da boşluklar
<u>EnvData</u> (ortam verileri)	Yok	Boş değerli dizilim ya da boşluklar
<u>UserData</u> (kullanıcı verileri)	Yok	Boş değerli dizilim ya da boşluklar



Çizelge 536. MQTMC2 içindeki alanlar (devamı var)

Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
QMgrName (kuyruk yöneticisi adı)	Yok	Boş değerli dizilim ya da boşluklar

**Notlar:**

1. ↯ simgesi, tek bir boş karakteri temsil eder.
2. Boş değerli dizilim ya da boşluk değeri, C dizgisindeki boş dizgiyi ve diğer programlama dillerindeki boş karakterleri gösterir.
3. C programlama dilinde, makro değişkeniMQTMC2\_DEFAULT , yukarıda listelenen değerleri içerir. Yapıdaki alanlar için başlangıç değerleri sağlamak üzere bunu aşağıdaki şekilde kullanın:

```
MQTMC2 MyTMC = {MQTMC2_DEFAULT};
```

## Dil bildirimleri

### MQTMC2 için C bildirimi

```
typedef struct tagMQTMC2 MQTMC2;
struct tagMQTMC2 {
    MQCHAR4    StrucId;        /* Structure identifier */
    MQCHAR4    Version;       /* Structure version number */
    MQCHAR48   QName;         /* Name of triggered queue */
    MQCHAR48   ProcessName;   /* Name of process object */
    MQCHAR64   TriggerData;   /* Trigger data */
    MQCHAR4    ApplType;      /* Application type */
    MQCHAR256  ApplId;        /* Application identifier */
    MQCHAR128  EnvData;       /* Environment data */
    MQCHAR128  UserData;      /* User data */
    MQCHAR48   QMgrName;     /* Queue manager name */
};
```

### MQTMC2 için COBOL bildirimi

```
** MQTMC2 structure
  10 MQTMC2.
**   Structure identifier
  15 MQTMC2-STRUCID   PIC X(4).
**   Structure version number
  15 MQTMC2-VERSION  PIC X(4).
**   Name of triggered queue
  15 MQTMC2-QNAME    PIC X(48).
**   Name of process object
  15 MQTMC2-PROCESSNAME PIC X(48).
**   Trigger data
  15 MQTMC2-TRIGGERDATA PIC X(64).
**   Application type
  15 MQTMC2-APPLTYPE  PIC X(4).
**   Application identifier
  15 MQTMC2-APPLID   PIC X(256).
**   Environment data
  15 MQTMC2-ENVDATA  PIC X(128).
**   User data
  15 MQTMC2-USERSDATA PIC X(128).
**   Queue manager name
  15 MQTMC2-QMGRNAME PIC X(48).
```

### MQTMC2 için PL/I bildirimi

```
dcl
  1 MQTMC2 based,
  3 StrucId    char(4), /* Structure identifier */
```

```

3 Version      char(4), /* Structure version number */
3 QName        char(48), /* Name of triggered queue */
3 ProcessName  char(48), /* Name of process object */
3 TriggerData  char(64), /* Trigger data */
3 ApplType     char(4), /* Application type */
3 ApplId       char(256), /* Application identifier */
3 EnvData      char(128), /* Environment data */
3 UserData     char(128), /* User data */
3 QMgrName     char(48); /* Queue manager name */

```

## MQTMC2 için High Level Assembler bildirimi

```

MQTMC2          DSECT
MQTMC2_STRUCID  DS   CL4   Structure identifier
MQTMC2_VERSION DS   CL4   Structure version number
MQTMC2_QNAME    DS  CL48   Name of triggered queue
MQTMC2_PROCESSNAME DS CL48 Name of process object
MQTMC2_TRIGGERDATA DS CL64 Trigger data
MQTMC2_APPLTYPE DS   CL4   Application type
MQTMC2_APPLID   DS  CL256 Application identifier
MQTMC2_ENVDATA  DS  CL128 Environment data
MQTMC2_USERDATA DS  CL128 User data
MQTMC2_QMGRNAME DS  CL48   Queue manager name
*
MQTMC2_LENGTH   EQU   *-MQTMC2
                ORG   MQTMC2
MQTMC2_AREA     DS   CL(MQTMC2_LENGTH)

```

## MQTMC2 için Visual Basic bildirimi

```

Type MQTMC2
  StrucId      As String*4   'Structure identifier'
  Version      As String*4   'Structure version number'
  QName        As String*48  'Name of triggered queue'
  ProcessName  As String*48  'Name of process object'
  TriggerData  As String*64  'Trigger data'
  ApplType     As String*4   'Application type'
  ApplId       As String*256 'Application identifier'
  EnvData      As String*128 'Environment data'
  UserData     As String*128 'User data'
  QMgrName     As String*48  'Queue manager name'
End Type

```

### **StrucId (MQCHAR4)**

Yapı tanıtıcısı.

Değer şu olmalıdır:

#### **MQTMC\_STRUC\_ID**

Tetikleyici ileti (karakter biçimi) yapısına ilişkin tanıtıcı.

C programlama dili için, sabit MQTMC\_STRUC\_ID\_ARRAY değişmezi de tanımlanır; bu, MQTMC\_STRUC\_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisi olur.

### **Sürüm (MQCHAR4)**

Yapı sürüm numarası.

Değer şu olmalıdır:

#### **MQTMC\_VERSION\_2**

Sürüm 2 tetikleme iletisi (karakter biçimi) yapısı.

For the C programming language, the constant MQTMC\_VERSION\_2\_ARRAY is also defined; this has the same value as MQTMC\_VERSION\_2, but is an array of characters instead of a string.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

#### **MQTMC\_CURRENT\_VERSION**

Tetikleyici iletisi (karakter biçimi) yapısının yürürlükteki sürümü.

**QName (MQCHAR48)**

Tetiklenen kuyruğun adı.

MQTM yapısındaki *QName* alanına bakın.

**ProcessName (MQCHAR48)**

Süreç nesnesinin adı.

MQTM yapısındaki *ProcessName* alanına bakın.

**TriggerData (MQCHAR64)**

Verileri tetikler.

MQTM yapısındaki *TriggerData* alanına bakın.

**ApplType (MQCHAR4)**

Uygulama tipi.

Bu alanda her zaman boşluk, özgün tetikleyici iletisinin MQTM yapısındaki *ApplType* alanındaki değer ne olursa olsun bulunur.

**ApplId (MQCHAR256)**

Uygulama tanıtıcısı.

MQTM yapısındaki *ApplId* alanına bakın.

**EnvData (MQCHAR128)**

Ortam verileri.

MQTM yapısındaki *EnvData* alanına bakın.

**UserData (MQCHAR128)**

Kullanıcı verileri.

MQTM yapısındaki *UserData* alanına bakın.

**QMgrName (MQCHAR48)**

Kuyruk yöneticisi adı.

Bu, tetikleme olayının ortaya çıktığı kuyruk yöneticisinin adıdır.

**MQWIH-İş bilgileri üstbilgisi**

Bir ileti z/OS iş yükü yöneticisi (WLM) tarafından işlenecek olursa, iletinin bir MQWIH yapısıyla başlaması gerekir. Bu yapı, WLM tarafından işlenecek bir iletinin başında bulunması gereken bilgileri açıklar.

**Kullanılabilirlik**

Tüm IBM MQ sistemleri ve bu sistemlere bağlı IBM MQ istemcileri.

**Biçim Adı**

MQFMT\_WORK\_INFO\_HEADER.

## Karakter kümesi ve kodlama

MQWIH yapısındaki alanlar, MQWIH ' den önce gelen üstbilgi yapısındaki *CodedCharSetId* ve *Encoding* alanları tarafından belirtilen karakter kümesinde ve kodlamasında ya da MQWIH uygulama iletilisi verilerinin başlangıcında ise MQMD yapısındaki bu alanlar tarafından bulunur.

Karakter takımı, kuyruk adlarında geçerli olan karakterler için tek baytlık karakterler içeren bir karakter olmalıdır.

## Kullanım

Desteklenen herhangi bir IBM MQ altyapısı için MQWIH yapısını içeren bir ileti yaratabilir ve iletebilir, ancak yalnızca bir IBM MQ for z/OS kuyruk yöneticisi WLM ile etkileşimde bulunabilir. Bu nedenle, iletinin z/OS dışı bir kuyruk yöneticisinden WLM ' ye ulaşabilmesi için kuyruk yöneticisi ağızının, iletinin yönetilebileceği en az bir z/OS kuyruk yöneticisi içermesi gerekir.

## Alanlar

**Not:** Aşağıdaki tabloda, alanlar alfabetik olarak değil, kullanıma göre gruplanır. Alt konular aynı sırayı izler.

Çizelge 537. MQWIH ' deki Alanlar		
Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
<u>StrucId</u> (yapı tanıtıcısı)	MQWIH_STRUC_ID	'WIH↵'
<u>Sürüm</u> (yapı sürümü numarası)	MQWIH_VERSION_1	1
<u>StrucLength</u> (MQWIH yapısının uzunluğu)	MQWIH_LENGTH_1	120
<u>Kodlama</u> (MQWIH ' yi izleyen verilerin sayısal kodlaması)	Yok	0
<u>CodedCharSetId</u> (MQWIH ' yi izleyen verilerin karakter kümesi tanıtıcısı)	MQCCSI_AYRILMAMIŞ	0
<u>Biçim</u> (MQWIH ' yi izleyen verilerin biçim adı)	MQFMT_NONE	Boşluklar
<u>İşaretler</u> (işaretler)	MQWIH_NONE	0
<u>ServiceName</u> (hizmet adı)	Yok	Boşluklar
<u>ServiceStep</u> (hizmet adımı adı)	Yok	Boşluklar
<u>MsgToken</u> (ileti simgesi)	MQMTOK_NONE	Boş Değerler
<u>Ayrılmış</u> (ayrılmış)	Yok	Boşluklar
<b>Notlar:</b> 1. ↵ simgesi, tek bir boş karakteri temsil eder. 2. C programlama dilinde, makro değişkeni MQWIH_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlar için başlangıç değerleri sağlamak üzere bunu aşağıdaki şekilde kullanın: <pre>MQWIH MyWIH = {MQWIH_DEFAULT};</pre>		

## Dil bildirimleri

MQWIH için C bildirimi

```
typedef struct tagMQWIH MQWIH;  
struct tagMQWIH {
```

```

MQCHAR4  StrucId;          /* Structure identifier */
MQLONG   Version;         /* Structure version number */
MQLONG   StrucLength;     /* Length of MQWIH structure */
MQLONG   Encoding;       /* Numeric encoding of data that follows
                           MQWIH */
MQLONG   CodedCharSetId; /* Character-set identifier of data that
                           follows MQWIH */
MQCHAR8  Format;          /* Format name of data that follows
                           MQWIH */
MQLONG   Flags;          /* Flags */
MQCHAR32 ServiceName;    /* Service name */
MQCHAR8  ServiceStep;    /* Service step name */
MQBYTE16 MsgToken;       /* Message token */
MQCHAR32 Reserved;       /* Reserved */
};

```

## MQWIH için COBOL bildirimi

```

** MQWIH structure
10 MQWIH.
** Structure identifier
15 MQWIH-STRUCID PIC X(4).
** Structure version number
15 MQWIH-VERSION PIC S9(9) BINARY.
** Length of MQWIH structure
15 MQWIH-STRUCLNGTH PIC S9(9) BINARY.
** Numeric encoding of data that follows MQWIH
15 MQWIH-ENCODING PIC S9(9) BINARY.
** Character-set identifier of data that follows MQWIH
15 MQWIH-CODEDCHARSETID PIC S9(9) BINARY.
** Format name of data that follows MQWIH
15 MQWIH-FORMAT PIC X(8).
** Flags
15 MQWIH-FLAGS PIC S9(9) BINARY.
** Service name
15 MQWIH-SERVICENAME PIC X(32).
** Service step name
15 MQWIH-SERVICESTEP PIC X(8).
** Message token
15 MQWIH-MSGTOKEN PIC X(16).
** Reserved
15 MQWIH-RESERVED PIC X(32).

```

## MQWIH için PL/I bildirimi

```

dcl
1 MQWIH based,
3 StrucId char(4), /* Structure identifier */
3 Version fixed bin(31), /* Structure version number */
3 StrucLength fixed bin(31), /* Length of MQWIH structure */
3 Encoding fixed bin(31), /* Numeric encoding of data that
                           follows MQWIH */
3 CodedCharSetId fixed bin(31), /* Character-set identifier of data
                           that follows MQWIH */
3 Format char(8), /* Format name of data that follows
                  MQWIH */
3 Flags fixed bin(31), /* Flags */
3 ServiceName char(32), /* Service name */
3 ServiceStep char(8), /* Service step name */
3 MsgToken char(16), /* Message token */
3 Reserved char(32); /* Reserved */

```

## MQWIH için High Level Assembler bildirimi

MQWIH	DSECT	
MQWIH_STRUCID	DS CL4	Structure identifier
MQWIH_VERSION	DS F	Structure version number
MQWIH_STRUCLNGTH	DS F	Length of MQWIH structure
MQWIH_ENCODING	DS F	Numeric encoding of data that follows MQWIH
*		
MQWIH_CODEDCHARSETID	DS F	Character-set identifier of data that follows MQWIH
*		
MQWIH_FORMAT	DS CL8	Format name of data that follows MQWIH
MQWIH_FLAGS	DS F	Flags
MQWIH_SERVICENAME	DS CL32	Service name

MQWIH_SERVICESTEP	DS	CL8	Service step name
MQWIH_MSGTOKEN	DS	XL16	Message token
MQWIH_RESERVED	DS	CL32	Reserved
*			
MQWIH_LENGTH	EQU	*-MQWIH	
	ORG	MQWIH	
MQWIH_AREA	DS	CL(MQWIH_LENGTH)	

## MQWIH için Visual Basic bildirimi

```

Type MQWIH
  StrucId      As String*4  'Structure identifier'
  Version     As Long      'Structure version number'
  StrucLength As Long      'Length of MQWIH structure'
  Encoding    As Long      'Numeric encoding of data that follows'
  CodedCharSetId As Long    'Character-set identifier of data that'
  follows MQWIH'
  Format      As String*8  'Format name of data that follows MQWIH'
  Flags      As Long      'Flags'
  ServiceName As String*32 'Service name'
  ServiceStep As String*8  'Service step name'
  MsgToken   As MQBYTE16 'Message token'
  Reserved   As String*32 'Reserved'
End Type

```

### **StrucId (MQCHAR4)**

Bu, yapı tanıtıcısıdır. Değer şu olmalıdır:

#### **MQWIH\_STRUC\_ID**

İş bilgileri üstbilgi yapısına ilişkin tanıtıcı.

C programlama dili için, MQWIH\_STRUC\_ID\_ARRAY değişmezi de tanımlanır; bu, MQWIH\_STRUC\_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisidir.

Bu alanın ilk değeri MQWIH\_STRUC\_ID ' dir.

### **Sürüm (MQUZE)**

Bu, yapı sürüm numarasıdır. Değer şu olmalıdır:

#### **MQWIH\_VERSION\_1**

Version-1 iş bilgileri üstbilgi yapısı.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

#### **MQWIH\_CURRENT\_VERSION**

İş bilgisi üstbilgi yapısının yürürlükteki sürümü.

Bu alanın ilk değeri MQWIH\_VERSION\_1' dir.

### **StrucLength (MQUZE)**

Bu, MQWIH yapısının uzunluğudur. Değer şu olmalıdır:

#### **MQWIH\_LENGTH\_1**

version-1 iş bilgileri üstbilgisi yapısı uzunluğu.

Aşağıdaki değişmez, yürürlükteki sürümün uzunluğunu belirtir:

#### **MQWIH\_CURRENT\_LENGTH**

İş bilgisi üstbilgisi yapısının geçerli sürümü uzunluğu.

Bu alanın ilk değeri MQWIH\_LENGTH\_1' dir.

### **Kodlama (MQUZE)**

Bu, MQWIH yapısını izleyen verilerin sayısal kodlamasını belirtir; bu, MQWIH yapısındaki sayısal veriler için geçerli değildir.

MQPOT ya da MQPUT1 çağrısında, uygulama bu alanı veri için uygun değere ayarlamalıdır.

Bu alanın ilk değeri 0 'tır.

### **CodedCharSetId (MQHOMER)**

Bu, MQWIH yapısını izleyen verilerin karakter kümesi tanıtıcısını belirtir; bu, MQWIH yapısındaki karakter verileri için geçerli değildir.

MQPOT ya da MQPUT1 çağrısında, uygulama bu alanı veri için uygun değere ayarlamalıdır. Aşağıdaki özel değeri kullanabilirsiniz:

#### **MQCCSI\_INHERIT**

Character data in the data *takip edilen* this structure is in the same character set as this structure.

Kuyruk yöneticisi, bu değeri, iletide belirtilen yapının gerçek karakter kümesi tanıtıcısına göre değiştirir. Herhangi bir hata oluşmaması koşuluyla, MQCCSI\_INHERIT değeri MQGET çağrısıyla döndürülmez.

MQMD 'deki *PutApplType* alanının değeri MQAT\_BROKER ise, MQCCSI\_INHERIT kullanılamaz.

Bu alanın ilk değeri MQCCSI\_UNDEFED değerine sahip.

### **Biçim (MQCHAR8)**

Bu, MQWIH yapısından sonra gelen verilerin biçim adını belirler.

MQPOT ya da MQPUT1 çağrısında, uygulama bu alanı veri için uygun değere ayarlamalıdır. Bu alanı kodlamaya ilişkin kurallar, MQMD 'deki *Format* alanı için yapılan kurallarla aynıdır.

Bu alanın uzunluğu MQ\_FORMAT\_LENGTH tarafından verilir. Bu alanın ilk değeri MQFMT\_NONE olur.

### **İşaretler (MQUZE)**

Değer şu olmalıdır:

#### **MQWIH\_NONE**

Bayrak yok.

Bu alanın ilk değeri MQWIH\_NONE olur.

### **ServiceName (MQCHAR32)**

Bu, iletiyi işlemek için kullanılan hizmetin adıdır.

Bu alanın uzunluğu MQ\_SERVICE\_NAME\_LENGTH tarafından verilir. Bu alanın ilk değeri 32 boş karakterdir.

### **ServiceStep (MQCHAR8)**

Bu ad, iletinin ilişkilendirdiği *ServiceName* adımının adıdır.

Bu alanın uzunluğu MQ\_SERVICE\_STEP\_LENGTH tarafından verilir. Bu alanın ilk değeri 8 boş karakterdir.

### **MsgToken (MQBYTE16)**

Bu, iletiyi benzersiz bir şekilde tanımlayan bir ileti belirtecidir.

MQPOT ve MQPUT1 çağrıları için bu alan yok sayılır. Bu alanın uzunluğuna MQ\_MSG\_TOKEN\_LENGTH değeri verilir. Bu alanın ilk değeri MQMTOK\_NONE olur.

### **Ayrılmış (MQCHAR32)**

Bu ayrılmış bir alandır; boş bırakılmalıdır.

## MQXP-Çıkış deęiřtirgesi öbeęi

MQXP yapısı, API-geçiş çıkışına giriş/çıkış parametresi olarak kullanılır. Bu çıkışla ilgili daha fazla bilgi için bkz. [API-geçiş çıkışı](#).

### Karakter kümesi ve kodlama

MQXP ' deki karakter verileri yerel kuyruk yöneticisinin karakter kümesinde; bu, **CodedCharSetId** kuyruk yöneticisi öznitelięi tarafından verilir. MQXP ' deki sayısal veriler yerli makine kodlamasında; bu, MQENC\_NATIVE tarafından verilir.

### Alanlar

**Not:** Ařaęıdaki tabloda, alanlar alfabetik olarak deęil, kullanıma göre gruplanır. Alt konular aynı sırayı izler.

Çizelge 538. MQXP ' deki Alanlar	
Alan adı ve açıklaması	Deęişmezin adı
StrucId (yapı tanıtıcısı)	MQXP_STRUC_ID
Sürüm (yapı sürümü numarası)	MQXP_VERSION_1
ExitId (çıkış tanıtıcısı)	MQXT_API_CROSSING_EXIT
ExitReason (çıkışın çağrılmasının nedeni)	Yok
ExitResponse (çıkıştan yanıt)	Yok
ExitCommand (API çağrı kodu)	Yok
ExitParmSayı (parametre sayısı)	Yok
Ayrılmış (ayrılmış)	Yok
ExitUserAlanı (kullanıcı alanı)	Yok

### Dil bildirimleri

MQXP için C bildirimini

```
typedef struct tagMQXP MQXP;
struct tagMQXP {
    MQCHAR4    StrucId;           /* Structure identifier */
    MQLONG     Version;          /* Structure version number */
    MQLONG     ExitId;           /* Exit identifier */
    MQLONG     ExitReason;       /* Reason for invocation of exit */
    MQLONG     ExitResponse;     /* Response from exit */
    MQLONG     ExitCommand;      /* API call code */
    MQLONG     ExitParmCount;    /* Parameter count */
    MQLONG     Reserved;         /* Reserved */
    MQBYTE16   ExitUserArea;    /* User area */
};
```

MQXP için COBOL bildirimini

```
** MQXP structure
10 MQXP.
** Structure identifier
15 MQXP-STRUCID PIC X(4).
** Structure version number
15 MQXP-VERSION PIC S9(9) BINARY.
** Exit identifier
15 MQXP-EXITID PIC S9(9) BINARY.
** Reason for invocation of exit
15 MQXP-EXITREASON PIC S9(9) BINARY.
```



```

**      Response from exit
15 MQXP-EXITRESPONSE PIC S9(9) BINARY.
**      API call code
15 MQXP-EXITCOMMAND  PIC S9(9) BINARY.
**      Parameter count
15 MQXP-EXITPARMCOUNT PIC S9(9) BINARY.
**      Reserved
15 MQXP-RESERVED      PIC S9(9) BINARY.
**      User area
15 MQXP-EXITUSERAREA PIC X(16) .

```

## MQXP için PL/I bildirimi

```

dcl
1 MQXP based,
3 StrucId      char(4),          /* Structure identifier */
3 Version      fixed bin(31),   /* Structure version number */
3 ExitId       fixed bin(31),   /* Exit identifier */
3 ExitReason   fixed bin(31),   /* Reason for invocation of exit */
3 ExitResponse fixed bin(31),   /* Response from exit */
3 ExitCommand  fixed bin(31),   /* API call code */
3 ExitParmCount fixed bin(31), /* Parameter count */
3 Reserved     fixed bin(31),   /* Reserved */
3 ExitUserArea char(16);        /* User area */

```

## MQXP için High Level Assembler (Üst Düzey Çevirici) bildirimi

```

MQXP          DSECT
MQXP_STRUCID  DS  CL4  Structure identifier
MQXP_VERSION  DS  F    Structure version number
MQXP_EXITID   DS  F    Exit identifier
MQXP_EXITREASON DS  F    Reason for invocation of exit
MQXP_EXITRESPONSE DS  F    Response from exit
MQXP_EXITCOMMAND DS  F    API call code
MQXP_EXITPARMCOUNT DS  F    Parameter count
MQXP_RESERVED DS  F    Reserved
MQXP_EXITUSERAREA DS  XL16 User area
*
MQXP_LENGTH   EQU  *-MQXP
               ORG  MQXP
MQXP_AREA     DS   CL(MQXP_LENGTH)

```

### **StrucId (MQCHAR4)**

Bu, yapı tanıtıcısıdır. Değer şu olmalıdır:

#### **MQXP\_STRUC\_ID**

Çıkış parametresi yapısıyla ilgili tanıtıcı.

C programlama dili için, sabit MQXP\_STRUC\_ID\_ARRAY değişmezi de tanımlanır; bu, MQXP\_STRUC\_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisi olur.

Bu, çıkışa giriş alanıdır.

### **Sürüm (MQUZE)**

Bu, yapı sürüm numarasıdır. Değer şu olmalıdır:

#### **MQXP\_VERSION\_1**

Çıkış parametresi-blok yapısı için sürüm numarası.

**Not:** Bu yapının yeni bir sürümü kullanıma sunulduğunda, var olan parçanın düzeni değiştirilmez. Bu nedenle, çıkışta, sürüm numarasının, çıkışa gereken alanları içeren en düşük sürüme eşit ya da bu sürümden daha büyük olduğunu kontrol etmelidir.

Bu, çıkışa giriş alanıdır.

### **ExitId (MQUZE)**

Bu, çıkış yordamlarına giriş olarak ayarlanır ve çıkış tipini belirtir:

## **MQXT\_API\_CROSSING\_EXIT**

CICS için API geçiş çıkışı.

Bu, çıkışa giriş alanıdır.

### ***ExitReason (MQUZE)***

Bu, çıkış yordamlarına giriş olarak ayarlanır. API geçidi çıkışı için, yordamın API çağrısının yürütülmesinden önce mi, yoksa sonra mı çağrılacağını gösterir:

#### **MQXR\_BEDEN**

API yürütmeden önce.

#### **MQXR\_AFTER**

API yürütmesinden sonra.

Bu, çıkışa giriş alanıdır.

### ***ExitResponse (MQUZE)***

Bu değer, çağırana iletilmek için çıkışa göre ayarlanır. Aşağıdaki değerler tanımlanır:

#### **MQXCC\_OK**

Çıkış başarıyla tamamlandı.

#### **MQXCC\_SUPPRESS\_FUNCTION**

İşlevi engelle.

Bu değer, API çağrısından *önce* çağrılan bir API geçidi çıkışı tarafından ayarlandığında, API çağrısı gerçekleştirilmez. Çağrı için *CompCode* , MQCC\_FAILED olarak ayarlıdır, *Reason* MQRC\_SUPPRESSED\_BY\_EXIT olarak ayarlıdır ve diğer tüm parametreler, çıkış olarak bunları bırakmaktan kalır.

Bu değer, API çağrısından *sonra* adı verilen bir API geçme çıkışa göre ayarlandığında, kuyruk yöneticisi tarafından yoksayılır.

#### **MQXCC\_SKIP\_FUNCTION**

İşlevi atlayın.

When this value is set by an API-crossing exit called *önce* the API call, the API call is not performed; the *CompCode* and *Reason* and all other parameters remain as the exit left them.

Bu değer, API çağrısından *sonra* adı verilen bir API geçme çıkışa göre ayarlandığında, kuyruk yöneticisi tarafından yoksayılır.

Bu, çıkıştan bir çıkış alanıdır.

### ***ExitCommand (MQUZA)***

Bu alan, çıkış yordamlarına giriş olarak ayarlanır. Çıkışa çağrılmasına neden olan API çağrısını tanımlar:

#### **MQXC\_CALLBACK**

CALLBACK çağrısı.

#### **MQXC\_MQI**

MQBACK çağrısı.

#### **MQXC\_MQCB**

MQCB çağrısı.

#### **MQXC\_MQCLOSE**

MQCLOSE çağrısı.

#### **MQXC\_MQCMIT**

MQCMIT çağrısı.

#### **MQXC\_MQCTL**

MQCTL çağrısı.

**MQXC\_MQGET**

MQGET çağrısı.

**MQXC\_MQINQ**

MQINQ çağrısı.

**MQXC\_MQAç**

MQOPEN çağrısı.

**MQXC\_MQPUT**

MQPUT çağrısı.

**MQXC\_MQPUT1**

MQPUT1 çağrısı.

**MQXC\_MQSET**

MQSET çağrısı.

**MQXC\_MQSTAT**

MQSTAT çağrısı.

**MQXC\_MQSUB**

MQSUB çağrısı.

**MQXC\_MQSUBRQ**

MQSUBRQ çağrısı.

Bu, çıkışa giriş alanıdır.

**ExitParmCount (MQLONG)**

Bu alan, çıkış yordamlarına giriş olarak ayarlanır. Bu, MQ çağrısının aldığı parametrelerin sayısını içerir.

*Çizelge 539. Her MQ çağrısına ilişkin parametre sayısı*

---

<b>Arama adı</b>	<b>Parametre sayısı</b>
MQBACK	3
MQCLOSE	5
MQCMIT	3
MQGet	9
MQINQ	10
MQOPEN	6
MQPUT	8
MQPUT1	8
MQSET	10

Bu, çıkışa giriş alanıdır.

**Ayrılmış (MQUZE)**

Bu ayrılmış bir alandır. Çıkış için değeri önemli değil.

**ExitUser(ExitUser) Alanı (MQBYTE16)**

Bu alan, çıkışta kullanılacak bir alandır. Göreve ilişkin çıkışa ilişkin ilk çağrıdan önce alan uzunluğu için ikili sıfır olarak başlatılır ve bundan sonra çıkışta bu alanda yapılan değişiklikler çıkışa yönelik çağrılar boyunca korunur. Aşağıdaki değer tanımlıdır:

**MQXUA\_NONE**

Kullanıcı bilgisi yok.

Değer, alanın uzunluğuna ilişkin ikili sıfır değeridir.

C programlama dili için, sabit MQXUA\_NONE\_DIZISI de tanımlanır; bu, MQXUA\_NONE ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisidir.

Bu alanın uzunluğu MQ\_EXIT\_USER\_AREA\_LENGTH tarafından verilir. Bu, çıkışa giriş/çıkış alanıdır.

## MQXQH-İletim kuyruğu üstbilgisi

MQXQH yapısı, iletim kuyruklarındayken iletilerin uygulama ileti verilerine önek olarak eklenen bilgileri tanımlar. İletim kuyruğu, uzak kuyruklara (yerel kuyruk yöneticisine ait olmayan kuyruklara) yönelik iletileri geçici olarak tutan özel bir yerel kuyruk tipidir. Bir iletim kuyruğu, MQUS\_ILETIM değerine sahip **Usage** kuyruk özneliğiyle gösterilir.

### Biçim Adı

MQFMT\_XMIT\_Q\_HEADER

### Karakter kümesi ve kodlama

MQXQH içindeki veriler, **CodedCharSetId** kuyruk yöneticisi özneliği tarafından belirtilen karakter kümesinde ve MQENC\_NATIVE tarafından verilen yerel kuyruk yöneticisinin kodlamasında olmalıdır.

MQXQH ' nin karakter kümesini ve kodlamasını aşağıdaki alanlardaki *CodedCharSetId* ve *Encoding* alanlarına ayarlayın:

- Ayrı MQMD (MQXQH yapısı ileti verilerinin başındadır) ya da
- MQXQH yapısından önce gelen üstbilgi yapısı (diğer tüm durumlar).

### Alanlar

**Not:** Aşağıdaki tabloda, alanlar alfabetik olarak değil, kullanıma göre gruplanır. Alt konular aynı sırayı izler.

Çizelge 540. MQXQH için MQXQH içindeki alanlar		
Alan adı ve açıklaması	Değişmezin adı	Değişmezin ilk değeri (varsa)
<u>StrucId</u> (yapı tanıtıcısı)	MQXQH_STRUC_ID	'XQH~'
<u>Sürüm</u> (yapı sürümü numarası)	MQXQH_VERSION_1	1
<u>RemoteQName</u> (hedef kuyruğun adı)	Yok	Boş değerli dizilim ya da boşluklar
<u>RemoteQMgrAd</u> (hedef kuyruk yöneticisinin adı)	Yok	Boş değerli dizilim ya da boşluklar
<u>MsgDesc</u> (özgün ileti tanımlayıcısı)	MQMD ile aynı adlar ve değerler; bkz. Çizelge 501 sayfa 418	-
<b>Notlar:</b> <ol style="list-style-type: none"><li>1. ~ simgesi, tek bir boş karakteri temsil eder.</li><li>2. Boş değerli dizilim ya da boşluk değeri, C dizgisindeki boş dizgiyi ve diğer programlama dillerindeki boş karakterleri gösterir.</li><li>3. C programlama dilinde, makro değişkeni MQXQH_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlar için başlangıç değerleri sağlamak üzere bunu aşağıdaki şekilde kullanın:<pre>MQXQH MyXQH = {MQXQH_DEFAULT};</pre></li></ol>		

## Dil bildirimleri

### MQXQH için C bildirimi

```
typedef struct tagMQXQH MQXQH;
struct tagMQXQH {
    MQCHAR4   StrucId;           /* Structure identifier */
    MQLONG    Version;         /* Structure version number */
    MQCHAR48  RemoteQName;     /* Name of destination queue */
    MQCHAR48  RemoteQMgrName; /* Name of destination queue manager */
    MQMD1     MsgDesc;         /* Original message descriptor */
};
```

### MQXQH için COBOL bildirimi

```
** MQXQH structure
10 MQXQH.
** Structure identifier
15 MQXQH-STRUCID PIC X(4).
** Structure version number
15 MQXQH-VERSION PIC S9(9) BINARY.
** Name of destination queue
15 MQXQH-REMOTEQNAME PIC X(48).
** Name of destination queue manager
15 MQXQH-REMOTEQMGRNAME PIC X(48).
** Original message descriptor
15 MQXQH-MSGDESC.
** Structure identifier
20 MQXQH-MSGDESC-STRUCID PIC X(4).
** Structure version number
20 MQXQH-MSGDESC-VERSION PIC S9(9) BINARY.
** Report options
20 MQXQH-MSGDESC-REPORT PIC S9(9) BINARY.
** Message type
20 MQXQH-MSGDESC-MSGTYPE PIC S9(9) BINARY.
** Expiry time
20 MQXQH-MSGDESC-EXPIRY PIC S9(9) BINARY.
** Feedback or reason code
20 MQXQH-MSGDESC-FEEDBACK PIC S9(9) BINARY.
** Numeric encoding of message data
20 MQXQH-MSGDESC-ENCODING PIC S9(9) BINARY.
** Character set identifier of message data
20 MQXQH-MSGDESC-CODEDCHARSETID PIC S9(9) BINARY.
** Format name of message data
20 MQXQH-MSGDESC-FORMAT PIC X(8).
** Message priority
20 MQXQH-MSGDESC-PRIORITY PIC S9(9) BINARY.
** Message persistence
20 MQXQH-MSGDESC-PERSISTENCE PIC S9(9) BINARY.
** Message identifier
20 MQXQH-MSGDESC-MSGID PIC X(24).
** Correlation identifier
20 MQXQH-MSGDESC-CORRELID PIC X(24).
** Backout counter
20 MQXQH-MSGDESC-BACKOUTCOUNT PIC S9(9) BINARY.
** Name of reply-to queue
20 MQXQH-MSGDESC-REPLYTOQ PIC X(48).
** Name of reply queue manager
20 MQXQH-MSGDESC-REPLYTOQMGR PIC X(48).
** User identifier
20 MQXQH-MSGDESC-USERIDENTIFIER PIC X(12).
** Accounting token
20 MQXQH-MSGDESC-ACCOUNTINGTOKEN PIC X(32).
** Application data relating to identity
20 MQXQH-MSGDESC-APPLIDENTITYDATA PIC X(32).
** Type of application that put the message
20 MQXQH-MSGDESC-PUTAPPLTYPE PIC S9(9) BINARY.
** Name of application that put the message
20 MQXQH-MSGDESC-PUTAPPLNAME PIC X(28).
** Date when message was put
20 MQXQH-MSGDESC-PUTDATE PIC X(8).
** Time when message was put
20 MQXQH-MSGDESC-PUTTIME PIC X(8).
** Application data relating to origin
20 MQXQH-MSGDESC-APPLORIGINDATA PIC X(4).
```

## MQXQH için PL/I bildirimi

```

dcl
  1 MQXQH based,
  3 StrucId          char(4),          /* Structure identifier */
  3 Version          fixed bin(31),    /* Structure version number */
  3 RemoteQName      char(48),        /* Name of destination queue */
  3 RemoteQMgrName   char(48),        /* Name of destination queue
                                     manager */
  3 MsgDesc,        /* Original message descriptor */
  5 StrucId          char(4),          /* Structure identifier */
  5 Version          fixed bin(31),    /* Structure version number */
  5 Report           fixed bin(31),    /* Report options */
  5 MsgType          fixed bin(31),    /* Message type */
  5 Expiry           fixed bin(31),    /* Expiry time */
  5 Feedback         fixed bin(31),    /* Feedback or reason code */
  5 Encoding         fixed bin(31),    /* Numeric encoding of message
                                     data */
  5 CodedCharSetId   fixed bin(31),    /* Character set identifier of
                                     message data */
  5 Format            char(8),          /* Format name of message data */
  5 Priority          fixed bin(31),    /* Message priority */
  5 Persistence      fixed bin(31),    /* Message persistence */
  5 MsgId            char(24),        /* Message identifier */
  5 CorrelId         char(24),        /* Correlation identifier */
  5 BackoutCount     fixed bin(31),    /* Backout counter */
  5 ReplyToQ         char(48),        /* Name of reply-to queue */
  5 ReplyToQMgr      char(48),        /* Name of reply queue manager */
  5 UserIdentifier   char(12),        /* User identifier */
  5 AccountingToken  char(32),        /* Accounting token */
  5 ApplIdentityData char(32),        /* Application data relating to
                                     identity */
  5 PutApplType      fixed bin(31),    /* Type of application that put the
                                     message */
  5 PutApplName      char(28),        /* Name of application that put the
                                     message */
  5 PutDate          char(8),          /* Date when message was put */
  5 PutTime          char(8),          /* Time when message was put */
  5 ApplOriginData   char(4);        /* Application data relating to
                                     origin */

```

## MQXQH için High Level Assembler (Üst Düzey Çevirici) bildirimi

```

MQXQH          DSECT
MQXQH_STRUCID  DS    CL4  Structure identifier
MQXQH_VERSION  DS    F    Structure version number
MQXQH_REMOTENAME DS    CL48 Name of destination queue
MQXQH_REMOTEQMGRNAME DS    CL48 Name of destination queue
                                     manager
*
MQXQH_MSGDESC  DS    0F   Force fullword alignment
MQXQH_MSGDESC_STRUCID DS    CL4  Structure identifier
MQXQH_MSGDESC_VERSION DS    F    Structure version number
MQXQH_MSGDESC_REPORT DS    F    Report options
MQXQH_MSGDESC_MSGTYPE DS    F    Message type
MQXQH_MSGDESC_EXPIRY DS    F    Expiry time
MQXQH_MSGDESC_FEEDBACK DS    F    Feedback or reason code
MQXQH_MSGDESC_ENCODING DS    F    Numeric encoding of message
                                     data
*
MQXQH_MSGDESC_CODEDCHARSETID DS    F    Character set identifier of
                                     message data
*
MQXQH_MSGDESC_FORMAT DS    CL8  Format name of message data
MQXQH_MSGDESC_PRIORITY DS    F    Message priority
MQXQH_MSGDESC_PERSISTENCE DS    F    Message persistence
MQXQH_MSGDESC_MSGID DS    XL24  Message identifier
MQXQH_MSGDESC_CORRELID DS    XL24  Correlation identifier
MQXQH_MSGDESC_BACKOUTCOUNT DS    F    Backout counter
MQXQH_MSGDESC_REPLYTOQ DS    CL48  Name of reply-to queue
MQXQH_MSGDESC_REPLYTOQMGR DS    CL48  Name of reply queue manager
MQXQH_MSGDESC_USERIDENTIFIER DS    CL12  User identifier
MQXQH_MSGDESC_ACCOUNTINGTOKEN DS    XL32  Accounting token
MQXQH_MSGDESC_APPLIDENTITYDATA DS    CL32  Application data relating to
                                     identity
*
MQXQH_MSGDESC_PUTAPPLTYPE DS    F    Type of application that put
                                     the message
*
MQXQH_MSGDESC_PUTAPPLNAME DS    CL28  Name of application that put
                                     the message
*
MQXQH_MSGDESC_PUTDATE DS    CL8  Date when message was put
MQXQH_MSGDESC_PUTTIME DS    CL8  Time when message was put

```

```

MQXQH_MSGDESC_APPLORIGINDATA  DS  CL4  Application data relating to
*                               origin
MQXQH_MSGDESC_LENGTH          EQU  *-MQXQH_MSGDESC
                                ORG  MQXQH_MSGDESC
MQXQH_MSGDESC_AREA            DS   CL(MQXQH_MSGDESC_LENGTH)
*
MQXQH_LENGTH                  EQU  *-MQXQH
                                ORG  MQXQH
MQXQH_AREA                     DS   CL(MQXQH_LENGTH)

```

## MQXQH için Visual Basic bildirimi

```

Type MQXQH
  StrucId      As String*4  'Structure identifier'
  Version      As Long      'Structure version number'
  RemoteQName  As String*48 'Name of destination queue'
  RemoteQMgrName As String*48 'Name of destination queue manager'
  MsgDesc     As MQMD1     'Original message descriptor'
End Type

```

## Ayrı ileti tanımlayıcısındaki alanlar

İletim kuyruğundaki bir iletide *iki* ileti tanımlayıcısı vardır:

- Bir ileti tanımlayıcısı, ileti verilerinden ayrı olarak saklanır; bu, *ayrı ileti tanımlayıcısı* olarak adlandırılır ve ileti iletim kuyruğuna yerleştirildiğinde kuyruk yöneticisi tarafından oluşturulur. Ayrı ileti tanımlayıcısındaki bazı alanlar, MQPUT ya da MQPUT1 çağrısındaki uygulama tarafından sağlanan ileti tanımlayıcısından kopyalanır.

Ayrı ileti tanımlayıcısı, ileti iletim kuyruğundan kaldırıldığında MQGET çağrısının **MsgDesc** değiştirilmesinde uygulamaya döndürülen iletidir.

- İleti verilerinin bir parçası olarak MQXQH yapısında ikinci bir ileti tanımlayıcısı saklanır; bu, *yerleşik ileti tanımlayıcısı* olarak adlandırılır ve MQPUT ya da MQPUT1 çağrısında (ikincil çeşitlemelerle) uygulama tarafından sağlanan ileti tanımlayıcısının bir kopyasıdır.

Yerleşik ileti tanımlayıcısı her zaman bir version-1 MQMD 'dir. Uygulama tarafından konan iletide MQMD 'deki version-2 alanlarından biri ya da daha fazlası için varsayılan olmayan değerler varsa, MQMDE yapısı MQXQH' yi izler ve ardından uygulama iletisi verileri (varsa) gelir. MQMDE aşağıdakilerden biri olabilir:

- Kuyruk yöneticisi tarafından üretilir (uygulama iletiyi koymak için version-2 MQMD kullanıyorsa) ya da
- Uygulama ileti verilerinin başlangıcında zaten var (uygulama iletiyi koymak için version-1 MQMD kullanıyorsa).

Yerleşik ileti tanımlayıcısı, ileti son hedef kuyruktan kaldırıldığında MQGET çağrısının **MsgDesc** değiştirilmesinde uygulamaya döndürülen iletidir.

Ayrı ileti tanımlayıcısındaki alanlar, kuyruk yöneticisi tarafından gösterildiği gibi ayarlanır. Kuyruk yöneticisi version-2 MQMD ' yi desteklemiyorsa, işlev kaybı olmadan version-1 MQMD kullanılır.

Çizelge 541. Ayrı MQMD ' deki alanlar için kullanılan değerler

Ayrı MQMD ' deki alan	Kullanılan değer
<i>StrucId</i>	MQMD_STRUC_ID
<i>Version</i>	MQMD_VERSION_2
<i>Report</i>	Yerleşik ileti tanımlayıcısından kopyalandı, ancak MQRO_ACCEPT_UNSUP_IF_XMIT_MASK tarafından tanımlanan bitler sıfır olarak ayarlandı. (Bu, bir ileti bir iletim kuyruğuna yerleştirildiğinde ya da iletiden kaldırıldığında bir COA ya da COD rapor iletisinin oluşturulmasını önler.)
<i>MsgType</i>	Yerleşik ileti tanımlayıcısından kopyalandı.
<i>Expiry</i>	Yerleşik ileti tanımlayıcısından kopyalandı.

Çizelge 541. Ayrı MQMD ' deki alanlar için kullanılan değerler (devamı var)

Ayrı MQMD ' deki alan	Kullanılan değer
<i>Feedback</i>	Yerleşik ileti tanımlayıcısından kopyalandı.
<i>Encoding</i>	MQENC_NATIVE (nota bakın)
<i>CodedCharSetId</i>	Kuyruk yöneticisinin <b>CodedCharSetId</b> özneliği.
<i>Format</i>	MQFMT_XMIT_Q_HEADER
<i>Priority</i>	Yerleşik ileti tanımlayıcısından kopyalandı.
<i>Persistence</i>	Yerleşik ileti tanımlayıcısından kopyalandı.
<i>MsgId</i>	Kuyruk yöneticisi tarafından yeni bir değer oluşturulur. Bu ileti tanıtıcısı, kuyruk yöneticisinin önceden açıklanan yerleşik ileti tanımlayıcısı için oluşturduğu <i>MsgId</i> ileteninden farklıdır.
<i>CorrelId</i>	Yerleşik ileti tanımlayıcısından <i>MsgId</i> . SYSTEM.CLUSTER.TRANSMIT.QUEUE, <i>CorrelId</i> iç kullanım için ayrılmıştır.
<i>BackoutCount</i>	0
<i>ReplyToQ</i>	Yerleşik ileti tanımlayıcısından kopyalandı.
<i>ReplyToQMGr</i>	Yerleşik ileti tanımlayıcısından kopyalandı.
<i>UserIdentifier</i>	Yerleşik ileti tanımlayıcısından kopyalandı.
<i>AccountingToken</i>	Yerleşik ileti tanımlayıcısından kopyalandı. SYSTEM.CLUSTER.TRANSMIT.QUEUE, <i>AccountingToken</i> iç kullanım için ayrılmıştır.
<i>ApplIdentityData</i>	Yerleşik ileti tanımlayıcısından kopyalandı.
<i>PutApplType</i>	MQAT_QMGR
<i>PutApplName</i>	Kuyruk yöneticisi adının ilk 28 baytı.
<i>PutDate</i>	İletinin iletim kuyruğuna konma tarihi.
<i>PutTime</i>	İletinin iletim kuyruğuna konma süresi.
<i>ApplOriginData</i>	Boşluklar
<i>GroupId</i>	MQGI_NONE
<i>MsgSeqNumber</i>	1
<i>Offset</i>	0
<i>MsgFlags</i>	MQMF_NONE
<i>OriginalLength</i>	MQOL_UNDEFINED

- Windowsüzerinde, Micro Focus COBOL için MQENC\_NATIVE değeri C değerinden farklıdır. Ayrı ileti tanımlayıcısındaki *Encoding* alanındaki değer, bu ortamlardaki C için her zaman değerdir; bu değer ondalık olarak 546 'dır. MQXQH yapısındaki tamsayı alanları da bu değere (yerel Intel kodlaması) karşılık gelen kodlamadır.

### Katıştırılmış ileti tanımlayıcısındaki alanlar

Gömülü ileti tanımlayıcısındaki alanlar, aşağıdakiler dışında, MQPUT ya da MQPUT1 çağrısının **MsgDesc** değiştirilmesiyle aynı değerlere sahiptir:

- *Version* alanında her zaman MQMD\_VERSION\_1değeri bulunur.



- *Priority* alanı MQPRI\_PRIORITY\_AS\_Q\_DEF değerine sahipse, kuyruğun **DefPriority** özneliğinin değeriyle değiştirilir.
- *Persistence* alanı MQPER\_PERSISTENCE\_AS\_Q\_DEF değerine sahipse, kuyruğun **DefPersistence** özneliğinin değeriyle değiştirilir.
- *MsgId* alanında MQMI\_NONE değeri varsa ya da MQPMO\_NEW\_MSG\_ID seçeneği belirlendiyse ya da ileti bir dağıtım listesi iletiliyse, *MsgId* yerine kuyruk yöneticisi tarafından oluşturulan yeni bir ileti tanıtıcısı konur.

Bir dağıtım listesi iletilisi, farklı iletim kuyruklarına yerleştirilen daha küçük dağıtım listesi iletilerine bölündüğünde, yeni yerleşik ileti tanımlayıcılarının her birindeki *MsgId* alanı, özgün dağıtım listesi iletilisindekiyle aynıdır.

- MQPMO\_NEW\_CORREL\_ID seçeneği belirtildiyse, *CorrelId* kuyruk yöneticisi tarafından oluşturulan yeni bir ilinti tanıtıcısıyla değiştirilir.
- Bağlam alanları, **PutMsgOpts** değiştirilmesinde belirtilen MQPMO\_\*\_CONTEXT seçenekleriyle belirtildiği şekilde ayarlanır; bağlam alanları şunlardır:
  - *AccountingToken*
  - *ApplIdentityData*
  - *ApplOriginData*
  - *PutApplName*
  - *PutApplType*
  - *PutDate*
  - *PutTime*
  - *UserIdentifier*
- version-2 alanları (varsa) MQMD ' den kaldırılır ve version-2 alanlarından birinin ya da daha fazlasının varsayılan olmayan bir değeri varsa MQMDE yapısına taşınır.

## Uzak kuyruklara ileti konulması

Bir uygulama uzak kuyruğa bir ileti koyduğunda (uzak kuyruğun adını doğrudan belirterek ya da uzak kuyruğun yerel tanımlamasını kullanarak), yerel kuyruk yöneticisi:

- Yerleşik ileti tanımlayıcısını içeren bir MQXQH yapısı yaratır
- Gerekliyse ve önceden yoksa bir MQMDE ekler
- Uygulama ileti verilerini ekler
- İletiyi uygun bir iletim kuyruğuna yerleştirir

## İletilerin doğrudan iletim kuyruklarına yerleştirilmesi

Bir uygulama, iletiyi doğrudan bir iletim kuyruğuna da yerleştirebilir. Bu durumda uygulama, uygulama iletilisi verilerine bir MQXQH yapısıyla örnek eklemeli ve alanları uygun değerlerle kullanıma hazırlamalıdır. Buna ek olarak, MQPUT ya da MQPUT1 çağrısının **MsgDesc** değiştirilmesindeki *Format* alanı MQFMT\_XMIT\_Q\_HEADER değerine sahip olmalıdır.

Uygulama tarafından yaratılan MQXQH yapısındaki karakter verileri, yerel kuyruk yöneticisinin ( **CodedCharSetId** kuyruk yöneticisi özneliği tarafından tanımlanan) karakter kümesinde olmalı ve tamsayı verileri yerel makine kodlamasında bulunmalıdır. Buna ek olarak, MQXQH yapısındaki karakter verileri, alanın tanımlı uzunluğuna kadar boşluklarla doldurulacaktır; kuyruk yöneticisi, boş ve sonraki karakterleri MQXQH yapısında boşluklara dönüştürmediği için, veriler boş bir karakter kullanılarak zamanından önce sona erdirilmemelidir.

Ancak, kuyruk yöneticisi bir MQXQH yapısının var olduğunu ya da alanlar için geçerli değerlerin belirtildiğini denetlemez.

Uygulamalar iletilerini doğrudan SYSTEM.CLUSTER.TRANSMIT.QUEUE.

## İletim kuyruklarından ileti alma

Bir iletim kuyruğundan ileti alan uygulamaların, MQXQH yapısındaki bilgileri uygun şekilde işlemesi gerekir. Uygulama iletileri verilerinin başında MQXQH yapısının varlığı, MQGET çağrısının **MsgDesc** değiştirgesindeki *Format* alanında döndürülen MQFMT\_XMIT\_Q\_HEADER değeriyle gösterilir. **MsgDesc** değiştirgesindeki *CodedCharSetId* ve *Encoding* alanlarında döndürülen değerler, MQXQH yapısındaki karakter kümesi ve tamsayı verilerinin kodlamasını gösterir. Uygulama iletileri verilerinin karakter kümesi ve kodlaması, yerleşik ileti tanımlayıcısındaki *CodedCharSetId* ve *Encoding* alanları tarafından tanımlanır.

### **StrucId (MQCHAR4)**

Bu, yapı tanıtıcısıdır. Değer şu olmalıdır:

#### **MQXQH\_STRUC\_ID**

İletim kuyruğu üstbilgi yapısına ilişkin tanıtıcı.

C programlama dili için, MQXQH\_STRUC\_ID\_ARRAY değişmezi de tanımlanır; bu, MQXQH\_STRUC\_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisidir.

Bu alanın ilk değeri MQXQH\_STRUC\_ID 'dir.

### **Sürüm (MQUZE)**

Bu, yapı sürüm numarasıdır. Değer şu olmalıdır:

#### **MQXQH\_VERSION\_1**

İletim kuyruğu üstbilgi yapısına ilişkin sürüm numarası.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

#### **MQXQH\_CURRENT\_VERSION**

İletim kuyruğu üstbilgi yapısının yürürlükteki sürümü.

Bu alanın ilk değeri MQXQH\_VERSION\_1 'dir.

### **RemoteQName (MQCHAR48)**

This is the name of the message queue that is the apparent eventual destination for the message (this might prove not to be the eventual destination if, for example, this queue is defined at *RemoteQMgrName* to be a local definition of another remote queue).

İleti bir dağıtım listesi iletiyse (yani, yerleşik ileti tanımlayıcısındaki *Format* alanı MQFMT\_DIST\_HEADER 'dir), *RemoteQName* boş olur.

Bu alanın uzunluğu MQ\_Q\_NAME\_LENGTH değeri tarafından verilir. Bu alanın başlangıç değeri, C ' de boş değer dizgisidir ve diğer programlama dillerindeki 48 boş karakter dizilimidir.

### **RemoteQMgrAd (MQCHAR48)**

Bu ad, ileti için nihai hedef olan kuyruğa sahip kuyruk yöneticisi ya da kuyruk paylaşım grubunun adıdır.

İleti bir dağıtım listesi iletisiyse, *RemoteQMgrName* boş olur.

Bu alanın uzunluğu MQ\_Q\_MGR\_NAME\_LENGTH değeri tarafından verilir. Bu alanın başlangıç değeri, C ' de boş değer dizgisidir ve diğer programlama dillerindeki 48 boş karakter dizilimidir.

### **MsgDesc (MQMD1)**

Bu, yerleşik ileti tanımlayıcısıdır ve ileti ilk olarak uzak kuyruğa konduğunda, MQPUT ya da MQPUT1 çağrısında **MsgDesc** değiştirgesi olarak belirtilen ileti tanımlayıcı MQMD ' nin bir kopyasını içerir.


**Not:** Bu bir version-1 MQMD ' dir.

Bu yapıdaki alanların ilk değerleri, MQMD yapısındaki alanlarla aynıdır.

## İşlev çağrıları

Bu kısım, mümkün olan tüm MQI çağrılarına ilişkin bilgi verir. Açıklamalar, sözdizimi, parametre bilgileri, kullanım notları ve her bir olası dile ilişkin dil çağrıları farklı çağrıların her biri için verilmiştir.

### İlgili başvurular

 MQI çağrılarından CEDF çıkışı örnekleri

### Arama açıklamaları

Bu bölümde, MQI çağrıları açıklanır.

- [“MQBACK-Değişiklikleri geri al” sayfa 617](#)
- [“MQBEGIN-İş birimini başlat” sayfa 621](#)
- [“MQBUFMH-Arabelleği ileti tutamaçına dönüştür” sayfa 624](#)
- [“MQCB-Geri çağırmaya yönet” sayfa 628](#)
- [“MQCB\\_FUNCTION-Callback işlevi” sayfa 637](#)
- [“MQCLOSE-Nesneyi kapat” sayfa 638](#)
- [“MQCMIT-Kesinleştirme değişiklikleri” sayfa 646](#)
- [“MQCONN-Bağlantı kuyruğu yöneticisi” sayfa 650](#)
- [“MQCONNX-Connect kuyruk yöneticisi \(genişletilmiş\)” sayfa 657](#)
- [“MQCRTMH-İleti tanıtıcısı yarat” sayfa 663](#)
- [“MQCTL-Denetim çağrıları” sayfa 666](#)
- [“MQDISC-Bağlantı kesme kuyruğu yöneticisi” sayfa 672](#)
- [“MQDLTMH-İleti silme tanıtıcısı” sayfa 676](#)
- [“MQDLTMP-İleti özelliğini sil” sayfa 678](#)
- [“MQGET-İletiyi al” sayfa 681](#)
- [“MQINQ-Nesne özniteliklerinin sorgulması” sayfa 693](#)
- [“MQINQMP-Questery ileti özelliği” sayfa 710](#)
- [“MQMHBUF-İleti tanıtıcısını arabelleğe dönüştür” sayfa 716](#)
- [“MQOpen-Open nesnesi” sayfa 720](#)
- [“MQPUT-İleti koy” sayfa 737](#)
- [“MQPUT1 -Bir ileti girin” sayfa 750](#)
- [“MQSET-Nesne özniteliklerinin ayarlanması” sayfa 761](#)
- [“MQSETMP-İleti özelliğini ayarla” sayfa 767](#)
- [“MQSTAT-Durum bilgilerini al” sayfa 771](#)
- [“MQMHBUF-İleti tanıtıcısını arabelleğe dönüştür” sayfa 716](#)
- [“MQSUB-Abonelikleri kaydet” sayfa 775](#)
- [“MQSUBRQ-Abonelik isteği” sayfa 782](#)

Online help on UNIX platforms, in the form of *adam* pages, is available for these calls.

**Not:** Veri dönüştürme, MQXCNCV ve MQ\_DATA\_CONV\_EXIT ile ilişkili çağrılar [“Veri dönüştürme çıkışı” sayfa 891](#) içinde yer alıyor.

### Çağrı açıklamalarında kullanılan kurallar

Her çağrı için, bu konu derlemi, çağrıyı programlama dilinden bağımsız bir biçimde değiştirmenin ve kullanımın bir tanımını verir. Bu, desteklenen programlama dillerinin her birinde, çağrıya ilişkin tipik çağrılar ve parametrelerinin tipik bildirimleriyle birlikte gelir.

**Önemli:** When coding IBM MQ API calls you must ensure that all relevant parameters (as described in the following sections) are provided. Bunun yapmaması, önceden kestirilemeyen sonuçlar üretebilir.

Her bir çağrıya ilişkin açıklama aşağıdaki bölümleri içerir:

#### **Arama adı**

Arama adı ve ardından arama amacına ilişkin kısa bir açıklama vardır.

#### **Parametreler**

Her parametre için, adın arkasından parantez içinde veri tipi gelir () ve aşağıdakilerden biri:

##### **Giriş**

Çağrıyı yaparken parametreye ilişkin bilgileri belirtmenizi sağlar.

##### **output**

Kuyruk yöneticisi, çağrı tamamlandığında ya da başarısız olduğunda parametreye ilişkin bilgileri döndürür.

##### **giriş/çıkış**

Çağrıyı yaparken parametreye ilişkin bilgileri belirttiğinizde ve kuyruk yöneticisi, arama tamamlandığında ya da başarısız olduğunda bilgileri değiştirir.

Örneğin:

*Compcode* (MQUVT)-çıkış

Bazı durumlarda, veri tipi bir yapıdır. Tüm durumlarda, [“Temel veri tipleri” sayfa 235](#) içindeki veri türü ya da yapısıyla ilgili daha fazla bilgi vardır.

Her çağrıdaki son iki parametre bir tamamlanma kodu ve neden kodudur. Tamamlama kodu, çağrılarının başarıyla tamamlanıp tamamlanmadığını, kısmen mi, yoksa hiç mi tamamlanmadığını gösterir. Kısmi başarıyla ilgili daha fazla bilgi ya da çağrı başarısızlığı neden kodunda verilmiştir. Her bir tamamlanma ve neden koduyla ilgili daha fazla bilgi için bkz. [“Dönüş kodları” sayfa 859](#).

#### **Kullanım notları**

Çağrıya ilgili ek bilgi, bunu nasıl kullanacağını ve kullanımıyla ilgili herhangi bir kısıtlamayı tanımlıyor.

#### **Çevirici dili çağırma**

Çağrıyı tipik olarak çağırma ve parametrelerinin bildirme dilini, çevirici dilde bildirme.

#### **C çağırma**

Çağrıyı tipik olarak çağırma ve parametrelerinin bildirgesi C ' de.

#### **Cobol çağırma**

Çağrıyı tipik olarak çağırma ve parametrelerinin bildirgesi COBOL ' de.

#### **PL/I çağırımı**

PL/I. ' da çağrılan ve parametrelerinin bildirimine ilişkin tipik çağrı.

Tüm parametreler başvuru yoluyla iletilir.

#### **Visual Basic çağırısı**

Çağrıyı tipik olarak çağırma ve parametrelerinin bildirimini, Visual Basic içinde.

Diğer gösterim kuralları şunlardır:

#### **Değişmezler**

Değişmezlerin adları büyük harfli olarak gösterilir; örneğin, MQOO\_OUTPUT. Aynı öneki taşıyan bir sabitler kümesi şu şekilde gösterilir: MQIA\_\*. Bir sabitin değeri için bkz. [“Değişmezler” sayfa 61](#).

#### **Diziler**

Bazı çağrılarda, parametreler sabit büyüklüklere sahip olmayan karakter dizgileri dizileridir. Bu parametrelerin açıklamalarında, küçük harf bir sayısal değişmezi gösterir. Bu parametreye ilişkin bildirim kodladığınızda, n değerini gereksinim duyduğunuz sayısal değerle değiştirin.

#### **C dilindeki çağrıları kullanma**

*yalnızca giriş* olan ve MQHCONN, MQHOBJ, MQHMSG ya da MQHOMEN tipinde olan parametreler değere göre geçirilir. Diğer tüm parametreler için, parametrenin *adres*i değeri değer olarak geçirilir.

Bir işlevi her çağırduğunuzda adresle geçirilen tüm parametreleri belirtmenize gerek yoktur. Belirli bir değiştirgeye gerek olmadığı durumlarda, parametre verilerinin adresi yerine, işlev çağırımında değiştirge olarak boş değerli bir gösterge belirtin. Bunun olası olduğu parametreler, çağrı açıklamalarında tanımlanır.

Çağrın değeri olarak bir parametre döndürülmez; C terminolojisinde, bu, tüm çağrıların geçersizdöndürdüğü anlamına gelir.

#### *Arabellek parametresinin bildirilmesi*

**MQGET**, **MQPUT** ve **MQPUT1** çağrıları her birinin tanımlanmamış bir veri tipine sahip bir parametresi vardır: *Arabellek* parametresi. Uygulamanın ileti verilerini göndermek ve almak için bu değiştirgeyi kullanın.

Bu sıralamayı içeren değiştirgeler, C örneklerinde MQBYTE dizisi olarak gösterilir. Parametreleri bu şekilde bildirebilirsiniz, ancak genellikle iletilerde verilerin yerleşim düzenini tanımlayan belirli bir yapı olarak bildirilmesi daha uygun olur. İşlev prototipi, parametreyi bir işaretçi olarak bildirir; böylece, çağrı çağrısındaki parametre olarak herhangi bir veri sırasının adresini belirleyebilirsiniz.

İşaretçi-boşluk, tanımsız biçimdeki verileri gösteren bir göstergedir. Şu şekilde tanımlanır:

```
typedef void *PMQVOID;
```

## **MQBACK-Değişiklikleri geri al**

MQBACK çağrısı, kuyruk yöneticisine, son eşitleme noktasının yedekleneceği için, tüm iletinin aldığı ve oluştuğunu belirten tüm iletileri gösterir.

Bir iş biriminin bir parçası olarak gönderilen iletiler silinir; iş biriminin bir parçası olarak alınan iletiler kuyruktan geri alınır.

- z/OS' ta bu çağrı yalnızca toplu programlar ( IMS toplu DL/I programları da içinde olmak üzere) tarafından kullanılır.

## **Sözdizimi**

MQBACK (*Hconn*, *Compcode*, *Reason*)

## **Parametreler**

### **Hconn**

Tip: MQHCONN-input

Bu tanıtıcı, kuyruk yöneticisiyle olan bağlantıyı gösterir. The value of *Hconn* was returned by a previous MQCONN or MQCONNX call.

### **KodKodu**

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Tamamlanma kodu; bu kod aşağıdakilerden biridir:

#### **MQCC\_OK**

İşlem başarıyla tamamlandı.

#### **MQCC\_UYARI**

Uyarı (kısmi tamamlama).

#### **MQCC\_FAILED**

Arama başarısız oldu.

### **Neden**

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

*CompCode* MQCC\_OK ise:

#### **MQRC\_NONE**

(0, X'000 ') Raporlamak için bir neden yok.

CompCode MQCC\_UYARI ise:

**MQRC\_OUTCOME\_PENDING**

(2124, X'84C') Geri dönüş işleminin sonucu beklemede.

CompCode MQCC\_FAILED ise:

**MQRC\_ADAPTER\_SERV\_LOAD\_ERROR**

(2130, X'852 ') Bağdaştırıcı hizmeti modülü yüklenemiyor.

**MQRC\_API\_EXIT\_ERROR**

(2374, X' 946 ') API çıkışı başarısız oldu.

**MQRC\_ASID\_MISSMACH**

(2157, X'86D') Birincil ve ana sunucu ASID değerleri farklı.

**MQRC\_CALL\_IN\_PROGRESS**

(2219, X'8AB') Önceki arama tamamlanmadan önce girilen MQI çağrısı.

**MQRC\_CF\_STRUC\_IN\_USE**

(2346, X'92A') Coupling-tesis yapısı kullanımda.

**MQRC\_CONNECTION\_BROKEN**

(2009, X'7D9') Kuyruk yöneticisine yönelik bağlantı kaybedildi.

**MQRC\_ENVIRONMENT\_ERROR**

(2012, X'7DC') Ortamda arama geçerli değil.

**MQRC\_HCONN\_ERROR**

(2018, X'7E2') Bağlantı tanıtıcısı geçerli değil.

**MQRC\_OBJECT\_ZARAR**

(2101, X'835 ') Nesne zarar gördü.

**MQRC\_OUTCOME\_MIXED**

(2123, X'84B') Kesinleştirme ya da geri dönüş işlemi sonucu karışık.

**MQRC\_Q\_MGR\_STOPPING**

(2162, X'872 ') Kuyruk yöneticisi sona erdiriliyor.

**MQRC\_RESOURCE\_SORUNU**

(2102, X'836 ') Yetersiz sistem kaynakları var.

**MQRC\_STORAGE\_MEDIUM\_FULL**

(2192, X'890 ') Dış depolama ortamı dolu.

**MQRC\_STORAGE\_NOT\_AVAM**

(2071, X'817 ') Kullanılabilir bellek yetersiz.

**MQRC\_UNEXPECTED\_ERROR**

(2195, X'893 ') Beklenmeyen bir hata oluştu.

Bu kodlara ilişkin ayrıntılı bilgi için [İletiler ve neden kodları](#) başlıklı konuya bakın.

## Kullanım notları

1. Bu çağrıyı, yalnızca kuyruk yöneticisinin iş birimini koordine ettiğinde kullanabilirsiniz. Bu durumda şunlar olabilir:

- Değişikliklerin yalnızca MQ kaynaklarını etkilediği yerel bir iş birimi.
- Değişikliklerin, diğer kaynak yöneticilerine ait kaynakları etkileyebileceği gibi, MQ kaynaklarını etkilediği genel bir iş birimi.

Yerel ve genel çalışma birimleriyle ilgili daha fazla ayrıntı için bkz. [“MQBEGIN-İş birimini başlat” sayfa 621.](#)

2. Kuyruk yöneticisinin iş birimini koordine etmediği ortamlarda, MQBACK yerine uygun geri çağırma çağrısını kullanın. Ortam, uygulamanın olağan dışı bir şekilde sonlandırılmasına neden olan örtük bir geri dönüş özelliğini de destekleyebilir.

- z/OS' ta aşağıdaki çağrıları kullanın:

- İş birimi yalnızca MQ kaynaklarını etkilediğinde, toplu iş programları ( IMS toplu DL/I programları da içinde olmak üzere) MQBACK çağrısını kullanabilir. Ancak, iş birimi hem MQ kaynaklarını ve diğer kaynak yöneticilerine ait kaynakları etkiliyorsa (örneğin, Db2 ), z/OS Kurtarılabılır Kaynak Hizmeti (RRS) tarafından sağlanan SRRBACK çağrısını kullanın. SRRBACK çağrısı, RRS eşgüdümü için etkinleştirilen kaynak yöneticilerine ait kaynaklarda yapılan değişiklikleri destekliyor.
  - CICS applications must use the EXEC CICS SYNCPOINT ROLLBACK command to back out the unit of work. Do not use the MQBACK call for CICS applications.
  - IMS uygulamaları (toplu DL/I programlarından farklı), iş birimini yedeklemek için ROLB gibi IMS çağrıları kullanılmalıdır. Do not use the MQBACK call for IMS applications (other than batch DL/I programs).
- IBM 'i ta, kuyruk yöneticisi tarafından koordine edilen yerel iş birimleri için bu çağrıyı kullanın. This means that a commitment definition must not exist at job level, that is, the STRCMTCTL command with the **CMTSCOPE (\*JOB)** parameter must not have been issued for the job.
3. Bir uygulama, bir iş biriminde kesinleştirilmemiş değişikliklerle sona ererse, bu değişikliklerin yok edilmesi, uygulamanın olağan ya da olağan dışı bir şekilde sona ermesine bağlıdır. Ek ayrıntılar için [“MQDISC-Bağlantı kesme kuyruğu yöneticisi” sayfa 672](#) içindeki kullanım notlarına bakın.
  4. Bir uygulama, gruplara ya da mantıksal ileti bölümlerine ileti yerleştirdiğinde ya da ileti aldığı anda, kuyruk yöneticisi, son başarılı MQPUT ve MQGET çağrılarına ilişkin ileti grubuyla ve mantıksal iletiyle ilgili bilgileri saklar. Bu bilgiler kuyruk tanıtıcısı ile ilişkilendirilir ve aşağıdaki gibi şeyler içerir:
    - MQMD ' de *GroupId, MsgSeqNumber, Offset* ve *MsgFlags* alanlarının değerleri.
    - İletinin bir iş biriminin parçası olup olmadığı.
    - MQPUT çağrısına ilişkin: İletinin kalıcı mı, yoksa kalıcı değil mi olduğu.

The queue manager keeps üç sets of group and segment information, one set for each of the following:

    - Son başarılı MQPUT çağrısı (bu, bir iş biriminin bir parçası olabilir).
    - Kuyruktan ileti kaldıran son başarılı MQGET çağrısı (bu, bir iş biriminin parçası olabilir).
    - Kuyruktan ileti alan son başarılı MQGET çağrısı (bu, bir iş biriminin parçası olamaz).
  5. MQGET çağrısıyla ilişkili bilgiler, yürürlükteki iş biriminde o kuyruk tanıtıcısı için ilk başarılı MQGET çağrısından önce sahip olduğu değere geri yüklenir.

Çalışma birimi başladıktan sonra uygulama tarafından güncellenen, ancak iş biriminin kapsamı dışında olan kuyruklar, iş birimi geriletilirse, grup ve bölüm bilgilerini geri yüklemez.

Bir iş birimi yedeklendiğinde grup ve kesim bilgilerinin önceki değerine geri yüklenmesi, uygulamanın çok sayıda iş biriminden oluşan büyük bir ileti grubu ya da büyük mantıksal ileti yayınlamasını ve iş birimlerinden biri başarısız olursa, ileti grubunun ya da mantıksal iletinin doğru noktasında yeniden başlatılmasına olanak tanır.

Yerel kuyruk yöneticisinin kuyruk depolaması sınırlı olması durumunda, birkaç iş birimi kullanılması avantajlı olabilir. Ancak, bir sistem hatası ortaya çıkarsa, uygulama, iletileri yeniden başlatabilmek ya da iletileri doğru noktaya getirmek için yeterli bilgileri sağlamalıdır.

Bir sistem hatasından sonra doğru noktada yeniden başlatılabilmeye ilişkin ayrıntılar için, [“MQPMO-İleti koyma seçenekleri” sayfa 492](#) içinde açıklanan MQPMO\_LOGICAL\_ORDER seçeneğine ve [“MQGMO-Get-message seçenekleri” sayfa 364](#) içinde açıklanan MQGMO\_LOGICAL\_ORDER seçeneğine bakın.

Kalan kullanım notları, yalnızca kuyruk yöneticisi iş birimlerini koordine ettiğinde geçerlidir.
  6. Bir iş birimi, bağlantı tanıtıcısı ile aynı kapsama sahiptir. Belirli bir iş birimini etkileyen tüm MQ çağrıları, aynı bağlantı tanıtıcısı kullanılarak gerçekleştirilmelidir. Farklı bir bağlantı tanıtıcısı kullanılarak verilen çağrılar (örneğin, başka bir uygulama tarafından yayınlanan çağrılar) farklı bir iş birimini etkiler. Bağlantı tutamaçlarının kapsamı hakkında bilgi için [“MQCONN-Bağlantı kuyruğu yöneticisi” sayfa 650](#) içinde açıklanan **Hconn** parametresine bakın.
  7. Bu çağrıdan etkilenecek, yalnızca yürürlükteki iş biriminin bir parçası olarak alınan ya da alınan iletiler etkilenir.

8. Bir iş birimi içinde MQGET, MQPUT ya da MQPUT1 tarafından çağrılan, ancak bir kesinleştirme ya da geriletme çağrısını hiçbir zaman yayınlamayan, uzun süredir çalışan bir uygulama, diğer uygulamaların kullanımına sunulmayan iletilerle kuyrukları doldurabilir. Bu olasılığa karşı koruma sağlamak için, denetiminin **MaxUncommittedMsgs** kuyruk yöneticisi özniteliğini, kuyrukları dolduran kaçak uygulamaları önleyecek kadar düşük bir değere ayarlaması gerekir, ancak beklenen ileti sistemi uygulamalarının doğru şekilde çalışmasına izin verecek kadar yüksek olmalıdır.

## C çağırma

```
MQBACK (Hconn, &CompCode, &Reason);
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
MQHCONN  Hconn;      /* Connection handle */
MQLONG   CompCode;   /* Completion code */
MQLONG   Reason;     /* Reason code qualifying CompCode */
```

## Cobol çağırma

```
CALL 'MQBACK' USING HCONN, COMPCODE, REASON.
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
** Connection handle
 01 HCONN    PIC S9(9) BINARY.
** Completion code
 01 COMPCODE PIC S9(9) BINARY.
** Reason code qualifying COMPCODE
 01 REASON   PIC S9(9) BINARY.
```

## PL/I çağırımı

```
call MQBACK (Hconn, CompCode, Reason);
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
dcl Hconn      fixed bin(31); /* Connection handle */
dcl CompCode   fixed bin(31); /* Completion code */
dcl Reason     fixed bin(31); /* Reason code qualifying CompCode */
```

## High Level Assembler çağırısı

```
CALL MQBACK,(HCONN,COMPCODE,REASON)
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
HCONN      DS F Connection handle
COMPCODE   DS F Completion code
REASON     DS F Reason code qualifying COMPCODE
```

## Visual Basic çağırısı

```
MQBACK Hconn, CompCode, Reason
```



Değişirgeleri aşağıdaki gibi bildirin:

```
Dim Hconn As Long 'Connection handle'  
Dim CompCode As Long 'Completion code'  
Dim Reason As Long 'Reason code qualifying CompCode'
```

## MQBEGIN-İş birimini başlat

MQBEGIN çağırısı, kuyruk yöneticisi tarafından eşgüdümlü bir iş birimi başlatır ve bu, dış kaynak yöneticilerini içerebilir.

### Sözdizimi

MQBEGIN (*Hconn*, *BeginOptions*, *Compcode*, *Reason*)

### Parametreler

#### Hconn

Tip: MQHCONN-input

Bu tanırtıcı, kuyruk yöneticisiyle olan bağlantıyı gösterir. The value of *Hconn* was returned by a previous MQCONN or MQCONNX call.

*Hconn* , paylaşılmayan bir bağlantı tanırtıcısı olmalıdır. Paylaşılan bir bağlantı tanırtıcısı belirtilirse, çağrı neden kodu MQRC\_HCONN\_ERROR ile başarısız olur. Paylaşılan ve paylaşılmayan tutamaçlar hakkında daha fazla bilgi için, [“MQCNO-Bağlantı seçenekleri” sayfa 314](#) içindeki MQCNO\_HANDLE\_SHARE\_\* seçeneklerinin açıklamasına bakın.

#### BeginOptions

Tip: MQBO-giriş/çıkış

Bunlar, [“MQBO-Başlangıç seçenekleri” sayfa 277](#) içinde açıklandığı şekilde, MQBEGIN eylemini denetleyen seçeneklerdir.

Herhangi bir seçenek gerekmiyorsa, C ya da S/390 çevirici programında yazılan programlar, bir MQBO yapısının adresini belirtmek yerine, boş değerli bir parametre adresi belirtebilir.

#### CompCode

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Tamamlanma kodu; bu kod aşağıdakilerden biridir:

##### MQCC\_OK

İşlem başarıyla tamamlandı.

##### MQCC\_UYARI

Uyarı (kısmi tamamlama).

##### MQCC\_FAILED

Arama başarısız oldu.

#### Neden

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

*CompCode* MQCC\_OK ise:

##### MQRC\_NONE

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_UYARI ise:

##### MQRC\_NO\_EXTERNAL\_KATILIMCILARI

(2121, X'849 ') Katılan kaynak yöneticisi kayıtlı değil.

**MQRC\_PARTICIPANT\_NOT\_VAR**

(2122, X'84A') Katılan kaynak yöneticisi kullanılamıyor.

*CompCode* MQCC\_FAILED ise:

**MQRC\_API\_EXIT\_ERROR**

(2374, X'946 ') API çıkışı başarısız oldu.

**MQRC\_BO\_HATASI**

(2134, X'856 ') Begin-options yapısı geçerli değil.

**MQRC\_CALL\_IN\_PROGRESS**

(2219, X'8AB') Önceki arama tamamlanmadan önce girilen MQI çağrısı.

**MQRC\_CONNECTION\_BROKEN**

(2009, X'7D9') Kuyruk yöneticisine yönelik bağlantı kaybedildi.

**MQRC\_ENVIRONMENT\_ERROR**

(2012, X'7DC') Ortamda arama geçerli değil.

**MQRC\_HCONN\_ERROR**

(2018, X'7E2') Bağlantı tanıtıcısı geçerli değil.

**MQRC\_OPTIONS\_ERROR**

(2046, X'7FE') Seçenekler geçerli değil ya da tutarlı değil.

**MQRC\_Q\_MGR\_STOPPING**

(2162, X'872 ') Kuyruk yöneticisi sona erdiriliyor.

**MQRC\_RESOURCE\_SORUNU**

(2102, X'836 ') Yetersiz sistem kaynakları var.

**MQRC\_STORAGE\_NOT\_AVAM**

(2071, X'817 ') Kullanılabilir bellek yetersiz.

**MQRC\_UNEXPECTED\_ERROR**

(2195, X'893 ') Beklenmeyen bir hata oluştu.

**MQRC\_UOW\_IN\_PROGRESS**

(2128, X'850 ') İş birimi zaten başlatıldı.

Bu kodlara ilişkin ek bilgi için [İletiler ve neden kodları](#) başlıklı konuya bakın.

## Kullanım notları

1. Kuyruk yöneticisi tarafından eşgüdümlü bir çalışma birimi başlatmak için MQBEGIN çağrısını kullanın ve diğer kaynak yöneticilerinin sahip olduğu kaynaklara ilişkin değişiklikleri içerebilecek bir iş birimi başlatın. Kuyruk yöneticisi üç tip iş birimi tipini destekler:
  - **Kuyruk yöneticisi-eşgüdümlü yerel iş birimi:** Kuyruk yöneticisinin katılan tek kaynak yöneticisi olduğu bir iş birimi ve kuyruk yöneticisi iş birimi eşgüdümçüsü olarak işlev görür.
    - Bu iş birimi tipini başlatmak için, ilk MQPUT, MQPUT1 ya da MQGET çağrısında MQPMO\_SYNCPOINT ya da MQGMO\_SYNCPOINT seçeneğini belirtin.
    - Bu iş birimi tipini kesinleştirmek ya da yedeklemek için, MQCMIT ya da MQBACK çağrısını kullanın.
  - **Kuyruk yöneticisi-eşgüdümlü genel iş birimi:** Kuyruk yöneticisinin iş birimi eşgüdümçüsü olarak işlev göreceği iş birimi; hem MQ kaynakları *hem de diğer kaynak yöneticilerine ait kaynaklar için* . Bu kaynak yöneticileri, iş birimindeki kaynaklarda yapılan tüm değişikliklerin kesinleştirildiğinden ya da yedeklendiğinden emin olmak için kuyruk yöneticisiyle işbirliği yapar.
    - Bu tip bir iş birimi başlatmak için, MQBEGIN çağrısını kullanın.
    - Bu iş birimi tipini kesinleştirmek ya da geri göndermek için, MQCMIT ve MQBACK çağrılarını kullanın.
  - **Dışarıdan eşgüdümlü genel iş birimi:** Kuyruk yöneticisinin bir katılımcı olduğu, ancak kuyruk yöneticisinin iş birimi eşgüdümçüsü olarak işlev görmediği bir iş birimi. Bunun yerine, kuyruk yöneticisinin işbirliği yaptığı bir iş birimi eşgüdümçüsü vardır.

- Bu tip bir iş birimi başlatmak için, dış iş birimi eşgüdümçüsü tarafından sağlanan ilgili çağrıyı kullanın.  
İş birimini başlatmayı denemek için MQSTART çağrısı kullanılırsa, çağrı neden kodu MQRCE\_ENVIRONMENT\_ERROR ile başarısız olur.
  - Bu tip bir iş birimini kesinleştirmek ya da yedeklemek için, dış iş birimi eşgüdümçüsü tarafından sağlanan kesinleştirme ve geri gönderme çağrılarını kullanın.  
MQCMIT ya da MQBACK çağrısını çalışma birimini kesinleştirmek ya da yedeklemek için kullanırsanız, çağrı neden kodu MQRCE\_ENVIRONMENT\_ERROR ile başarısız olur.
2. Uygulama bir iş biriminde kesinleştirilmemiş değişikliklerle sona ererse, bu değişikliklerin yok edilmesi, uygulamanın olağan ya da olağandışı bir şekilde sona ermesine bağlıdır. Ek ayrıntılar için “MQDISC-Bağlantı kesme kuyruğu yöneticisi” sayfa 672 içindeki kullanım notlarına bakın.
  3. Bir uygulama, aynı anda yalnızca bir iş birimine katılabilir. Uygulamaya ilişkin var olan iş birimi tipine bakılmaksızın, MQRCE\_UOW\_IN\_PROGRESS neden kodu ile MQBEGIN çağrısı başarısız olur.
  4. Bir MQ MQI istemci ortamında MQBEGIN çağrısı geçerli değil. Çağrıyı kullanma girişimi başarısız oldu; neden kodu MQRCE\_ENVIRONMENT\_ERROR.
  5. Kuyruk yöneticisi, genel iş birimleri için iş birimi eşgüdümçüsü olarak işlev gördüğünde, iş birimine katılabilen kaynak yöneticileri kuyruk yöneticisi yapılanış kütüğünde tanımlanır.
  6. IBM i' ta, üç tip iş birimi aşağıdaki şekilde desteklenir:
    - **Kuyruk yöneticisi-eşgüdümlü yerel iş birimi** yalnızca, bir kesinleştirme tanımı iş düzeyinde bulunmadığında kullanılabilir. Bu durumda, iş için **CMTSCOPE (\*JOB)** parametresine sahip STRCMTCTL komutu verilmemiş olmalıdır.
    - **Kuyruk yöneticisi-eşgüdümlü genel iş birimi** desteklenmiyor.
    - **Dışarıdan eşgüdümlü genel iş birimi** yalnızca, bir kesinleştirme tanımı iş düzeyinde olduğunda kullanılabilir; bu durumda, iş için **CMTSCOPE (\*JOB)** parametresiyle STRCMTCTL parametresine sahip bir komut verilmelidir. Bu işlem yapıldıysa, IBM i COMMIT ve ROLLBACK işlemleri, diğer katılımcı kaynak yöneticilerine ait kaynakların yanı sıra MQ kaynakları için de geçerlidir.

## C çağırma

```
MQBEGIN (Hconn, &BeginOptions, &CompCode, &Reason);
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
MQHCONN Hconn;          /* Connection handle */
MQBO     BeginOptions;  /* Options that control the action of MQBEGIN */
MQLONG   CompCode;     /* Completion code */
MQLONG   Reason;       /* Reason code qualifying CompCode */
```

## Cobol çağırma

```
CALL 'MQBEGIN' USING HCONN, BEGINOPTIONS, COMPCODE, REASON.
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
** Connection handle
01 HCONN          PIC S9(9) BINARY.
** Options that control the action of MQBEGIN
01 BEGINOPTIONS.
   COPY CMQBOV.
** Completion code
01 COMPCODE      PIC S9(9) BINARY.
** Reason code qualifying COMPCODE
01 REASON        PIC S9(9) BINARY.
```

## PL/I çağırımı

```
call MQBEGIN (Hconn, BeginOptions, CompCode, Reason);
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
dcl Hconn          fixed bin(31); /* Connection handle */
dcl BeginOptions  like MQBO;     /* Options that control the action of
                                MQBEGIN */
dcl CompCode      fixed bin(31); /* Completion code */
dcl Reason        fixed bin(31); /* Reason code qualifying CompCode */
```

## Visual Basic çağırısı

```
MQBEGIN Hconn, BeginOptions, CompCode, Reason
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
Dim Hconn          As Long 'Connection handle'
Dim BeginOptions  As MQBO 'Options that control the action of MQBEGIN'
Dim CompCode      As Long 'Completion code'
Dim Reason        As Long 'Reason code qualifying CompCode'
```

## MQBUFMH-Arabelleği ileti tutamaçına dönüştür

MQBUFMH işlev çağırısı bir arabelleği ileti tanıtıcısı olarak dönüştürür ve MQMHBUF çağırısının tersi olur.

Bu çağrı, arabelleğindeki bir ileti tanımlayıcısını ve MQRFH2 özelliklerini alır ve bunları bir ileti tanıtıcısı aracılığıyla kullanılabilir kılar. İsteğe bağlı olarak, ileti verilerindeki MQRFH2 özellikleri kaldırılır. Özellikler kaldırıldıktan sonra arabelleğin içeriğini doğru şekilde tanımlamak için gerekirse, ileti tanımlayıcısının *Encoding, CodedCharSetIdve Format* alanları güncellenir.

## Sözdizimi

MQBUFMH (*Hconn, Hmsg, BufMsgHOpts, MsgDesc, BufferLength, Buffer, DataLength, Compcode, Reason*)

## Parametreler

### Hconn

Tip: MQHCONN-input

Bu tanıtıcı, kuyruk yöneticiyle olan bağlantıyı gösterir. **Hconn** değeri, **Hmsg** değiştirgesinde belirtilen ileti tanıtıcısını yaratmak için kullanılan bağlantı tanıtıcısı ile eşleşmelidir.

İleti tanıtıcısı MQHC\_UNASSOCIATED\_HCONN kullanılarak yaratıldıysa, bir arabelleği ileti tutamaçına dönüştüren iş parçacığıda geçerli bir bağlantı kurulmalıdır. Geçerli bir bağlantı kurulmamışsa, çağrı MQRC\_CONNECTION\_BROKEN ile başarısız olur.

### İt

Tip: MQHMQSG-input

Bu, arabelleğin doldurulması gereken ileti tanıtıcısıdır. Değer önceki bir MQCRTMH çağırısı tarafından döndürüldü.

### BufMsgHOpts

Tip: MQBMHO-input

MQBMHO yapısı, uygulamaların, ileti tutamaçlarının arabelleklerden nasıl üretildiğini denetleyen seçenekleri belirlemesine olanak sağlar.

Ayrıntılar için bkz. [“MQBMHO-İleti işleme seçenekleri için arabellek” sayfa 273.](#)

## MsgDesc

Tip: MQMD-giriş/çıkış

*MsgDesc* yapısı, ileti tanımlayıcı özelliklerini içerir ve arabellek alanının içeriğini açıklar.

Çağrıdan çıktıda, özellikler isteğe bağlı olarak arabellek alanından kaldırılır ve bu durumda, arabellek alanını doğru şekilde tanımlamak için ileti tanımlayıcısı güncellenir.

Bu yapıdaki veriler, uygulamanın karakter kümesi ve kodlamasında olmalıdır.

## BufferLength

Tip: MQHOME-girişi

*BufferLength*, arabellek alanının bayt cinsinden uzunluğudur.

Sıfır baytlık bir *BufferLength* değeri geçerlidir ve arabellek alanının veri içermediği anlamına gelir.

## Arabellek

Tip: MQBYTEExBufferUzunluk-giriş/çıkış

Bunlar, “MQBEGIN-İş birimini başlat” sayfa 621’inde açıklandığı şekilde, MQBEGIN eylemini denetleyen seçeneklerdir.

**Buffer**, ileti arabelleğinin bulunduğu alanı tanımlar. Çoğu veri için, arabelleği 4 byte 'lık bir sınır üzerinde hizalamanız gerekir.

**Buffer**, karakter ya da sayısal veri içeriyorsa, **MsgDesc** parametresindeki *CodedCharSetId* ve *Encoding* alanlarını, veriye uygun değerlere ayarlayın; bu, gerekirse verilerin dönüştürülmesini sağlar.

Özellikler ileti arabelleğinde bulunursa, bunlar isteğe bağlı olarak kaldırılır; daha sonra, çağrıdan dönüşte ileti tanıtıcısından kullanılabilir duruma gelir.

C programlama dilinde, parametre, parametre olarak belirtilebilecek herhangi bir veri tipinin adresi anlamına gelen, işaretçi (pointer-to-void) olarak bildirilir.

**BufferLength** parametresi sıfır ise, **Buffer** bu durumda gönderilmez; bu durumda, C ya da System/390 çevirici programlarında yazılan programlarla geçirilen parametre adresi boş değerli olabilir.

## DataLength

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Özelliklerin kaldırılmasına neden olabilecek arabelleğin bayt cinsinden uzunluğu.

## CompCode

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Tamamlanma kodu; bu kod aşağıdakilerden biridir:

### MQCC\_OK

İşlem başarıyla tamamlandı.

### MQCC\_FAILED

Arama başarısız oldu.

## Neden

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

*CompCode* MQCC\_OK ise:

### MQRC\_NONE

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_FAILED ise:

### MQRC\_ADAPTER\_NOT\_AVAM

(2204, X'089C') Bağdaştırıcı kullanılmıyor.

### MQRC\_ADAPTER\_SERV\_LOAD\_ERROR

(2130, X'852 ') Bağdaştırıcı hizmeti modülü yüklenemiyor.

**MQRC\_ASID\_MISSMACH**

(2157, X'86D') Birincil ve ana sunucu ASID değerleri farklı.

**MQRC\_BMHO\_ERROR**

(2489, X'09B9') Arabellek için ileti işleme seçenekleri yapısı geçerli değil.

**MQRC\_BUFFER\_ERROR**

(2004, X'07D4') Arabellek parametresi geçerli değil.

**MQRC\_BUFFER\_LENGTH\_ERROR**

(2005, X'07D5') Arabellek uzunluğu parametresi geçerli değil.

**MQRC\_CALL\_IN\_PROGRESS**

(2219, X'08AB') Önceki arama tamamlanmadan önce girilen MQI çağırısı.

**MQRC\_CONNECTION\_BROKEN**

(2009, X'07D9') Kuyruk yöneticisine yönelik bağlantı kaybedildi.

**MQRC\_HMSG\_ERROR**

(2460, X'099C') İleti tanıtıcısı geçerli değil.

**MQRC\_MD\_ERROR**

(2026, X'07EA') İleti tanımlayıcısı geçerli değil.

**MQRC\_MSG\_HANDLE\_IN\_USE**

(2499, X'09C3') İleti tanıtıcısı zaten kullanılıyor.

**MQRC\_OPTIONS\_ERROR**

(2046, X'07FE') Seçenekler geçerli değil ya da tutarlı değil.

**MQRC\_RFH\_ERROR**

(2334, X'091E') MQRFH2 yapısı geçerli değil.

**MQRC\_RFH\_FORMAT\_ERROR**

(2421, X'0975 ') Özellikler içeren bir MQRFH2 klasörü ayrıştırılmadı.

**MQRC\_UNEXPECTED\_ERROR**

(2195, X'893 ') Beklenmeyen bir hata oluştu.

Bu kodlarla ilgili ayrıntılı bilgi için [İletiler ve neden kodları](#) başlıklı konuya bakın.

## Kullanım notları

MQBUFMMH çağrılarını API çıkışlarıyla algılayamaz; arabellek, uygulama alanındaki bir ileti tanıtıcısı içine dönüştürülür; çağrı kuyruk yöneticisine ulaşmaz.

## C çağırma

```
MQBUFMMH (Hconn, Hmsg, &BufMsgHOpts, &MsgDesc, BufferLength, Buffer,  
&DataLength, &CompCode, &Reason);
```

Değiştiricileri aşağıdaki gibi bildirin:

```
MQHCONN Hconn;          /* Connection handle */  
MQHMSG  Hmsg;          /* Message handle */  
MQBMHO  BufMsgHOpts;  /* Options that control the action of MQBUFMMH */  
MQMD    MsgDesc;      /* Message descriptor */  
MQLONG  BufferLength;  /* Length in bytes of the Buffer area */  
MQBYTE  Buffer[n];     /* Area to contain the message buffer */  
MQLONG  DataLength;   /* Length of the output buffer */  
MQLONG  CompCode;     /* Completion code */  
MQLONG  Reason;       /* Reason code qualifying CompCode */
```

## Cobol çağırma

```
CALL 'MQBUFMH' USING HCONN, HMSG, BUFMSGHOPTS, MSGDESC, BUFFERLENGTH,  
BUFFER, DATALENGTH, COMPCODE, REASON.
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
** Connection handle  
01 HCONN          PIC S9(9) BINARY.  
** Message handle  
01 HMSG          PIC S9(18) BINARY.  
** Options that control the action of MQBUFMH  
01 BUFMSGHOPTS.  
   COPY CMQBMHOV.  
** Message descriptor  
01 MSGDESC.  
   COPY CMQMD.  
** Length in bytes of the Buffer area  
01 BUFFERLENGTH PIC S9(9) BINARY.  
** Area to contain the message buffer  
01 BUFFER        PIC X(n).  
** Length of the output buffer  
01 DATALENGTH  PIC S9(9) BINARY.  
** Completion code  
01 COMPCODE     PIC S9(9) BINARY.  
** Reason code qualifying COMPCODE  
01 REASON       PIC S9(9) BINARY.
```

## PL/I çağırımı

```
call MQBUFMH (Hconn, Hmsg, BufMsgHOpts, MsgDesc, BufferLength, Buffer,  
DataLength, CompCode, Reason);
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
dcl Hconn          fixed bin(31); /* Connection handle */  
dcl Hmsg           fixed bin(63); /* Message handle */  
dcl BufMsgHOpts   like MQBMHO; /* Options that control the action of  
                               MQBUFMH */  
dcl MsgDesc       like MQMD; /* Message descriptor */  
dcl BufferLength   fixed bin(31); /* Length in bytes of the Buffer area */  
dcl Buffer         char(n); /* Area to contain the message buffer */  
dcl DataLength    fixed bin(31); /* Length of the output buffer */  
dcl CompCode      fixed bin(31); /* Completion code */  
dcl Reason        fixed bin(31); /* Reason code qualifying CompCode */
```

## High Level Assembler çağırısı

```
CALL MQBUFMH, (HCONN, HMSG, BUFMSGHOPTS, MSGDESC, BUFFERLENGTH, BUFFER,  
DATALENGTH, COMPCODE, REASON)
```

Değiştirgeleri aşağıdaki gibi bildirin:

HCONN	DS	F	Connection handle
HMSG	DS	D	Message handle
BUFMSGHOPTS	CMQBMHOA	,	Options that control the action of MQBUFMH
MSGDESC	CMQMDA	,	Message descriptor
BUFFERLENGTH	DS	F	Length in bytes of the BUFFER area
BUFFER	DS	CL(n)	Area to contain the properties
DATALENGTH	DS	F	Length of the output buffer
COMPCODE	DS	F	Completion code
REASON	DS	F	Reason code qualifying COMPCODE

## MQCB-Geri çağırma yönet

MQCB çağrısı, belirtilen nesne tanıtıcısı için bir geri çağırma kaydını kaydeder ve geri bildirme için etkinleştirmeyi ve değişiklikleri denetler.

Geri çağrı, belirli olaylar oluştuğunda IBM MQ tarafından çağrılan bir kod parçasıdır (dinamik olarak bağlantılı ya da işlev olarak işlev görebilen bir işlevin adı olarak belirtilir).

Bir istemciye MQCB ve MQCTL ' yi kullanmak için, kanaldaki anlaşmalı **SHARECNV** parametresinin sıfır dışında bir değer kabul ettiği bir sunucuya bağlanmanız gerekir.

Tanımlanabilen geri çağırma tipleri şunlardır:

### İleti tüketicisi

Bir ileti tüketicisi geri bildirme işlevi, bir ileti, belirtilen seçim ölçütlerine uygun olarak bir nesne tanıtıcısında kullanılabilir olduğunda çağrılır.

Her nesne tanıtıcısı için yalnızca bir geri çağırma işlevi kaydedilebilir. Tek bir kuyruk birden çok seçim ölçütüyle okunacaksa, kuyruğun birden çok kez açılması ve her bir tutamaçla ilgili bir tüketici işlevi kaydedilmelidir.

### Olay işleyici

Olay işleyici, tüm geri bildirme ortamını etkileyen koşullar için çağrılır.

Bu işlev, bir olay koşulu oluştuğunda (örneğin, bir kuyruk yöneticisi ya da bağlantı durdurma ya da susturucu) çağrılır.

Bu işlev, tek bir ileti tüketicisine özgü koşullar için çağrılmaz; örneğin, MQRC\_GET\_INHIBITE;, ancak bir geri çağırma işlevi olağan şekilde bitmezse çağrılır.

## Sözdizimi

MQCB (*Hconn, Operation, CallbackDesc, Hobj, MsgDesc, GetMsgOpts, CompCode, Reason*)

## Parametreler

### Hconn

Tip: MQHCONN-input

Bu tanıtıcı, kuyruk yöneticisiyle olan bağlantıyı gösterir. The value of *Hconn* was returned by a previous MQCONN or MQCONNX call.

On z/OS for CICS applications you can specify the following special value for *MQHC\_DEF\_HCONN* to use the connection handle associated with this execution unit.

### İşlem

Tip: MQHOME-girişi

Belirtilen nesne tanıtıcısı için tanımlanan geri bildirme işleminde işlem yapılıyor. Aşağıdaki seçeneklerden birini belirlemelisiniz. birden çok seçenek belirtmek için, değerleri bir araya ekleyin (aynı değışımezi bir kereden fazla eklemeyin) ya da bit ya da işlemi kullanarak değerleri birleştirin (programlama dili, bit işlemleri destekliyse).

### MQOP\_KAYDOL

Belirtilen nesne tanıtıcısı için geri bildirme işlevini tanımlayın. Bu işlem, çağrılacak işlevi ve kullanılacak seçim ölçütlerinin tanımlarını tanımlar.

Nesne tanıtıcısı için önceden tanımlanmış bir geri bildirme işlevi tanımlanırsa, tanım değıştirilir. Geri çağırma değıştirilirken bir hata saptanırsa, işlev kayıttan kaldırılır.

Bir geri çağrı, daha önce kaydı kaldırıldığı aynı geri çağırma işlevinde kaydedilirse, bu bir değıştirme işlemi olarak işlem görür; ilk ya da son çağrılar çağrılmaz.

MQOP\_REGISTER ile MQOP\_SUSPEND ya da MQOP\_RESUME ile birlikte MQOP\_REGISTER kullanabilirsiniz.



## **MQOP\_DEREGISTER**

Nesne tanıtıcısı için ileti tüketmeyi durdurun ve geri bildirme için uygun olan tanıtıcısı kaldırır.

İlişkili tanıtıcı kapatılırsa, geri çağırma otomatik olarak kayıttan kaldırılır.

Bir tüketici içinden MQOP\_DEREGISTER çağrılırsa ve geri çağırmaya bir durdurma çağrısı tanımlandıysa, bu çağrı tüketiciden geri döndükten sonra çağrılır.

Bu işlem kayıtlı bir tüketicisi olmayan bir *Hobj* için verilirse, çağrı MQRC\_CALLBACK\_NOT\_REGISTERED ile döndürülür.

## **MQOP\_SUSPEND**

Nesne tanıtıcısı için iletilerin tüketilmesinin askıya alınması.

Bu işlem bir olay işleyicisine uygulanırsa, olay işleyici askıya alındığında olay almaz ve askıya alınan durumdaysa kaçırılan olaylar devam edildiğinde işleme sağlanmaz.

Askıya alma işlemi askıya alındığında, tüketici işlevi denetim tipi geri çağrılarını almaya devam eder.

## **MQOP\_RESUME**

Nesne tanıtıcısı için ileti tüketmeye devam edin.

Bu işlem bir olay işleyicisine uygulanırsa, olay işleyici askıya alındığında olay almaz ve askıya alınan durumdaysa kaçırılan olaylar devam edildiğinde işleme sağlanmaz.

## **CallbackDesc**

Tip: MQCBD-input

Bu yapı, uygulama tarafından kaydedilmekte olan geri bildirme işlevini ve kaydı kaydettirirken kullanılan seçenekleri tanımlayan bir yapıdır.

Yapıyla ilgili ayrıntılar için [MQCBD](#) başlıklı konuya bakın.

Geri çağırma tanımlayıcısı yalnızca MQOP\_REGISTER seçeneği için gereklidir; tanımlayıcı gerekli değilse, geçirilen parametre adresi boş değerli olabilir.

## **Hobj**

Tip: MQHOBJ-input

Bu tanıtıcı, bir iletinin tüketileceği nesne için kurulmuş olan erişimi gösterir. Bu, önceki bir [MQOPED](#) ya da [MQSUB](#) çağrısından (**Hobj** parametresindeki) döndürülen bir tanıtıcıdır.

Bir olay işleyici yordamı (MQCBT\_EVENT\_HANDLER) tanımlanırken *Hobj* gerekli değildir ve MQHO\_NONE olarak belirlenmelidir.

*Hobj*, bir MQOPER çağrısından döndürülürse, kuyruk aşağıdaki seçeneklerden biriyle ya da birkaçında açılmış olmalıdır:

- MQOO\_INPUT\_SHARED
- MQOO\_INPUT\_EXCLUSIVE
- MQOO\_INPUT\_AS\_Q\_DEF
- MQOO\_BROWSE

## **MsgDesc**

Tip: MQMD-input

Bu yapı, gerekli iletinin özniteliklerini ve alınan iletinin özniteliklerini açıklamadır.

**MsgDesc** parametresi, tüketicinin gerektirdiği iletilerin özniteliklerini ve ileti tüketicisine geçirilecek MQMD 'nin sürümünü tanımlar.

MQMD 'deki *MsgId*, *CorrelId*, *GroupId*, *MsgSeqNumber* ve *Offset*, **GetMsgOpts** parametresindeki seçeneklere bağlı olarak, ileti seçimi için kullanılır.

The *Encoding* and *CodedCharSetId* are used for message conversion if you specify the MQGMO\_CONVERT option.

Ayrıntılar için bkz. [MQMD](#) .

*MsgDesc* , MQOP\_REGISTER için kullanılır ve herhangi bir alan için varsayılan değer dışında bir değer istiyorsanız kullanılır. *MsgDesc* , bir olay işleyicisi için kullanılmaz.

Tanımlayıcı gerekli değilse, geçirilen parametre adresi boş değerli olabilir.

Birden çok tüketici çakışan seçicilerle aynı kuyruğa kayıt yaptıysa, her ileti için seçilen tüketici tanımsız olduğunu unutmayın.

### **GetMsgSeçenekleri**

Tip: MQGMO-input

**GetMsgOpts** parametresi, ileti tüketicisinin iletileri nasıl alacağını denetler. Bu değıştirgenin tüm seçenekleri, bir MQGET çağrısında kullanıldığında, "[MQGMO-Get-message seçenekleri](#)" sayfa 364'inde açıklandığı gibi anlamlara sahiptir.

#### **MQGMO\_SET\_SIGNAL**

Bu seçeneğe izin verilmez.

#### **MQGMO\_BROWSE\_FIRST, MQGMO\_BROWSE\_NEXT, MQGMO\_MARK\_\***

Bir göz atma tüketicisine teslim edilen iletilerin sırası, bu seçeneklerin birleşimleri tarafından belirlenir. Önemli birleşimler şunlardır:

##### **MQGMO\_BROWSE\_FIRST**

Kuyruktaki ilk ileti tüketicie sürekli olarak teslim edilir. Bu, tüketicinin yok edici olarak geri bildirimde iletiyi tüketmesi yararlı olur. Bu seçeneği dikkatli kullanın.

##### **MQGMO\_BROWSE\_NEXT**

Kuyruktaki her iletiye, yürürlükteki imleç konumundan, kuyruğun sonuna ulaşıncaya kadar her ileti verilir.

##### **MQGMO\_BROWSE\_FIRST + MQGMO\_BROWSE\_NEXT**

İmleç, kuyruğun başlangıcına sıfırlanır. Daha sonra, imleç kuyruğun sonuna ulaşıncaya kadar tüketici her iletiyi verilir.

##### **MQGMO\_BROWSE\_FIRST + MQGMO\_MARK\_\***

Kuyruğun başlangıcından başlayarak, tüketicie kuyrukta ilk işaretlenmemiş ilk ileti verilir ve bu ileti bu tüketici için işaretlenir. Bu birleşim, tüketicinin geçerli imleç noktasının arkasına eklenen yeni iletileri alabilmesini sağlar.

##### **MQGMO\_BROWSE\_NEXT + MQGMO\_MARK\_\***

İmleç konumundan başlayarak, tüketicie kuyrukta bir sonraki işaretli olmayan ileti verilir; bu ileti, bu tüketici için işaretlenir. İletiler geçerli imleç konumunun ardındaki kuyruğa eklenebildiğinden, bu bileşimi dikkatli kullanın.

##### **MQGMO\_BROWSE\_FIRST + MQGMO\_BROWSE\_NEXT + MQGMO\_MARK\_\***

Bu birleşmeye izin verilmez. Kullanılırsa, çağrı MQRC\_OPTIONS\_ERROR değerini döndürür.

#### **MQGMO\_NO\_WAIT, MQGMO\_WAIT ve WaitInterval**

Bu seçenekler, tüketicinin nasıl çağrılacağını denetler.

##### **MQGMO\_NO\_BEKLEME**

Tüketici, MQRC\_NO\_MSG\_AVAILABLE ile hiç çağrılmaz. Tüketici yalnızca iletiler ve olaylar için çağrılır.

##### **Bir sıfır WaitInterval olan MQGMO\_WAN**

Kullanılabilir bir ileti olmadığında ve tüketici başlatılmış ya da tüketici son "ileti yok" neden kodundan sonra en az bir ileti teslim edildiğinde, MQRC\_NO\_MSG\_AVAILABLE kodu tüketicie iletilir.

Bu, sıfır bekleme aralığı belirtildiğinde, tüketicinin meşgul bir döngüde yaklaşmasını önler.

##### **MQGMO\_WAIT ve artı WaitInterval**

Tüketici, belirtilen bekleme aralığından sonra çağrılır; neden kodu MQRC\_NO\_MSG\_AVAILABLE. Bu arama, herhangi bir iletinin tüketicie teslim edilip edilmediği dikkate alınmaksızın yapılır. Bu, kullanıcının sağlıklı işletim bildirimini ya da toplu iş tipi işleme gerçekleştirmesini sağlar.

## **MQGMO\_WAIT ve WaitInterval /MQWI\_UNSNISININ**

Bu, MQRC\_NO\_MSG\_AVAILABLE döndürmeden önce sonsuz beklenti belirtir. Tüketici, MQRC\_NO\_MSG\_AVAILABLE ile hiç çağrılmaz.

*GetMsgOpts* yalnızca MQOP\_REGISTER için kullanılır ve herhangi bir alan için varsayılan değer dışında değer gerekiyorsa kullanılır. *GetMsgOpts* , bir olay işleyicisi için kullanılmaz.

*GetMsgOpts* gerekmiyorsa, geçirilen parametre adresi boş değerli olabilir. Bu değıştirgenin kullanılması, MQGMO\_DEFAULT ile birlikte MQGMO\_FAIL\_IF\_QUIESCING ile birlikte DEFAULT belirtimiyle aynıdır.

MQGMO yapısında bir ileti özellikleri tanıtıcısı sağlandıysa, tüketici geri bildiriminde geçirilen MQGMO yapısında bir kopya sağlanır. MQCB çağrısından geri dönerek, uygulama ileti özellikleri tanıtıcısını silebilir.

### **CompCode**

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Tamamlanma kodu; bu kod aşağıdakilerden biridir:

#### **MQCC\_OK**

İşlem başarıyla tamamlandı.

#### **MQCC\_UYARI**

Uyarı (kısmi tamamlama).

#### **MQCC\_FAILED**

Arama başarısız oldu.

### **Neden**

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Aşağıdaki listede yer alan neden kodları, kuyruk yöneticisinin **Reason** parametresi için döndürülebileceği listelerdir.

*CompCode* MQCC\_OK ise:

#### **MQRC\_NONE**

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_FAILED ise:

#### **MQRC\_ADAPTER\_NOT\_AVAM**

(2204, X'89C') Bağdaştırıcı kullanılmıyor.

#### **MQRC\_ADAPTER\_CONV\_LOAD\_ERROR**

(2133, X'855 ') Veri dönüştürme hizmetleri modüllerinin yüklenmesi gerçekleştirilemiyor.

#### **MQRC\_ADAPTER\_SERV\_LOAD\_ERROR**

(2130, X'852 ') Bağdaştırıcı hizmeti modülü yüklenemiyor.

#### **MQRC\_API\_EXIT\_ERROR**

(2374, X' 946 ') API çıkışı başarısız oldu.

#### **MQRC\_API\_EXIT\_LOAD\_ERROR**

(2183, X'887 ') API çıkışı yüklenemiyor.

#### **MQRC\_ASID\_MISSMACH**

(2157, X'86D') Birincil ve ana sunucu ASID değerleri farklı.

#### **MQRC\_BUFFER\_LENGTH\_ERROR**

(2005, X'7D5') Arabellek uzunluğu parametresi geçerli değil.

#### **MQRC\_CALL\_IN\_PROGRESS**

(2219, X'8AB') Önceki arama tamamlanmadan önce girilen MQI çağrısı.

#### **MQRC\_CALLBACK\_LINK\_ERROR**

(2487, X'9B7') Geri çağrı tipi alanı yanlış.

**MQR\_CALLBACK\_NOT\_REGISTERED**

(2448, X'990') Kayıtlı bir geri çağırım olmadığı için kaydı kaldırılamıyor, askıya alınamıyor ya da devam ettirilemiyor.

**MQR\_CALLBACK\_ROUTINE\_ERROR**

(2486, X'9B6') *CallbackFunction* ya da *CallbackName* belirtilmeli, ancak her ikisi belirtilmemelidir.

**MQR\_CALLBACK\_TYPE\_ERROR**

(2483, X'9B3') Yanlış geri çağırma tipi alanı.

**MQR\_CBD\_OPTIONS\_ERROR**

(2484, X'9B4') MQR\_CBD seçenekleri alanı yanlış.

**MQR\_CICS\_WAIT\_FAILED**

(2140, X'85C') Bekleme isteği CICS tarafından reddedildi.

**MQR\_CONNECTION\_BROKEN**

(2009, X'7D9') Kuyruk yöneticisine yönelik bağlantı kaybedildi.

**MQR\_CONNECTION\_NOT\_YETKILI**

(2217, X'8A9') Bağlantı için yetkili değil.

**MQR\_CONNECTION QUIESCING**

(2202, X'89A') Bağlantı susturulmuş durumda.

**MQR\_CONNECTION\_DURATION**

(2203, X'89B') Bağlantı sona erdiriliyor.

**MQR\_CORREL\_ID\_ERROR**

(2207, X'89F') İtilendirme tanıtıcısı hatası.

**MQR\_DATA\_LENGTH\_ERROR**

(2010, X'7DA') Veri uzunluğu parametresi geçerli değil.

**MQR\_ENVIRONMENT\_ERROR**

(2012, X'7DC') Ortamda arama geçerli değil.

**MQR\_FUNCTION\_NOT\_SUPPORTED**

(2298, X'8FA') İstenen işlev geçerli ortamda yok.

**MQR\_GET\_INHIBITED**

(2016, X'7E0') Kuyruğun engellenmesini sağlar.

**MQR\_GLOBAL\_UOW\_CONFLICTIONS**

(2351, X'92F') Genel iş çakışmaları birimleri.

**MQR\_GMO\_ERROR**

(2186, X'88A') Al-ileti seçenekleri yapısı geçerli değil.

**MQR\_HANDLE\_IN\_USE\_FOR\_UOW**

(2353, X'931') Genel iş birimi için kullanılan tanıtıcı.

**MQR\_HCONN\_ERROR**

(2018, X'7E2') Bağlantı tanıtıcısı geçerli değil.

**MQR\_HOBJ\_ERROR**

(2019, X'7E3') Nesne tanıtıcısı geçerli değil.

**MQR\_INCONTINUED\_BROWSE**

(2259, X'8D3') Tutarsız göz atma belirtimi.

**MQR\_INCONSISTENT\_UOW**

(2244, X'8C5') Tutarsız iş birliği-iş belirtimi.

**MQR\_INVALID\_MSG\_UNDER\_CURSOR**

(2246, X'8C6') İmleç altındaki ileti alma için geçerli değil.

**MQR\_LOCAL\_UOW\_CONFLICTID**

(2352, X'930') Genel iş birimi yerel iş birimi ile çakışıyor.

**MQR\_MATCH\_OPTIONS\_ERROR**

(2247, X'8C7') Eşleştirme seçenekleri geçerli değil.

**MQRC\_MAX\_MSG\_LENGTH\_ERROR**

(2485, X'9B4') Yanlıř *MaxMsgLength* alanı.

**MQRC\_MD\_ERROR**

(2026, X'7EA') İleti tanımlayıcısı geçerli deęil.

**MQRC\_MODULE\_ENTRY\_NOT\_FOUND**

(2497, X'9C1') Belirtilen iřlev giriř noktası modülde bulunamadı.

**MQRC\_MODULE\_INVALID**

(2496, X'9C0') Modül bulundu, ancak yanlıř tipte; 32 bit, 64 bit deęil ya da geçerli bir dinamik baęlantı kitaplıęı.

**MQRC\_MODULE\_NOT\_FOUND**

(2495, X'9BF') Modül, arama yolunda bulunamadı ya da yükleme yetkisine sahip deęil.

**MQRC\_MSG\_SEQ\_NUMBER\_ERROR**

(2250, X'8CA') İleti sıra numarası geçerli deęil.

**MQRC\_MSG\_TOKEN\_HATASI**

(2331, X'91B') İleti simgesinin kullanımı geçerli deęil.

**MQRC\_NO\_MSG\_AVAILABLE**

(2033, X'7F1') İleti yok.

**MQRC\_NO\_MSG\_UNDER\_CURSOR**

(2034, X'7F2') Browse imleci iletide konumlandırılmamıř.

**MQRC\_NOT\_OPEN\_FOR\_BROWSE**

(2036, X'7F4') Kuyruk göz atma için açık deęil.

**MQRC\_NOT\_OPEN\_FOR\_INPUT**

(2037, X'7F5') Kuyruk giriř için açık deęil.

**MQRC\_OBJECT\_CHANGED**

(2041, X'7F9') Nesne tanımlaması açıldıęından beri deęiřtirildi.

**MQRC\_OBJECT\_ZARAR**

(2101, X'835 ') Nesne zarar gördü.

**MQRC\_OPERATION\_HATASI**

(2206, X'89E') API Çaęırısında yanlıř iřlem kodu.

**MQRC\_OPTIONS\_ERROR**

(2046, X'7FE') Seçenekler geçerli deęil ya da tutarlı deęil.

**MQRC\_PAGEES\_HATASI**

(2193, X'891 ') Sayfa kümesi veri kümesine eriřilirken hata oluřtu.

**MQRC\_Q\_DELETED**

(2052, X'804 ') Kuyruk silinmiřtir.

**MQRC\_Q\_INDEX\_TYPE\_ERROR**

(2394, X'95A') Kuyruk yanlıř dizin tipi ięeriyor.

**MQRC\_Q\_MGR\_NAME\_ERROR**

(2058, X'80A') Kuyruk yöneticisi adı geçerli deęil ya da bilinmiyor.

**MQRC\_Q\_MGR\_NOT\_VAR**

(2059, X'80B') Kuyruk yöneticisi baęlantı için kullanılabilir deęil.

**MQRC\_Q\_MGR\_QUIESCING**

(2161, X'871 ') Kuyruk yöneticisi susturuyor.

**MQRC\_Q\_MGR\_STOPPING**

(2162, X'872 ') Kuyruk yöneticisi sona erdiriliyor.

**MQRC\_RESOURCE\_SORUNU**

(2102, X'836 ') Yetersiz sistem kaynakları var.

**MQRC\_SIGNAL\_INSTANT**

(2069, X'815 ') Bu tutamaç için bekleyen sinyal.

**MQRC\_STORAGE\_NOT\_AVAM**

(2071, X'817 ') Kullanılabilir bellek yetersiz.

**MQRC\_SUPPRESSED\_BY\_EXIT**

(2109, X'83D') Çıkış programı tarafından çağrı engellendi.

**MQRC\_SYNCPOINT\_LIMIT\_UVARD**

(2024, X'7E8') Yürürlükteki iş birimi içinde başka ileti işlenemez.

**MQRC\_SYNCPOINT\_NOT\_AVAM**

(2072, X'818 ') Eşitleme noktası desteği yok.

**MQRC\_UNEXPECTED\_ERROR**

(2195, X'893 ') Beklenmeyen bir hata oluştu.

**MQRC\_UOWENLISTMENT\_ERROR**

(2354, X' 932 ') Genel iş birimi içindeki bilgiler başarısız oldu.

**MQRC\_UOW\_MIX\_NOT\_SUPPORTANT**

(2355, X' 933 ') İşlerin birim içi çağrıları karışımı desteklenmiyor.

**MQRC\_UOW\_NOT\_VAR**

(2255, X'8CF') Kuyruk yöneticisinin kullanması için kullanılacak iş birimi.

**MQRC\_WAIT\_INTERVAL\_ERROR**

(2090, X'82A') MQGMO' da bekleme aralığı geçerli değil.

**MQRC\_WRONG\_GMO\_SÜRÜMÜ**

(2256, X'8D0') MQGMO' nun yanlış sürümü sağlandı.

**MQRC\_WRONG\_MD\_VERSION**

(2257, X'8D1') MQMD' nin yanlış sürümü sağlandı.

Bu kodlarla ilgili ayrıntılı bilgi için [İletiler ve neden kodları](#) başlıklı konuya bakın.

**Kullanım notları**

1. MQCB, kuyruğun kullanılabilir olduğu belirtilen ölçütlerle eşleşen her ileti için çağrılacak işlemi tanımlamak için kullanılır. İşlem işlendiğinde, ileti kuyruktan kaldırılır ve tanımlı ileti tüketicisine geçirilir ya da iletiyi almak için kullanılan bir ileti simgesi sağlanır.
2. MQCB, MQCTL ile tüketime başlamadan önce geri çağırma yordamlarını tanımlamak için kullanılabilir ya da bir geri çağrı yordamından kullanılabilir.
3. MQCB ' yi bir geri bildirme yordamından uzak kullanmak için önce MQCTL kullanarak ileti tüketimini askıya almanız ve daha sonra tüketime devam etmeniz gerekir.
4. MQCB, IMS bağdaştırıcısı içinde desteklenmiyor.

**İleti tüketicisi geri çağırma sırası**

Tüketiciyi, tüketicinin yaşam çevrimi boyunca anahtar noktalarında geri çağırmaya çağırma için bir tüketici yapılandırabilirsiniz. Örneğin:

- tüketicinin ilk kayıt yaptırdığı zaman,
- bağlantı başlatıldığında,
- bağlantı durdurulduğunda ve
- Tüketici, bir MQCLOSE ile belirtik olarak ya da örtük olarak silindiğinde.

Çizelge 542. MQCTL yüklemi tanımlamaları	
Komut	Anlamı
MQCTL (BAŞLAT)	MQOP_START İşlemi kullanılarak MQCTL çağrısı
MQCTL (DURDUR)	MQOP_STOP İşlemi kullanılarak MQCTL çağrısı

<i>Çizelge 542. MQCTL yüklemi tanımlamaları (devamı var)</i>	
<b>Komut</b>	<b>Anlamı</b>
MQCTL (BEKLE)	MQOP_START_WHOP İşlemi kullanılarak MQCTL çağırısı

Bu, tüketicinin tüketiciyle ilişkili durumu korumasını sağlar. Bir uygulama tarafından bir geri çağırma istendiğinde, tüketici çağırısına ilişkin kurallar şunlardır:

#### **Kaydettir**

Her zaman geri çağırma çağırısının ilk çağırısıdır.

MQCB (REGISTER) çağırısının her zaman aynı iş parçacığıyla çağırılır.

#### **START**

Her zaman MQCTL (START) komutu ile zamanuyumlu olarak çağırılır.

- MQCTL (START) komutu döndürmeden önce, tüm START çağıruları tamamlanır.

THREAD\_AFFINITY istenirse, ileti teslimiyle aynı iş parçacığıda yer alıyor.

Örneğin, önceki bir geri çağırma MQCTL (STOP) sırasında MQCTL (STOP) gibi bir çağırma, başlatma ile çağırılan arama garanti edilmez.

#### **DUR**

Bağlantı yeniden başlatılıncaya kadar, bu çağrıdan sonra başka ileti ya da olay teslim edilmez.

Uygulama, daha önce START ya da bir ileti ya da bir olay için çağırıldıysa, STOP garantilendir.

#### **DEREGISTER**

Her zaman geri çağırma çağırısının son tipi olur.

Uygulamanızın START ve STOP geri çağırılarında iş parçacığı tabanlı kullanıma hazırlama ve temizleme gerçekleştirdiğinden emin olun. REGISTER ve DEREGISTER geri çağırılarıyla iş parçacığı tabanlı olmayan kullanıma hazırlama ve temizleme işlemi yapabilirsiniz.

Belirtilenler dışında iş parçacığının yaşam ve kullanılabilirliği hakkında herhangi bir varsayımda bulunmaz. Örneğin, DEREGISTER ' a yapılan son çağrıdan sonra canlı olarak kalan bir iş parçacığına güvenmeyin. Benzer şekilde, THREAD\_AFFINITY seçeneğini kullanmamayı seçtiyseniz, bağlantı her başlatıldığında iş parçacığında var olduğunu varsaymayın.

Uygulamanızın iş parçacığı özellikleri için belirli gereksinimleri varsa, her zaman bir iş parçacığı buna uygun olarak yaratılabilir, sonra MQCTL (WAIT) seçeneğini kullanabilirsiniz. Bu, zamanuyumsuz ileti teslimi için iş parçacığının IBM MQ 'e ilişkin' donaleme ' etkisine sahiptir.

### **İleti tüketici bağlantısı kullanımı**

Tüketicuyu, tüketicinin yaşam çevrimi boyunca anahtar noktalarında geri çağırma çağırma için bir tüketici yapılandırabilirsiniz. Örneğin:

- tüketicinin ilk kayıt yaptırdığı zaman,
- bağlantı başlatıldığında,
- bağlantı durdurulduğunda ve
- Tüketici, bir MQCLOSE ile belirttik olarak ya da örtük olarak silindiğinde.

<i>Çizelge 543. MQCTL yüklemi tanımlamaları</i>	
<b>Komut</b>	<b>Anlamı</b>
MQCTL (BAŞLAT)	MQOP_START İşlemi kullanılarak MQCTL çağırısı
MQCTL (DURDUR)	MQOP_STOP İşlemi kullanılarak MQCTL çağırısı
MQCTL (BEKLE)	MQOP_START_WHOP İşlemi kullanılarak MQCTL çağırısı

Bu, tüketicinin tüketiciyle ilişkili durumu korumasını sağlar. Bir uygulama tarafından bir geri çağırma istendiğinde, tüketici çağırısına ilişkin kurallar şunlardır:

### **Kaydettir**

Her zaman geri çağırma çağırısının ilk çağırısıdır.

MQCB (REGISTER) çağırısının her zaman aynı iş parçacığıyla çağrılır.

### **START**

Her zaman MQCTL (START) komutu ile zamanuyumlu olarak çağrılır.

- MQCTL (START) komutu döndürmeden önce, tüm START çağrıları tamamlanır.

THREAD\_AFFINITY istenirse, ileti teslimiyle aynı iş parçacığıda yer alıyor.

Örneğin, önceki bir geri çağrı MQCTL (STOP) sırasında MQCTL (STOP) gibi bir çağrıysa, başlatma ile çağrılan arama garanti edilmez.

### **DUR**

Bağlantı yeniden başlatılıncaya kadar, bu çağrıdan sonra başka ileti ya da olay teslim edilmez.

Uygulama, daha önce START ya da bir ileti ya da bir olay için çağrıldıysa, STOP garantilendir.

### **DEREGISTER**

Her zaman geri çağırma çağırısının son tipi olur.

Uygulamanızın START ve STOP geri çağrılarında iş parçacığı tabanlı kullanıma hazırlama ve temizleme gerçekleştirdiğinden emin olun. REGISTER ve DEREGISTER geri çağrılarıyla iş parçacığı tabanlı olmayan kullanıma hazırlama ve temizleme işlemi yapabilirsiniz.

Belirtilenler dışında iş parçacığının yaşam ve kullanılabilirliği hakkında herhangi bir varsayımda bulunmaz. Örneğin, DEREGISTER ' a yapılan son çağrıdan sonra canlı olarak kalan bir iş parçacığına güvenmeyin. Benzer şekilde, THREAD\_AFFINITY seçeneğini kullanmamayı seçtiyseniz, bağlantı her başlatıldığında iş parçacığında var olduğunu varsaymayın.

Uygulamanızın iş parçacığı özellikleri için belirli gereksinimleri varsa, her zaman bir iş parçacığı buna uygun olarak yaratılabilir, sonra MQCTL (WAIT) seçeneğini kullanabilirsiniz. Bu, zamanuyumsuz ileti teslimi için iş parçacığının IBM MQ 'e ilişkin' donaleme ' etkisine sahiptir.

## **C çağırma**

```
MQCB (Hconn, Operation, CallbackDesc, Hobj, MsgDesc,  
GetMsgOpts, &CompCode, &Reason);
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
MQHCONN  Hconn;          /* Connection handle */  
MQLONG   Operation;     /* Operation being processed */  
MQCBD    CallbackDesc;  /* Callback descriptor */  
MQHOBJ   HObj           /* Object handle */  
MQMD     MsgDesc        /* Message descriptor attributes */  
MQGMO    GetMsgOpts     /* Message options */  
MQLONG   CompCode;      /* Completion code */  
MQLONG   Reason;        /* Reason code qualifying CompCode */
```

## **Cobol çağırma**

```
CALL 'MQCB' USING HCONN, OPERATION, CBDESC, HOBJ, MSGDESC,  
GETMSGOPTS, COMPCODE, REASON.
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
** Connection handle  
01 HCONN PIC S9(9) BINARY.  
** Operation
```



```

01 OPERATION PIC S9(9) BINARY.
** Callback Descriptior
01 CBDESC.
   COPY CMQCBDV.
01 HOBJ      PIC S9(9) BINARY.
** Message Descriptior
01 MSGDESC.
   COPY CMQMDV.
** Get Message Options
01 GETMSGOPTS.
   COPY CMQGMV.
** Completion code
01 COMPCODE PIC S9(9) BINARY.
** Reason code qualifying COMPCODE
01 REASON   PIC S9(9) BINARY.

```

## PL/I çağırımı

```

call MQCB(Hconn, Operation, CallbackDesc, Hobj, MsgDesc, GetMsgOpts,
          CompCode, Reason)

```

Değiştirgeleri aşağıdaki gibi bildirin:

```

dcl Hconn      fixed bin(31); /* Connection handle */
dcl Operation  fixed bin(31); /* Operation */
dcl CallbackDesc like MQCBD; /* Callback Descriptor */
dcl Hobj       fixed bin(31); /* Object Handle */
dcl MsgDesc    like MQMD; /* Message Descriptor */
dcl GetMsgOpts like MQGMO; /* Get Message Options */
dcl CompCode   fixed bin(31); /* Completion code */
dcl Reason     fixed bin(31); /* Reason code qualifying CompCode */

```

## MQCB\_FUNCTION-Callback işlevi

MQCB\_FUNCTION işlev çağırısı, olay işleme ve zamanuyumsuz ileti tüketimine ilişkin geri bildirme işlevidir.

MQCB\_FUNCTION çağrı tanımlaması yalnızca, geri çağırma işlevine geçirilen değiştirgeleri tanımlamak için sağlanır. Kuyruk yöneticisi tarafından MQCB\_FUNCTION adlı bir giriş noktası sağlanmadı.

Çağrılacak gerçek işlevin belirtimi, [MQCB](#) çağırısına giriş olup [MQCBD](#) yapısından geçirilir.

## Sözdizimi

MQCB\_FUNCTION (*Hconn*, *MsgDesc*, *GetMsgOpts*, *Buffer*, *Context*)

## Parametreler

### Hconn

Tip: MQHCONN-input

Bu tanıtıcı, kuyruk yöneticisiyle olan bağlantıyı gösterir. The value of *Hconn* was returned by a previous MQCONN or MQCONNX call. On z/OS for CICS applications the MQCONN call can be omitted, and the following value specified for Hconn:

### MQHC\_DEF\_CONN

Varsayılan bağlantı tanıtıcısı.

### MsgDesc

Tip: MQMD-input

Bu yapı, alınan iletinin özniteliklerini tanımlar.

Ayrıntılar için bkz. [“MQMD-İleti tanımlayıcı” sayfa 416.](#)

İletilen MQMD sürümü, tüketici işlevini tanımlayan MQCB çağırısına iletiildiği sürümle aynı.

Sürüm 4 MQGMO, MQMD yerine bir ileti tanıtıcısı döndürülmesini istemek için kullanıldıysa, MQMD 'nin adresi boş değerli karakter olarak geçirilir.

Bu, ileti tüketicisi işlevine ilişkin bir giriş alanıdır; bir olay işleyici işleviyle ilgili değildir.

### **GetMsgSeçenekleri**

Tip: MQGMO-input

İleti tüketicisinin eylemlerini denetlemek için kullanılan seçenekler. Bu parametre, döndürülen iletiye ilişkin ek bilgi de içerir.

Ayrıntılar için bkz. [MQGMO](#).

İletilen MQGMO sürümü, desteklenen en son sürümdür.

Bu, ileti tüketicisi işlevine ilişkin bir giriş alanıdır; bir olay işleyici işleviyle ilgili değildir.

### **Arabellek**

Tip: MQBYTExBufferUzunluğu-giriş

Bu alan, ileti verilerini içeren alandır.

Bu çağrıya ilişkin herhangi bir ileti yoksa ya da ileti ileti verisi içermiyorsa, *Buffer* adresinin adresi boş değer olarak geçirilir.

Bu, ileti tüketicisi işlevine ilişkin bir giriş alanıdır; bir olay işleyici işleviyle ilgili değildir.

### **Bağlam**

Tip: MQCBC-giriş/çıkış

Bu yapı, geri bildirme işlevlerine bağlam bilgileri sağlar. Ayrıntılar için bkz. "[MQCBC-Geri çağırma bağlamı](#)" sayfa 279.

## **Kullanım notları**

1. Geri çağırım yordamlarınız iş parçacığının geciktirilebilecek ya da engellenecek hizmetleri kullanırsa, örneğin, MQGET ile bekleme süresi, diğer geri çağırılarının dağıtımınızı geciktirebilir.
2. Bir geri çağrı yordamının her çağrılması için ayrı bir iş birimi otomatik olarak oluşturulmaz; bu nedenle, yordamlar bir mantıksal iş kümesi işleninceye kadar bir kesinleştirme çağrısı yayınlatabilir ya da kesinleştirmeyi erteleyebilir. İş kümesi kesinleştirildiğinde, son eşitleme noktasından bu yana çağrılan tüm geri bildirme işlevlerine ilişkin iletileri kesinleştirir.
3. CICS LINK ya da CICS START tarafından çağrılan programlar, kanal kapsayıcıları olarak bilinen adlandırılmış nesnelere aracılığıyla CICS hizmetlerini kullanarak parametreleri alır. Kap adları, değiştirge adlarıyla aynıdır. Daha fazla bilgi için CICS belgelerinize bakın.
4. Geri bildirme yordamları bir MQDISC çağrısı yayınlatabilir, ancak kendi bağlantıları için değil. Örneğin, bir geri çağrı yordamı bir bağlantı yarattıysa, bağlantının bağlantısını da kesebilir.
5. Genel olarak, bir geri çağırma yordamı, her seferinde aynı iş parçacığından çağrılmaya güvenilmemelidir. Gerekliyse, bağlantı başlatıldığında MQCTLO\_THREAD\_AFFINITY seçeneğini kullanın.
6. Bir geri bildirme yordamı sıfır olmayan bir neden kodu aldığı anda, uygun işlemi yapması gerekir.
7. MQCB\_FUNCTION, IMS bağdaştırıcısı içinde desteklenmiyor.

## **MQCLOSE-Nesneyi kapat**

MQCLOSE çağrısı bir nesneye erişimi durdurarak MQOPEN ve MQSUB çağrılarının tersini oluşturur.

### **Sözdizimi**

MQCLOSE (*Hconn, Hobj, Seçenekler, CompCode, Neden*)

## Parametreler

### Hconn.

Tip: MQHCONN-giriş

Bu tanıttıcı, kuyruk yöneticisine yönelik bağlantıyı gösterir. *Hconn* değeri önceki bir MQCONN ya da MQCONNX çağrısı tarafından döndürüldü.

z/OS for CICS uygulamalarında MQCONN çağrılarını atlayabilir ve *Hconn* için aşağıdaki değeri belirtebilirsiniz:

#### MQHC\_DEF\_HCONN

Varsayılan bağlantı tanıttıcısı.

### Hobj.

Tip: MQHOBJ-giriş/çıkış

Bu tanıttıcı, kapatılan nesneyi gösterir. Nesne herhangi bir tipte olabilir. *Hobj* değeri önceki bir MQOPEN çağrısı tarafından döndürüldü.

Çağrı başarıyla tamamlandığında, kuyruk yöneticisi bu parametreyi ortam için geçerli bir tanıttıcı olmayan bir değere ayarlar. Bu değer:

#### MQHO\_UNUSABLE\_HOBJ

Kullanılmayan nesne tanıttıcısı.

z/OS' da *Hobj* , tanımlanmamış bir değere ayarlanır.

### Seçenekler

Tip: MQLONG-input

Bu parametre, nesnenin nasıl kapatılacağını denetler.

Yalnızca kalıcı dinamik kuyruklar ve abonelikler birden çok şekilde kapatılabilir; bunların alıkonması ya da silinmesi gerekir; bunlar, MQQDT\_PERMANENT\_DYNAMIC değerine sahip **DefinitionType** özniteliğine sahip kuyruklardır ( "Kuyruklara ilişkin öznitelikler" sayfa 821 içinde açıklanan **DefinitionType** özniteliğine bakın). Kapatma seçenekleri bu konuda özetlenir.

Sürekli abonelikler alıkonabilir ya da kaldırılabilir; bunlar MQSO\_DURABLE seçeneğiyle MQSUB çağrısı kullanılarak yaratılır.

Tanıttıcıyı yönetilen bir hedefe (MQSO\_MANAGED seçeneğini kullanan bir MQSUB çağrısında döndürülen **Hobj** değiştirgesi) kapatırken, kuyruk yöneticisi ilişkili abonelik de kaldırdığında alınmamış yayınları temizler. Abonelik, bir MQSUB çağrısında döndürülen **Hsub** değiştirgesindeki MQCO\_REMOVE\_SUB seçeneği kullanılarak kaldırılır. MQCO\_REMOVE\_SUB öğesinin kalıcı olmayan bir abonelik için MQCLOSE üzerinde varsayılan davranış olduğunu unutmayın.

Bir tutamacı yönetilmeyen bir hedefe kapatırken, yayınların gönderildiği kuyruğu temizlemek sizin sorumluluğunuzdadır. Önce MQCO\_REMOVE\_SUB kullanarak aboneliği kapatın ve sonra ileti kalmayınca kadar kuyruktaki iletileri işleyin.

Aşağıdaki seçeneklerden yalnızca birini belirtmeniz gerekir:

**Dinamik kuyruk seçenekleri:** Bu seçenekler, kalıcı dinamik kuyrukların nasıl kapatılacağını denetler.

#### MQCO\_DELETE

Aşağıdakilerden biri doğruysa, kuyruk silinir:

- Bu, önceki bir MQOPEN çağrısı tarafından yaratılan kalıcı bir dinamik kuyruktur ve kuyruktaki ileti yoktur ve kuyruk için kesinleştirilmemiş alma ya da yerleştirme istekleri yoktur (yürürlükteki görev ya da başka bir görev için).
- *Hobj* öğesini döndüren, MQOPEN çağrısıyla yaratılan geçici dinamik kuyruktur. Bu durumda, kuyruktaki tüm iletiler temizlenir.

*Hobj* ' in bir MQSUB çağrısında döndürüldüğü vaka da içinde olmak üzere diğer tüm durumlarda, çağrı MQRC\_OPTION\_NOT\_VALID\_FOR\_TYPE neden koduyla başarısız olur ve nesne silinmez.

z/OS' da, kuyruk mantıksal olarak silinmiş dinamik bir kuyruksa ve bu, kuyruk için son tanıtıcıysa, fiziksel olarak silinir. Daha fazla ayrıntı için bkz. "Kullanım notları" sayfa 644 .

### **MQCO\_DELETE\_PURGE**

Aşağıdakilerden biri doğruysa, kuyruk silinir ve üzerindeki iletiler temizlenir:

- Bu, önceki bir MQOPEN çağrısı tarafından yaratılan kalıcı dinamik bir kuyruktur ve kuyruk için (yürürlükteki görev ya da başka bir görev için) kesinleştirilmemiş alma ya da koyma isteği yoktur.
- *Hobj* ögesini döndüren, MQOPEN çağrısıyla yaratılan geçici dinamik kuyruktur.

*Hobj* ' in bir MQSUB çağrısında döndürüldüğü vaka da içinde olmak üzere diğer tüm durumlarda, çağrı MQRC\_OPTION\_NOT\_VALID\_FOR\_TYPE neden koduyla başarısız olur ve nesne silinmez.

Nesne ya da kuyruk tipi	MQCO_NONE	MQCO_DELETE	MQCO_DELETE_PURGE
Kuyruk dışındaki bir nesne	Alıkondur	Geçerli değil	Geçerli değil
Önceden tanımlı kuyruk	Alıkondur	Geçerli değil	Geçerli değil
Kalıcı dinamik kuyruk	Alıkondur	Boşsa ve bekleyen güncelleme yoksa silindi	İletiler silindi; bekleyen güncelleme yoksa kuyruk silindi
Geçici dinamik kuyruk (kuyruk yaratan tarafından çağrı yayınlandı)	Silindiği Tarih	Silindiği Tarih	Silindiği Tarih
Geçici dinamik kuyruk (kuyruk yaratan tarafından çağrı yayınlanmadı)	Alıkondur	Geçerli değil	Geçerli değil
Dağıtım listesi	Alıkondur	Geçerli değil	Geçerli değil
Yönetilen abonelik hedefi	Alıkondur	Geçerli değil	Geçerli değil
Dağıtım listesi (abonelik kaldırıldı)	İletiler silindi; kuyruk silindi	Geçerli değil	Geçerli değil

**Abonelik kapatma seçenekleri:** Bu seçenekler, tutamaç kapatıldığında sürekli aboneliklerin kaldırılıp kaldırılmayacağını ve uygulama tarafından okunmayı bekleyen yayınların temizlenip temizlenmeyeceğini denetler. Bu seçenekler yalnızca, MQSUB çağrısının **Hsub** değiştirgesinde döndürülen bir nesne tanıtıcısıyla kullanılmak üzere geçerlidir.

### **MQCO\_KEEP\_SUB**

Abonelik tanıtıcısı kapatıldı, ancak yapılan abonelik korundu. Yayınlar, abonelikte belirtilen hedefe gönderilmeye devam eder. Bu seçenek, abonelik MQSO\_DURABLE seçeneğiyle yapıldıysa geçerlidir.

Abonelik dayanıklıysa, MQCO\_KEEP\_SUB varsayılan değerdir

### **MQCO\_REMOVE\_SUB**

Abonelik kaldırılır ve abonelik tanıtıcısı kapatılır.

MQSUB çağrısının **Hobj** değiştirgesi **Hsub** değiştirgesinin kapatılmasıyla geçersiz kılınmaz ve geri kalan yayınları almak üzere MQGET ya da MQCB için kullanılmaya devam edebilir. MQSUB çağrısının **Hobj** değiştirgesi de kapatıldığında, yönetilen bir hedefse, alınmayan yayınlar kaldırılır.

Abonelik dayanıklı değilse, MQCO\_REMOVE\_SUB varsayılan değerdir.

MQCO\_REMOVE\_SUB işleminin başarıyla tamamlanması, işlemin tamamlandığı anlamına gelmez. Bu aramanın tamamlanıp tamamlanmadığını denetlemek için Dağıtılmış ağlara ilişkin zamanuyumsuz komutların tamamlanıp tamamlanmadığını denetleme başlıklı konudaki DELETE SUB adımına bakın.

Bu abonelik kapatma seçenekleri aşağıdaki tablolarda özetlenmiştir.

Çizelge 545. Sürekli abonelik tanıtıcısını kapatma, ancak aboneliği koruma seçenekleri	
Görev	Abonelik kapatma seçeneği
Yayınları MÖOPENed tutamacında tut	MÖCO_KEEP_SUB
MÖOPENed tanıtıcısındaki yayınları kaldır	İşleme izin verilmiyor
Yayınları MÖSO_MANAGED tutamacında tut	MÖCO_KEEP_SUB
MÖSO_MANAGED tanıtıcısındaki yayınları kaldır	İşleme izin verilmiyor

Sürekli bir abonelik tanıtıcısını kapatıp aboneliğini kaldırarak ya da kalıcı olmayan bir abonelik tanıtıcısını kapatarak aboneliğinizi kaldırmak için aşağıdaki abonelik kapatma seçeneklerini kullanın:

Çizelge 546. Aboneliği kaldırma seçenekleri	
Görev	Abonelik kapatma seçeneği
Yayınları MÖOPENed tutamacında tut	MÖCO_REMOVE_SUB
MÖOPENed tanıtıcısındaki yayınları kaldır	İşleme izin verilmiyor
Yayınları MÖSO_MANAGED tutamacında tut	MÖCO_REMOVE_SUB

**Önden okuma seçenekleri:** Aşağıdaki seçenekler, bir uygulama tarafından istenmeden önce istemciye gönderilen ve henüz uygulama tarafından tüketilmeyen kalıcı olmayan iletilere ne olacağını denetler. Bu iletiler, uygulama tarafından istenmeyi bekleyen istemci önden okuma arabelleğinde saklanır ve MÖCLOSE tamamlanmadan önce atılabilir ya da kuyruktan tüketilebilir.

#### MÖCO\_IMMEDIATE

Nesne hemen kapatılır ve istenmeden önce istemciye gönderilen iletiler atılır ve herhangi bir uygulama tarafından tüketilemez. Bu varsayılan değerdir.

#### MÖCO\_QUIESCE

Nesneyi kapatma isteğinde bulunursanız, ancak bir uygulama bunları istemeden önce istemciye gönderilen iletiler istemcinin önden okuma arabelleğinde bulunmaya devam ediyorsa, MÖCLOSE çağrısı MÖRC\_READ\_AHEAD\_MSGS uyarısıyla geri döner ve nesne tanıtıcısı geçerli kalır.

Daha sonra uygulama, ileti almak için nesne tanıtıcısını kullanmaya devam edebilir ve daha sonra, nesneyi yeniden kapatabilir. İstemciye, istekte bulunan bir uygulamadan önce başka ileti gönderilmez, önden okuma işlevi kapatılmış olur.

Son MÖGET çağrısı ile MÖCO\_IMMEDIATE kullanılırsa atılacak sonraki MÖCLOSE arasında bir ileti olabileceğinden, uygulamaların istemcinin önden okuma arabelleğinde başka ileti olmadığı bir noktaya ulaşmaya çalışmak yerine MÖCO\_QUIESCE kullanmaları önerilir.

Zamanuyumsuz bir geri çağırma işlevinden MÖCO\_QUIESCE içeren bir MÖCLOSE yayınlandıysa, önden okuma iletileriyle aynı davranış geçerli olur. MÖRC\_READ\_AHEAD\_MSGS uyarısı döndürülürse, geri çağırma işlevi en az bir kez daha çağrılır. Önden okunan son ileti geri çağırma işlevine geçirildiğinde, MÖCBC ConsumerFlags alanı MÖCBCF\_READA\_BUFFER\_EMPTY olarak ayarlanır.

**Varsayılan seçenek:** Önceden açıklanan seçeneklerin hiçbirine gerek duymazsanız, aşağıdaki seçeneği kullanabilirsiniz:

#### MÖCO\_NONE

İsteğe bağlı kapatma işlemi gerekmez.

Bu, aşağıdakiler için belirtilmelidir:

- Kuyruklar dışındaki nesnelere
- Önceden tanımlı kuyruklar
- Geçici dinamik kuyruklar (ancak, *Hobj* ' in kuyruğu yaratan MÖOPEN çağrısıyla döndürülen tanıtıcı olmadığı durumlarda).

- Dağıtım listeleri

Yukarıdaki tüm durumlarda, nesne korunur ve silinmez.

Geçici dinamik kuyruk için bu seçenek belirtilirse:

- Kuyruk, *Hobj* ; döndüren MQOPEN çağrısıyla yaratıldıysa silinir. kuyruktaki iletiler temizlenir.
- Diğer tüm durumlarda kuyruk (ve üzerindeki iletiler) alıkonur.

Kalıcı dinamik kuyruk için bu seçenek belirtilirse, kuyruk alıkonur ve silinmez.

z/OS' da, kuyruk mantıksal olarak silinmiş dinamik bir kuyruksa ve bu, kuyruk için son tanıtıcıysa, fiziksel olarak silinir. Daha fazla ayrıntı için bkz. [“Kullanım notları” sayfa 644](#) .

### CompCode

Tip: MQLONG-output

Tamamlanma kodu; aşağıdakilerden biridir:

#### **MQCC\_OK**

Başarıyla tamamlandı.

#### **MQCC\_UYARISI**

Uyarı (kısmi tamamlama).

#### **MQCC\_FAILED**

Arama başarısız oldu.

### Neden

Tip: MQLONG-output

Listelenen neden kodları, kuyruk yöneticisinin **Reason** parametresi için döndürebileceği kodlardır.

*CompCode* MQCC\_OK ise:

#### **MQRC\_NONE**

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_WARNING ise:

#### **MQRC\_INCOMPLETE\_GROUP**

(2241, X'8C1') İleti grubu tamamlanmadı.

#### **MQRC\_INCOMPLETE\_MSG**

(2242, X'8C2') Mantıksal ileti tamamlanmadı.

#### **MQRC\_READ\_AHEAD\_MSGS**

(nnnn, X'xxx ') İstemci, henüz uygulama tarafından kullanılmayan önden okuma iletileri aldı.

*CompCode* MQCC\_FAILED ise:

#### **MQRC\_ADAPTER\_NOT\_KULLANILAMIYOR**

(2204, X'89C') Bağdaştırıcı kullanılmıyor.

#### **MQRC\_ADAPTER\_SERV\_LOAD\_ERROR (UYGULAMA\_HATASI)**

(2130, X'852 ') Bağdaştırıcı hizmet modülü yüklenemiyor.

#### **MQRC\_API\_EXIT\_ERROR (UYGULAMA\_HATASI)**

(2374, X' 946 ') API çıkışı başarısız oldu.

#### **MQRC\_API\_EXIT\_LOAD\_ERROR (UYGULAMA\_HATASI)**

(2183, X'887 ') API çıkışı yüklenemiyor.

#### **MQRC\_ASID\_MISMATCH**

(2157, X'86D') Birincil ve ana ASID' ler farklıdır.

#### **MQRC\_CALL\_IN\_PROGRESS**

(2219, X'8AB') Önceki arama tamamlanmadan önce girilen MQI çağrısı.

#### **MQRC\_CF\_NOT\_YOK**

(2345, X' 929 ') Bağlama olanağı kullanılmıyor.

**MQRC\_CF\_STRUC\_FAILED**

(2373, X'945 ') Bağlama tesisi yapısı başarısız oldu.

**MQRC\_CF\_STRUC\_IN\_USE**

(2346, X'92A') Coupling-facility yapısı kullanımda.

**MQRC\_CICS\_WAIT\_FAILED**

(2140, X'85C') Bekleme isteği CICStarafından reddedildi.

**MQRC\_CONNECTION\_BROKEN**

(2009, X'7D9') Kuyruk yöneticisine bağlantı kaybedildi.

**MQRC\_CONNECTION\_NOT\_YETKILI**

(2217, X'8A9') Bağlantı için yetkili değil.

**MQRC\_CONNECTION\_DURDURULUYOR**

(2203, X'89B') Bağlantı kapatılıyor.

**MQRC\_DB2\_NOT\_AVAILABLE**

(2342, X'926 ') Db2 altsistemi kullanılmıyor.

**MQRC\_HCONN\_HATA**

(2018, X'7E2') Bağlantı tanıtıcısı geçersiz.

**MQRC\_HOBJ\_HATA**

(2019, X'7E3') Nesne tanıtıcısı geçerli değil.

**MQRC\_NOT\_YETKILI**

(2035, X'7F3') Erişim için yetkili değil.

**MQRC\_OBJECT\_HASARLI**

(2101, X'835 ') Nesne zarar gördü.

**MQRC\_OPTION\_NOT\_VALID\_FOR\_TYPE**

(2045, X'7FD') MQOPEN ya da MQCLOSE çağrısında: seçeneği nesne tipi için geçerli değil.

**MQRC\_OPTIONS\_HATA**

(2046, X'7FE') Seçenekler geçerli değil ya da tutarlı değil.

**MQRC\_PAGESET\_HATA**

(2193, X'891 ') Sayfa kümesi veri kümesine erişilirken hata oluştu.

**MQRC\_Q\_MGR\_AD\_HATASI**

(2058, X'80A') Kuyruk yöneticisi adı geçerli değil ya da bilinmiyor.

**MQRC\_Q\_MGR\_YOK**

(2059, X'80B') Kuyruk yöneticisi bağlantı için kullanılmıyor.

**MQRC\_Q\_MGR\_DURDURULUYOR**

(2162, X'872 ') Kuyruk yöneticisi kapatılıyor.

**MQRC\_Q\_NOT\_EMPTY**

(2055, X'807 ') Kuyruk bir ya da daha fazla ileti ya da kesinleştirilmemiş koyma ya da alma isteği içeriyor.

**MQRC\_RESOURCE\_PROBLEM**

(2102, X'836 ') Sistem kaynakları yetersiz.

**MQRC\_SECURITY\_HATA**

(2063, X'80F') Güvenlik hatası oluştu.

**MQRC\_STORAGE\_KULLANILAMIYOR**

(2071, X'817 ') Kullanılabilir bellek yetersiz.

**MQRC\_SUPPRESSED\_BY\_EXIT**

(2109, X'83D') Çağrı çıkış programı tarafından engellendi.

**MQRC\_UNEXPECTED\_HATASI**

(2195, X'893 ') Beklenmeyen hata oluştu.

Bu kodlarla ilgili ayrıntılı bilgi için bkz. [İletiler ve neden kodları](#).

## Kullanım notları

1. Bir uygulama MQDISC çağrısıyla ilgili bir sorun oluşturduğunda ya da olağan dışı ya da olağandışı bir şekilde sona erdiğinde, uygulama tarafından açılan ve hala açık olan nesnelere MQCO\_NONE seçeneğiyle otomatik olarak kapatılır.
2. Kapatılan nesne bir *kuyruk* ise aşağıdaki noktalar geçerlidir:
  - Kuyruktaki işlemler bir iş biriminin parçası olarak gerçekleştirilirse, kuyruk, eşitleme noktasının sonucunu etkilemeden, eşitleme noktasından önce ya da sonra kapatılabilir. Kuyruk tetiklenirse, kuyruğu kapatmadan önce geriye işleme gerçekleştirilmesi bir tetikleyici iletilerinin yayınlanmasına neden olabilir. Tetikleyici iletilerine ilişkin ek bilgi için [Tetikleyici iletilerinin özellikleribaşlıklı konuya](#) bakın.
  - Kuyruk MQOO\_BROWSE seçeneğiyle açıldıysa, göz atma imleci yok edilir. Kuyruk daha sonra MQOO\_BROWSE seçeneğiyle yeniden açılırsa, yeni bir göz atma imleci yaratılır (bkz. [MQOO\\_BROWSE](#) ).
  - MQCLOSE çağrısı sırasında bu tanıtıcı için bir ileti kilitliyse, kilit serbest bırakılır (bkz. [MQGMO\\_LOCK](#) ).
  - z/OS sistemlerinde, kuyruk tanıtıcısıyla ilgili olarak MQGMO\_SET\_SIGNAL seçeneği bekleyen bir MQGET isteği varsa, istek iptal edilir ( [MQGMO\\_SET\\_SIGNAL](#) konusuna bakın). Aynı kuyruğa ilişkin, ancak farklı tanıtıcılara (*Hobj*) karşı yerleştirilen sinyal istekleri etkilenmez (dinamik bir kuyruk silinmedikçe, bu durumda bunlar da iptal edilir).

3. Kapatılan nesne bir *dinamik kuyruk* (kalıcı ya da geçici) ise aşağıdaki noktalar geçerlidir:

- Devingen bir kuyruk için, ilgili MQOPEN çağrısında belirtilen seçeneklerden bağımsız olarak MQCO\_DELETE ve MQCO\_DELETE\_PURGE seçeneklerini belirtebilirsiniz.
- Dinamik bir kuyruk silindiğinde, kuyruk için bekleyen MQGMO\_WAIT seçeneğiyle tüm MQGET çağrılarını iptal edilir ve neden kodu MQRC\_Q\_DELETED döndürülür. Bkz. [MQGMO\\_WAIT](#).

Uygulamalar silinmiş bir kuyruğa erişemese de, kuyruk sistemden kaldırılmaz ve kuyruğa başvuran tüm tanıtıcıları kapatıncaya ve kuyruğu etkileyen tüm iş birimleri kesinleştirilinceye ya da yedekleninceye kadar, ilişkili kaynaklar serbest bırakılmaz.

z/OS sistemlerinde, mantıksal olarak silinmiş, ancak henüz sistemden kaldırılmamış bir kuyruk, silinen kuyrukla aynı ada sahip yeni bir kuyruk yaratılmasını önler; MQOPEN çağrısı bu durumda MQRC\_NAME\_IN\_USE neden koduyla başarısız olur. Ayrıca, bu tür bir kuyruk, uygulamalar tarafından erişilemese de MQSC komutları kullanılarak görüntülenmeye devam edebilir.

- Kalıcı bir dinamik kuyruk silindiğinde, MQCLOSE çağrısında belirtilen *Hobj* tanıtıcısı, kuyruğu yaratan MQOPEN çağrısının döndürdüğü tanıtıcı değilse, MQOPEN çağrısının geçerliliğini denetlemek için kullanılan kullanıcı kimliğinin kuyruğu silme yetkisi olduğu denetlenir. MQOPEN çağrısında MQOO\_ALTERNATE\_USER\_AUTHORITY seçeneği belirtildiyse, denetlenen kullanıcı kimliği *AlternateUserId* olur.

Bu denetim aşağıdaki durumda gerçekleştirilmez:

- Belirtilen tanıtıcı, kuyruğu yaratan MQOPEN çağrısıyla döndürülen tanıtıcı.
- Silinmekte olan kuyruk geçici bir dinamik kuyruk.
- Geçici bir dinamik kuyruk kapatıldığında, MQCLOSE çağrısında belirtilen *Hobj* tanıtıcısı, kuyruğu yaratan MQOPEN çağrısıyla döndürülen tanıtıcıysa, kuyruk silinir. Bu durum, MQCLOSE çağrısında belirtilen kapatma seçeneklerinden bağımsız olarak oluşur. Kuyrukta ileti varsa, bunlar atılır; rapor iletileri oluşturulmaz.

Kuyruğu etkileyen kesinleştirilmemiş iş birimleri varsa, kuyruk ve iletileri silinmeye devam eder, ancak iş birimleri başarısız olmaz. Ancak, daha önce açıklandığı gibi, iş birimleriyle ilişkili kaynaklar, her bir iş birimi kesinleştirilinceye ya da geri çekilinceye kadar serbest bırakılmaz.

4. Kapatılan nesne bir *dağıtım listesi* ise aşağıdaki noktalar geçerlidir:



- Dağıtım listesi için geçerli olan tek kapatma seçeneği MQCO\_NONE 'dir; başka seçenekler belirtilirse, çağrı MQRC\_OPTIONS\_ERROR ya da MQRC\_OPTION\_NOT\_VALID\_FOR\_TYPE neden koduyla başarısız olur.
- Bir dağıtım listesi kapatıldığında, listedeki kuyruklar için tek tek tamamlama kodları ve neden kodları döndürülmez; tanılama amacıyla yalnızca çağrıya ilişkin **CompCode** ve **Reason** parametreleri kullanılabilir.

Kuyruklardan biri kapatılırken bir hata oluşursa, kuyruk yöneticisi işlemeye devam eder ve dağıtım listesinde kalan kuyrukları kapatmayı dener. Çağrıya ilişkin **CompCode** ve **Reason** parametreleri, hatayı açıklayan bilgileri döndürecek şekilde ayarlanır. Kuyrukların çoğu başarıyla kapatılmış olsa da, tamamlama kodunun MQCC\_FAILED olması mümkündür. Hatayla karşılaşan kuyruk tanımlanmadı.

Birden çok kuyruktaki bir hata varsa, **CompCode** ve **Reason** parametrelerinde hangi hatanın bildirildiği tanımlanmaz.

## C çağrısı

```
MQCLOSE (Hconn, &Hobj, Options, &CompCode, &Reason);
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
MQHCONN  Hconn;      /* Connection handle */
MQHOBJ   Hobj;       /* Object handle */
MQLONG   Options;    /* Options that control the action of MQCLOSE */
MQLONG   CompCode;   /* Completion code */
MQLONG   Reason;     /* Reason code qualifying CompCode */
```

## COBOL çağrısı

```
CALL 'MQCLOSE' USING HCONN, HOBJ, OPTIONS, COMPCODE, REASON.
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
** Connection handle
01 HCONN      PIC S9(9) BINARY.
** Object handle
01 HOBJ       PIC S9(9) BINARY.
** Options that control the action of MQCLOSE
01 OPTIONS    PIC S9(9) BINARY.
** Completion code
01 COMPCODE   PIC S9(9) BINARY.
** Reason code qualifying COMPCODE
01 REASON     PIC S9(9) BINARY.
```

## PL/I çağrısı

```
call MQCLOSE (Hconn, Hobj, Options, CompCode, Reason);
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
dcl Hconn      fixed bin(31); /* Connection handle */
dcl Hobj       fixed bin(31); /* Object handle */
dcl Options    fixed bin(31); /* Options that control the action of
                               MQCLOSE */
dcl CompCode   fixed bin(31); /* Completion code */
dcl Reason     fixed bin(31); /* Reason code qualifying CompCode */
```

## High Level Assembler (Yüksek Düzeyli çevirici) çağrısı

```
CALL MQCLOSE, (HCONN, HOBJ, OPTIONS, COMPCODE, REASON)
```

Değiştirgeleri aşağıdaki gibi bildirin:

HCONN	DS	F	Connection handle
HOBJ	DS	F	Object handle
OPTIONS	DS	F	Options that control the action of MQCLOSE
COMPCODE	DS	F	Completion code
REASON	DS	F	Reason code qualifying COMPCODE

## Visual Basic çağrısı

```
MQCLOSE Hconn, Hobj, Options, CompCode, Reason
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
Dim Hconn As Long 'Connection handle'  
Dim Hobj As Long 'Object handle'  
Dim Options As Long 'Options that control the action of MQCLOSE'  
Dim CompCode As Long 'Completion code'  
Dim Reason As Long 'Reason code qualifying CompCode'
```

## MQCMIT-Kesinleştirme değişiklikleri

MQCMIT çağrısı, uygulamanın kuyruk yöneticisine, uygulamanın bir eşitleme noktasına ulaştığını ve son eşitleme noktasının kalıcı kılınacağı için tüm ileti alıkonacağı ve oluştuğunu gösterir.

Bir iş biriminin bir parçası olarak sunulan iletiler, diğer uygulamalar tarafından kullanılabilir kılınsın; bir iş biriminin bir parçası olarak alınan iletiler silinir.

- **z/OS** z/OS üzerinde arama yalnızca toplu programlar (IMS toplu DL/I programları da içinde olmak üzere) tarafından kullanılır.

## Sözdizimi

MQCMIT (*Hconn*, *CompCode*, *Reason*)

## Parametreler

### Hconn

Tip: MQHCONN-input

Bu tanırtıcı, kuyruk yöneticisiyle olan bağlantıyı gösterir. The value of *Hconn* was returned by a previous MQCONN or MQCONNX call.

### CompCode

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Tamamlanma kodu; bu kod aşağıdakilerden biridir:

#### MQCC\_OK

İşlem başarıyla tamamlandı.

#### MQCC\_UYARI

Uyarı (kısmi tamamlama).

#### MQCC\_FAILED

Arama başarısız oldu.

## Neden

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Listelenen neden kodları, kuyruk yöneticisinin **Reason** parametresi için döndürülebileceği kodlardır.

*CompCode* MQCC\_OK ise:

### **MQRC\_NONE**

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_UYARI ise:

### **MQRC\_BACKED\_OUT**

(2003, X'7D3') İş birimi yedeklendi.

### **MQRC\_OUTCOME\_PENDING**

(2124, X'84C') Kesinleştirme işleminin sonucu beklemede.

*CompCode* MQCC\_FAILED ise:

### **MQRC\_ADAPTER\_SERV\_LOAD\_ERROR**

(2130, X'852 ') Bağdaştırıcı hizmeti modülü yüklenemiyor.

### **MQRC\_API\_EXIT\_ERROR**

(2374, X' 946 ') API çıkışı başarısız oldu.

### **MQRC\_ASID\_MISSMACH**

(2157, X'86D') Birincil ve ana sunucu ASID değerleri farklı.

### **MQRC\_CALL\_IN\_PROGRESS**

(2219, X'8AB') Önceki arama tamamlanmadan önce girilen MQI çağrısı.

### **MQRC\_CALL\_INTERRUPT**

(2549, X'9F5') MQPUT ya da MQCMIT kesintiye uğradı ve yeniden bağlanma işlemi kesin bir sonucu yeniden oluşturamıyor.

### **MQRC\_CF\_STRUC\_IN\_USE**

(2346, X'92A') Coupling-tesis yapısı kullanımda.

### **MQRC\_CONNECTION\_BROKEN**

(2009, X'7D9') Kuyruk yöneticisine yönelik bağlantı kaybedildi.

### **MQRC\_ENVIRONMENT\_ERROR**

(2012, X'7DC') Ortamda arama geçerli değil.

### **MQRC\_HCONN\_ERROR**

(2018, X'7E2') Bağlantı tanıtıcısı geçerli değil.

### **MQRC\_OBJECT\_ZARAR**

(2101, X'835 ') Nesne zarar gördü.

### **MQRC\_OUTCOME\_MIXED**

(2123, X'84B') Kesinleştirme ya da geri dönüş işlemi sonucu karışık.

### **MQRC\_Q\_MGR\_STOPPING**

(2162, X'872 ') Kuyruk yöneticisi sona erdiriliyor.

### **MQRC\_RECONNECT\_FAILED**

(2548, X'9F4') Yeniden bağlandıktan sonra, yeniden bağlanabilir bağlantı için tutamaçlar geri döndürülürken bir hata oluştu.

### **MQRC\_RESOURCE\_SORUNU**

(2102, X'836 ') Yetersiz sistem kaynakları var.

### **MQRC\_STORAGE\_MEDIUM\_FULL**

(2192, X'890 ') Dış depolama ortamı dolu.

### **MQRC\_STORAGE\_NOT\_AVAM**

(2071, X'817 ') Kullanılabilir bellek yetersiz.

### **MQRC\_UNEXPECTED\_ERROR**

(2195, X'893 ') Beklenmeyen bir hata oluştu.

Bu kodlarla ilgili ayrıntılı bilgi için [İletiler ve neden kodları](#) başlıklı konuya bakın.

## Kullanım notları

1. Bu çağrıyı, yalnızca kuyruk yöneticisinin iş birimini koordine ettiğinde kullanın. Bu durumda şunlar olabilir:

- Değişikliklerin yalnızca IBM MQ kaynaklarını etkilediği yerel bir iş birimi.
- Değişikliklerin, diğer kaynak yöneticilerine ait kaynakları etkileyebileceği gibi, IBM MQ kaynaklarını etkilediği genel bir iş birimi.

Yerel ve genel çalışma birimleriyle ilgili daha fazla ayrıntı için bkz. "[MQBEGIN-İş birimini başlat](#)" sayfa 621.

2. Kuyruk yöneticisinin iş birimini koordine etmediği ortamlarda, MQCMIT yerine uygun kesinleştirme çağrısı kullanılmalıdır. Ortam, uygulamanın olağan şekilde sonlandırılmasına neden olan örtük bir kesinleştirmeyi de destekler.

- z/OS' ta aşağıdaki çağrıları kullanın:
  - İş birimi yalnızca IBM MQ kaynaklarını etkilediğinde, toplu iş programları ( IMS toplu DL/I programları da içinde olmak üzere) MQCMIT çağrısını kullanabilir. Ancak, iş birimi hem IBM MQ kaynaklarını, hem de diğer kaynak yöneticilerine ait kaynakları etkiliyorsa (örneğin, Db2 ), z/OS Kurtarılabilir Kaynak Hizmeti (RRS) tarafından sağlanan SRRCMIT çağrısını kullanın. SRRCMIT çağrısı, RRS eşgüdümü için etkinleştirilen kaynak yöneticilerine ait kaynaklarda değişiklik sağlar.
  - CICS applications must use the EXEC CICS SYNCPOINT command to commit the unit of work explicitly. Diğer bir seçenek olarak, işlemin sona erdirilmesinin, iş biriminin örtük olarak kesinleştirilmesine neden olması gerekir. MQCMIT çağrısı, CICS uygulamaları için kullanılamaz.
  - IMS applications (other than batch DL/I programs) must use IMS calls such as GU and CHKP to commit the unit of work. MQCMIT çağrısı, IMS uygulamaları (toplu DL/I programlarından başka) için kullanılamaz.
- IBM i' ta, kuyruk yöneticisi tarafından koordine edilen yerel iş birimleri için bu çağrıyı kullanın. This means that a commitment definition must not exist at job level, that is, the STRCMTCTL command with the **CMTSCOPE (\*JOB)** parameter must not have been issued for the job.

3. Bir uygulama, bir iş biriminde kesinleştirilmemiş değişikliklerle sona ererse, bu değişikliklerin yok edilmesi, uygulamanın olağan ya da olağan dışı bir şekilde sona ermesine bağlıdır. Ek ayrıntılar için [MQDISC kullanım notları](#) başlıklı konuya bakın.

4. Bir uygulama, gruplara ya da mantıksal ileti bölümlerine ileti yerleştirdiğinde ya da ileti aldığı anda, kuyruk yöneticisi, son başarılı MQPUT ve MQGET çağrılarına ilişkin ileti grubuyla ve mantıksal iletiyle ilgili bilgileri saklar. Bu bilgiler kuyruk tanıtıcısı ile ilişkilendirilir ve aşağıdaki gibi şeyler içerir:

- MQMD ' de *GroupId*, *MsgSeqNumber*, *Offset* ve *MsgFlags* alanlarının değerleri.
- İletinin bir iş biriminin parçası olup olmadığı.
- MQPUT çağrısına ilişkin: İletinin kalıcı mı, yoksa kalıcı değil mi olduğu.

Bir iş birimi kesinleştirildiğinde, kuyruk yöneticisi grup ve bölüm bilgilerini saklar ve uygulama yürürlükteki ileti grubuna ya da mantıksal iletiye ileti yerleştirmeye ya da ileti almaya devam edebilir.

Bir iş birimi kesinleştirildiğinde, grubun ve kesim bilgilerinin saklanması, uygulamanın büyük bir ileti grubu ya da çok sayıda iş birimi üzerinden birçok kesimden oluşan büyük bir mantıksal ileti yayınlamasına olanak tanır. Yerel kuyruk yöneticisinin kuyruk depolaması sınırlı olması durumunda, birkaç iş birimi kullanılması avantajlıdır. Ancak, bir sistem hatası ortaya çıkarsa, uygulama, iletileri yeniden başlatmak ya da doğru noktaya ileti almak için yeterli bilgileri sağlamalıdır. Sistem hatasından sonra doğru noktada yeniden başlatılabilmeye ilişkin ayrıntılar için bakınız: [MQPMO\\_LOGICAL\\_ORDER](#) ve [MQGMO\\_LOGICAL\\_ORDER](#).

Kalan kullanım notları, yalnızca kuyruk yöneticisi iş birimlerini koordine ettiğinde geçerlidir:

5. Bir iş birimi, bağlantı tanıtıcısı ile aynı kapsamı içerir; belirli bir iş birimini etkileyen tüm IBM MQ çağrıları, aynı bağlantı tanıtıcısı kullanılarak gerçekleştirilmelidir. Farklı bir bağlantı tanıtıcısı

kullanılarak verilen çağrılar (örneğin, başka bir uygulama tarafından yayınlanan çağrılar) farklı bir iş birimini etkiler. Bağlantı tutamaçlarının kapsamı hakkında bilgi için MQCONN ' de açıklanan **Hconn** parametresine bakın.

6. Bu çağrıdan etkilenecek, yalnızca yürürlükteki iş biriminin bir parçası olarak alınan ya da alınan iletiler etkilenir.
7. Bir iş birimi içinde MQGET, MQPUT ya da MQPUT1 tarafından çağrılan, ancak bir kesinleştirme ya da geri alma çağrısını hiçbir zaman yayınlamayan, uzun süredir çalışan bir uygulama, diğer uygulamaların kullanımına sunulmayan iletilerle kuyukları doldurabilir. Buna karşı koruma sağlamak için, denetimcinin **MaxUncommittedMsgs** kuyruk yöneticisi özneliğini, kuyukları dolduran kaçak uygulamaları önleyecek kadar düşük bir değere ayarlaması gerekir, ancak beklenen ileti sistemi uygulamalarının doğru şekilde çalışmasına izin verecek kadar yüksek olmalıdır.
8. **ALW** AIX, Linux, and Windows sistemlerinde, **Reason** değıştirgesi MQRC\_CONNECTION\_BROKEN ( *CompCode* MQCC\_FAILED ile birlikte) ya da MQRC\_UNEXPECTED\_ERROR ise, iş biriminin başarıyla kesinleştirilmesinin olanaklı olduğunu.

## C çağırma

```
MQCMIT (Hconn, &CompCode, &Reason);
```

Değıştirgeleri aşağıdaki gibi bildirin:

```
MQHCONN Hconn; /* Connection handle */
MQLONG CompCode; /* Completion code */
MQLONG Reason; /* Reason code qualifying CompCode */
```

## Cobol çağırma

```
CALL 'MQCMIT' USING HCONN, COMPCODE, REASON.
```

Değıştirgeleri aşağıdaki gibi bildirin:

```
** Connection handle
01 HCONN PIC S9(9) BINARY.
** Completion code
01 COMPCODE PIC S9(9) BINARY.
** Reason code qualifying COMPCODE
01 REASON PIC S9(9) BINARY.
```

## PL/I çağırımı

```
call MQCMIT (Hconn, CompCode, Reason);
```

Değıştirgeleri aşağıdaki gibi bildirin:

```
dcl Hconn fixed bin(31); /* Connection handle */
dcl CompCode fixed bin(31); /* Completion code */
dcl Reason fixed bin(31); /* Reason code qualifying CompCode */
```

## High Level Assembler çağırısı

```
CALL MQCMIT,(HCONN,COMPCODE,REASON)
```

Değıştirgeleri aşağıdaki gibi bildirin:

```
HCONN      DS  F  Connection handle
COMPCODE   DS  F  Completion code
REASON     DS  F  Reason code qualifying COMPCODE
```

## Visual Basic çağrısı

```
MQCMIT Hconn, CompCode, Reason
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
Dim Hconn      As Long 'Connection handle'
Dim CompCode   As Long 'Completion code'
Dim Reason     As Long 'Reason code qualifying CompCode'
```

## MQCONN-Bağlantı kuyruğu yöneticisi

MQCONN çağrısı, bir uygulama programını bir kuyruk yöneticisine bağlar.

Uygulamanın sonraki ileti kuyruklama çağrılarında kullandığı bir kuyruk yöneticisi bağlantı tanıtıcısı sağlar.

- z/OS' da CICS uygulamalarının bu çağrıyı yapması gerekmez. Bu uygulamalar, CICS sisteminin bağlı olduğu kuyruk yöneticisine otomatik olarak bağlanır. Ancak, MQCONN ve MQDISC çağrıları CICS uygulamalarından kabul edilir.
- IBM üzerinde, uygulamalar kuyruk yöneticisine bağlanmak için MQCONN ya da MQCONNX çağrıyı ve kuyruk yöneticisiyle bağlantıyı kesmek için MQDISC çağrıyı kullanmalıdır.

Yalnızca sunucu kuruluşunda istemci bağlantısı kurulamaz ve yalnızca istemci kuruluşunda yerel bağlantı kurulamaz.

## Sözdizimi

MQCONN (*QMgrName*, *Hconn*, *CompCode*, *Reason*)

## Parametreler

### QMgrName

Tip: MQCHAR48 -giriş

Bu, uygulamanın bağlanmak istediği kuyruk yöneticisinin adıdır. Ad aşağıdaki karakterleri içerebilir:

- Büyük harf alfabetik karakterler (A-Z)
- Küçük harf alfabetik karakterler (a-z)
- Sayısal basamaklar (0-9)
- Nokta (.), eğik çizgi (/), altçizgi (\_), yüzde (%)

Ad başta ya da gömülü boşluklar içermemelidir, ancak sonda boşluklar içerebilir. Addaki önemli verilerin sonunu göstermek için boş karakter kullanılabilir; boş değer ve ardından gelen karakterler boşluk olarak değerlendirilir. Belirtilen ortamlarda aşağıdaki kısıtlamalar geçerlidir:

- EBCDIC Katakana kullanan sistemlerde küçük harfli karakterler kullanılamaz.
- z/OS üzerinde, alt çizgiyle başlayan ya da biten adlar, işlemler ve denetim panoları tarafından işlenemez. Bu nedenle, bu tür isimlerden kaçınınız.
- IBM üzerinde, komutlarda belirtildiğinde küçük harf, eğik çizgi ya da yüzde işareti içeren adları tırnak içine alın. **QMgrName** değiştirgesinde bu tırnak işaretlerini belirtmeyin.

Ad tamamen boşluklardan oluşuyorsa, *varsayılan* kuyruk yöneticisinin adı kullanılır. Ancak, IBM MQ MQI client uygulamalarındaki bölümde açıklanan boş kuyruk yöneticisi adlarının kullanımına dikkat edin.

*QMGrName* için belirtilen ad bir *bağlanabilir* kuyruk yöneticisinin adı ya da kuyruk yöneticisi grupları kullanılıyorsa, kuyruk yöneticisi grubunun adı olmalıdır.

z/OS üzerinde, bağlanabildiği kuyruk yöneticileri ortam tarafından belirlenir:

- CICS için yalnızca CICS sisteminin bağlı olduğu kuyruk yöneticisini kullanabilirsiniz. **QMGrName** parametresi yine de belirtilmelidir, ancak değeri yoksayılmıştır; boş karakterler uygun bir seçenektir.
- IMS için, IMS içinde SSM çizelgesinde listelenen yalnızca altsistem tanımlama çizelgesinde (CSQQDEFV), ve kuyruk yöneticileri bağlanabilir (kullanım notuna bakın 6).
- z/OS toplu iş ve TSO için, yalnızca uygulamayla aynı sistemde bulunan kuyruk yöneticileri bağlanabilir (kullanım notuna bakın 6).

**Kuyruk paylaşım grupları:** Birden çok kuyruk yöneticisinin bulunduğu ve bir kuyruk paylaşım grubu oluşturmak üzere yapılandırıldığı sistemlerde, kuyruk yöneticisi adı yerine *QMGrName* için kuyruk paylaşım grubunun adı belirtilebilir. Bu, uygulamanın, kuyruk paylaşım grubunda bulunan ve uygulamayla aynı z/OS görüntüsünde bulunan *herhangi bir* kuyruk yöneticisine bağlanmasını sağlar. Sistem, boş bir *QMGrName* kullanılması varsayılan kuyruk yöneticisi yerine kuyruk paylaşım grubuna bağlanacak şekilde de yapılandırılabilir.

*QMGrName* , kuyruk paylaşım grubunun adını belirtirse, ancak sistemde bu adı taşıyan bir kuyruk yöneticisi de varsa, öncekiyle bağlantı kurulur. Yalnızca bu bağlantı başarısız olursa, denenen kuyruk paylaşım grubundaki kuyruk yöneticilerinden biriyle bağlantı kurulur.

Bağlantı başarılı olursa, MQCONN ya da MQCONNX çağrısıyla döndürülen tanıtıcı değeri kullanarak, bağlantının kurulduğu kuyruk yöneticisine ait olan *tüm* kaynaklara (paylaşılan ve paylaşılmayan) erişebilirsiniz. Bu kaynaklara erişim, tipik yetki denetimlerine tabidir.

Uygulama eşzamanlı bağlantı kurmak için iki MQCONN ya da MQCONNX çağrısı yaparsa ve/ya da her iki çağrı kuyruk paylaşım grubunun adını belirtirse, ikinci çağrı ilk çağrıyla aynı kuyruk yöneticisine bağlandığında MQCC\_WARNING ve MQRC\_ALREADY\_CONNECTED neden kodunu döndürür.

Kuyruk paylaşım grupları yalnızca z/OS üzerinde desteklenir. Bir kuyruk paylaşım grubuna bağlantı yalnızca toplu iş, RRS toplu işi, CICS ve TSO ortamlarında desteklenir. CICS için yalnızca CICS sisteminin bağlı olduğu kuyruk paylaşım grubunu kullanabilirsiniz. **QMGrName** değiştirgesini yine de belirtmeniz gerekir, ancak değeri yoksayılr; boş karakterler uygun bir seçenektir.



**Uyarı:** IMS , bir kuyruk paylaşım grubuna bağlanamıyor.

**IBM MQ MQI client applications:** IBM MQ MQI client uygulamaları için, her istemci-bağlantı kanalı tanımlaması için belirtilen kuyruk yöneticisi adıyla, başarılı oluncaya kadar bağlantı girişiminde bulunulur. Ancak, kuyruk yöneticisi belirtilen adla aynı ada sahip olmalıdır. Tümü boş bir ad belirtilirse, kuyruk yöneticisi adı boş olan her istemci bağlantı kanalı, biri başarılı oluncaya kadar denenir; bu durumda, kuyruk yöneticisinin gerçek adıyla ilgili bir denetim olmaz.

IBM MQ istemci uygulamaları z/OS ürününde desteklenmez, ancak z/OS , IBM MQ istemci uygulamalarının bağlanabileceği bir IBM MQ sunucusu olarak işlev görür.

**IBM MQ MQI client Kuyruk yöneticisi grupları:** Belirtilen ad bir yıldız işaretiyle (\*) başlıyorsa, bağlantının kurulduğu kuyruk yöneticisinin adı uygulama tarafından belirtilenden farklı olabilir. Belirtilen ad (yıldız işareti olmadan), bağlantı için uygun kuyruk yöneticilerinden oluşan bir *grubu* tanımlar. Uygulama, bağlantı kurulabilecek bir bağlantı bulununcaya kadar her birini sırayla deneyerek gruptan bir tane seçer. Bağlantıların denendiği sıra, aday kanalların istemci kanal ağırlığı ve bağlantı benzerliği değerlerinden etkilenir. Gruptaki kuyruk yöneticilerinden hiçbiri bağlantı için uygun değilse, çağrı başarısız olur. Her kuyruk yöneticisi yalnızca bir kez denenir. Ad için tek başına bir yıldız imi belirtilirse, somutlama tanımlı bir varsayılan kuyruk yöneticisi grubu kullanılır.

Kuyruk yöneticisi grupları yalnızca MQ-istemcisi ortamında çalışan uygulamalar için desteklenir; istemci olmayan bir uygulama yıldız işaretiyle başlayan bir kuyruk yöneticisi adını belirtirse çağrı başarısız olur. Grup, gruptaki her kuyruk yöneticisiyle iletişim kurmak için, aynı kuyruk yöneticisi adıyla (yıldız imi olmadan belirtilen ad) birden çok istemci bağlantı kanalı tanımlaması sağlanarak tanımlanır. Varsayılan grup, her biri boş bir kuyruk yöneticisi adı olan bir ya da daha çok istemci bağlantı kanalı

tanımı sağlanarak tanımlanır (bu nedenle, tümü boş bir ad belirtilmesi, istemci uygulaması için ad için tek bir yıldız imi belirlenmesiyle aynı etkiye sahiptir).

Bir grubun bir kuyruk yöneticisine bağlandıktan sonra, bir uygulama, uygulamanın bağlandığı kuyruk yöneticisinin adını ( *yemel kuyruk yöneticisi* ) belirtmek için, iletideki ve nesne tanımlayıcılarındaki kuyruk yöneticisi adı alanlarında tipik olarak boşluk belirtebilir. Uygulamanın bu adı bilmesi gerekiyorsa, **QMGRName** kuyruk yöneticisi özniteliğini sorgulamak için MQINQ çağrısına bakın.

Bağlantı adına bir yıldız imi konması, uygulamanın gruptaki belirli bir kuyruk yöneticisine bağlanmasına bağlı olmadığı anlamına gelir. Uygun uygulamalar şunlardır:

- İletileri koyan, ancak ileti almayan uygulamalar.
- İstek iletileri koyan ve yanıt iletilerini *geçici dinamik* kuyruktan alan uygulamalar.

Uygun olmayan uygulamalar, belirli bir kuyruk yöneticisindeki belirli bir kuyruktan ileti alması gerekenlerdir; bu tür uygulamalar adın başına yıldız işareti koymamalıdır.

Bir yıldız işareti belirlerseniz, adın geri kalan kısmının uzunluk üst sınırı 47 karakterdir.

Bu değiştirgenin uzunluğu MQ\_Q\_MGR\_NAME\_LENGTH tarafından verilir.

### Hconn.

Tip: MQHCONN-çıkış

Bu tanıtıcı, kuyruk yöneticisine yönelik bağlantıyı gösterir. Bunu, uygulama tarafından yayınlanan sonraki tüm ileti kuyruklama çağrılarında belirtin. MQDISC çağrısı yayınlandığında ya da tanıtıcı kapsamını tanımlayan işlem birimi sona erdiğinde geçerli olmaz.

IBM MQ artık mqm kitaplığını sunucu paketlerinin yanı sıra istemci paketleriyle de sağlar. Bu, mqm kitaplığında bulunan bir MQI çağrısı yapıldığında, bağlantı tipinin bir istemci mi, yoksa sunucu bağlantısı mı olduğunu görmek için denetlendiği ve ardından doğru temel çağrı yapıldığı anlamına gelir. Bu nedenle, *Hconn* iletilen bir çıkış artık mqm kitaplığına bağlanabilir, ancak bir istemci kuruluşunda kullanılabilir.

*İşle kapsamı:* Döndürülen tanıtıcı kapsamı, kuyruk yöneticisine (MQCONN ya da MQCONNX) bağlanmak için kullanılan çağrıya bağlıdır. Kullanılan çağrı MQCONNX ise, tanıtıcı kapsamı MQCNO yapısının *Options* alanında belirtilen MQCNO\_HANDLE\_SHARE\_\* seçeneğine de bağlıdır.

- Çağrı MQCONN ise ya da MQCNO\_HANDLE\_SHARE\_NONE seçeneği belirtildiyse, döndürülen tanıtıcı *paylaşılmayan* bir tanıtıcıdır.

Paylaşılmayan bir tutamacın kapsamı, uygulamanın çalıştığı platform tarafından desteklenen en küçük koşut işleme birimidir (ayrıntılar için bkz. [Çizelge 547 sayfa 652](#) ); tanıtıcı, çağrı yayınlandığı koşut işleme birimi dışında geçerli değildir.

- MQCNO\_HANDLE\_SHARE\_BLOCK ya da MQCNO\_HANDLE\_SHARE\_NO\_BLOCK seçeneğini belirtirseniz, döndürülen tanıtıcı *paylaşılan* bir tanıtıcıdır.

Paylaşılan tanıtıcı kapsamı, çağrının verildiği iş parçacığına sahip olan işlemdir; tanıtıcı, o işleme ait olan herhangi bir iş parçacığından kullanılabilir. Her platform iş parçacıklarını desteklemez.

- MQCONN ya da MQCONNX çağrısı MQCC\_FAILED ' e eşit tamamlama koduyla başarısız olursa, Hconn değeri tanımsız olur.

<i>Çizelge 547. Çeşitli platformlardaki paylaşılmayan tutamaçların kapsamı</i>	
<b>Hizmet olarak sunulan</b>	<b>Paylaşılmayan tanıtıcı kapsamı</b>
z/OS	<ul style="list-style-type: none"><li>• CICS: CICS görevi</li><li>• IMS: görev, bir sonraki eşitleme noktasına kadar (görevin alt görevleri hariç)</li><li>• z/OS toplu iş ve TSO: görev (görevin alt görevleri hariç)</li></ul>
IBM i	İş



Çizelge 547. Çeşitli platformlardaki paylaşılmayan tutamaçların kapsamı (devamı var)	
Hizmet olarak sunulan	Paylaşılmayan tanıtıcı kapsamı
AIX and Linux	Başlık
32 bit Windows uygulamaları	Başlık
64 bit Windows uygulamaları	Başlık

z/OS ' da CICS uygulamaları için döndürülen değer:

**MQHC\_DEF\_HCONN**

Varsayılan bağlantı tanıtıcısı.

**CompCode**

Tip: MQLONG-output

Tamamlanma kodu; aşağıdakilerden biridir:

**MQCC\_OK**

Başarıyla tamamlandı.

**MQCC\_UYARISI**

Uyarı (kısmi tamamlama).

**MQCC\_FAILED**

Arama başarısız oldu.

**Neden**

Tip: MQLONG-output

CompCode MQCC\_OK ise:

**MQRC\_NONE**

(0, X'000 ') Raporlamak için bir neden yok.

CompCode MQCC\_WARNING ise:

**MQRC\_ALREADY\_CONNECTED**

(2002, X'7D2') Uygulama zaten bağlı.

**MQRC\_CLUSTER\_EXIT\_LOAD\_ERROR (Çıkış)**

(2267, X'8DB') Küme iş yükü çıkışı yüklenemiyor.

**MQRC\_SSL\_ALREADY\_KULLANIMA\_HAZIRLANDI**

(2391, X' 957 ') SSL zaten başlatıldı.

CompCode MQCC\_FAILED ise:

**MQRC\_ADAPTER\_CONN\_LOAD\_ERROR (UYGULAMA\_HATASI)**

(2129, X'851 ') Bağdaştırıcı bağlantı modülü yüklenemiyor.

**MQRC\_ADAPTER\_DEFS\_HATA**

(2131, X'853 ') Bağdaştırıcı altsistemi tanımlama modülü geçersiz.

**MQRC\_ADAPTER\_DEFS\_LOAD\_ERROR**

(2132, X'854 ') Bağdaştırıcı altsistemi tanımlama modülü yüklenemiyor.

**MQRC\_ADAPTER\_NOT\_KULLANILAMIYOR**

(2204, X'89C') Bağdaştırıcı kullanılmıyor.

**MQRC\_ADAPTER\_SERV\_LOAD\_ERROR (UYGULAMA\_HATASI)**

(2130, X'852 ') Bağdaştırıcı hizmet modülü yüklenemiyor.

**MQRC\_ADAPTER\_STORAGE\_SHORTAGE**

(2127, X'84F') Bağdaştırıcı için bellek yetersiz.

**MQRC\_ANOTHER\_Q\_MGR\_CONNECTED**

(2103, X'837 ') Başka bir kuyruk yöneticisi zaten bağlandı.

**MQRC\_API\_EXIT\_ERROR (UYGULAMA\_HATASI)**

(2374, X'946 ') API çıkışı başarısız oldu.

**MQRC\_API\_EXIT\_INIT\_ERROR (UYGULAMA\_HATASI)**

(2375, X'947 ') API çıkışını başlatma başarısız oldu.

**MQRC\_API\_EXIT\_TERM\_HATA**

(2376, X'948 ') API çıkışı sonlandırılmadı.

**MQRC\_ASID\_MISMATCH**

(2157, X'86D') Birincil ve ana ASID' ler farklıdır.

**MQRC\_BUFFER\_LENGTH\_ERROR**

(2005, X'7D5') Arabellek uzunluğu parametresi geçersiz.

**MQRC\_CALL\_IN\_PROGRESS**

(2219, X'8AB') Önceki arama tamamlanmadan önce girilen MQI çağrısı.

**MQRC\_CONN\_ID\_IN\_USE**

(2160, X'870 ') Bağlantı tanıtıcısı kullanımda.

**MQRC\_CONNECTION\_BROKEN**

(2009, X'7D9') Kuyruk yöneticisine bağlantı kaybedildi.

**MQRC\_CONNECTION\_HATA**

(2273, X'8E1') MQCONN çağrısı işlenirken hata oluştu.

**MQRC\_CONNECTION\_KULLANILAMIYOR**

(2568, X'A08') Kuyruk yöneticisi yürürlükteki kuruluştaki istenen bağlantı tipini sağlayamadığı zaman MQCONN ya da MQCONNX çağrısında oluşur. Yalnızca sunucu kuruluşunda istemci bağlantısı kurulamaz. Yalnızca istemci kuruluşunda yerel bağlantı kurulamaz.

**MQRC\_CONNECTION QUIESCING**

(2202, X'89A') Bağlantı susturuyor.

**MQRC\_CONNECTION DURDURULUYOR**

(2203, X'89B') Bağlantı kapatılıyor.

**MQRC\_CRYPTO\_HARDWARE\_HATA**

(2382, X'94E') Şifreleme donanımı yapılandırma hatası.

**MQRC\_DUPLICATE\_RECOV\_COORD**

(2163, X'873 ') Kurtarma koordinatörü var.

**MQRC\_ENVIRONMENT\_ERROR**

(2012, X'7DC') Çağrısı ortamda geçerli değil.

Buna ek olarak, MQCONNX çağrısında, bir CICS ya da IMS uygulamasından "MQCSP-Güvenlik değiştiricileri" sayfa 335 denetim öbeğini iletiyor.

**MQRC\_HCONN\_HATA**

(2018, X'7E2') Bağlantı tanıtıcısı geçersiz.

**MQRC\_ANASISTEM\_YOK**

(2538, X'9EA') Bir istemciden kuyruk yöneticisine bağlanmak için MQCONN çağrısı yayınlandı, ancak uzak sisteme etkileşim ayırma girişimi başarısız oldu.

**MQRC\_INSTALLATION\_MISMATCH**

(2583, X'A17') Kuyruk yöneticisi kuruluşu ile seçilen kitaplık arasında uyumsuzluk var.

**MQRC\_KEY\_REPOSITORY\_HATA**

(2381, X'94D') Anahtar havuzu geçersiz.

**MQRC\_MAX\_CONNS\_LIMIT\_SINIRINA ulaşıldı**

(2025, X'7E9') Bağlantı sayısı üst sınırına ulaşıldı.

**MQRC\_NOT\_YETKILI**

(2035, X'7F3') Erişim için yetkili değil.

**MQRC\_OPEN\_FAILED**

(2137, X'859 ') Nesne başarıyla açılmadı.

**MQRC\_Q\_MGR\_AD\_HATASI**

(2058, X'80A') Kuyruk yöneticisi adı geçerli değil ya da bilinmiyor.

**MQRC\_Q\_MGR\_YOK**

(2059, X'80B') Kuyruk yöneticisi bağlantı için kullanılmıyor.

**MQRC\_Q\_MGR\_QUIESCING**

(2161, X'871 ') Kuyruk yöneticisi susturuyor.

**MQRC\_Q\_MGR\_DURDURULUYOR**

(2162, X'872 ') Kuyruk yöneticisi kapatılıyor.

**MQRC\_RESOURCE\_PROBLEM**

(2102, X'836 ') Sistem kaynakları yetersiz.

**MQRC\_SECURITY\_HATA**

(2063, X'80F') Güvenlik hatası oluştu.

**MQRC\_SSL\_INITIALIZION\_HATA**

(2393, X' 959 ') SSL kullanıma hazırlama hatası.

**MQRC\_STORAGE\_KULLANILAMIYOR**

(2071, X'817 ') Kullanılabilir bellek yetersiz.

**MQRC\_UNEXPECTED\_HATASI**

(2195, X'893 ') Beklenmeyen hata oluştu.

Bu kodlarla ilgili ayrıntılı bilgi için bkz. [İletiler ve neden kodları](#).

**Kullanım notları**

1. MQCONN çağrısı kullanılarak bağlantı kurulan kuyruk yöneticisine *yemel kuyruk yöneticisi* adı verilir.
2. Yerel kuyruk yöneticisinin sahip olduğu kuyruklar, uygulamada yerel kuyruklar olarak görünür. İletiler takılabilir ve bu kuyruklardan ileti alabilir.  
  
Yerel kuyruk yöneticisinin ait olduğu kuyruk paylaşım grubunun sahip olduğu paylaşılan kuyruklar, uygulamaya yerel kuyruklar olarak görünür. İletiler takılabilir ve bu kuyruklardan ileti alabilir.  
  
Uzak kuyruk yöneticilerinin iyeliğindeki kuyruklar uzak kuyruklar olarak görünür. Bu kuyruklara ileti koymak mümkündür, ancak bu kuyruklardan ileti almak mümkün değildir.
3. Bir uygulama çalışırken kuyruk yöneticisi başarısız olursa, uygulamanın sonraki IBM MQ çağrılarında kullanılacak yeni bir bağlantı tanıtıcısı elde etmek için MQCONN çağrısını yeniden yürütmesi gerekir. Uygulama, çağrı başarılı oluncaya kadar düzenli aralıklarla MQCONN çağrısına devam edebilir.  
  
Bir uygulama kuyruk yöneticisine bağlı olup olmadığından emin değilse, uygulama bağlantı tanıtıcısı elde etmek için güvenli bir şekilde bir MQCONN çağrısı verebilir. Uygulama zaten bağlıysa, döndürülen tanıtıcı önceki MQCONN çağrısıyla aynıdır, ancak tamamlama kodu MQCC\_WARNING ve neden kodu MQRC\_ALREADY\_CONNECTED.
4. Uygulama IBM MQ çağrılarını kullanmayı bitirdiğinde, uygulamanın kuyruk yöneticisiyle bağlantısını kesmek için MQDISC çağrılarını kullanması gerekir.
5. MQCONN çağrısı MQCC\_FAILED ' e eşit tamamlama koduyla başarısız olursa, Hconn değeri tanımsız olur.
6. z/OS'ta:
  - Toplu iş, TSO ve IMS uygulamaları, diğer IBM MQ çağrılarını kullanmak için MQCONN çağrılarını yayınlamalıdır. Bu uygulamalar eşzamanlı olarak birden çok kuyruk yöneticisine bağlanabilir.  
  
Kuyruk yöneticisi başarısız olursa, yeni bir bağlantı tanıtıcısı elde etmek için uygulamanın kuyruk yöneticisi yeniden başlatıldıktan sonra çağrıyı yeniden yürütmesi gerekir.  
  
IMS uygulamaları MQCONN çağrılarını, önceden bağlanmış olsalar bile, yinelenen olarak yürütebilse de, çevrimiçi ileti işleme programları (MPPs) için bu önerilmez.


- CICS uygulamalarının diğer IBM MQ çağrılarını kullanmak için MQCONN çağrısı yayınlamak zorunda değildir, ancak isterlerse bunu yapabilir; hem MQCONN çağrısı hem de MQDISC çağrısı kabul edilir. Ancak, aynı anda birden çok kuyruk yöneticisiyle bağlantı kurulamaz.

Kuyruk yöneticisi başarısız olursa, kuyruk yöneticisi yeniden başlatıldığında bu uygulamalar otomatik olarak yeniden bağlanır ve MQCONN çağrısına gerek yoktur.

7. z/OS' da kullanılabilir kuyruk yöneticilerini tanımlamak için:

- Toplu iş uygulamalarında, sistem programcılarını CSQBDEF makrosunu kullanarak varsayılan kuyruk yöneticisi adını ya da kuyruk paylaşım grubu adını tanımlayan bir birim (CSQBDEFV) yaratabilir.
- IMS uygulamalarında, sistem programcılarını kullanılabilir kuyruk yöneticilerinin adlarını tanımlayan ve varsayılan kuyruk yöneticisini belirten bir modül (CSQQDEFV) yaratmak için CSQQDEFX makrosunu kullanabilir.

Ayrıca, her kuyruk yöneticisi IMS denetim bölgesinde ve o kuyruk yöneticisine erişen her bağımlı bölgede tanımlanmalıdır. Bunu yapmak için IMS içinde bir altsistem üyesi oluşturmanız gerekir. PROCLIB kitaplığı ve altsistem üyesini ilgili IMS bölgelerine tanıtır. Bir uygulama, IMS bölgesi için altsistem üyesinde tanımlı olmayan bir kuyruk yöneticisine bağlanmayı denerse, uygulama olağandışı biter.

 **z/OS** Bu makroları kullanma hakkında daha fazla bilgi için bkz. [Müşteri kullanımı için tasarlanmış makrolar](#).

8. IBM üzerinde, olağandışı sona eren programların kuyruk yöneticisiyle bağlantısı otomatik olarak kesilmez. MQCONN ya da MQCONNX çağrısının tamamlama kodu MQCC\_WARNING ve neden kodu MQRC\_ALREADY\_CONNECTED döndürme olasılığına izin verecek uygulamaları yazın. Bu durumda döndürülen bağlantı tanıtıcısını olağan biçimde kullanın.

## C çağrısı

```
MQCONN (QMgrName, &Hconn, &CompCode, &Reason);
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
MQCHAR48 QMgrName; /* Name of queue manager */
MQHCONN Hconn; /* Connection handle */
MQLONG CompCode; /* Completion code */
MQLONG Reason; /* Reason code qualifying CompCode */
```

## COBOL çağrısı

```
CALL 'MQCONN' USING QMGRNAME, HCONN, COMPCODE, REASON.
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
** Name of queue manager
01 QMGRNAME PIC X(48).
** Connection handle
01 HCONN PIC S9(9) BINARY.
** Completion code
01 COMPCODE PIC S9(9) BINARY.
** Reason code qualifying COMPCODE
01 REASON PIC S9(9) BINARY.
```

## PL/I çağrısı

```
call MQCONN (QMGrName, Hconn, CompCode, Reason);
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
dcl QMgrName char(48); /* Name of queue manager */
dcl Hconn fixed bin(31); /* Connection handle */
dcl CompCode fixed bin(31); /* Completion code */
dcl Reason fixed bin(31); /* Reason code qualifying CompCode */
```

## High Level Assembler (Yüksek Düzeyli çevirici) çağrısı

```
CALL MQCONN, (QMGRNAME, HCONN, COMPCODE, REASON)
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
QMGRNAME DS CL48 Name of queue manager
HCONN DS F Connection handle
COMPCODE DS F Completion code
REASON DS F Reason code qualifying COMPCODE
```

## Visual Basic çağrısı

```
MQCONN QMgrName, Hconn, CompCode, Reason
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
Dim QMgrName As String*48 'Name of queue manager'
Dim Hconn As Long 'Connection handle'
Dim CompCode As Long 'Completion code'
Dim Reason As Long 'Reason code qualifying CompCode'
```

## MQCONNX-Connect kuyruk yöneticisi (genişletilmiş)

MQCONNX çağrısı bir uygulama programını kuyruk yöneticisine bağlar. Bu, sonraki IBM MQ çağrılarında uygulama tarafından kullanılan bir kuyruk yöneticisi bağlantı tanıtıcısı sağlar.

MQCONNX çağrısı, MQCONNX çağrısının, çağrılarının çalışma şeklini denetlemek için belirtilmesine izin vermesi dışında, MQCONN çağrısına benzer.

- Bu çağrı, tüm IBM MQ sistemlerinde ve bu sistemlere bağlı IBM MQ istemcilerinde desteklenir.

Yalnızca bir sunucu kuruluşunda istemci bağlantısı kurulamaz ve yalnızca istemci kuruluşunda yerel bir bağlantı kurulamaz.

## Sözdizimi

MQCONNX (*QMgrName*, *ConnectOpts*, *Hconn*, *CompCode*, *Reason*)

### Parametreler

#### QMgrName

Tip: MQCHAR48 -giriş

Ayrıntılar için “MQCONN-Bağlantı kuyruğu yöneticisi” sayfa 650 içinde açıklanan **QMgrName** parametresine bakın.

#### ConnectOpts

Tip: MQCNO-giriş/çıkış

Ayrıntılar için bkz. “MQCNO-Bağlantı seçenekleri” sayfa 314.

## Hconn.

Tip: MQHCONN-çıkış

Bu tanıtıcı, kuyruk yöneticisine yönelik bağlantıyı gösterir. Bunu, uygulama tarafından yayınlanan sonraki tüm ileti kuyruklama çağrılarında belirtin. MQDISC çağrısı yayınlandığında ya da tanıtıcı kapsamını tanımlayan işlem birimi sona erdiğinde geçerli olmaz.

IBM MQ artık mqm kitaplığını sunucu paketlerinin yanı sıra istemci paketleriyle de sağlar. Bu, mqm kitaplığında bulunan bir MQI çağrısı yapıldığında, bağlantı tipinin bir istemci mi, yoksa sunucu bağlantısı mı olduğunu görmek için denetlendiği ve ardından doğru temel çağrı yapıldığı anlamına gelir. Bu nedenle, *Hconn* iletilen bir çıkış artık mqm kitaplığına bağlanabilir, ancak bir istemci kuruluşunda kullanılabilir.

*İşle kapsamı:* Döndürülen tanıtıcı kapsamı, kuyruk yöneticisine (MQCONN ya da MQCONNX) bağlanmak için kullanılan çağrıya bağlıdır. Kullanılan çağrı MQCONNX ise, tanıtıcı kapsamı MQCNO yapısının *Options* alanında belirtilen MQCNO\_HANDLE\_SHARE\_\* seçeneğine de bağlıdır.

- Çağrı MQCONN ise ya da MQCNO\_HANDLE\_SHARE\_NONE seçeneği belirtildiyse, döndürülen tanıtıcı *paylaşılmayan* bir tanıtıcıdır.

Paylaşılmayan bir tutamacın kapsamı, uygulamanın çalıştığı platform tarafından desteklenen en küçük koşut işleme birimidir (ayrıntılar için bkz. [Çizelge 548 sayfa 658](#)); tanıtıcı, çağrı yayınlandığı koşut işleme birimi dışında geçerli değildir.

- MQCNO\_HANDLE\_SHARE\_BLOCK ya da MQCNO\_HANDLE\_SHARE\_NO\_BLOCK seçeneğini belirtirseniz, döndürülen tanıtıcı *paylaşılan* bir tanıtıcıdır.

Paylaşılan tanıtıcı kapsamı, çağrının verildiği iş parçacığına sahip olan işlemdir; tanıtıcı, o işleme ait olan herhangi bir iş parçacığından kullanılabilir. Her platform iş parçacıklarını desteklemez.

- MQCONN ya da MQCONNX çağrısı MQCC\_FAILED ' e eşit tamamlama koduyla başarısız olursa, Hconn değeri tanımsız olur.

Hizmet olarak sunulan	Paylaşılmayan tanıtıcı kapsamı
z/OS	<ul style="list-style-type: none"><li>• CICS: CICS görevi</li><li>• IMS: görev, bir sonraki eşitleme noktasına kadar (görevin alt görevleri hariç)</li><li>• z/OS toplu iş ve TSO: görev (görevin alt görevleri hariç)</li></ul>
IBM i	İş
AIX and Linux	Başlık
32 bit Windows uygulamaları	Başlık
64 bit Windows uygulamaları	Başlık

z/OS ' da CICS uygulamaları için döndürülen değer:

### **MQHC\_DEF\_HCONN**

Varsayılan bağlantı tanıtıcısı.

## CompCode

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Ayrıntılar için “MQCONN-Bağlantı kuyruğu yöneticisi” sayfa 650 içinde açıklanan **CompCode** parametresine bakın.

## Neden

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

İzleyen kodlar, MQCONN ve MQCONNX çağrıları tarafından döndürülebilmektedir. MQCONNX çağrısı tarafından döndürülebilecek ek kodların listesi için aşağıdaki kodlara bakın.

*CompCode* MQCC\_OK ise:

**MQRC\_NONE**

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_WARNING ise:

**MQRC\_ALREADY\_CONNECTED**

(2002, X'7D2') Uygulama zaten bağlı.

**MQRC\_CLUSTER\_EXIT\_LOAD\_ERROR (Çıkış)**

(2267, X'8DB') Küme iş yükü çıkışı yüklenemiyor.

**MQRC\_SSL\_ALREADY\_KULLANIMA\_HAZIRLANDI**

(2391, X' 957 ') SSL zaten başlatıldı.

*CompCode* MQCC\_FAILED ise:

**MQRC\_ADAPTER\_CONN\_LOAD\_ERROR (UYGULAMA\_HATASI)**

(2129, X'851 ') Bağdaştırıcı bağlantı modülü yüklenemiyor.

**MQRC\_ADAPTER\_DEFS\_HATA**

(2131, X'853 ') Bağdaştırıcı altsistemi tanımlama modülü geçersiz.

**MQRC\_ADAPTER\_DEFS\_LOAD\_ERROR**

(2132, X'854 ') Bağdaştırıcı altsistemi tanımlama modülü yüklenemiyor.

**MQRC\_ADAPTER\_NOT\_KULLANILAMIYOR**

(2204, X'89C') Bağdaştırıcı kullanılmıyor.

**MQRC\_ADAPTER\_SERV\_LOAD\_ERROR (UYGULAMA\_HATASI)**

(2130, X'852 ') Bağdaştırıcı hizmet modülü yüklenemiyor.

**MQRC\_ADAPTER\_STORAGE\_SHORTAGE**

(2127, X'84F') Bağdaştırıcı için bellek yetersiz.

**MQRC\_ANOTHER\_Q\_MGR\_CONNECTED**

(2103, X'837 ') Başka bir kuyruk yöneticisi zaten bağlandı.

**MQRC\_API\_EXIT\_ERROR (UYGULAMA\_HATASI)**

(2374, X' 946 ') API çıkışı başarısız oldu.

**MQRC\_API\_EXIT\_INIT\_ERROR (UYGULAMA\_HATASI)**

(2375, X' 947 ') API çıkışını başlatma başarısız oldu.

**MQRC\_API\_EXIT\_TERM\_HATA**

(2376, X' 948 ') API çıkışı sonlandırılmadı.

**MQRC\_ASID\_MISMATCH**

(2157, X'86D') Birincil ve ana ASID' ler farklıdır.

**MQRC\_BUFFER\_LENGTH\_ERROR**

(2005, X'7D5') Arabellek uzunluğu parametresi geçersiz.

**MQRC\_CALL\_IN\_PROGRESS**

(2219, X'8AB') Önceki arama tamamlanmadan önce girilen MQI çağrısı.

**MQRC\_CONN\_ID\_IN\_USE**

(2160, X'870 ') Bağlantı tanıtıcısı kullanımda.

**MQRC\_CONNECTION\_BROKEN**

(2009, X'7D9') Kuyruk yöneticisine bağlantı kaybedildi.

**MQRC\_CONNECTION\_HATA**

(2273, X'8E1') MQCONN çağrısı işlenirken hata oluştu.

**MQRC\_CONNECTION\_KULLANILAMIYOR**

(2568, X'A08') Kuyruk yöneticisi yürürlükteki kuruluşta istenen bağlantı tipini sağlayamadığı zaman MQCONN ya da MQCONNX çağrısında oluşur. Yalnızca sunucu kuruluşunda istemci bağlantısı kurulamaz. Yalnızca istemci kuruluşunda yerel bağlantı kurulamaz.

**MQRC\_CONNECTION\_QUIESCING**

(2202, X'89A') Bağlantı susturuyor.

**MQRC\_CONNECTION\_DURDURULUYOR**

(2203, X'89B') Bağlantı kapatılıyor.

**MQRC\_CRYPTO\_HARDWARE\_HATA**

(2382, X'94E') Şifreleme donanımı yapılandırma hatası.

**MQRC\_DUPLICATE\_RECOV\_COORD**

(2163, X'873 ') Kurtarma koordinatörü var.

**MQRC\_ENVIRONMENT\_ERROR**

(2012, X'7DC') Çağrısı ortamda geçerli değil.

Buna ek olarak, MQCONNX çağrısında, bir CICS ya da IMS uygulamasından “MQCSP-Güvenlik değiştirgeleri” sayfa 335 denetim öbeğini iletiyor.

**MQRC\_HCONN\_HATA**

(2018, X'7E2') Bağlantı tanıtıcısı geçersiz.

**MQRC\_ANASISTEM\_YOK**

(2538, X'9EA') Bir istemciden kuyruk yöneticisine bağlanmak için MQCONN çağrısı yayınlandı, ancak uzak sisteme etkileşim ayırma girişimi başarısız oldu.

**MQRC\_INSTALLATION\_MISMATCH**

(2583, X'A17') Kuyruk yöneticisi kurulumu ile seçilen kitaplık arasında uyumsuzluk var.

**MQRC\_KEY\_REPOSITORY\_HATA**

(2381, X'94D') Anahtar havuzu geçersiz.

**MQRC\_MAX\_CONNS\_LIMIT\_SINIRINA ulaşıldı**

(2025, X'7E9') Bağlantı sayısı üst sınırına ulaşıldı.

**MQRC\_NOT\_YETKILI**

(2035, X'7F3') Erişim için yetkili değil.

**MQRC\_OPEN\_FAILED**

(2137, X'859 ') Nesne başarıyla açılmadı.

**MQRC\_Q\_MGR\_AD\_HATASI**

(2058, X'80A') Kuyruk yöneticisi adı geçerli değil ya da bilinmiyor.

**MQRC\_Q\_MGR\_YOK**

(2059, X'80B') Kuyruk yöneticisi bağlantı için kullanılmıyor.

**MQRC\_Q\_MGR\_QUIESCING**

(2161, X'871 ') Kuyruk yöneticisi susturuyor.

**MQRC\_Q\_MGR\_DURDURULUYOR**

(2162, X'872 ') Kuyruk yöneticisi kapatılıyor.

**MQRC\_RESOURCE\_PROBLEM**

(2102, X'836 ') Sistem kaynakları yetersiz.

**MQRC\_SECURITY\_HATA**

(2063, X'80F') Güvenlik hatası oluştu.

**MQRC\_SSL\_INITIALIZION\_HATA**

(2393, X' 959 ') SSL kullanıma hazırlama hatası.

**MQRC\_STORAGE\_KULLANILAMIYOR**

(2071, X'817 ') Kullanılabilir bellek yetersiz.

**MQRC\_UNEXPECTED\_HATASI**

(2195, X'893 ') Beklenmeyen hata oluştu.

İzleyen ek neden kodları, MQCONNX çağrısı tarafından döndürülebilir:

*CompCode* MQCC\_FAILED ise:

**MQRC\_AIR\_ERROR**

(2385, X' 951 ') Kimlik Doğrulama bilgileri kaydı geçerli değil.



**MQRC\_AUTH\_INFO\_CONN\_NAME\_ERROR**

(2387, X' 953 ') Kimlik doğrulama bilgileri bağlantı adı geçerli değil.

**MQRC\_AUTH\_INFO\_REC\_COUNT\_ERROR**

(2383, X'94F') Kimlik doğrulama bilgileri kayıt sayısı geçerli değil.

**MQRC\_AUTH\_INFO\_REC\_ERROR**

(2384, X' 950 ') Kimlik doğrulama bilgileri kayıt alanları geçerli değil.

**MQRC\_AUTH\_INFO\_TYPE\_ERROR**

(2386, X' 952 ') Kimlik doğrulama bilgileri tipi geçerli değil.

**MQRC\_CD\_HATASI**

(2277, X'8E5') Kanal tanımlaması geçerli değil.

**MQRC\_CLIENT\_CONN\_ERROR**

(2278, X'8E6') İstemci bağlantı alanları geçerli değil.

**MQRC\_CNO\_HATASI**

(2139, X'85B') Connect-options yapısı geçerli değil.

**MQRC\_CONN\_TAG\_IN\_USE**

(2271, X'8DF') Bağlantı etiketi kullanımda.

**MQRC\_CONN\_TAG\_NOT\_USABLE**

(2350, X'92E') Bağlantı etiketi kullanılabilir değil.

**MQRC\_LDAP\_PASSWORD\_ERROR**

(2390, X' 956 ') LDAP parolası geçerli değil.

**MQRC\_LDAP\_USER\_NAME\_ERROR**

(2388, X' 954 ') LDAP kullanıcı adı alanları geçerli değil.

**MQRC\_LDAP\_USER\_NAME\_LENGTH\_ERR**

(2389, X' 955 ') LDAP kullanıcı adı uzunluğu geçerli değil.

**MQRC\_OPTIONS\_ERROR**

(2046, X'7FE') Seçenekler geçerli değil ya da tutarlı değil.

**MQRC\_SCO\_ERROR**

(2380, X'94C') SSL yapılandırma seçenekleri yapısı geçerli değil.

**MQRC\_SSL\_CONFIG\_ERROR**

(2392, X' 958 ') SSL yapılandırma hatası.

Bu kodlarla ilgili ayrıntılı bilgi için [İletiler ve neden kodları](#) başlıklı konuya bakın.

## Kullanım notları

Visual Basic programlama dili için, aşağıdaki nokta geçerli olur:

- **ConnectOpts** parametresi, MQCNO tipinde olduğu için bildirilir. If the application is running as an IBM MQ MQI client, and you want to specify the parameters of the client-connection channel, declare the **ConnectOpts** parameter as being of type Any, so that the application can specify an MQCNOCD structure on the call in place of an MQCNO structure. Ancak, bu, **ConnectOpts** parametresinin doğru veri tipinde olduğundan emin olmak için denetlenemez anlamına gelir.

## C çağırma

```
MQCONN (QMgrName, &ConnectOpts, &Hconn, &CompCode, &Reason);
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
MQCHAR48 QMgrName; /* Name of queue manager */
MQCNO ConnectOpts; /* Options that control the action of MQCONN */
MQHCONN Hconn; /* Connection handle */
```

```
MQLONG   CompCode;    /* Completion code */
MQLONG   Reason;      /* Reason code qualifying CompCode */
```

## Cobol çağırma

```
CALL 'MQCONN' USING QMGRNAME, CONNECTOPTS, HCONN, COMPCODE,
REASON.
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
** Name of queue manager
01 QMGRNAME PIC X(48).
** Options that control the action of MQCONN
01 CONNECTOPTS.
   COPY CMQCNOV.
** Connection handle
01 HCONN PIC S9(9) BINARY.
** Completion code
01 COMPCODE PIC S9(9) BINARY.
** Reason code qualifying COMPCODE
01 REASON PIC S9(9) BINARY.
```

## PL/I çağırımı

```
call MQCONN (QMgrName, ConnectOpts, Hconn, CompCode, Reason);
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
dcl QMgrName char(48); /* Name of queue manager */
dcl ConnectOpts like MQCNO; /* Options that control the action of
MQCONN */
dcl Hconn fixed bin(31); /* Connection handle */
dcl CompCode fixed bin(31); /* Completion code */
dcl Reason fixed bin(31); /* Reason code qualifying CompCode */
```

## High Level Assembler çağırısı

```
CALL MQCONN, (QMGRNAME,CONNECTOPTS,HCONN,COMPCODE,REASON)
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
QMGRNAME DS CL48 Name of queue manager
CONNECTOPTS CMQCNOA , Options that control the action of MQCONN
HCONN DS F Connection handle
COMPCODE DS F Completion code
REASON DS F Reason code qualifying COMPCODE
```

## Visual Basic çağırısı

```
MQCONN QMgrName, ConnectOpts, Hconn, CompCode, Reason
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
Dim QMgrName As String*48 'Name of queue manager'
Dim ConnectOpts As MQCNO 'Options that control the action of'
' MQCONN'
Dim Hconn As Long 'Connection handle'
```

Dim CompCode	As Long	'Completion code'
Dim Reason	As Long	'Reason code qualifying CompCode'

## MQCRTMH-İleti tanıtıcısı yarat

MQCRTMH çağrısı bir ileti tanıtıcısı döndürür.

Bir uygulama sonraki ileti kuyruklama çağrılarında MQCRTMH çağrısı kullanabilir:

- İleti tanıtıcısının bir özelliğini ayarlamak için [MQSETMP](#) çağrısını kullanın.
- İleti tanıtıcısının bir özelliğinin değerini sorgulamak için [MQINQMP](#) çağrısını kullanın.
- İleti tanıtıcısının bir özelliğini silmek için [MQDLTMP](#) çağrısını kullanın.

İleti tanıtıcısı MQPUT ve MQPUT1 çağrılarında, ileti tanıtıcısının özelliklerini, konmakta olan iletiyle ilişkilendirmek için kullanılabilir. Benzer şekilde, MQGET çağrısında bir ileti tanıtıcısı belirtilerek, MQGET çağrısı tamamlandığında ileti tanıtıcısı kullanılarak, alınmakta olan iletinin özelliklerine erişilebilir.

İleti tanıtıcısını silmek için [MQDLTMH](#) komutunu kullanın.

## Sözdizimi

MQCRTMH (*Hconn*, *CrtMsgHOpts*, *Hmsg*, *CompCode*, *Reason*)

## Parametreler

### Hconn.

Tip: MQHCONN-giriş

Bu tanıtıcı, kuyruk yöneticisine yönelik bağlantıyı gösterir. *Hconn* değeri önceki bir MQCONN ya da MQCONNX çağrısı tarafından döndürüldü. Kuyruk yöneticisine yönelik bağlantı geçerliliğini keserse ve ileti tanıtıcısında IBM MQ çağrısı çalışmıyorsa, iletiyi silmek için [MQDLTMH](#) örtük olarak çağrılır.

Diğer bir seçenek olarak, aşağıdaki değeri belirtebilirsiniz:

### MQHC\_UNASSOCIATED\_HCONN

Bağlantı tanıtıcısı, belirli bir kuyruk yöneticisine yönelik bir bağlantıyı göstermiyor.

Bu değer kullanıldığında, kendisine ayrılan saklama alanını serbest bırakmak için ileti tanıtıcısı [MQDLTMH](#) 'e yönelik belirtik bir çağrıyla silinmelidir; IBM MQ ileti tanıtıcısını hiçbir zaman örtük olarak silmez.

İleti tanıtıcısını yaratan iş parçacığında kurulmuş bir kuyruk yöneticisiyle en az bir geçerli bağlantı olmalıdır; tersi durumda, çağrı MQRC\_HCONN\_ERROR ile başarısız olur.

Tek bir sistemde birden çok kuruluşu olan bir ortamda, MQHC\_UNASSOCIATED\_HCONN değeri, sürece yüklenen ilk kuruluşla birlikte kullanımla sınırlıdır. İleti tanıtıcısı farklı bir kuruluşla sağlanırsa, MQRC\_HMSG\_NOT\_AVAILABLE neden kodu döndürülür.

z/OS for CICS uygulamalarında MQCONN çağrısı atlanabilir ve *Hconn* için aşağıdaki değeri belirtebilirsiniz:

### MQHC\_DEF\_CONN

Varsayılan bağlantı tanıtıcısı

### CrtMsgHOpts

Tip: MQCMHO-giriş

MQCRTMH işlemini denetleyen seçenekler. Ayrıntılar için bkz. [MQCMHO](#) .

### İlt

Tip: MQHMSG-çıkış

Çıkışta, ileti tanıtıcısının özelliklerini ayarlamak, sorgulamak ve silmek için kullanılacak bir ileti tanıtıcısı döndürülür. Başlangıçta ileti tanıtıcısı özellik içeriyor.

İleti tanıtıcısının ilişkili bir ileti tanımlayıcısı da vardır. Başlangıçta bu, varsayılan değerleri içerir. İlişkili ileti tanımlayıcı alanlarının değerleri MQSETMP ve MQINQMP çağrıları kullanılarak ayarlanabilir ve sorilebilir. MQDLTMP çağrısı, ileti tanımlayıcısının bir alanını varsayılan değerine geri çevirir.

*Hconn* değiştirgesi MQHC\_UNASSOCIATED\_HCONN değeri olarak belirtilirse, döndürülen ileti tanıtıcısı MQGET, MQPUT ya da MQPUT1 çağrılarında işlem birimi içindeki herhangi bir bağlantıyla kullanılabilir, ancak bir kerede tek bir IBM MQ çağrısı tarafından kullanılabilir. İkinci bir IBM MQ çağrısı aynı ileti tanıtıcısını kullanmayı denediğinde tanıtıcı kullanılırsa, ikinci IBM MQ çağrısı MQRC\_MSG\_HANDLE\_IN\_USE neden koduyla başarısız olur.

*Hconn* değiştirgesi MQHC\_UNASSOCIATED\_HCONN değilse, döndürülen ileti tanıtıcısı yalnızca belirtilen bağlantıda kullanılabilir.

Bu ileti işlecinin kullanıldığı sonraki MQI çağrılarında aynı *Hconn* değiştirge değeri kullanılmalıdır:

- MQDLTMH
- MQSETMP
- MQINQMP
- MQDLTMP
- MQMHBUF
- MQBUFMH

İleti tanıtıcısı için MQDLTMH çağrısı yayınlandığında ya da tanıtıcı kapsamını tanımlayan işlem birimi sona erdiğinde, döndürülen ileti tanıtıcısı geçerli olmaktan çıkar. İleti tanıtıcısı yaratıldığında belirli bir bağlantı sağlanırsa ve kuyruk yöneticisine yönelik bağlantı geçerliliğini keserse (örneğin, MQDBC çağrılırsa) MQDLTMH örtük olarak çağrılır.

### CompCode

Tip: MQLONG-output

Tamamlanma kodu; aşağıdakilerden biridir:

#### **MQCC\_OK**

Başarıyla tamamlandı.

#### **MQCC\_FAILED**

Arama başarısız oldu.

### Neden

Tip: MQLONG-output

*CompCode* MQCC\_OK ise:

#### **MQRC\_NONE**

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_FAILED ise:

#### **MQRC\_ADAPTER\_NOT\_KULLANILAMIYOR**

(2204, X'089C') Bağdaştırıcı kullanılmıyor.

#### **MQRC\_ADAPTER\_SERV\_LOAD\_ERROR (UYGULAMA\_HATASI)**

(2130, X'852 ') Bağdaştırıcı hizmet modülü yüklenemiyor.

#### **MQRC\_ASID\_MISMATCH**

(2157, X'86D') Birincil ve ana ASID' ler farklıdır.

#### **MQRC\_CALL\_IN\_PROGRESS**

(2219, X'08AB') Önceki arama tamamlanmadan önce girilen MQI çağrısı.

#### **MQRC\_CMHO\_HATA**

(2461, X'099D') İleti tanıtıcısı yaratma seçenekleri yapısı geçersiz.

#### **MQRC\_CONNECTION\_BROKEN**

(2273, X'7D9') Kuyruk yöneticisiyle bağlantı kesildi.

**MQRC\_HANDLE\_NOT\_YOK**

(2017, X'07E1') Başka tanıtıcı yok.

**MQRC\_HCONN\_HATA**

(2018, X'7E2') Bağlantı tanıtıcısı geçersiz.

**MQRC\_HMSG\_HATA**

(2460, X'099C') İleti tanıtıcısı göstergesi geçersiz.

**MQRC\_OPTIONS\_HATA**

(2046, X'07FE') Seçenekler geçerli değil ya da tutarlı değil.

**MQRC\_STORAGE\_KULLANILAMIYOR**

(2071, X'817 ') Kullanılabilir bellek yetersiz.

**MQRC\_UNEXPECTED\_HATASI**

(2195, X'893 ') Beklenmeyen hata oluştu.

Bu kodlarla ilgili ayrıntılı bilgi için bkz. [İletiler ve neden kodları](#).

**C**

```
MQCRTMH (Hconn, &CrtMsgHOpts, &Hmsg, &CompCode, &Reason);
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
MQHCONN  Hconn;          /* Connection handle */
MQCMHO   CrtMsgHOpts;   /* Options that control the action of MQCRTMH */
MQHMSG   Hmsg;          /* Message handle */
MQLONG   CompCode;     /* Completion code */
MQLONG   Reason;       /* Reason code qualifying CompCode */
```

**COBOL**

```
CALL 'MQCRTMH' USING HCONN, CRTMSGHOPTS, HMSG, COMPCODE, REASON.
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
** Connection handle
01 HCONN      PIC S9(9) BINARY.
** Options that control the action of MQCRTMH
01 CRTMSGHOPTS.
   COPY CMQCMHOV.
** Message handle
01 HMSG      PIC S9(18) BINARY.
** Completion code
01 COMPCODE  PIC S9(9) BINARY.
** Reason code qualifying COMPCODE
01 REASON    PIC S9(9) BINARY.
```

**PL/I**

```
call MQCRTMH (Hconn, CrtMsgHOpts, Hmsg, CompCode, Reason);
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
dcl Hconn      fixed bin(31); /* Connection handle */
dcl CrtMsgHOpts like MQCMHO; /* Options that control the action of MQCRTMH */
dcl Hmsg       fixed bin(63); /* Message handle */
dcl CompCode   fixed bin(31); /* Completion code */
dcl Reason     fixed bin(31); /* Reason code qualifying CompCode */
```

## High Level Assembler

```
CALL MQCRTMH, (HCONN, CRTMSGHOPTS, HMSG, COMPCODE, REASON)
```

Değiştirgeleri aşağıdaki gibi bildirin:

HCONN	DS	F	Connection handle
CRTMSGHOPTS	CMQCMHOA	,	Options that control the action of MQCRTMH
HMSG	DS	D	Message handle
COMPCODE	DS	F	Completion code
REASON	DS	F	Reason code qualifying COMPCODE

## MQCTL-Denetim çağruları

MQCTL çağrısı, geri çağrılar üzerinde denetim işlemlerini gerçekleştirir ve bağlantı için açılan nesne tanıtıcılarını gerçekleştirir.

### Sözdizimi

MQCTL (*Hconn, Operation, ControlOpts, CompCode, Reason*)

### Parametreler

#### Hconn

Tip: MQHCONN-input

Bu tanıtıcı, kuyruk yöneticisiyle olan bağlantıyı gösterir. The value of *Hconn* was returned by a previous MQCONN or MQCONNX call.

On z/OS for CICS applications the MQCONN call can be omitted, and you can specify the following special value for *Hconn* :

#### MQHC\_DEF\_HCONN

Varsayılan bağlantı tanıtıcısı.

#### İşlem

Tip: MQHOME-girişi

Belirtilen nesne tanıtıcısı için tanımlanan geri bildirme işleminde işlem yapılıyor. Aşağıdaki seçeneklerden birini ve yalnızca birini belirtmeniz gerekir:

#### MQOP\_START

Belirtilen bağlantı tanıtıcısı için, tanımlı tüm ileti tüketicisi işlevleri için iletilerin tütütmesini başlatın.

Geri çağrılar, sistem tarafından başlatılan bir iş parçacığının üzerinde çalışır; bu, uygulama iş parçacıklarının hiçbirinden farklıdır.

Bu işlem, sistem tarafından sağlanan bağlantı tanıtıcısı denetimini verir. Tüketici iş parçacığı dışındaki bir iş parçacığı tarafından yayınlanabilen yalnızca MQI çağruları şunlardır:

- MQOP\_STOP İşlemi ile MQCTL
- MQOP\_SUSPEND İşlemi ile MQCTL
- MQDISC-HConn bağlantısını kesmeden önce MQOP\_STOP işlemi ile MQCTL ' yi gerçekleştirir.

Bağlantı tanıtıcısı başlatıldığında ve çağrı bir ileti tüketicisi işlevinden kaynaklanmazsa, bir IBM MQ API çağrısı yayınlanırsa MQRC\_HCONN\_ASYNC\_ETKİN değeri döndürülür.

Bir ileti tüketicisi, MQCBCT\_START\_CALL işlemi sırasında bağlantıyı durdurursa, MQCTL çağrısı, MQRC\_CONNECTION\_STOP hata neden koduyla geri döner.

Bu, bir tüketici işlevinde yayınlanabilir. Geri çağırma yordamıyla aynı bağlantı için, tek amacı daha önce verilen bir MQOP\_STOP işlemi iptal etmek.

Bu seçenek aşağıdaki ortamlarda desteklenmez: z/OS üzerinde CICS ya da uygulama iş parçacıklı bir IBM MQ kitaplığından bağlıysa.

### **MQOP\_START\_SABE**

Belirtilen bağlantı tanıtıcısı için, tanımlı tüm ileti tüketicisi işlevleri için iletilerin tütütmesini başlatın.

İleti tüketicileri aynı iş parçacığı üzerinde çalışır ve denetim MQCTL ' nin çağrısına döndürülünceye kadar döndürülmez:

- MQCTL MQOP\_STOP ya da MQOP\_SUSPEND işlemleri tarafından serbest bırakıldı ya da
- Tüm tüketici yordamları kayıttan kaldırıldı ya da askıya alındı.

Tüm tüketiciler kayıttan kaldırılırsa ya da askıya alınırsa, örtük bir MQOP\_STOP işlemi yayınlanır.

Bu seçenek, yürürlükteki bağlantı tanıtıcısı ya da başka bir bağlantı tanıtıcısı için, geri bildirme yordamından kullanılamaz. Arama girişiminde bulunulursa, MQRC\_ENVIRONMENT\_ERROR ile dönüş işlemi döndürülür.

Bir MQOP\_START\_WAIT işlemi sırasında herhangi bir zamanda kayıtlı olmayan, askıya alınmış olmayan tüketicilere çağrı, bir MQRC\_NO\_CALLBACKS\_ETKİN bir neden kodunda başarısız olur.

Bir MQOP\_START\_WEKE işlemi sırasında, bağlantı askıya alınır, MQCTL çağrısı bir uyarı neden kodu (MQRC\_CONNECTION\_SUSPENDED) döndürür; bağlantı 'başlatıldı' kalır.

Uygulama, MQOP\_STOP ya da MQOP\_RESUME komutunu yayınlamayı seçebilir. Bu örnekte, MQOP\_RESUME işlem öbekleri.

Tek bir iş parçacıklı istemcide bu seçenek desteklenmez.

### **MQOP\_STOP**

İletilerin tüketmesini durdurun ve tüm tüketicilerin bu seçenek tamamlanmadan operasyonlarını tamamlaması için bekleyin. Bu işlem bağlantı tanıtıcısını serbest bırakır.

Bu seçenek, bir geri çağrı yordamından yayınlandıysa, yordam çıkılıncaya kadar bu seçenek yürürlüğe girmez. Zaten okunan iletiler için tüketici yordamlarından sonra ve geri çağırma yordamlarıyla çağrılan çağrılar (istendiye) sonrasında, başka ileti tüketici yordamı çağrılanmaz.

Bir geri çağrı yordamı dışında yayınlanırsa, denetim, önceden okunan iletiler için tüketici rutinleri tamamlanincaya kadar ve çağrılar durdurulduktan sonra (istendiye) çağrılara geri dönmez (istenirse). Ancak geri çağrılar, kayıtlı olarak kalmaya devam ediyor.

Bu işlevin önünde okuma yazma hiçbir etkisi yoktur. Teslim edilecek başka iletilerin olup olmadığını belirlemek için geri bildirme işlevi içinden, tüketicinin MQclose (MQCO QUIESCE) komutunu çalıştırdığından emin olmalısınız.

### **MQOP\_SUSPEND**

İletilerin tüketilmesini duraklatın. Bu işlem bağlantı tanıtıcısını serbest bırakır.

Bu, uygulamaya ilişkin iletilerin okunmasındaki okuma üzerinde herhangi bir etkiye sahip değildir. Uzun süre iletileri tüketmeyi durdurmak istiyorsanız, kuyruğu kapatmayı ve tüketimin devam etmesi durumunda yeniden açmayı düşünebilirsiniz.

Bir geri çağrı yordamından yayınlanırsa, yordam çıkışlarına kadar yürürlüğe girmez. Yürürlükteki yordam çıkışlarından sonra ileti tüketici yordamlarından başka bir yordam çağrılacak.

Bir geri çağırma dışında yayınlanırsa, denetim, yürürlükteki tüketici yordamı tamamlanincaya kadar arayanın çağrısına dönmez ve daha fazla çağrılanmaz.

### **MQOP\_RESUME**

İletilerin tüketilmesine devam edin.

Bu seçenek genellikle ana uygulama iş parçacığından verilir, ancak aynı yordamda yayınlanan daha önceki bir askıya alma isteğini iptal etmek için bir geri çağırma yordamından da kullanılabilir.

MQOP\_RESUME bir MQOP\_START\_SOLT işlemi sürdürmek için kullanıldıysa, işlem öbekleri.

## ControlOpts

Tip: MQCTLO-input

MQCTL ' nin işlemini denetleyen seçenekler

Yapıyla ilgili ayrıntılar için [MQCTLO](#) başlıklı konuya bakın.

## CompCode

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Tamamlanma kodu; bu kod aşağıdakilerden biridir:

### MQCC\_OK

İşlem başarıyla tamamlandı.

### MQCC\_UYARI

Uyarı (kısmi tamamlama).

### MQCC\_FAILED

Arama başarısız oldu.

## Neden

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

*CompCode* MQCC\_OK ise:

### MQRC\_NONE

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_FAILED ise:

### MQRC\_ADAPTER\_CONV\_LOAD\_ERROR

(2133, X'855 ') Veri dönüştürme hizmetleri modüllerinin yüklenmesi gerçekleştirilemiyor.

### MQRC\_ADAPTER\_NOT\_AVAM

(2204, X'89C') Bağdaştırıcı kullanılmıyor.

### MQRC\_ADAPTER\_SERV\_LOAD\_ERROR

(2130, X'852 ') Bağdaştırıcı hizmeti modülü yüklenemiyor.

### MQRC\_API\_EXIT\_ERROR

(2374, X' 946 ') API çıkışı başarısız oldu.

### MQRC\_API\_EXIT\_LOAD\_ERROR

(2183, X'887 ') API çıkışı yüklenemiyor.

### MQRC\_ASID\_MISSMACH

(2157, X'86D') Birincil ve ana sunucu ASID değerleri farklı.

### MQRC\_BUFFER\_LENGTH\_ERROR

(2005, X'7D5') Arabellek uzunluğu parametresi geçerli değil.

### MQRC\_CALLBACK\_LINK\_ERROR

(2487, X'9B7') Geri çağırma yordamı çağırılmıyor

### MQRC\_CALLBACK\_NOT\_KAYDEDILDI

(2448, X' 990 ') Kayıtlı bir geri çağırma olmadığı için Deregister, Suspend ya da Sürdürme işlemi gerçekleştirilemiyor

### MQRC\_CALLBACK\_ROUTINE\_ERROR

(2486, X'9B6') Ya CallbackFunction , hem de CallbackName bir MQOP\_REGISTER çağrısında belirtildi.

Ya da CallbackFunction ya da CallbackName belirtildi, ancak kayıtlı geri bildirme işleviyle eşleşmedi.

### MQRC\_CALLBACK\_TYPE\_ERROR

(2483, X'9B3') Yanlış CallBackTip alanı.

### MQRC\_CALL\_IN\_PROGRESS

(2219, X'8AB') Önceki arama tamamlanmadan önce girilen MQI çağrısı.



**MQRC\_CBD\_ERROR**

(2444, X'98C') Seçenek bloğu yanlış.

**MQRC\_CBD\_OPTIONS\_ERROR**

(2484, X'9B4') MQCBD seçenekleri alanı yanlış.

**MQRC\_CICS\_WAIT\_FAILED**

(2140, X'85C') Bekleme isteği CICStarafından reddedildi.

**MQRC\_CONNECTION\_BROKEN**

(2009, X'7D9') Kuyruk yöneticisine yönelik bağlantı kaybedildi.

**MQRC\_CONNECTION\_NOT\_YETKILI**

(2217, X'8A9') Bağlantı için yetkili değil.

**MQRC\_CONNECTION\_QUIESCING**

(2202, X'89A') Bağlantı susturulmuş durumda.

**MQRC\_CONNECTION\_DURATION**

(2203, X'89B') Bağlantı sona erdiriliyor.

**MQRC\_CORREL\_ID\_ERROR**

(2207, X'89F') İntilendirme tanıtıcısı hatası.

**MQRC\_ENVIRONMENT\_ERROR**

(2012, X'7DC') Ortamda arama geçerli değil.

**MQRC\_FUNCTION\_NOT\_SUPPORTUL**

(2298, X'8FA') İstenen işlev geçerli ortamda yok.

**MQRC\_GET\_INHIBITED**

(2016, X'7E0') Kuyruğun engellenmesini sağlar.

**MQRC\_GLOBAL\_UOW\_CONFLICTIONS**

(2351, X'92F') Genel iş çakışmaları birimleri.

**MQRC\_GMO\_ERROR**

(2186, X'88A') Al-ileti seçenekleri yapısı geçerli değil.

**MQRC\_ANDLE\_IN\_USE\_FOR\_UOW**

(2353, X' 931 ') Genel iş birimi için kullanılan tanıtıcı.

**MQRC\_HCONN\_ERROR**

(2018, X'7E2') Bağlantı tanıtıcısı geçerli değil.

**MQRC\_HOBJ\_ERROR**

(2019, X'7E3') Nesne tanıtıcısı geçerli değil.

**MQRC\_INCONTINUED\_BROWSE**

(2259, X'8D3') Tutarsız göz atma belirtimi.

**MQRC\_INCONSISTENT\_UOW**

(2244, X'8C5') Tutarsız iş birliği-iş belirtimi.

**MQRC\_INVALID\_MSG\_UNDER\_CURSOR**

(2246, X'8C6') İmleç altındaki ileti alma için geçerli değil.

**MQRC\_LOCAL\_UOW\_CONFLICTID**

(2352, X' 930 ') Genel iş birimi yerel iş birimi ile çakışıyor.

**MQRC\_MATCH\_OPTIONS\_ERROR**

(2247, X'8C7') Eşleştirme seçenekleri geçerli değil.

**MQRC\_MAX\_MSG\_LENGTH\_ERROR**

(2485, X'9B5') Yanlış MaxMsgUzunluk alanı

**MQRC\_MD\_ERROR**

(2026, X'7EA') İleti tanımlayıcısı geçerli değil.

**MQRC\_MODULE\_ENTRY\_NOT\_FOUND**

(2497, X'9C1') Belirtilen işlev giriş noktası modülde bulunamadı.

**MQRC\_MODULE\_INVALID**

(2496, X'9C0') Modül bulundu, ancak yanlış tipte (32 bit/64 bit) ya da geçerli bir dll değil.

**MQRD\_MODULE\_NOT\_FOUND**  
(2495, X'9BF') Modül, arama yolunda bulunamadı ya da yükleme yetkisine sahip değil.

**MQRD\_MSG\_ID\_ERROR**  
(2206, X'89E') İleti-tanıtıcısı hatası.

**MQRD\_MSG\_SEQ\_NUMBER\_ERROR**  
(2250, X'8CA') İleti sıra numarası geçerli değil.

**MQRD\_MSG\_TOKEN\_HATASI**  
(2331, X'91B') İleti simgesinin kullanımı geçerli değil.

**MQRD\_NOT\_OPEN\_FOR\_BROWSE**  
(2036, X'7F4') Kuyruk göz atma için açık değil.

**MQRD\_NOT\_OPEN\_FOR\_INPUT**  
(2037, X'7F5') Kuyruk giriş için açık değil.

**MQRD\_OBJECT\_CHANGED**  
(2041, X'7F9') Nesne tanımlaması açıldığından beri değiştirildi.

**MQRD\_OBJECT\_ZARAR**  
(2101, X'835 ') Nesne zarar gördü.

**MQRD\_OPERATION\_HATASI**  
(2488, X'9B8') API Çağrısında Hatalı İşlem kodu

**MQRD\_OPTIONS\_ERROR**  
(2046, X'7FE') Seçenekler geçerli değil ya da tutarlı değil.

**MQRD\_PAGEES\_HATASI**  
(2193, X'891 ') Sayfa kümesi veri kümesine erişilirken hata oluştu.

**MQRD\_Q\_DELETED**  
(2052, X'804 ') Kuyruk silinmiştir.

**MQRD\_Q\_INDEX\_TYPE\_ERROR**  
(2394, X'95A') Kuyruk yanlış dizin tipi içeriyor.

**MQRD\_Q\_MGR\_NAME\_ERROR**  
(2058, X'80A') Kuyruk yöneticisi adı geçerli değil ya da bilinmiyor.

**MQRD\_Q\_MGR\_NOT\_VAR**  
(2059, X'80B') Kuyruk yöneticisi bağlantı için kullanılabilir değil.

**MQRD\_Q\_MGR QUIESCING**  
(2161, X'871 ') Kuyruk yöneticisi susturuyor.

**MQRD\_Q\_MGR\_STOPPING**  
(2162, X'872 ') Kuyruk yöneticisi sona erdiriliyor.

**MQRD\_RESOURCE\_SORUNU**  
(2102, X'836 ') Yetersiz sistem kaynakları var.

**MQRD\_SIGNAL\_INSTANT**  
(2069, X'815 ') Bu tutamaç için bekleyen sinyal.

**MQRD\_STORAGE\_NOT\_AVAM**  
(2071, X'817 ') Kullanılabilir bellek yetersiz.

**MQRD\_SUPPRESSED\_BY\_EXIT**  
(2109, X'83D') Çıkış programı tarafından çağrı engellendi.

**MQRD\_SYNCPOINT\_NOT\_AVAM**  
(2072, X'818 ') Syncpoint desteği yok.

**MQRD\_UNEXPECTED\_ERROR**  
(2195, X'893 ') Beklenmeyen bir hata oluştu.

**MQRD\_UOWENLISTMENT\_ERROR**  
(2354, X' 932 ') Genel iş birimi içindeki bilgiler başarısız oldu.

**MQRD\_UOW\_MIX\_NOT\_SUPPORTANT**  
(2355, X' 933 ') İşlerin birim içi çağrıları karışımı desteklenmiyor.

### **MQRC\_UOW\_NOT\_VAR**

(2255, X'8CF') Kuyruk yöneticisinin kullanması için kullanılabilir iş birimi.

### **MQRC\_WAIT\_INTERVAL\_ERROR**

(2090, X'82A') MQGMO' da bekleme aralığı geçerli değil.

### **MQRC\_WRONG\_GMO\_SÜRÜMÜ**



(2256, X'8D0') MQGMO' nun yanlış sürümü sağlandı.

### **MQRC\_WRONG\_MD\_VERSION**

(2257, X'8D1') MQMD' nin yanlış sürümü sağlandı.

Bu kodlarla ilgili ayrıntılı bilgi için [İletiler ve neden kodları](#) başlıklı konuya bakın.

## **Kullanım notları**

1. Geri bildirme yordamları, çağırdıkları tüm hizmetlerden gelen yanıtları denetlememeli ve yordam çözülemeyen bir koşul saptarsa, geri çağırma yordamlarına yinelenen çağrılar önlemek için bir MQCB MQOP\_DEREGITER komutu yayınlamalıdır.
2. XA Transaction Manager 'ın genel işlemleri yönetmekte olduğu bir uygulamada zamanuyumsuz olarak tüketmeyi kullanıyorsanız, IBM MQ ile ilgili güncellemeler de içinde olmak üzere, aşağıdaki ek noktaları göz önünde bulundurmanız gerekir:
  - a. It is not valid to call MQCTL(MQOP\_START) for an **HConn**, after it has been created, after calling **xa\_open**.  
Bunun nedeni, **HConn** ' un bir XA bağlamına eklendiği ve bu nedenle zamanuyumsuz tükeme mekanizması tarafından kullanılan ayrı iş parçacığının ya da iş parçacıklarının üzerine erişilememesi.
  - b. Bu senaryoda MQCTL ' yi (MQOP\_START) çağırırsanız, çağrı neden kodu MQRC\_ASYNC\_XA\_CONFLICE (2350) koduyla başarısız olur.
  - c. It is valid to call MQCTL(MQOP\_START\_WAIT) for an **HConn**, after it has been created, after calling **xa\_open**.  
Bunun nedeni, zamanuyumsuz tükeme mekanizmasının başlatılması yönteminin, **HConn** için, MQCTL çağrısının yapıldığı iş parçacığın üzerinde çalışmasına neden olur. Bu nedenle, **HConn** ile iş parçacığı arasındaki bağlantı kaybolmaz.
3.  On z/OS, when Operation is MQOP\_START:
  - Zamanuyumsuz geri bildirme yordamlarını kullanan programların z/OS UNIX System Services (z/OS UNIX) olanağını kullanma yetkisi olmalıdır.
  - Language Environment (LE) programs which use asynchronous callback routines must use the LE runtime option POSIX(ON).
  - Zamanuyumsuz geri bildirme yordamlarını kullanan LE dışı programlar, z/OS UNIX pthread\_create arabirimini kullanmamalıdır (çağrılabilir hizmet BPX1PTC).
4.  MQCTL, IMS bağdaştırıcısı içinde desteklenmiyor.

**Not:** CICS' ta MQOP\_START desteklenmez. Bunun yerine, MQOPT\_START\_WAIT işlev çağrısını kullanın.

## **C çağırma**

```
MQCTL (Hconn, Operation, &ControlOpts, &CompCode, &Reason)
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
MQHCONN Hconn;          /* Connection handle */
MQLONG  Operation;    /* Operation being processed */
MQCTL0  ControlOpts   /* Options that control the action of MQCTL */
```

```
MQLONG  CompCode;      /* Completion code */
MQLONG  Reason;        /* Reason code qualifying CompCode */
```

## Cobol çağırma

```
CALL 'MQCTL' USING HCONN, OPERATION, CTLOPTS, COMPCODE, REASON.
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
** Connection handle
01 HCONN PIC S9(9) BINARY.
** Operation
01 OPERATION PIC S9(9) BINARY.
** Control Options
01 CTLOPTS.
   COPY CMQCTLOV.
** Completion code
01 COMPCODE PIC S9(9) BINARY.
** Reason code qualifying COMPCODE
01 REASON PIC S9(9) BINARY.
```

## PL/I çağırımı

```
call MQCTL(Hconn, Operation, CtlOpts, CompCode, Reason)
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
dcl Hconn          fixed bin(31); /* Connection handle */
dcl Operation      fixed bin(31); /* Operation */
dcl CtlOpts like   MQCTLO;        /* Options that control the action of MQCTL */
dcl CompCode       fixed bin(31); /* Completion code */
dcl Reason         fixed bin(31); /* Reason code qualifying CompCode */
```

## MQDISC-Bağlantı kesme kuyruğu yöneticisi

MQDISC çağırısı, kuyruk yöneticisi ile uygulama programı arasındaki bağlantıyı keser ve MQCONN ya da MQCONNX çağırısının tersi olur.

- z/OS' ta, zamanuyumsuz ileti tüketimi, olay işleme ya da geri çağırma kullanan tüm uygulamalar, sona erdirilmeden önce ana denetim iş parçacığının bir MQDISC çağırısı yayınlaması gerekir. Daha fazla ayrıntı için bkz. [Zamanuyumsuz tüketim IBM MQ iletileri](#) .
- z/OSüzerinde, CICS uygulamalarının kuyruk yöneticisinden bağlantıyı kesmek için bu çağırısı yayınlamaya gerek yoktur.

Bir CICS uygulaması bu çağırısı gerçekleştirirse, aşağıdaki belirtilerden birini belirterek, daha önceki bir MQCONNX çağırısı yapılmadıkça hiçbir etkisi olmaz:

```
MQCNO_SERIALIZE_CONN_TAG_Q_MGR
MQCNO_SERIALIZE_CONN_TAG_QSG
MQCNO_RESTRICT_CONN_TAG_Q_MGR ya da
MQCNO_RESTRICT_CONN_TAG_QSG
```

seçenekler; bu durumda, açık olan tüm nesne tanıtıcıları kapanır.

## Sözdizimi

MQDISC (*Hconn, CompCode, Reason*)

## Parametreler

### Hconn

Tip: MQHCONN-giriş/çıkış

Bu tanıttıcı, kuyruk yöneticisiyle olan bağlantıyı gösterir. The value of *Hconn* was returned by a previous MQCONN or MQCONNX call.

On z/OS for CICS applications you can omit the MQCONN call, and specify the following value for *Hconn* :

#### **MQHC\_DEF\_HCONN**

Varsayılan bağlantı tanıttıcısı.

Çağrı başarıyla tamamlandığında, kuyruk yöneticisi *Hconn* değerini, ortam için geçerli bir tanıttıcı olmayan bir değere ayarlar. Bu değer:

#### **MQHC\_UNUSABLE\_HCONN**

Kullanılmayan bağlantı tanıttıcısı.

z/OSüzerinde, *Hconn* tanımsız bir değere ayarlıdır.

### CompCode

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Tamamlanma kodu; bu kod aşağıdaki kodlardan biridir:

#### **MQCC\_OK**

İşlem başarıyla tamamlandı.

#### **MQCC\_UYARI**

Uyarı (kısmi tamamlama).

#### **MQCC\_FAILED**

Arama başarısız oldu.

### Neden

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

*CompCode* MQCC\_OK ise:

#### **MQRC\_NONE**

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_UYARI ise:

#### **MQRC\_BACKED\_OUT**

(2003, X'7D3') İş birimi yedeklendi.

#### **MQRC\_CONN\_TAG\_NOT\_SERBEST**

(2344, X'928 ') Bağlantı etiketi serbest bırakılmadı.

#### **MQRC\_OUTCOME\_PENDING**

(2124, X'84C') Kesinleştirme işleminin sonucu beklemede.

*CompCode* MQCC\_FAILED ise:

#### **MQRC\_ADAPTER\_DISC\_LOAD\_ERROR**

(2138, X'85A') Bağdaştırıcı bağlantı kesme modülü yüklenemiyor.

#### **MQRC\_ADAPTER\_NOT\_AVAM**

(2204, X'89C') Bağdaştırıcı kullanılmıyor.

#### **MQRC\_ADAPTER\_SERV\_LOAD\_ERROR**

(2130, X'852 ') Bağdaştırıcı hizmeti modülü yüklenemiyor.

#### **MQRC\_API\_EXIT\_ERROR**

(2374, X'946 ') API çıkışı başarısız oldu.

#### **MQRC\_API\_EXIT\_INIT\_ERROR**

(2375, X'947 ') API çıkış başlatma başarısız oldu.

**MQRC\_API\_EXIT\_TERM\_ERROR**

(2376, X'948 ') API çıkış sonlandırması başarısız oldu.

**MQRC\_ASID\_MISSMACH**

(2157, X'86D') Birincil ve ana sunucu ASID değerleri farklı.

**MQRC\_CALL\_IN\_PROGRESS**

(2219, X'8AB') Önceki arama tamamlanmadan önce girilen MQI çağrısı.

**MQRC\_CONNECTION\_BROKEN**

(2009, X'7D9') Kuyruk yöneticisine yönelik bağlantı kaybedildi.

**MQRC\_CONNECTION\_DURATION**

(2203, X'89B') Bağlantı sona erdiriliyor.

**MQRC\_HCONN\_ERROR**

(2018, X'7E2') Bağlantı tanıtıcısı geçerli değil.

**MQRC\_OUTCOME\_MIXED**

(2123, X'84B') Kesinleştirme ya da geri dönüş işlemi sonucu karışık.

**MQRC\_PAGEES\_HATASI**

(2193, X'891 ') Sayfa kümesi veri kümesine erişilirken hata oluştu.

**MQRC\_Q\_MGR\_NAME\_ERROR**

(2058, X'80A') Kuyruk yöneticisi adı geçerli değil ya da bilinmiyor.

**MQRC\_Q\_MGR\_NOT\_VAR**

(2059, X'80B') Kuyruk yöneticisi bağlantı için kullanılabilir değil.

**MQRC\_Q\_MGR\_STOPPING**

(2162, X'872 ') Kuyruk yöneticisi sona erdiriliyor.

**MQRC\_RESOURCE\_SORUNU**

(2102, X'836 ') Yetersiz sistem kaynakları var.

**MQRC\_STORAGE\_NOT\_AVAM**

(2071, X'817 ') Kullanılabilir bellek yetersiz.

**MQRC\_UNEXPECTED\_ERROR**

(2195, X'893 ') Beklenmeyen bir hata oluştu.

Bu kodlarla ilgili ayrıntılı bilgi için [İletiler ve neden kodları](#) başlıklı konuya bakın.

**Kullanım notları**

- Bağlantı o bağlantı altında açık olan nesnelere varsa, MQDISC çağrısı yayınlanırsa, kuyruk yöneticisi bu nesnelere kapatır ve kapama seçenekleri MQCO\_NONE olarak ayarlanmış olur.
- Uygulama bir iş birimindeki kesinleştirilmemiş değişikliklerle sona ererse, bu değişikliklerin yok edilmesi, uygulamanın sona erme şekline bağlıdır:
  - Uygulama, bitmeden önce MQDISC çağrısını yayınlarsa:
    - Kuyruk yöneticisi-eşgüdümlü iş birimi için, kuyruk yöneticisi uygulama adına MQCMIT çağrısını yayınlar. İş birimi olanaklıysa, kesinleştirilir ve desteklenmiyorsa yedeklenir.
    - Dışarıdan eşgüdümlü bir iş birimi için, iş biriminin durumunda bir değişiklik yoktur; ancak, kuyruk yöneticisi genellikle iş birimi eşgüdümcüsü tarafından istendiğinde iş biriminin kesinleştirilip kesinleştirildiğini gösterir.

z/OS, CICS, IMS (toplu DL/1 programlarından farklı) ve RRS uygulamaları bu şekilde.
  - Uygulama olağan bir şekilde sona ererse, ancak MQDISC çağrısını vermeden sona erdirilirse, işlem ortama bağlıdır:
    - On z/OS, except for MQ Java or MQ JMS applications, the actions described in note 2a occur.
    - In all other cases, the actions described in note 2c occur.

Ortamlar arasındaki farklar nedeniyle, bağlantı noktasına dönmek istediğiniz uygulamaların, iş birimini sona erdirmeden önce kesinleştirmelerini ya da yedeklemenizi sağlar.

c. Uygulama, MQDISC çağrısını yayınlamadan *anormal* ' i sona erdirirse, iş birimi geriletilir.

3. z/OS' ta aşağıdaki noktalar geçerlidir:

- CICS applications do not have to issue the MQDISC call to disconnect from the queue manager, because the CICS system itself connects to the queue manager, and the MQDISC call has no effect on this connection.
- CICS, IMS (toplu DL/1 programları dışında) ve RRS uygulamaları, bir dış iş birimi eşgüdümçüsü tarafından koordine edilen iş birimlerini kullanır. Sonuç olarak, MQDISC çağrısı, çağrı yayınlandığında var olan iş biriminin durumunu (varsa) etkilemez.

Ancak, MQDISC çağrısı *yaptığı* , uygulama tarafından yayınlanan bir önceki MQCONNX çağrısıyla bağlantıyla ilişkili *ConnTag* bağlantı etiketinin kullanımının sona erdiğini gösterir. MQDISC çağrısı yayınlandığında bağlantı etiketine başvuruda bulunan etkin bir iş birimi varsa, çağrı tamamlanma kodu MQCC\_UYARY ile tamamlanır ve neden kodu MQRC\_CONN\_TAG\_NOT\_SCREEN ile tamamlanır. Dış birim iş birimi eşgüdümçüsü iş birimini çözünceye kadar bağlantı etiketi yeniden kullanılmak üzere kullanılamaz.

**Not:** CICS' ta MQOP\_START desteklenmez. Bunun yerine, MQOPT\_START\_WAIT işlev çağrısını kullanın.

## C çağırma

```
MQDISC (&Hconn, &CompCode, &Reason);
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
MQHCONN Hconn;      /* Connection handle */
MQLONG  CompCode;   /* Completion code */
MQLONG  Reason;     /* Reason code qualifying CompCode */
```

## Cobol çağırma

```
CALL 'MQDISC' USING HCONN, COMPCODE, REASON.
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
** Connection handle
01 HCONN PIC S9(9) BINARY.
** Completion code
01 COMPCODE PIC S9(9) BINARY.
** Reason code qualifying COMPCODE
01 REASON PIC S9(9) BINARY.
```

## PL/I çağırımı

```
call MQDISC (Hconn, CompCode, Reason);
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
dcl Hconn      fixed bin(31); /* Connection handle */
dcl CompCode   fixed bin(31); /* Completion code */
dcl Reason     fixed bin(31); /* Reason code qualifying CompCode */
```

## System/390 çevirici çağırma

```
CALL MQDISC, (HCONN, COMPCODE, REASON)
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
HCONN      DS  F  Connection handle
COMPCODE   DS  F  Completion code
REASON     DS  F  Reason code qualifying COMPCODE
```

## Visual Basic çağırısı

```
MQDISC Hconn, CompCode, Reason
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
Dim Hconn      As Long 'Connection handle'
Dim CompCode   As Long 'Completion code'
Dim Reason     As Long 'Reason code qualifying CompCode'
```

## MQDLTMH-İleti silme tanıtıcısı

MQDLTMH çağırısı bir ileti tanıtıcısını siler ve MQCRTMH çağırısının tersidir.

### Sözdizimi

```
MQDLTMH (Hconn, Hmsg, DltMsgHOpts, CompCode, Reason)
```

### Parametreler

#### Hconn.

Tip: MQHCONN-giriş

Bu tanıtıcı, kuyruk yöneticisine yönelik bağlantıyı gösterir.

Değer, **Hmsg** değiştirgesinde belirtilen ileti tanıtıcısını yaratmak için kullanılan bağlantı tanıtıcısıyla eşleşmelidir.

İleti tanıtıcısı MQHC\_UNASSOCIATED\_HCONN kullanılarak yaratıldıysa, ileti tanıtıcısını silme iş parçacığında geçerli bir bağlantı kurulmalıdır; tersi durumda, çağrı MQRC\_CONNECTION\_BROKEN ile başarısız olur.

#### İlt

Tip: MQHMSG-giriş/çıkış

Bu, silinecek ileti tanıtıcısıdır. Değer, önceki bir MQCRTMH çağırısı tarafından döndürüldü.

Çağrı başarıyla tamamlandığında, tanıtıcı ortam için geçersiz bir değere ayarlanır. Bu değer:

#### **MQHM\_HABLE\_HMSG**

Kullanılmayan ileti tanıtıcısı.

Aynı ileti tanıtıcısı iletilen başka bir IBM MQ çağırısı devam ediyorsa, ileti tanıtıcısı silinemez.

#### DltMsgHOpts

Tip: MQDMHO-giriş

Ayrıntılar için bkz. [MQDMHO](#) .

#### CompCode

Tip: MQLONG-output



Tamamlanma kodu; aşağıdakilerden biridir:

**MQCC\_OK**

Başarıyla tamamlandı.

**MQCC\_FAILED**

Arama başarısız oldu.

**Neden**

Tip: MQLONG-output

*CompCode* MQCC\_OK ise:

**MQRC\_NONE**

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_FAILED ise:

**MQRC\_ADAPTER\_NOT\_KULLANILAMIYOR**

(2204, X'089C') Bağdaştırıcı kullanılamıyor.

**MQRC\_ADAPTER\_SERV\_LOAD\_ERROR (UYGULAMA\_HATASI)**

(2130, X'852 ') Bağdaştırıcı hizmet modülü yüklenemiyor.

**MQRC\_ASID\_MISMATCH**

(2157, X'86D') Birincil ve ana ASID' ler farklıdır.

**MQRC\_CALL\_IN\_PROGRESS**

(2219, X'08AB') Önceki arama tamamlanmadan önce girilen MQI çağrısı.

**MQRC\_CONNECTION\_BROKEN**

(2009, X'07D9') Kuyruk yöneticisine bağlantı kaybedildi.

**MQRC\_DMHO\_ERROR**

(2462, X'099E') İleti silme tanıtıcısı seçenekleri yapısı geçersiz.

**MQRC\_HMSG\_HATA**

(2460, X'099C') İleti tanıtıcısı göstergesi geçersiz.

**MQRC\_MSG\_HANDLE\_IN\_USE**

(2499, X'09C3') İleti tanıtıcısı kullanımda.

**MQRC\_OPTIONS\_HATA**

(2046, X'07FE') Seçenekler geçerli değil ya da tutarlı değil.

**MQRC\_STORAGE\_KULLANILAMIYOR**

(2071, X'817 ') Kullanılabilir bellek yetersiz.

**MQRC\_UNEXPECTED\_HATASI**

(2195, X'893 ') Beklenmeyen hata oluştu.

Bu kodlarla ilgili ayrıntılı bilgi için bkz. [İletiler ve neden kodları](#).

## C çağrısı

```
MQDLTMH (Hconn, &Hmsg, &DltMsgHOpts, &CompCode, &Reason);
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
MQHCONN  Hconn;          /* Connection handle */
MQHMSG    Hmsg;          /* Message handle */
MQDMHO    DltMsgHOpts; /* Options that control the action of MQDLTMH */
MQLONG    CompCode;     /* Completion code */
MQLONG    Reason;       /* Reason code qualifying CompCode */
```

## COBOL çağırısı

```
CALL 'MQDLTMH' USING HCONN, HMSG, DLTMSGHOPTS, COMPCODE, REASON.
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
** Connection handle
01 HCONN PIC S9(9) BINARY.

** Options that control the action of MQDLTMH
01 DLTMSGHOPTS.
COPY CMQDMHOL.

** Completion code
01 COMPCODE PIC S9(9) BINARY.

** Reason code qualifying COMPCODE
01 REASON PIC S9(9) BINARY.
```

## PL/I çağırısı

```
call MQDLTMH (Hconn, Hmsg, DltMsgHOpts, CompCode, Reason);
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
dcl Hconn          /* Connection handle */
dcl Hmsg           /* Message handle */
dcl DltMsgHOpts   like MQDMHO; /* Options that control the action of MQDLTMH */
dcl CompCode      /* Completion code */
dcl Reason        /* Reason code qualifying CompCode */
```

## High Level Assembler (Yüksek Düzeyli çevirici) çağırısı

```
CALL MQDLTMH, (HCONN, HMSG, DLTMSGHOPTS, COMPCODE, REASON)
```

Değiştirgeleri aşağıdaki gibi bildirin:

HCONN	DS	F	Connection handle
HMSG	DS	D	Message handle
DLTMSGHOPTS	CMQDMHOA	,	Options that control the action of MQDLTMH
COMPCODE	DS	F	Completion code
REASON	DS	F	Reason code qualifying COMPCODE

## MQDLTMP-İleti özelliğini sil

MQDLTMP çağırısı bir özelliği ileti tanıtıcısından siler ve MQSETMP çağırısının tersidir.

### Sözdizimi

MQDLTMP (*Hconn, Hmsg, DltPropOpts, Name, CompCode, Reason*)

### Parametreler

#### Hconn.

Tip: MQHCONN-giriş

Bu tanıtıcı, kuyruk yöneticisine yönelik bağlantıyı gösterir. Değer, **Hmsg** değiştirgesinde belirtilen ileti tanıtıcısını yaratmak için kullanılan bağlantı tanıtıcısıyla eşleşmelidir.

İleti tanıtıcısı MQHC\_UNASSOCIATED\_HCONN kullanılarak yaratıldıysa, ileti tanıtıcısını silen iş parçacığında geçerli bir bağlantı kurulmalıdır; tersi durumda, çağrı MQRC\_CONNECTION\_BROKEN ile başarısız olur.

### İlt

Tip: MQHMSG-giriş

Bu, silinecek özelliği içeren ileti tanıtıcısıdır. Değer, önceki bir MQCRTMH çağrısı tarafından döndürüldü.

### DltProp(DltProp) Seçmeler

Tip: MQDMPO-giriş

Ayrıntılar için [MQDMPO](#) veri tipine bakın.

### Ad

Tip: MQCHARV-giriş

Silinecek özelliğin adı. Özellik adlarıyla ilgili ek bilgi için [Özellik adları](#) konusuna bakın.

Özellik adında genel arama karakterlerine izin verilmez.

### CompCode

Tip: MQLONG-output

Tamamlanma kodu; aşağıdakilerden biridir:

#### **MQCC\_OK**

Başarıyla tamamlandı.

#### **MQCC\_UYARISI**

Uyarı (kısmi tamamlama).

#### **MQCC\_FAILED**

Arama başarısız oldu.

### Neden

Tip: MQLONG-output

*CompCode* MQCC\_OK ise:

#### **MQRC\_NONE**

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_WARNING ise:

#### **MQRC\_PROPERTY\_NOT\_VAR**

(2471, X'09A7') Özellik kullanılamıyor.

#### **MQRC\_RFH\_FORMAT\_HATA**

(2421, X'0975 ') Özellikleri içeren bir MQRFH2 klasörü ayrıştırılmadı.

*CompCode* MQCC\_FAILED ise:

#### **MQRC\_ADAPTER\_NOT\_KULLANILAMIYOR**

(2204, X'089C') Bağdaştırıcı kullanılamıyor.

#### **MQRC\_ADAPTER\_SERV\_LOAD\_ERROR (UYGULAMA\_HATASI)**

(2130, X'0852 ') Bağdaştırıcı hizmet modülü yüklenemiyor.

#### **MQRC\_ASID\_MISMATCH**

(2157, X'086D') Birincil ve ana ASID' ler farklıdır.

#### **MQRC\_CALL\_IN\_PROGRESS**

(2219, X'08AB') Önceki arama tamamlanmadan önce girilen MQI çağrısı.

#### **MQRC\_CONNECTION\_BROKEN**

(2009, X'07D9') Kuyruk yöneticisine bağlantı kaybedildi.

#### **MQRC\_DMPO\_ERROR**

(2481, X'09B1') İleti silme özelliği seçenekleri yapısı geçersiz.

**MQRC\_HMSG\_HATA**

(2460, X'099C') İleti tanıtıcısı geçersiz.

**MQRC\_MSG\_HANDLE\_IN\_USE**

(2499, X'09C3') İleti tanıtıcısı kullanımda.

**MQRC\_OPTIONS\_HATA**

(2046, X'07FE') Seçenekler geçerli değil ya da tutarlı değil.

**MQRC\_NAMETY\_NAME\_ERROR**

(2442, X'098A') Özellik adı geçersiz.

**MQRC\_SOURCE\_CCSID\_HATA**

(2111, X'083F') Özellik adı kodlanmış karakter takımı tanıtıcısı geçersiz.

**MQRC\_UNEXPECTED\_HATASI**

(2195, X'0893 ') Beklenmeyen hata oluştu.

Bu kodlarla ilgili ayrıntılı bilgi için bkz. [İletiler ve neden kodları](#).

## C çağırısı

```
MQDLTMP (Hconn, Hmsg, &DltPropOpts, &Name, &CompCode, &Reason)
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
MQHCONN Hconn;          /* Connection handle */
MQHMSG  Hmsg;           /* Message handle */
MQDMPO  DltPropOpts;   /* Options that control the action of MQDLTMP */
MQCHARV Name;          /* Property name */
MQLONG  CompCode;      /* Completion code */
MQLONG  Reason;        /* Reason code qualifying CompCode */
```

## COBOL çağırısı

```
CALL 'MQDLTMP' USING HCONN, HMSG, DLTPROPOPTS, NAME, COMPCODE, REASON.
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
** Connection handle
01 HCONN PIC S9(9) BINARY.
** Message handle
01 HMSG PIC S9(18) BINARY.
** Options that control the action of MQDLTMP
01 DLTPROPOPTS.
COPY CMQDMPOV.
** Property name
01 NAME.
COPY CMQCHRVV.
** Completion code
01 COMPCODE PIC S9(9) BINARY.
** Reason code qualifying COMPCODE
01 REASON PIC S9(9) BINARY.
```

## PL/I çağırısı

```
call MQDLTMP (Hconn, Hmsg, DltPropOpts, Name, CompCode, Reason);
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
dcl Hconn      fixed bin(31); /* Connection handle */
dcl Hmsg       fixed bin(63); /* Message handle */
dcl DltPropOpts like MQDMPO; /* Options that control the action of MQDLTMP */
```

```
dc1 Name      like MQCHARV; /* Property name */
dc1 CompCode  fixed bin(31); /* Completion code */
dc1 Reason    fixed bin(31); /* Reason code qualifying CompCode */
```

## High Level Assembler (Yüksek Düzeyli çevirici) çağrısı

```
CALL MQDLTMP, (HCONN,HMSG,DLTPROPOPTS,NAME,COMPCODE,REASON)
```

Değişirgeleri aşağıdaki gibi bildirin:

HCONN	DS	F	Connection handle
HMSG	DS	D	Message handle
DLTPROPOPTS	CMQDMP0A	,	Options that control the action of MQDLTMP
NAME	CMQCHRVA	,	Property name
COMPCODE	DS	F	Completion code
REASON	DS	F	Reason code qualifying COMPCODE

## MQGET-İletiyi al

MQGET çağrısı, yerel bir kuyruktan gelen bir iletiyi, MQOPEN çağrısıyla birlikte alır.

### Sözdizimi

MQGET (*Hconn*, *Hobj*, *MsgDesc*, *GetMsgOpts*, *BufferLength*, *Buffer*, *DataLength*, *CompCode*, *Reason*)

### Parametreler

#### Hconn

Tip: MQHCONN-input

Bu tanıtıcı, kuyruk yöneticisiyle olan bağlantıyı gösterir. The value of *Hconn* was returned by a previous MQCONN or MQCONNX call.

On z/OS for CICS applications the MQCONN call can be omitted, and the following value specified for *Hconn* :

#### MQHC\_DEF\_HCONN

Varsayılan bağlantı tanıtıcısı.

#### Hobj

Tip: MQHOBJ-input

Bu tanıtıcı, bir iletinin alınacağı kuyruğu gösterir. The value of *Hobj* was returned by a previous MQOPEN call. Kuyruk, aşağıdaki seçeneklerden biriyle ya da daha fazlasıyla açılmış olmalıdır (ayrıntılar için [“MQOpen-Open nesnesi” sayfa 720](#) konusuna bakın):

- MQOO\_INPUT\_SHARED
- MQOO\_INPUT\_EXCLUSIVE
- MQOO\_INPUT\_AS\_Q\_DEF
- MQOO\_BROWSE

#### MsgDesc

Tip: MQMD-giriş/çıkış

Bu yapı, gerekli iletinin özniteliklerini ve alınan iletinin özniteliklerini açıklamadır. Ayrıntılar için bkz. [“MQMD-İleti tanımlayıcı” sayfa 416](#).

*BufferLength* ileti uzunluğundan küçükse, kuyruk yöneticisi *MsgDesc* tarafından doldurulur, **GetMsgOpts** parametresinde MQGMO\_ACCEPT\_TRUNCATED\_MSG belirtilirse (bkz. [MQGMO- Seçenekler alanı](#)).

Uygulama bir version-1 MQMD sağlıyorsa, döndürülen iletinin uygulama iletisi verilerine önek olarak bir MQMDE öneki vardır; ancak, MQMDE içindeki alanlardan biri ya da daha fazlasının varsayılan dışı bir değeri varsa, bu ileti bir MQMDE öneki içerir. MQMDE ' deki tüm alanlarda varsayılan değerler varsa, MQMDE atlanır. MQMD 'deki *Biçim* alanında MQFMT\_MD\_EXTENSION 'ın biçim adı, bir MQMDE' nin var olduğunu gösterir.

*MsgHandle* alanında geçerli bir ileti tanıtıcısı sağlanırsa, uygulamanın bir MQMD yapısı sağlaması gerekmez. Bu alanda hiçbir şey sağlanmıyorsa, iletinin tanımlayıcı, ileti tanıtıcılarıyla ilişkili tanımlayıcıdan alınır.

Uygulama bir MQMD yapısı yerine bir ileti tanıtıcısı sağlıyorsa ve MQGMO\_PROPERTIES\_FORCE\_MQRFH2 belirtirse, çağrı neden kodu MQRC\_MD\_ERROR ile başarısız olur. Uygulama bir MQMD yapısı sağlamazsa ve MQGMO\_PROPERTIES\_AS\_Q\_DEF değerini belirtmezse ve **PropertyControl** kuyruk özneliği MQPROP\_FORCE\_MQRFH2 ise, çağrı da başarısız olur. Neden kodu: MQRC\_MD\_ERROR.

Eşleşme seçenekleri belirlendiyse ve ileti tanıtıcısı ile ilişkilendirilen ileti tanımlayıcısı kullanılıyorsa, ileti tanıtıcısından eşleşen giriş alanları için kullanılır.

### GetMsgSeçenekleri

Tip: MQGMO-giriş/çıkış

Ayrıntılar için bkz. [“MQGMO-Get-message seçenekleri”](#) sayfa 364.

### BufferLength

Tip: MQHOME-giriş

Bu, *Buffer* alanının bayt cinsinden uzunluğudur. Veri içermeyen iletiler için sıfır belirtin ya da ileti kuyruktan kaldırılacaksa ve atılan veri varsa (bu durumda MQGMO\_ACCEPT\_TRUNCATED\_MSG belirtmeniz gerekir).

**Not:** Kuyruktan okunmanın mümkün olduğu en uzun iletinin uzunluğu **MaxMsgLength** kuyruk özneliği tarafından verilir; bkz. [“Kuyruklara ilişkin öznelikler”](#) sayfa 821.

### Arabellek

Tip: MQBYTEExBufferUzunluğu-çıkış

Bu alan, ileti verilerini içermek için kullanılan alandır. Arabelleği, iletteki verilerin niteine uygun bir sınırla hizalayın. 4 baytlık hizalama çoğu ileti için uygundur ( IBM MQ üstbilgi yapılarını içeren iletiler de içinde olmak üzere), ancak bazı iletiler daha sıkı hizalamayı gerektirebilir. Örneğin, 64 bit ikili tamsayı içeren bir ileti 8 byte 'lık hizalama gerektirebilir.

*BufferLength* ileti uzunluğundan küçükse, iletinin olabildiğince çoğu **Buffer** ' a taşınır. Bu durum, **GetMsgOpts** parametresinde MQGMO\_ACCEPT\_TRUNCATED\_MSG parametresinin belirtilip belirtilmediğini (ek bilgi için [MQGMO-Options alanı](#) başlıklı konuya bakın).

**Buffer** içindeki verilerin karakter kümesi ve kodlaması, **MsgDesc** parametresine döndürülen *CodedCharSetId* ve *Encoding* alanları tarafından verilir. Bu değerler, alıcının gerektirdiği değerlerden farklıysa, alıcı, uygulama iletisi verilerini karakter takımı ve kodlamaya dönüştürmelidir. İleti verilerini dönüştürmek için MQGMO\_CONVERT seçeneği kullanılabilir (gerekirse kullanıcı tarafından yazılmış bir çıkışa sahip); bu seçeneğin ayrıntıları için bkz. [“MQGMO-Get-message seçenekleri”](#) sayfa 364 .

**Not:** MQGET çağrısındaki diğer tüm parametreler, yerel kuyruk yöneticisinin karakter kümesinde ve kodlamasında ( **CodedCharSetId** kuyruk yöneticisi özneliği ve MQENC\_NATIVE ile verilir) kodlanır.

Arama başarısız olursa, arabelleğin içeriği değişmeye devam edebilir.

C programlama dilinde parametre, bir işaretçi olarak bildirilir: Herhangi bir veri tipinin adresi, parametre olarak belirtilebilir.

**BufferLength** parametresi sıfır ise, *Buffer* bu durumda gönderilmez; bu durumda, C ya da System/390 çevirici programlarında yazılan programlarla geçirilen parametre adresi boş değerli olabilir.

## DataLength

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

This is the length in bytes of the application data *İletide*. Bu değer *BufferLength* değerinden büyükse, **Buffer** parametresine yalnızca *BufferLength* byte 'ları döndürülür (yani ileti kısaltılır). Değer sıfır ise, ileti uygulama verisi içermez.

If *BufferLength* is less than the message length, *DataLength* is still completed by the queue manager, whether MQGMO\_ACCEPT\_TRUNCATED\_MSG is specified on the **GetMsgOpts** parameter (see MQGMO-Seçenekler alanı for more information). Bu, uygulamanın ileti verilerini barındırmak için gereken arabelleğin büyüklüğünü belirlemesine olanak tanır ve daha sonra, aramayı uygun boyutta bir arabellekle yeniden yayınlayabilir.

Ancak, MQGMO\_CONVERT seçeneği belirtilirse ve dönüştürülen ileti verilerinin *Buffer* içine sığması çok uzun olursa, *DataLength* için döndürülen değer şudur:

- Kuyruk yöneticisi tanımlı biçimler için *dönüştürülemeyen* verilerin uzunluğu.

In this case, if the nature of the data causes it to expand during conversion, the application must allocate a buffer bigger than the value returned by the queue manager for *DataLength*.

- Uygulama tanımlı biçimler için, veri dönüştürme çıkışıyla döndürülen değer.

## CompCode

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Tamamlanma kodu; bu kod aşağıdakilerden biridir:

### MQCC\_OK

İşlem başarıyla tamamlandı.

### MQCC\_UYARI

Uyarı (kısmi tamamlama).

### MQCC\_FAILED

Arama başarısız oldu.

## Neden

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Listelenen neden kodları, kuyruk yöneticisinin **Reason** parametresi için döndürülebileceği kodlardır. If the application specifies the MQGMO\_CONVERT option, and a user-written exit is invoked to convert some or all the message data, the exit decides what value is returned for the **Reason** parameter. Sonuç olarak, belgelenen değerlerin dışındaki değerler de mümkündür.

*CompCode* MQCC\_OK ise:

### MQRC\_NONE

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_UYARI ise:

### MQRC\_CONVERTED\_MSG\_TOO\_BÜYÜK

(2120, X'848 ') Dönüştürülen veriler arabellek için çok büyük.

### MQRC\_CONVERTED\_STRING\_TOO\_BÜYÜK

(2190, X'88E') Dönüştürülen dize, alan için çok büyük.

### MQRC\_DBCS\_HATASI

(2150, X'866 ') DBCS dizgisi geçerli değil.

### MQRC\_FORMAT\_ERROR

(2110, X'83E') İleti biçimi geçerli değil.

### MQRC\_INCOMPLE\_GROUP

(2241, X'8C1') İleti grubu tamamlanmadı.

### MQRC\_INCOMPLE\_MSG

(2242, X'8C2') Mantıksal ileti tamamlanmadı.

**MQRC\_INCONSISTENT\_CCSDS**

(2243, X'8C3') İleti kesimleri farklı CCSID' lere sahip.

**MQRC\_INCONTINUENT\_ENCODINGS**

(2244, X'8C4') İleti kesimleri farklı kodlamalara sahiptir.

**MQRC\_INCONSISTENT\_UOW**

(2244, X'8C5') Tutarsız iş birliği-iş belirtimi.

**MQRC\_MSG\_TOKEN\_HATASI**

(2331, X'91B') İleti belirtecinin kullanımı geçersiz.

**MQRC\_NO\_MSG\_LOCKED**

(2209, X'8A1') İleti kilitlemedi.

**MQRC\_NOT\_CONVERTED**

(2119, X'847 ') İleti verileri dönüştürülmedi.

**MQRC\_OPTIONS\_CHANGED**

(nnnn, X'xxx ') Tutarlı olması zorunlu olan seçenekler değiştirildi.

**MQRC\_PARTIALY\_CONVERTED**

(2272, X'8E0') İleti verileri kısmen dönüştürüldü.

**MQRC\_SIGNAL\_REQUEST\_ACCEPTED**

(2070, X'816 ') İleti döndürülmedi (ancak sinyal isteği kabul edildi).

**MQRC\_SOURCE\_BUFFER\_ERROR**

(2145, X'861 ') Kaynak arabellek parametresi geçerli değil.

**MQRC\_SOURCE\_CCSD\_ERROR**

(2111, X'83F') Kaynak kodlanmış karakter takımı tanıtıcısı geçerli değil.

**MQRC\_SOURCE\_DECIMAL\_ENC\_ERROR**

(2113, X'841 ') İletide paketli ondalık kodlama tanınmadı.

**MQRC\_SOURCE\_FLOATENC\_ERROR**

(2114, X'842 ') İletide kayan noktalı kodlama tanınmadı.

**MQRC\_SOURCE\_INTEGER\_ENC\_ERROR**

(2112, X'840 ') Kaynak tamsayı kodlaması tanınmadı.

**MQRC\_SOURCE\_LENGTH\_ERROR**

(2143, X'85F') Kaynak uzunluğu parametresi geçerli değil.

**MQRC\_TARGET\_BUFF\_ERROR**

(2146, X'862 ') Hedef arabellek parametresi geçerli değil.

**MQRC\_TARGET\_CCSD\_HATASI**

(2115, X'843 ') Hedef kodlanmış karakter takımı tanıtıcısı geçerli değil.

**MQRC\_TARGET\_DECIMALIM\_ERROR**

(2117, X'845 ') Packed-receiver tarafından belirtilen paket-ondalık kodlama tanınmıyor.

**MQRC\_TARGET\_FLOATENC\_ERROR**

(2118, X'846 ') Alıcı tarafından belirtilen kayan noktalı kodlama tanınmıyor.

**MQRC\_TARGET\_INTEGER\_ENC\_ERROR**

(2116, X'844 ') Hedef tamsayı kodlaması tanınmadı.

**MQRC\_TRUNCATED\_MSG\_KABUL edildi**

(2079, X'81F') Kesilen ileti döndürüldü (işleme tamamlandı).

**MQRC\_TRUNCATED\_MSG\_FAILED**

(2080, X'820 ') Kesilen ileti döndürüldü (işleme tamamlanmadı).

*CompCode* MQCC\_FAILED ise:

**MQRC\_ADAPTER\_NOT\_AVAM**

(2204, X'89C') Bağdaştırıcı kullanılmıyor.

**MQRC\_ADAPTER\_CONV\_LOAD\_ERROR**

(2133, X'855 ') Veri dönüştürme hizmetleri modüllerinin yüklenmesi gerçekleştirilemiyor.



**MQRC\_ADAPTER\_SERV\_LOAD\_ERROR**  
(2130, X'852 ') Baędařtırıcı hizmeti modülü yüklenemiyor.

**MQRC\_API\_EXIT\_ERROR**  
(2374, X' 946 ') API çıkışı başarısız oldu.

**MQRC\_API\_EXIT\_LOAD\_ERROR**  
(2183, X'887 ') API çıkışı yüklenemiyor.

**MQRC\_ASID\_MISSMACH**  
(2157, X'86D') Birincil ve ana sunucu ASID deęerleri farklı.

**MQRC\_BACKED\_OUT**  
(2003, X'7D3') İş birimi yedeklendi.

**MQRC\_BUFFER\_ERROR**  
(2004, X'7D4') Arabellek parametresi geçerli deęil.

**MQRC\_BUFFER\_LENGTH\_ERROR**  
(2005, X'7D5') Arabellek uzunluęu parametresi geçerli deęil.

**MQRC\_CALL\_IN\_PROGRESS**  
(2219, X'8AB') Önceki arama tamamlanmadan önce girilen MQI çağırısı.

**MQRC\_CF\_NOT\_VAR**  
(2345, X' 929 ') Coupling tesisi kullanılmıyor.

**MQRC\_CF\_STRUC\_FAILED**  
(2373, X' 945 ') Coupling-tesis yapısı başarısız oldu.

**MQRC\_CF\_STRUC\_IN\_USE**  
(2346, X'92A') Coupling-tesis yapısı kullanımda.

**MQRC\_CF\_STRUC\_LIST\_HDR\_IN\_USE**  
(2347, X'92B') Coupling-tesis yapısı listesi-üstbilgi kullanımda.

**MQRC\_CICS\_WAIT\_FAILED**  
(2140, X'85C') Bekleme isteęi CICStarafından reddedildi.

**MQRC\_CONNECTION\_BROKEN**  
(2009, X'7D9') Kuyruk yöneticisine yönelik baęlantı kaybedildi.

**MQRC\_CONNECTION\_NOT\_YETKILI**  
(2217, X'8A9') Baęlantı için yetkili deęil.

**MQRC\_CONNECTION QUIESCING**  
(2202, X'89A') Baęlantı susturulmuş durumda.

**MQRC\_CONNECTION\_DURATION**  
(2203, X'89B') Baęlantı sona erdiriliyor.

**MQRC\_CORREL\_ID\_ERROR**  
(2207, X'89F') İlintilendirme tanıtıcısı hatası.

**MQRC\_DATA\_LENGTH\_ERROR**  
(2010, X'7DA') Veri uzunluęu parametresi geçerli deęil.

**MQRC\_DB2\_NOT\_AVAILABLE**  
(2342, X' 926 ') Db2 altsistemi kullanılmıyor.

**MQRC\_GET\_INHIBITED**  
(2016, X'7E0') Kuyruęun engellenmesini saęlar.

**MQRC\_GLOBAL\_UOW\_CONFLICTIONS**  
(2351, X'92F') Genel iş çakışmaları birimleri.

**MQRC\_GMO\_ERROR**  
(2186, X'88A') Al-ileti seęenekleri yapısı geçerli deęil.

**MQRC\_ANDLE\_IN\_USE\_FOR\_UOW**  
(2353, X' 931 ') Genel iş birimi için kullanılan tanıtıcı.

**MQRC\_HCONN\_ERROR**  
(2018, X'7E2') Baęlantı tanıtıcısı geçerli deęil.

**MQR\_C\_HOBJ\_ERROR**

(2019, X'7E3') Nesne tanıtıcısı geçerli değil.

**MQR\_C\_INCONTINUED\_BROWSE**

(2259, X'8D3') Tutarsız göz atma belirtimi.

**MQR\_C\_INCONSISTENT\_UOW**

(2244, X'8C5') Tutarsız iş birliği-iş belirtimi.

**MQR\_C\_INVALID\_MSG\_UNDER\_CURSOR**

(2246, X'8C6') İmleç altındaki ileti alma için geçerli değil.

**MQR\_C\_LOCAL\_UOW\_CONFLICTID**

(2352, X' 930 ') Genel iş birimi yerel iş birimi ile çakışıyor.

**MQR\_C\_MATCH\_OPTIONS\_ERROR**

(2247, X'8C7') Eşleştirme seçenekleri geçerli değil.

**MQR\_C\_MD\_ERROR**

(2026, X'7EA') İleti tanımlayıcısı geçerli değil.

**MQR\_C\_MSG\_ID\_ERROR**

(2206, X'89E') İleti-tanıtıcısı hatası.

**MQR\_C\_MSG\_SEQ\_NUMBER\_ERROR**

(2250, X'8CA') İleti sıra numarası geçerli değil.

**MQR\_C\_MSG\_TOKEN\_HATASI**

(2331, X'91B') İleti simgesinin kullanımı geçerli değil.

**MQR\_C\_NO\_MSG\_AVAILABLE**

(2033, X'7F1') İleti yok.

**MQR\_C\_NO\_MSG\_UNDER\_CURSOR**

(2034, X'7F2') Browse imleci iletide konumlandırılmamış.

**MQR\_C\_NOT\_OPEN\_FOR\_BROWSE**

(2036, X'7F4') Kuyruk göz atma için açık değil.

**MQR\_C\_NOT\_OPEN\_FOR\_INPUT**

(2037, X'7F5') Kuyruk giriş için açık değil.

**MQR\_C\_OBJECT\_CHANGED**

(2041, X'7F9') Nesne tanımlaması açıldığından beri değiştirildi.

**MQR\_C\_OBJECT\_ZARAR**

(2101, X'835 ') Nesne zarar gördü.

**MQR\_C\_OPTIONS\_ERROR**

(2046, X'7FE') Seçenekler geçerli değil ya da tutarlı değil.

**MQR\_C\_PAGEES\_HATASI**

(2193, X'891 ') Sayfa kümesi veri kümesine erişilirken hata oluştu.

**MQR\_C\_Q\_DELETED**

(2052, X'804 ') Kuyruk silinmiştir.

**MQR\_C\_Q\_INDEX\_TYPE\_ERROR**

(2394, X'95A') Kuyruk yanlış dizin tipi içeriyor.

**MQR\_C\_Q\_MGR\_NAME\_ERROR**

(2058, X'80A') Kuyruk yöneticisi adı geçerli değil ya da bilinmiyor.

**MQR\_C\_Q\_MGR\_NOT\_VAR**

(2059, X'80B') Kuyruk yöneticisi bağlantı için kullanılabilir değil.

**MQR\_C\_Q\_MGR\_QUIESCING**

(2161, X'871 ') Kuyruk yöneticisi susturuyor.

**MQR\_C\_Q\_MGR\_STOPPING**

(2162, X'872 ') Kuyruk yöneticisi sona erdiriliyor.

**MQR\_C\_RESOURCE\_SORUNU**

(2102, X'836 ') Yetersiz sistem kaynakları var.

**MQRC\_SECOND\_MARK\_NOT\_ALLOWED**

(2062, X'80E') Bir ileti önceden işaretlendi.

**MQRC\_SIGNAL\_INSTANT**

(2069, X'815 ') Bu tutamaç için bekleyen sinyal.

**MQRC\_SIGNAL1\_ERROR**

(2099, X'833 ') Sinyal alanı geçerli değil.

**MQRC\_STORAGE\_MEDIUM\_FULL**

(2192, X'890 ') Dış depolama ortamı dolu.

**MQRC\_STORAGE\_NOT\_AVAM**

(2071, X'817 ') Kullanılabilir bellek yetersiz.

**MQRC\_SUPPRESSED\_BY\_EXIT**

(2109, X'83D') Çıkış programı tarafından çağrı engellendi.

**MQRC\_SYNCPOINT\_LIMIT\_UVARD**

(2024, X'7E8') Yürürlükteki iş birimi içinde başka ileti işlenemez.

**MQRC\_SYNCPOINT\_NOT\_AVAM**

(2072, X'818 ') eşitleme noktası desteği yok.

**MQRC\_UNEXPECTED\_ERROR**

(2195, X'893 ') Beklenmeyen bir hata oluştu.

**MQRC\_UOWENLISTMENT\_ERROR**

(2354, X' 932 ') Genel iş birimi içindeki bilgiler başarısız oldu.

**MQRC\_UOW\_MIX\_NOT\_SUPPORTANT**

(2355, X' 933 ') İşlerin birim içi çağrıları karışımı desteklenmiyor.

**MQRC\_UOW\_NOT\_VAR**

(2255, X'8CF') Kuyruk yöneticisinin kullanması için kullanılacak iş birimi.

**MQRC\_WAIT\_INTERVAL\_ERROR**

(2090, X'82A') MQGMO' da bekleme aralığı geçerli değil.

**MQRC\_WRONG\_GMO\_SÜRÜMÜ**

(2256, X'8D0') MQGMO' nun yanlış sürümü sağlandı.

**MQRC\_WRONG\_MD\_VERSION**

(2257, X'8D1') MQMD' nin yanlış sürümü sağlandı.

Bu kodlarla ilgili ayrıntılı bilgi için [İletiler ve neden kodları](#) başlıklı konuya bakın.

**Kullanım notları**

1. Alınan ileti olağan durumda kuyruktan silinir. Bu silme, MQGET çağrısının bir parçası olarak ya da bir eşitleme noktasının bir parçası olarak ortaya çıkabilir.  
Göz atma seçenekleri şunlardır: MQGMO\_BROWSE\_FIRST, MQGMO\_BROWSE\_NEXT ve MQGMO\_BROWSE\_MSG\_UNDER\_CURSOR.
2. Göz atma seçeneklerinden biri ile MQGMO\_LOCK seçeneği belirtilirse, göz atma ileti yalnızca bu tutamaçla görünür olacak şekilde kilitlenir.  
MQGMO\_UNLOCK seçeneği belirtilirse, önceden kilitlenmiş bir iletinin kilidi açılır. Bu durumda hiçbir ileti alınmadı; **MsgDesc**, **BufferLength**, **Buffer** ve **DataLength** parametreleri denetlenmez ya da değiştirilmez.
3. Bir MQGET çağrısı yayınlayan uygulamalar için, uygulama olağandışı bir şekilde sona erdirilirse ya da çağrı işlenirken kopan bağlantı kesilirse, alınan ileti kaybedilebilir. Bu sorun, uygulama adına MQGET çağrısını konu alan kuyruk yöneticisiyle aynı platformda çalışan vekil sunucu uygulaması, ileti kuyruktan kaldırıldıktan sonra, taşıyıcı iletiyi uygulamaya döndürmek üzere oluncaya kadar uygulamanın kaybını algılayamadığından ortaya çıkar. Bu sorun hem kalıcı iletiler, hem de kalıcı olmayan iletiler için oluşabilir.

İletilerin bu şekilde kaybolma riskini ortadan kaldırmak için her zaman iş birimleri içinde iletileri alın. Yani, MQGET çağrısındaki MQGMO\_SYNCPOINT seçeneğini belirterek ve ileti işleme tamamlandığında iş birimini kesinleştirmek ya da geri almak için MQCMIT ya da MQBACK çağrılarını kullanın. MQGMO\_SYNCPOINT belirtildiyse ve istemci olağan dışı bir şekilde sona erdirilirse ya da bağlantı kesilirse, vekil, kuyruk yöneticilikteki çalışma birimini yedekler ve ileti kuyruktan geri gelir. Eşitleme noktaları hakkında daha fazla bilgi için bkz. [IBM MQ uygulamalarındaki eşitleme noktası konuları](#).

Bu durum, kuyruk yöneticisiyle aynı platformda çalışan uygulamalarla birlikte IBM MQ istemcileriyle de ortaya çıkabilir.

4. Bir uygulama belirli bir ileti dizisini belirli bir yere koyarsatek bir iş birimi içinde kuyruğa girin ve bu iş birimini başarıyla kesinleştirerek, iletiler aşağıdaki şekilde alınabilmekte kullanılabilir olur:

- Kuyruk *paylaşılmayan kuyruksa* (yani, yerel bir kuyruksa), iş birimi içindeki tüm iletiler aynı anda kullanılabilir olur.
- Kuyruk *paylaşılan bir kuyruksa*, iş birimi içindeki iletiler, bunların yerleştirdikleri sırayla kullanılabilir olur, ancak tümü aynı anda kullanılamaz. Sistem yoğun olarak yüklü olduğunda, iş birimindeki ilk iletinin başarılı bir şekilde alınması mümkündür; ancak, MQRC\_NO\_MSG\_AVAILABLE biriminde ikinci ya da sonraki ileti için MQGET çağrısının başarısız olması mümkün olur. Bu sorun ortaya çıkarsa, uygulamanın kısa bir süre beklemesi ve sonra işlemi yeniden denemesi gerekir.

5. Bir uygulama, ileti grupları kullanılmadan aynı kuyruğa ileti dizisi koyarsa, Belirli koşullar karşılanırsa, bu iletilerin sırası korunur. Ayrıntılar için bkz. [MQPUT kullanım notları](#) . Koşullar yerine getirildiyse, iletiler gönderildikleri sırayla, teslim alma uygulamasına sunulur:

- Kuyruktan yalnızca bir alıcı ileti alıyor.

Kuyruktan ileti alan iki ya da daha fazla uygulama varsa, bunlar, bir sıraya ait iletileri tanımlamak için kullanılacak mekanizmayı gönderen ile kabul etmelidir. Örneğin, gönderici, iletilerde bulunan tüm CorrelId alanlarını, bu ileti dizisi için benzersiz olan bir değere ayarlayabilir.

- Alıcı, örneğin belirli bir MsgId ya da CorrelIdbelirtilerek, alma sırasını kasıtlı olarak değiştirmez.

Gönderme uygulaması iletileri bir ileti grubu olarak koyarsa, alma uygulaması MQGET çağrısında MQGMO\_LOGICAL\_ORDER seçeneğini belirtiyorsa, iletiler doğru sırayla teslim alma uygulamasına sunulur. İleti gruplarına ilişkin ek bilgi için aşağıdaki başlıklara bakın:

- [MQMD- MsgFlags alanı](#)
- [MQPMO\\_LOGICAL\\_ORDER](#)
- [MQGMO\\_LOGICAL\\_ORDER](#)

Kullanıcı, eşitleme noktası altındaki bir gruptaki iletileri alıyorsa, işlemi bitirmeye çalışmadan önce tüm grubun işlendiğinden emin olmalıdır.

6. Uygulamalar, **MsgDesc** parametresinin Feedback alanındaki geribildirim kodu MQFB\_QUIT için test etmeli ve bu değeri bulurlarsa sona ermelidir. Ek bilgi için bkz. [MQMD-Feedback alanı](#) .

7. Hobj ile tanıtilen kuyruk, MQOO\_SAVE\_ALL\_CONTEXT seçeneğiyle açıldıysa ve MQGET çağrısından tamamlanma kodu MQCC\_OK ya da MQCC\_UYARI ise, Hobj kuyruk işleciyle ilişkilendirilen bağlam, alınan iletinin bağlamına ayarlanır (MQGMO\_BROWSE\_FIRST, MQGMO\_BROWSE\_NEXT ya da MQGMO\_BROWSE\_MSG\_UNDER\_CURSOR seçeneği ayarlandıysa, bu durumda bağlam kullanılabilir değil olarak işaretlenir).

MQPMO\_PASS\_IDENTITY\_CONTEXT ya da MQPMO\_PASS\_ALL\_CONTEXT seçeneklerini belirterek, saklanmış bağlamı sonraki bir MQPUT ya da MQPUT1 çağrısında kullanabilirsiniz. Bu işlem, alınan iletinin içeriğinin tamamını ya da bir kısmı başka bir iletiye (örneğin, ileti başka bir kuyruğa iletilindiğinde) aktarılmasını sağlar. İleti bağlamına ilişkin ek bilgi için [İleti bağlamı](#) başlıklı konuya bakın.

8. **GetMsgOpts** parametresine MQGMO\_CONVERT seçeneğini eklerseniz, veriler **Buffer** parametresine yerleştirilmeden önce, uygulama iletileri alan uygulama tarafından istenen gösteriye dönüştürülür.

- İletideki denetim bilgilerindeki Format alanı, uygulama verilerinin yapısını tanımlar ve iletteki denetim bilgilerindeki CodedCharSetId ve Encoding alanları, karakter kümesi tanıtıcısını ve kodlamasını belirtir.

- MQGET çağrısını yayınlayan uygulama, **MsgDesc** parametresindeki CodedCharSetId ve Encoding alanlarında, uygulama iletisi verilerinin dönüştürülebileceği karakter kümesi tanıtıcısını ve kodlamasını belirtir.

İleti verilerinin dönüştürülmesi gerekliyse, dönüştürme işlemi kuyruk yöneticisinin kendisi tarafından ya da kullanıcı tarafından yazılan bir çıkışa göre gerçekleştirilir; bu durumda, iletteki denetim bilgilerindeki Format alanının değerine bağlı olarak değişir:

- Aşağıdaki biçim adları, kuyruk yöneticisi tarafından dönüştürülen biçimlerdir; bu biçimler "yerleşik" biçimler olarak adlandırılır:

- MQFMT\_ADMIN
- MQFMT\_CICS (yalnızca z/OS)
- MQFMT\_COMMAND\_1
- MQFMT\_COMMAND\_2
- MQFMT\_DEAD\_LETTER\_HEADER
- MQFMT\_DIST\_HEADER
- MQFMT\_EVENT sürüm 1
- MQFMT\_EVENT sürüm 2 (yalnızca z/OS)
- MQFMT\_IMS
- MQFMT\_IMS\_VAR\_STRING
- MQFMT\_MD\_EXTENSION
- MQFMT\_PCF
- MQFMT\_REF\_MSG\_HEADER
- MQFMT\_RF\_HEADER
- MQFMT\_RF\_HEADER\_2
- MQFMT\_STRING
- MQFMT\_TETIKLEYICISI
- MQFMT\_WORK\_INFO\_HEADER (yalnızca z/OS)
- MQFMT\_XMIT\_Q\_HEADER

- MQFMT\_NONE biçim adı, iletteki verilerin niteliğin tanımsız olduğunu gösteren özel bir değerdir. Sonuç olarak, kuyruk yöneticisi ileti kuyruktan alındığında dönüştürme girişiminde bulunmaz.

**Not:** MQGET çağrısında MQFMT\_NONE biçim adı bulunan bir ileti için MQGMO\_CONVERT belirtilirse ve iletinin karakter kümesi ya da kodlaması **MsgDesc** değiştirilmesinde belirtilenden farklıysa, ileti **Buffer** parametresine döndürülür (başka hata varsayılarak), ancak çağrı tamamlanma kodu MQCC\_UYANCE ve neden kodu MQRC\_FORMAT\_ERROR ile tamamlanır.

MQFMT\_NONE ' u ileti verilerinin niteliği, dönüştürme gerektirmediği anlamına geldiğinde ya da gönderme ve alma uygulamaları, kendileriyle ileti verilerinin gönderileceği form arasında anlaşıldığında, bu işlemi kullanabilirsiniz.

- Diğer tüm biçim adları, iletiyi dönüştürmek için kullanıcı tarafından yazılmış bir çıkışa iletir. Çıkış, ortama özgü eklemeler dışında, biçimle aynı adı içerir. Kullanıcı tarafından belirtilen biçim adları IBM MQharfleriyle başlamamalıdır.

Veri dönüştürme çıkışıyla ilgili ayrıntılar için bkz. [“Veri dönüştürme çıkışı” sayfa 891](#) .

İletideki kullanıcı verileri, desteklenen karakter kümeleri ve kodlamalar arasında dönüştürülebilir. Ancak, ileti bir ya da daha fazla IBM MQ üstbilgi yapısı içeriyorsa, ileti, kuyruk adlarında geçerli olan karakterlerden herhangi biri için çift byte ya da çok byte karakterler içeren bir karakter kümesinden ya da bir karakter takımından dönüştürülemez. Bu denirse neden kodu MQRC\_SOURCE\_CCSD\_ERROR ya da MQRC\_TARGET\_CCSD\_ERROR sonucu, bu denirse ve ileti dönüştürülmemiş olarak döndürülür. Unicode karakter kümesi UTF-16 , bu tür bir karakter kümesine bir örnektir.

MQGET ' den dönüşte, aşağıdaki neden kodu iletinin başarıyla dönüştürülmüş olduğunu gösterir:

- MQRC\_NONE

The following reason code indicates that the message might have been converted successfully; the application must check the CodedCharSetId and Encoding fields in the **MsgDesc** parameter to find out:

- MQRC\_TRUNCATED\_MSG\_KABUL edildi

Diğer tüm neden kodları, iletinin dönüştürülmediğini gösterir.

**Not:** Bu neden kodunun yorumlanması, yalnızca çıkışta “Veri dönüştürme çıkışı” sayfa 891’inde açıklanan işleme yönergelerine uyması durumunda, kullanıcı tarafından yazılan bir çıkışta gerçekleştirilen dönüştürmeler için geçerlidir.

9. İletileri almak için nesne yönelimli arabirimi kullanırken, bir MQGET çağrısına ilişkin ileti verilerini tutmak için bir arabellek belirtmemeyi seçebilirsiniz. Ancak, IBM WebSphere MQ 7.0 sürümünden önceki IBM MQ sürümlerinde, bir arabellek belirtilmediğinde bile MQGET ' in neden kodu MQRC\_CONVERTED\_MSG\_TO\_BIG ile başarısız olması mümkün oldu. IBM WebSphere MQ 7.0' dan, ileti alma arabelleğindeki boyutu kısıtlamadan nesne yönelimli bir uygulama kullanarak bir ileti aldığınızda, uygulama MQRC\_CONVERTED\_MSG\_TOO\_BIG ile başarısız olmaz ve dönüştürülen iletiyi alır. Bu, aşağıdaki ortamlardan doğrudur:

- .NET, including fully managed applications
- C++
- Java ( IBM MQ classes for Java )

**Not:** Tüm istemciler için, sharingConversations değeri sıfırsa, kanal, IBM WebSphere MQ 7.0 tarihinden önce olduğu gibi çalışır ve ileti işleme IBM WebSphere MQ 6 davranışına geri döner. Bu durumda, arabellek dönüştürülen iletiyi alamayacak kadar küçükse, dönüştürülemez ileti döndürülür; neden kodu MQRC\_CONVERTED\_MSG\_TOO\_BÜYÜK değerine sahip. sharingConversations ile ilgili daha fazla bilgi için bkz. [Bir istemci uygulamasında sohbetlerin paylaşılmasını kullanma.](#)

10. Yerleşik biçimler için, kuyruk yöneticisi, MQGMO\_CONVERT seçeneği belirtildiğinde, ileti içindeki karakter dizgilerinin varsayılan dönüştürmesini gerçekleştirebilir. Varsayılan dönüştürme, kuyruk yöneticisinin dizgi verilerini dönüştürürken gerçek karakter kümesine yaklaşan, kuruluş tarafından belirlenmiş bir varsayılan karakter kümesini kullanmasına olanak sağlar. Sonuç olarak, MQGET çağrısı MQCC\_UYARI ile tamamlanacağına ve MQRC\_SOURCE\_CCSD\_ERROR ya da MQRC\_TARGET\_CCSD\_ERROR neden kodunu tamamlamak yerine, MQCC\_OK tamamlanma kodu ile başarılı olabilir.

**Not:** Dizgi verilerini dönüştürmek için yaklaşık bir karakter kümesi kullanılmasının sonucu bazı karakterlerin yanlış dönüştürülebileceğini ifade eder. Bunu önlemek için, dizgide hem gerçek karakter kümesi hem de varsayılan karakter kümesi için ortak karakterler kullanın.

Varsayılan dönüştürme hem uygulama iletisi verileri hem de MQMD ve MQMDE yapılarındaki karakter alanları için geçerlidir:

- Uygulama iletisi verilerinin varsayılan dönüşümü, yalnızca aşağıdaki tüm deyimler doğru olduğunda gerçekleşir:
  - Uygulama MQGMO\_CONVERT değerini belirtiyor.
  - İleti, desteklenmeyen bir karakter kümesinden ya da desteklenmeyen bir karakter kümesine dönüştürülmesi gereken veriler içeriyor.
  - Varsayılan dönüştürme, kuyruk yöneticisi kurulu ya da yeniden başlatıldığında etkinleştirildi.
- Kuyruk yöneticisi için varsayılan dönüştürme etkinleştirildiyse, MQMD ve MQMDE yapılarındaki karakter alanlarının varsayılan dönüştürülmesi gerektiği gibi gerçekleşir. Dönüştürme işlemi, MQGET çağrısındaki uygulama tarafından MQGMO\_CONVERT seçeneği belirtilmediyse de gerçekleştirilir.

11. Visual Basic programlama dili için, aşağıdaki noktalar geçerlidir:

- **Buffer** parametresinin boyutu, **BufferLength** parametresi tarafından belirtilen uzunluktan azsa, çağrı, MQRC\_STORAGE\_NOT\_AVAM neden kodunda başarısız olur.
- **Buffer** parametresi, Stringtipinde olduğu için bildirilir. Kuyruktan alınacak veriler Stringtipinden değilse, MQGEN MQGET yerine çağrılıyor.

MQGETAny çağrısı, MQGET çağrıyla aynı parametrelere sahiptir; ancak, **Buffer** parametresinin Anytipi olarak bildirilmiş olması dışında, herhangi bir veri tipinin alınmasını sağlar. Ancak bu, **Buffer** 'in büyüklük olarak en az BufferLength byte olduğundan emin olmak için denetleyemediği anlamına gelir.

12. Önceden okuma etkinleştirildiğinde, tüm MQGET seçenekleri desteklenmez. Aşağıdaki çizelge, izin verilen seçeneklerin ve MQGET çağrıları arasında değiştirilip değiştirilmediklerini belirtmektedir.

Çizelge 549. İleriye okurken izin verilen MQGET seçenekleri geçerli kıldığında			
	Önceden okuma etkinleştirildiğinde ve MQGET çağrıları arasında değiştirilebilir olduğunda izin verilir	Okuma önden okuma etkinleştirildiğinde izin verilir, ancak MQGET çağrıları arasında değiştirilemez <sup>a</sup>	Önceden okuma özelliği etkinleştirildiğinde izin verilmeyen MQGET seçenekleri <sup>b</sup>
MQGET MD değerleri	MsgId <sup>c</sup> CorrelId <sup>c</sup>	Kodlama CodedCharSetId	
MQGET MQGMO seçenekleri	MQGMO_BEKLE MQGMO_NO_BEKLEME MQGMO_FAIL_IF_QUIESCING MQGMO_BROWSE_FIRST <sup>d</sup> MQGMO_BROWSE_NEXT <sup>d</sup> MQGMO_BROWSE_MESSAGE_UNDER_CURSOR <sup>d</sup>	MQGMO_SYNCPOINT_IF_PERSISTENT MQGMO_NO_SYNCPOINT MQGMO_ACCEPT_TRUNCATED_MSG MQGMO_CONVERT MQGMO_LOGICAL_ORDER MQGMO_COMPLE_MSG MQGMO_ALL_MSGS_AVALABILIR MQGMO_ALL_SEGMENTS_AVALABILIR MQGMO_MARK_BROWSE_HANDLE MQGMO_MARK_BROWSE_CO_OP MQGMO_UNMARK_BROWSE_CO_OP MQGMO_UNMARK_BROWSE_HANDLE MQGMO_UNMARKET_BROWSE_MSG MQGMO_PROPERTIES_FORCE_MQRFH2 MQGMO_NO_ÖZELLİKLERİ MQGMO_PROPERTIES_IN_HANDLE MQGMO_PROPERTIES_COMPATIBILITY	MQGMO_SET_SIGNAL MQGMO_SYNCPOINT MQGMO_MARK_SKIP_BACKUT MQGMO_MSG_ALT_CURSOR <sup>d</sup> MQGMO_LOCK MQGMO_UNLOCK
MQGMO değerleri		MsgHandle	

- MQGET çağrıları arasında bu seçenekler değiştirilirse, bir MQRC\_OPTIONS\_CHANGED neden kodu döndürülür.
  - Bu seçenekler ilk MQGET çağrısında belirtilirse, önden okuma geçersiz kılır. Sonraki bir MQGET çağrısında bu seçenekler belirtilirse, MQRC\_OPTIONS\_ERROR neden kodu döndürülür.
  - İstemci uygulamalarının, MsgId ve CorrelId değerleri önceki değerlerle MQGET çağrıları iletileri arasında değiştirilirse, istemciye önceden gönderilmiş olabileceğini ve tüketilinceye (ya da otomatik olarak temizleninceye) kadar istemcide önden okuma arabelleğinde kalacağını bilmeleri gerekir.
  - İlk MQGET çağrısı, önden okuma etkinleştirildiğinde iletilere göz atılacağını ya da kuyruktan ileti alınacağını belirler. Uygulama göz atma ve alma birleşimi kullanmayı denerse, MQRC\_OPTIONS\_CHANGED neden kodu döndürülür.
  - MQGMO\_MSG\_UNDER\_CURSOR, önden okuma ile olanaklı değil. Önden okuma etkinleştirildiğinde iletilere göz atılabilir ya da iletiler alınabilir, ancak her ikisinin birleşimi olamaz.
13. Uygulamalar, yalnızca bu iletiler alma işlemi ile aynı yerel iş birimine konulursa, kesinleştirilmemiş iletileri ortadan kaldırılabilir. Uygulamalar, kesinleştirilmemiş iletileri yok edici olmayan bir şekilde alamıyor.
14. Göz atma imlecinin altındaki iletiler bir iş biriminde alınabilir. Kesinleştirilmemiş bir iletiyi bu şekilde almak olanaklı değildir.

## C çağırma

```
MQGET (Hconn, Hobj, &MsgDesc, &GetMsgOpts, BufferLength, Buffer,
&DataLength, &CompCode, &Reason);
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
MQHCONN Hconn; /* Connection handle */
```

```

MQHOBj  Hobj;          /* Object handle */
MQMD    MsgDesc;      /* Message descriptor */
MQGMO   GetMsgOpts;   /* Options that control the action of MQGET */
MQLONG  BufferLength;  /* Length in bytes of the Buffer area */
MQBYTE  Buffer[n];     /* Area to contain the message data */
MQLONG  DataLength;   /* Length of the message */
MQLONG  CompCode;     /* Completion code */
MQLONG  Reason;       /* Reason code qualifying CompCode */

```

## Cobol çağırma

```

CALL 'MQGET' USING HCONN, HOBj, MSGDESC, GETMSGOPTS, BUFFERLENGTH,
BUFFER, DATALENGTH, COMPCODE, REASON.

```

Değiştirgeleri aşağıdaki gibi bildirin:

```

** Connection handle
01 HCONN          PIC S9(9) BINARY.
** Object handle
01 HOBj          PIC S9(9) BINARY.
** Message descriptor
01 MSGDESC.
   COPY CMQMDV.
** Options that control the action of MQGET
01 GETMSGOPTS.
   COPY CMQGMV.
** Length in bytes of the BUFFER area
01 BUFFERLENGTH PIC S9(9) BINARY.
** Area to contain the message data
01 BUFFER        PIC X(n).
** Length of the message
01 DATALENGTH  PIC S9(9) BINARY.
** Completion code
01 COMPCODE     PIC S9(9) BINARY.
** Reason code qualifying COMPCODE
01 REASON       PIC S9(9) BINARY.

```

## PL/I çağırımı

```

call MQGET (Hconn, Hobj, MsgDesc, GetMsgOpts, BufferLength, Buffer,
DataLength, CompCode, Reason);

```

Değiştirgeleri aşağıdaki gibi bildirin:

```

dcl Hconn      fixed bin(31); /* Connection handle */
dcl Hobj       fixed bin(31); /* Object handle */
dcl MsgDesc    like MQMD;     /* Message descriptor */
dcl GetMsgOpts like MQGMO;    /* Options that control the action of
MQGET */
dcl BufferLength fixed bin(31); /* Length in bytes of the Buffer
area */
dcl Buffer      char(n);       /* Area to contain the message data */
dcl DataLength fixed bin(31); /* Length of the message */
dcl CompCode   fixed bin(31); /* Completion code */
dcl Reason     fixed bin(31); /* Reason code qualifying CompCode */

```

## High Level Assembler çağırısı

```

CALL MQGET,(HCONN,HOBj,MSGDESC,GETMSGOPTS,BUFFERLENGTH,
BUFFER,DATALENGTH,COMPCODE,REASON)

```

Değiştirgeleri aşağıdaki gibi bildirin:

```

HCONN      DS      F      Connection handle

```



HOBJ	DS	F	Object handle
MSGDESC	CMQMDA	,	Message descriptor
GETMSGOPTS	CMQGMOA	,	Options that control the action of MQGET
BUFFERLENGTH	DS	F	Length in bytes of the BUFFER area
BUFFER	DS	CL(n)	Area to contain the message data
DATALength	DS	F	Length of the message
COMPCode	DS	F	Completion code
REASON	DS	F	Reason code qualifying COMPCode

## Visual Basic çağırısı

```
MQGET Hconn, Hobj, MsgDesc, GetMsgOpts, BufferLength, Buffer,
DataLength, CompCode, Reason
```

Değişirgeleri aşağıdaki gibi bildirin:

```
Dim Hconn As Long 'Connection handle'
Dim Hobj As Long 'Object handle'
Dim MsgDesc As MQMD 'Message descriptor'
Dim GetMsgOpts As MQGMO 'Options that control the action of MQGET'
Dim BufferLength As Long 'Length in bytes of the Buffer area'
Dim Buffer As String 'Area to contain the message data'
Dim DataLength As Long 'Length of the message'
Dim CompCode As Long 'Completion code'
Dim Reason As Long 'Reason code qualifying CompCode'
```

## MQINQ-Nesne özniteliklerinin sorgulması

MQINQ çağırısı, bir tamsayılar dizisi ve bir nesnenin özniteliklerini içeren bir karakter dizgileri kümesini döndürür.

Aşağıdaki nesne tipleri geçerlidir:

- Kuyruk yöneticisi
- Kuyruk
- Ad Listesi
- Süreç tanımlaması

## Sözdizimi

MQINQ (*Hconn, Hobj, SelectorCount, Selectors, IntAttrCount, IntAttrs, CharAttrLength, CharAttrs, CompCode, Reason*)

## Parametreler

### Hconn

Tip: MQHCONN -giriş

Bu tanırtıcı, kuyruk yöneticisiyle olan bağlantıyı gösterir. The value of *Hconn* was returned by a previous MQCONN or MQCONNX call.

On z/OS for CICS applications the MQCONN call can be omitted, and the following value specified for *Hconn* :

### MQHC\_DEF\_HCONN

Varsayılan bağlantı tanırtıcısı.

### Hobj

Tip: MQHOBJ -giriş

Bu tanırtıcı, gerekli özniteliklere sahip nesneyi (herhangi bir tipte) gösterir. Tanırtıcı, MQ00\_SORGULAMA seçeneğini belirten önceki bir MQOPEN çağırısı tarafından döndürülmelidir.

## SelectorCount

Tip: MQLONG -giriş

Bu, *Selectors* dizisinde sağlanan seçicilerin sayısıdır. Döndürülemekte olan özniteliklerin sayısıdır. Sıfır geçerli bir değerdir. İzin verilen sayı üst sınırı 256 'tır.

## Seçiciler

Tip: MQLONG x *SelectorCount* -giriş

Bu, **SelectorCount** öznitelik seçicilerinden oluşan bir dizidir; her seçici, bir özniteliği (tamsayı ya da karakter) zorunlu bir değerle tanımlar.

Her seçici, *Hobj* ' in temsil ettiği nesne türü için geçerli olmalıdır; aksi halde çağrı, tamamlanma kodu MQCC\_FAILED ve neden kodu MQRC\_SELECTOR\_ERROR ile başarısız olur.

Özel kuyruklar halinde:

- Seçici herhangi bir tipteki kuyruklar için geçerli değilse, çağrı tamamlanma kodu MQCC\_FAILED ve neden kodu MQRC\_SELECTOR\_ERROR ile başarısız olur.
- Seçici yalnızca nesnenin tipi dışındaki tiplerin kuyrukları için geçerliyse, çağrı tamamlanma kodu MQCC\_WARNING ve neden kodu MQRC\_SELECTOR\_NOT\_FOR\_TYPE ile başarılı olur.
- Sorgulanmakta olan kuyruk bir küme kuyruğuna, geçerli olan seçiciler, kuyruğun nasıl çözümlendiğine bağlıdır; ayrıntılı bilgi için "[Kullanım notları](#)" sayfa 707 konusuna bakın.

Seçicileri herhangi bir siparişte belirtebilirsiniz. Attribute values that correspond to integer attribute selectors (MQIA\_\* selectors) are returned in *IntAttrs* in the same order in which these selectors occur in *Selectors*. Attribute values that correspond to character attribute selectors (MQCA\_\* selectors) are returned in *CharAttrs* in the same order in which those selectors occur. MQIA\_\* seçicileri, MQCA\_\* seçicileri ile birlikte kullanılabilir; yalnızca her tip içindeki görelî sıralama önemlidir.

### Not:

1. Tamsayı ve karakter özniteliği seçicileri farklı iki aralık içinde ayrılır; MQIA\_\* seçicileri, MQIA\_FIRST - MQIA\_LAST aralığında ve MQCA\_FIRST - MQCA\_LAST aralığındaki MQCA\_\* seçicileri arasında yer alır.  
Her bir aralık için, MQIA\_LAST\_USED ve MQCA\_LAST\_USED değişmezleri kuyruk yöneticisinin kabul ettiği en yüksek değeri tanımlar.
2. Tüm MQIA\_\* seçicileri önce gerçekleşirse, *Selectors* ve *IntAttrs* dizilerinde karşılık gelen öğeleri karşılamak için aynı öğe numaraları kullanılabilir.
3. **SelectorCount** parametresi sıfırsa, *Selectors* ifadesine başvurulmaz. Bu durumda, C ya da S/390 çevirici programlarında yazılan programlarla geçirilen parametre adresi boş değerli olabilir.

Sorgulanabilen öznitelikler aşağıdaki tablolarda listelenir. For the MQCA\_\* selectors, the constant that defines the length in bytes of the resulting string in *CharAttrs* is provided in parentheses.

Aşağıdaki gibi alfabetik sırayla, seçicileri listeleyen tablolar alfabetik sırada:

- Kuyruklar için [Çizelge 550 sayfa 695](#) MQINQ öznitelik seçicileri
- Ad listeleri için [Çizelge 551 sayfa 697](#) MQINQ öznitelik seçicileri
- Süreç tanımlamaları için [Çizelge 552 sayfa 697](#) MQINQ öznitelik seçicileri
- Kuyruk yöneticisi için [Çizelge 553 sayfa 698](#) MQINQ öznitelik seçicileri

Tüm seçiciler, **Not** sütununda belirtildiği şekilde, tüm IBM MQ platformlarında aşağıdaki gibi desteklenir:

### DEĞİLz/OS

hariç z/OS tüm platformlarda desteklenir

### z/OS

z/OS üzerinde desteklenen **yalnızca**

Çizelge 550. Kuyruklar içinMQINQ öznelik seçicileri			
Seçici	Alan uzunluğu	Tanım	Not
MQCA_ALTERATION_DATE	MQ_DATE_LENGTH	En son değiştirilme tarihi	
MQCA_ALTERATION_TIME	MQ_TIME_LENGTH	En son değiştirilme zamanı	
MQCA_BACKOUT_REQ_Q_NAME	MQ_Q_NAME_LENGTH	Gereğinden fazla geri alma isteği adı	
MQCA_BASE_Q_NAME	MQ_Q_NAME_LENGTH	Diğer adın çözdüğü kuyruğun adı	
MQCA_CF_STRUC_NAME	MQ_CF_STRUC_NAME_LENGTH	Bağlaşım olanağı yapısı adı	z/OS
MQCA_CLUS_CHL_NAME	MQ_CHANNEL_NAME_LENGTH	Bu kuyruğu bir iletim kuyruğu olarak kullanan, kümeli gönderen kanalının adı.	
MQCA_CLUSTER_NAME	MQ_CLUSTER_NAME_LENGTH	Küme adı	
MQCA_CLUSTER_NAMELIST	MQ_NAMELIST_NAME_LENGTH	Küme adı listesi	
MQCA_CREATION_DATE	MQ_CREATION_DATE_LENGTH	Kuyruk yaratma tarihi	
MQCA_CREATION_TIME	MQ_CREATION_TIME_LENGTH	Kuyruk yaratma zamanı	
MQCA_CUSTOM	MQ_CUSTOM_LENGTH	Yeni özellikler için özel öznelik	
MQCA_INITIATION_Q_NAME	MQ_Q_NAME_LENGTH	Başlatma kuyruğu adı	
MQCA_PROCESS_NAME	MQ_PROCESS_NAME_LENGTH	Süreç tanımlamasının adı	
MQCA_Q_DESC	MQ_Q_DESC_LENGTH	Kuyruk tanımlaması	
MQCA_Q_NAME	MQ_Q_NAME_LENGTH	Kuyruk adı	
MQCA_REMOTE_Q_MGR_NAME	MQ_Q_MGR_NAME_LENGTH	Uzak kuyruk yöneticisinin adı	
MQCA_REMOTE_Q_NAME	MQ_Q_NAME_LENGTH	Uzak kuyruk yöneticisinde bulunduğu şekilde uzak kuyruğun adı	
MQCA_STORAGE_CLASS	MQ_STORAGE_CLASS_LENGTH	Depolama sınıfının adı	z/OS
MQCA_TRIGGER_DATA	MQ_TRIGGER_DATA_LENGTH	Verileri tetikle	
MQCA_XMIT_Q_NAME	MQ_Q_NAME_LENGTH	İletim kuyruğu adı	
MQIA_ACCOUNTING_Q	MQLONG	Kuyruğa ilişkin muhasebe verilerinin toplanmasını denetler	DEĞİL z/OS
MQIA_BACKOUT_THRESHOLD	MQLONG	Geriletme eşiği	
MQIA_CLWL_Q_PRIORITY	MQLONG	Kuyruk önceliği	
MQIA_CLWL_Q_RANK	MQLONG	Kuyruk sırası	
MQIA_CLWL_USEQ	MQLONG	Uzak kuyrukları kullan	
MQIA_CURRENT_Q_DEPTH	MQLONG	Kuyruktaki ileti sayısı	
MQIA_DEF_BIND	MQLONG	Varsayılan bağ tanımı	

Çizelge 550. Kuyruklar içinMQINQ öznitelik seçicileri (devamı var)			
Seçici	Alan uzunluğu	Tanım	Not
MQIA_DEF_INPUT_OPEN_OPTION	MQLONG	Giriş için varsayılan açma seçeneği	
MQIA_DEF_PERSISTENCE	MQLONG	Varsayılan ileti kalıcılığı	
MQIA_DEF_PRIORITY	MQLONG	Varsayılan ileti önceliği	
MQIA_DEFINITION_TYPE	MQLONG	Kuyruk tanımlaması tipi	
MQIA_DIST_LISTS	MQLONG	Dağıtım listesi desteği	DEĞİL z/OS
MQIA_HARDEN_GET_BACKOUT	MQLONG	Arka arkaya sayma sayılıp sayılmayacağını	
MQIA_INDEX_TYPE	MQLONG	Kuyruk için bakım yapılan dizin tipi	z/OS
MQIA_INHIBIT_GET	MQLONG	Alma işlemlerine izin verilip verilmediği	
MQIA_INHIBIT_PUT	MQLONG	Koyma işlemlerine izin verilip verilmediği	
MQIA_MAX_MSG_LENGTH	MQLONG	İleti uzunluğu üst sınırı	
MQIA_MAX_Q_DEPTH	MQLONG	Kuyruktan izin verilen ileti sayısı üst sınırı	
MQIA_MSG_DELIVERY_SEQUENCE	MQLONG	İleti önceliğinin ilgili olup olmadığı	
MQIA_NPM_CLASS	MQLONG	Kalıcı olmayan iletiler için güvenilirlik düzeyi	
MQIA_OPEN_INPUT_COUNT	MQLONG	Kuyruğun giriş için açık olan MQOPEN çağrılarının sayısı	
MQIA_OPEN_OUTPUT_COUNT	MQLONG	Kuyruk çıkış için açık olan MQOPEN çağrılarının sayısı	
MQIA_PROPERTY_CONTROL	MQLONG	Özellik denetimi özniteliği	
MQIA_Q_DEPTH_HIGH_EVENT	MQLONG	Kuyruk derinliği yüksek olayları için denetim özniteliği	DEĞİL z/OS
MQIA_Q_DEPTH_HIGH_LIMIT	MQLONG	Kuyruk derinliği üst sınırı	DEĞİL z/OS
MQIA_Q_DEPTH_LOW_EVENT	MQLONG	Kuyruk derinliği düşük olayları için denetim özniteliği	DEĞİL z/OS
MQIA_Q_DEPTH_LOW_LIMIT	MQLONG	Kuyruk derinliği için alt sınır	DEĞİL z/OS
MQIA_Q_DEPTH_MAX_EVENT	MQLONG	Kuyruk derinliği olayları üst sınırı için denetim özniteliği	DEĞİL z/OS
MQIA_Q_SERVICE_INTERVAL	MQLONG	Kuyruk hizmeti aralığı sınırı	DEĞİL z/OS
MQIA_Q_SERVICE_INTERVAL_EVENT	MQLONG	Kuyruk hizmeti aralığı olayları için denetim özniteliği	DEĞİL z/OS
MQIA_Q_TYPE	MQLONG	Kuyruk tipi	

Çizelge 550. Kuyruklar içinMQINQ öznitelik seçicileri (devamı var)

Seçici	Alan uzunluğu	Tanım	Not
MQIA_QSG_DISP	MQLONG	Kuyruk paylaşımı grubu yok etme	z/OS
MQIA_RETENTION_INTERVAL	MQLONG	Kuyruk alıkoyma aralığı	
MQIA_SCOPE	MQLONG	Kuyruk tanımlaması kapsamı	DEĞİL z/OS
MQIA_SHAREABILITY	MQLONG	Kuyruğun giriş için paylaşılıp paylaşılacağı	
MQIA_STATISTICS_Q	MQLONG	Kuyruğa ilişkin istatistik verilerinin toplanmasını denetler	DEĞİL z/OS
MQIA_TRIGGER_CONTROL	MQLONG	Tetikleyici denetimi	
MQIA_TRIGGER_DEPTH	MQLONG	Tetikleyici derinliği	
MQIA_TRIGGER_MSG_PRIORITY	MQLONG	Tetikleyiciler için eşik iletisi önceliği	
MQIA_TRIGGER_TYPE	MQLONG	Tetikleme Tipi	
MQIA_USAGE	MQLONG	Kullanım	

Çizelge 551. Ad listeleri içinMQINQ öznitelik seçicileri

Seçici	Alan uzunluğu	Tanım	Not
MQCA_ALTERATION_DATE	MQ_DATE_LENGTH	En son değiştirilme tarihi	
MQCA_ALTERATION_TIME	MQ_TIME_LENGTH	En son değiştirilme zamanı	
MQCA_NAMELIST_DESC	MQ_NAMELIST_DESC_LENGTH	Ad listesi açıklaması	
MQCA_NAMELIST_NAME	MQ_NAMELIST_NAME_LENGTH	Ad listesi nesnesinin adı	
MQIA_NAMELIST_TYPE	MQLONG	Ad listesi tipi	z/OS
MQCA_NAMES	MQ_Q_NAME_LENGTH x Number of names in the list	Ad listesindeki adlar	
MQIA_NAME_COUNT	MQLONG	Namelist 'teki ad sayısı	
MQIA_QSG_DISP	MQLONG	Kuyruk paylaşımı grubu yok etme	z/OS

Çizelge 552. Süreç tanımlamaları içinMQINQ öznitelik seçicileri

Seçici	Alan uzunluğu	Tanım	Not
MQCA_ALTERATION_DATE	MQ_DATE_LENGTH	En son değiştirilme tarihi	
MQCA_ALTERATION_TIME	MQ_TIME_LENGTH	En son değiştirilme zamanı	
MQCA_APPL_ID	MQ_PROCESS_APPL_ID_LENGTH	Uygulama tanıtıcısı	
MQCA_ENV_DATA	MQ_PROCESS_ENV_DATA_LENGTH	Ortam verileri	

<i>Çizelge 552. Süreç tanımlamaları içinMQINQ öznitelik seçicileri (devamı var)</i>			
<b>Seçici</b>	<b>Alan uzunluğu</b>	<b>Tanım</b>	<b>Not</b>
MQCA_PROCESS_DESC	MQ_PROCESS_DESC_LENGTH	Süreç tanımlamasının tanımı	
MQCA_PROCESS_NAME	MQ_PROCESS_NAME_LENGTH	Süreç tanımlamasının adı	
MQCA_USER_DATA	MQ_PROCESS_USER_DATA_LENGTH	Kullanıcı verileri	
MQIA_APPL_TYPE	MQLONG	Uygulama tipi	
MQIA_QSG_DISP	MQLONG	Kuyruk paylaşımı grubu yok etme	z/OS

<i>Çizelge 553. Kuyruk yöneticisi içinMQINQ öznitelik seçicileri</i>			
<b>Seçici</b>	<b>Alan uzunluğu</b>	<b>Tanım</b>	<b>Not</b>
MQCA_ALTERATION_DATE	MQ_DATE_LENGTH	En son değiştirilme tarihi	
MQCA_ALTERATION_TIME	MQ_TIME_LENGTH	En son değiştirilme zamanı	
MQCA_CHANNEL_AUTO_DEF_EXIT	MQ_EXIT_NAME_LENGTH	Otomatik kanal tanımlama çıkış adı	
MQCA_CHINIT_SERVICE_PARM		IBMtarafından kullanılmak üzere ayrılmış	
MQCA_CLUSTER_WORKLOAD_DATA	MQ_EXIT_DATA_LENGTH	Küme iş yükü çıkışa aktarılan veriler	
MQCA_CLUSTER_WORKLOAD_EXIT	MQ_EXIT_NAME_LENGTH	Küme iş yükü çıkışının adı	
MQCA_COMMAND_INPUT_Q_NAME	MQ_Q_NAME_LENGTH	Sistem komutu giriş kuyruğu adı	
MQCA_CUSTOM	MQ_CUSTOM_LENGTH	Yeni özellikler için özel öznitelik	
MQCA_DEAD_LETTER_Q_NAME	MQ_Q_NAME_LENGTH	Ölü-mektup kuyruğunun adı	
MQCA_DEF_XMIT_Q_NAME	MQ_Q_NAME_LENGTH	Varsayılan iletim kuyruğu adı	
MQCA_DNS_GROUP	MQ_DNS_GROUP_NAME_LENGTH	Birleştirme işlemi için kuyruk paylaşım grubu için gelen iletimleri işleyen TCP dinleyicisine ilişkin grubun adı. Ad, İş Yükü Yöneticisi Dinamik Etki Alanı Adı Hizmetleri 'ni kullanırken geçerlidir.	z/OS
MQCA_IGQ_USER_ID	MQ_USER_ID_LENGTH	Grup içi kuyruğa alma kullanıcı kimliği	z/OS
MQCA_INSTALLATION_DESC	MQ_INSTALLATION_DESC_LENGTH	İlişkili kuruluşun açıklaması	z/OSd eğil. DEĞİ LIBM i


Çizelge 553. Kuyruk yöneticisi için MQINQ öznelik seçicileri (devamı var)

Seçici	Alan uzunluğu	Tanım	Not
MQCA_INSTALLATION_NAME	MQ_INSTALLATION_NAME_LENGTH	Kuyruk yöneticisiyle ilişkilendirilmiş kuruluşun adı	z/OSd eğil. DEĞİ LIBM i
MQCA_INSTALLATION_PATH	MQ_INSTALLATION_PATH_LENGTH	İlişkili IBM MQ ' in kurulu olduğu yol	z/OSd eğil. DEĞİ LIBM i
MQCA_LU_GROUP_NAME	MQ_LU_NAME_LENGTH	Kullanılacak kuyruk paylaşım grubu için gelen iletimleri işleyen LU 6.2 dinleyicisine ilişkin genel LU adı	z/OS
MQCA_LU_NAME	MQ_LU_NAME_LENGTH	Giden LU 6.2 iletimleri için kullanılacak LU ' nun adı. Bu adı, dinleyici tarafından gelen iletimler için kullandığı LU ' ya ayarlayın	z/OS
MQCA_LU62_ARM_SUFFIX	MQ_ARM_SUFFIX_LENGTH	Suffix of the SYS1 . PARMLIB member APPCPM <i>xx</i> , that nominates the LUADD for this channel initiator	z/OS
MQCA_PARENT	MQ_Q_MGR_NAME_LENGTH	Bu kuyruk yöneticisinin üst ögesi olarak aday gösterilen, sıradüzensel olarak bağlı bir kuyruk yöneticisinin adı.	
MQCA_Q_MGR_DESC	MQ_Q_MGR_DESC_LENGTH	Kuyruk yöneticisi tanımı	
MQCA_Q_MGR_IDENTIFIER	MQ_Q_MGR_IDENTIFIER_LENGTH	Kuyruk yöneticisi tanıtıcısı (H)	
MQCA_Q_MGR_NAME	MQ_Q_MGR_NAME_LENGTH	Yerel kuyruk yöneticisinin adı	
MQCA_QSG_NAME	MQ_QSG_NAME_LENGTH	Kuyruk paylaşım grubu adı	z/OS
MQCA_REPOSITORY_NAME	MQ_CLUSTER_NAME_LENGTH	Kuyruk yöneticisinin havuz hizmetleri sağladığı kümenin adı	
MQCA_REPOSITORY_NAMELIST	MQ_NAMELIST_NAME_LENGTH	Kuyruk yöneticisinin havuz hizmetleri sağladığı kümelerin adlarını içeren ad listesi nesnesinin adı	
MQCA_TCP_NAME	MQ_TCP_NAME_LENGTH	Kullanmakta olduğunuz TCP/IP sisteminin adı	z/OS
MQIA_ACCOUNTING_CONN_OVERRIDE	MQLONG	Hesap ayarlarını geçersiz kıl	DEĞİ LZ/O S
MQIA_ACCOUNTING_INTERVAL	MQLONG	ara muhasebe kayıtları ne sıklıkta yazılıyor?	DEĞİ LZ/O S

Çizelge 553. Kuyruk yöneticisi içinMQINQ öznelik seçicileri (devamı var)			
Seçici	Alan uzunluğu	Tanım	Not
MQIA_ACCOUNTING_MQI	MQLONG	MQI verilerine ilişkin muhasebe bilgilerinin toplanmasını denetler	DEĞİ Lz/O S
MQIA_ACCOUNTING_Q	MQLONG	Kuyruklara ilişkin muhasebe bilgilerinin toplanmasını denetler	DEĞİ Lz/O S
MQIA_ACTIVE_CHANNELS	MQLONG	Herhangi bir zamanda etkin olabilecek kanal sayısı üst sınırı	z/OS
MQIA_ADOPTNEWMCA_CHECK	MQLONG	MCA ' yı kabul edip etmeyeceğini belirlemek için denetlenen öğeler. Denetim, önceden etkin bir MCA ile aynı ada sahip yeni bir gelen kanal saptandığında gerçekleştirilir.	z/OS
MQIA_ADOPTNEWMCA_INTERVAL	MQLONG	Yeni kanalın artık kalan kanalı sona erdirmek için bekleyeceği süre (saniye olarak)	DEĞİ Lz/O S
MQIA_ADOPTNEWMCA_TYPE	MQLONG	AdoptNewMCACheck değıştirmeleriyle eşleşen yeni bir gelen kanal isteęi saptandığında, belirli bir kanal tipinin MCA ' nın artık otomatik olarak yeniden başlatılıp başlatılmayacağı	z/OS
MQIA_AUTHORITY_EVENT	MQLONG	Yetki olaylarına ilişkin denetim öznelięi	DEĞİ Lz/O S
MQIA_BRIDGE_EVENT	MQLONG	IMS köprüsü olaylarına ilişkin denetim öznelięi	z/OS
MQIA_CHANNEL_AUTO_DEF	MQLONG	Otomatik kanal tanımlamasına ilişkin denetim öznelięi	DEĞİ Lz/O S
MQIA_CHANNEL_AUTO_DEF_EVENT	MQLONG	Otomatik kanal tanımlaması olaylarına ilişkin denetim öznelięi	DEĞİ Lz/O S
MQIA_CHANNEL_EVENT	MQLONG	Kanal olaylarına ilişkin denetim öznelięi	
MQIA_CHINIT_ADAPTERS	MQLONG	IBM MQ çağrılarını işlemek için kullanılacak bağdařtırıcı alt görevlerinin sayısı	z/OS
MQIA_CHINIT_DISPATCHERS	MQLONG	Kanal başlatıcısı için kullanılacak dağıtıcıların sayısı	z/OS
MQIA_CHINIT_TRACE_AUTO_START	MQLONG	Kanal başlatıcı izlemesinin otomatik olarak başlatılıp başlatılmayacağını	z/OS
MQIA_CHINIT_TRACE_TABLE_SIZE	MQLONG	Kanal başlatıcısının izleme veri alanının büyüklüęü (MB)	z/OS
MQIA_CLUSTER_WORKLOAD_LENGTH	MQLONG	Küme iş yükü uzunluęu.	



Çizelge 553. Kuyruk yöneticisi içinMQINQ öznelik seçicileri (devamı var)			
Seçici	Alan uzunluğu	Tanım	Not
MQIA_CLWL_MRU_CHANNELS	MQLONG	Küme iş yükü dengelemesi için en son kullanılan kanalların sayısı	
MQIA_CLWL_USEQ	MQLONG	Uzak kuyrukları kullan	
MQIA_CODED_CHAR_SET_ID	MQLONG	Kodlanmış karakter takımı tanıtıcısı	
MQIA_COMMAND_EVENT	MQLONG	Komut olaylarına ilişkin denetim özneliği	
MQIA_COMMAND_LEVEL	MQLONG	Kuyruk yöneticisi tarafından desteklenen komut düzeyi	
MQIA_CONFIGURATION_EVENT	MQLONG	Yapılandırma olaylarına ilişkin denetim özneliği	DEĞİ Lz/O S
MQIA_DEF_CLUSTER_XMIT_Q_TYPE	MQLONG	Kümeyi gönderen kanallar için kullanılacak varsayılan iletim kuyruğu tipi.	
MQIA_DIST_LISTS	MQLONG	Dağıtım listesi desteği	DEĞİ Lz/O S
MQIA_DNS_WLM	MQLONG	Kuyruk paylaşım grubu için gelen iletimleri işleyen TCP dinleyicisinin, Dinamik Etki Alanı Adı Hizmetleri için İş Yükü Yöneticisi ile kaydolup olmadığını	z/OS
MQIA_EXPIRY_INTERVAL	MQLONG	Süresi dolan iletiler için taramalar arasındaki aralık	z/OS
MQIA_GROUP_UR	MQLONG	Bu kuyruk yöneticisi için, grup kurtarma birimlerinin etkinleştirilip etkinleştirilmediğine ilişkin denetim özneliği. GROUP birimi kurtarma işlemi, yalnızca kuyruk yöneticisi bir kuyruk paylaşım grubunun üyesiye kullanılabilir.	z/OS
MQIA_IGQ_PUT_AUTHORITY	MQLONG	Grup içi kuyruğa alma yetkisi	z/OS
MQIA_INHIBIT_EVENT	MQLONG	Olayları engelleyici için denetim özneliği	DEĞİ Lz/O S
MQIA_INTRA_GROUP_queuing	MQLONG	Grup içi kuyruğa alma desteği	z/OS
MQIA_LISTENER_TIMER	MQLONG	APPC ya da TCP/IP başarısız olursa, IBM MQ dinleyicisini yeniden başlatma girişimleri arasındaki zaman aralığı (saniye olarak).	z/OS
MQIA_LOCAL_EVENT	MQLONG	Yerel olaylara ilişkin denetim özneliği	DEĞİ Lz/O S

Çizelge 553. Kuyruk yöneticisi içinMQINQ öznitelik seçicileri (devamı var)			
Seçici	Alan uzunluğu	Tanım	Not
MQIA_LOGGER_EVENT	MQLONG	Olayları engelleyici için denetim özniteliği	DEĞİ Lz/O S
MQIA_LU62_CHANNELS	MQLONG	LU 6.2 iletim protokolü kullanılarak, geçerli olabilecek kanal sayısı üst sınırı ya da bağlanabilen istemci sayısı	z/OS
MQIA_MSG_MARK_BROWSE_INTERVAL	MQLONG	Kuyruk yöneticisinin, iletilere göz atma iminden otomatik olarak bir işareti kaldırabileceği zaman aralığı (milisaniye).  <b>Uyarı:</b> Bu değeri varsayılan 5000 değerinin altında ayarlamamalısınız.	
MQIA_MAX_CHANNELS	MQLONG	Geçerli olabilecek kanal sayısı üst sınırı (bağlı istemcilere sahip sunucu bağlantı kanalları da içinde olmak üzere)	z/OS
MQIA_MAX_HANDLES	MQLONG	Tanıtıcı sayısı üst sınırı	
MQIA_MAX_MSG_LENGTH	MQLONG	İleti uzunluğu üst sınırı	
MQIA_MAX_PRIORITY	MQLONG	En yüksek öncelik	
MQIA_MAX_UNCOMMITTED_MESSAGES	MQLONG	Bir iş birimi içinde kesinleştirilmemiş ileti sayısı üst sınırı	
MQIA_OUTBOUND_PORT_MAX	MQLONG	MQIA_OUTBOUND_PORT_MINile, giden kanalları bağlarken kullanılacak kapı numaralarının aralığını tanımlar	z/OS
MQIA_OUTBOUND_PORT_MIN	MQLONG	MQIA_OUTBOUND_PORT_MAXile, giden kanalları bağlarken kullanılacak kapı numaralarının aralığını tanımlar	z/OS
MQIA_PERFORMANCE_EVENT	MQLONG	Performans olayları için denetim özniteliği	DEĞİ Lz/O S
MQIA_PLATFORM	MQLONG	Kuyruk yöneticisinin bulunduğu altyapı	
MQIA_PROT_POLICY_CAPABILITY	MQLONG	Bir kuyruk yöneticisi için Advanced Message Security güvenlik yeteneklerinin kullanılabilir olup olmadığını belirtir.	
MQIA_PUBSUB_MAXMSG_RETRY_COUNT	MQLONG	Eşitleme noktası altında başarısız olan bir komut iletisini yeniden işleme girişimi sayısı	

Çizelge 553. Kuyruk yöneticisi içinMQINQ öznelik seçicileri (devamı var)

Seçici	Alan uzunluğu	Tanım	Not
MQIA_PUBSUB_MODE	MQLONG	Yayınlama/abone olma motorunun ve kuyruğa alınan yayınlama/abone olma arabiriminin çalışıp çalışmadığı. Uygulama programlama arabirimini kullanarak yayınlamak ya da abone olmak için uygulamalar yayınlama/abone olma motorunu gerektirir. Kuyruğa alınan yayınlama/abone olma arabirimi tarafından izlenen kuyruklar, kuyruğa alınan yayınlama/abone olma arabirimini çalışır durumda gerektirir.	
MQIA_PUBSUB_NP_MSG	MQLONG	Teslim edilmemiş bir giriş iletisinin atılıp atılmayacağını (ya da alıkoymak)	
MQIA_PUBSUB_NP_RESP	MQLONG	Teslim edilemeyen yanıt iletilerinin davranışını denetler	
MQIA_PUBSUB_SYNC_PT	MQLONG	Yalnızca kalıcı (ya da tüm) iletilerin eşitleme noktası altında işlenip işlenmediği	
MQIA_QMGR_CFCONLOS	MQLONG	Kuyruk yöneticisi yönetim yapısıyla ya da CFCONLOS ile ASQMGRdeğerine ayarlanmış CF yapılarına bağlılığı kaybettiğinde yapılacak işlemi belirtir.	z/OS
MQIA_RECEIVE_TIMEOUT	MQLONG	Bir TCP/IP kanalının, iş ortasından, etkin olmayan duruma dönmeden önce kalp atışları da içinde olmak üzere veri almayı yaklaşık olarak ne kadar bekleyeceğini belirler. Değer, MQIA_RECEIVE_TIMEOUT_TYPEile nitelenen sayısal değerdir.	z/OS
MQIA_RECEIVE_TIMEOUT_MIN	MQLONG	Etkin olmayan duruma dönmeden önce, bir TCP/IP kanalının iş ortasından kalp atışları da içinde olmak üzere veri almak için bekleyeceği en az süre	z/OS
MQIA_RECEIVE_TIMEOUT_TYPE	MQLONG	Bir TCP/IP kanalının, iş ortasından, etkin olmayan duruma dönmeden önce kalp atışları da içinde olmak üzere veri almayı yaklaşık olarak ne kadar bekleyeceğini belirler. MQIA_RECEIVE_TIMEOUT_TYPE , MQIA_RECEIVE_TIMEOUTiçin uygulanan niteleyicidir.	z/OS
MQIA_REMOTE_EVENT	MQLONG	Uzak olaylar için denetim özneliği	DEĞİ Lz/O S
MQIA_SECURITY_CASE	MQLONG	Güvenlik profilleri vakası	z/OS
MQIA_SSL_EVENT	MQLONG	Kanal olaylarına ilişkin denetim özneliği	

Çizelge 553. Kuyruk yöneticisi için MQINQ öznitelik seçicileri (devamı var)			
Seçici	Alan uzunluğu	Tanım	Not
MQIA_SSL_FIPS_REQUIRED	MQLONG	Kriptografi için yalnızca FIPS onaylı algoritmalar kullan	
MQIA_SSL_RESET_COUNT	MQLONG	TLS anahtarı sıfırlama sayısı	
MQIA_START_STOP_EVENT	MQLONG	Durdurma olaylarının başlatılması için denetim özniteliği	DEĞİ Lz/O S
MQIA_STATISTICS_AUTO_CLUSTER	MQLONG	Küme gönderen kanallarına ilişkin istatistik izleme bilgilerinin derlemeni denetler	
MQIA_STATISTICS_CHANNEL	MQLONG	Kanallar için istatistik verilerinin toplanmasını denetler	
MQIA_STATISTICS_INTERVAL	MQLONG	İstatistik izleme verileri yazma sıklığı	DEĞİ Lz/O S
MQIA_STATISTICS_MQI	MQLONG	Kuyruk yöneticisine ilişkin istatistik izleme bilgilerinin derlemeni denetler	DEĞİ Lz/O S
MQIA_STATISTICS_Q	MQLONG	Kuyruklar için istatistik verilerinin toplanmasını denetler	DEĞİ Lz/O S
MQIA_SYNCPOINT	MQLONG	eşitleme noktası kullanılabilirliği	
MQIA_TCP_CHANNELS	MQLONG	TCP/IP iletim protokolünü kullanarak, geçerli olabilecek kanal sayısı üst sınırı ya da bağlanabilen istemci sayısı	z/OS
MQIA_TCP_KEEP_ALIVE	MQLONG	Bağlantının diğer ucunun hala kullanılabilir olup olmadığını denetlemek için TCP KEPESAVE tesisinin kullanılmasının gerekip gerekmediğini	z/OS
MQIA_TCP_STACK_TYPE	MQLONG	Kanal başlatıcının yalnızca TCPNAME içinde belirlenen TCP/IP adres alanını kullanıp kullanamayacağı ya da isteğe bağlı olarak seçilen herhangi bir TCP/IP adresine bağ tanımlayıp kullanamayacağı	z/OS
MQIA_TRACE_ROUTE_RECORDING	MQLONG	İzleme rotası bilgilerinin kaydedilmesini denetler	z/OS
MQIA_TREE_LIFE_TIME	MQLONG	Kullanılmayan yönetimle ilgili olmayan konuların geçerlilik süresi	
MQIA_TRIGGER_INTERVAL	MQLONG	Tetikleme aralığı	

#### IntAttrSayı

Tip: MQLONG -giriş

Bu, *IntAttrs* dizindeki öğelerin sayısıdır. Sıfır geçerli bir değerdir.

`IntAttrCount` , en az **Selectors** parametresindeki `MQIA_*` seçicilerinin sayısını gösteriyorsa, istenen tüm tamsayı öznitelikleri döndürülür.

### IntAttrs

Tip: `MQLONG` x `IntAttrCount` -çıkış

Bu, `IntAttrCount` tamsayı öznitelik değerlerinden oluşan bir dizidir.

Tamsayı öznitelik değerleri, **Selectors** parametresindeki `MQIA_*` seçicileriyle aynı sırayla döndürülür. Dizi, `MQIA_*` seçicilerinin sayısından daha fazla öğe içeriyorsa, fazla öğeler değiştirilmez.

If `Hobj` represents a queue, but an attribute selector does not apply to that type of queue, the specific value `MQIAV_NOT_APPLICABLE` is returned. `IntAttrs` dizisine karşılık gelen öğe için döndürülür.

**IntAttrCount** ya da **SelectorCount** parametresi sıfırsa, `IntAttrs` bu parametreye başvuruda bulunmaz. Bu durumda, C ya da S/390 çevirici programlarında yazılan programlarla geçirilen parametre adresi boş değerli olabilir.

### CharAttrUzunluğu

Tip: `MQLONG` -giriş

Bu, **CharAttrs** parametresindeki bayt cinsinden uzunluğudur.

`CharAttrUzunluğu` , istenen karakter özniteliklerinin uzunluklarının en az toplamını olmalıdır (bkz. Seçiciler ). Sıfır geçerli bir değerdir.

### CharAttrs

Tip: `MQCHAR` x `CharAttrLength` -çıkış

Bu arabellek, karakter özniteliklerinin birlikte döndürüldüğü, birleştirilen arabelleğidir. Arabellek uzunluğu **CharAttrLength** parametresiyle verilir.

Karakter öznitelikleri, **Selectors** parametresindeki `MQCA_*` seçicileriyle aynı sırayla döndürülür. Her bir öznitelik için her öznitelik dizgisinin uzunluğu değişmez (bkz. Seçiciler ) ve bunun içindeki değer, gerekirse boşluklarla doldurulmuştur. İstenen tüm karakter özniteliklerini ve doldurmaları içermek için gerekenden daha büyük bir arabellek sağlamanız gerekir. Döndürülen son öznitelik değerinin ötesindeki byte 'lar değiştirilmeden kalır.

`Hobj` bir kuyruğu temsil ediyorsa, ancak bir öznitelik seçici o kuyruk tipine uygulanmaz ise, tamamen yıldız işaretleri (\*) içeren bir karakter dizgisi döndürülür. Yıldız işareti, `CharAttrs` içindeki bu öznitelik değeri olarak döndürülür.

`CharAttrLength` ya da **SelectorCount** parametresi sıfırsa, `CharAttrs` bu parametreye başvuruda bulunmaz. Bu durumda, C ya da S/390 çevirici programlarında yazılan programlarla geçirilen parametre adresi boş değerli olabilir.

### CompCode

Tip: `MQLONG` -çıkış

Tamamlanma kodu:

#### **MQCC\_OK**

İşlem başarıyla tamamlandı.

#### **MQCC\_WARNING**

Uyarı (kısmi tamamlama).

#### **MQCC\_FAILED**

Arama başarısız oldu.

### Neden

Tip: `MQXX_ENCODE_CASE_ONE` long-output

`CompCode` `MQCC_OK` ise:

#### **MQRC\_NONE**

(0, X'000') Raporlamak için herhangi bir neden yok.

`CompCode` `MQCC_WARNING` ise:

**MQRD\_CHAR\_ATTRS\_TOO\_SHORT**

(2008, X'7D8') Karakter öznitelikleri için yeterli yer yok.

**MQRD\_INT\_ATTR\_COUNT\_TOO\_SMALL**

(2022, X'7E6') Tamsayı öznitelikleri için yeterli yer izni yok.

**MQRD\_SELECTOR\_NOT\_FOR\_TYPE**

(2068, X'814') Seçici, kuyruk tipi için geçerli değil.

*CompCode* MQRD\_FAILEDise:

**MQRD\_ADAPTER\_NOT\_AVAILABLE**

(2204, X'89C') Bağdaştırıcı kullanılmıyor.

**MQRD\_ADAPTER\_SERV\_LOAD\_ERROR**

(2130, X'852') Bağdaştırıcı hizmeti modülü yüklenemiyor.

**MQRD\_API\_EXIT\_ERROR**

(2374, X'946') API çıkışı başarısız oldu.

**MQRD\_API\_EXIT\_LOAD\_ERROR**

(2183, X'887') API çıkışı yüklenemiyor.

**MQRD\_ASID\_MISMATCH**

(2157, X'86D') Birincil ve ana ASID ' ler farklı.

**MQRD\_CALL\_IN\_PROGRESS**

(2219, X'8AB') Önceki arama tamamlanmadan önce girilen MQI çağrısı.

**MQRD\_CF\_STRUC\_FAILED**

(2373, X'945') Coupling-tesis yapısı başarısız oldu.

**MQRD\_CF\_STRUC\_IN\_USE**

(2346, X'92A') Coupling-tesis yapısı kullanımda.

**MQRD\_CHAR\_ATTR\_LENGTH\_ERROR**

(2006, X'7D6') Karakter özniteliklerinin uzunluğu geçerli değil.

**MQRD\_CHAR\_ATTRS\_ERROR**

(2007, X'7D7') Karakter öznitelikleri dizisi geçerli değil.

**MQRD\_CICS\_WAIT\_FAILED**

(2140, X'85C') Bekleme isteği CICS tarafından reddedildi.

**MQRD\_CONNECTION\_BROKEN**

(2009, X'7D9') Kuyruk yöneticisine bağlantı kaybedildi.

**MQRD\_CONNECTION\_NOT\_AUTHORIZED**

(2217, X'8A9') Bağlantı için yetkili değil.

**MQRD\_CONNECTION\_STOPPING**

(2203, X'89B') Bağlantı sona erdiriliyor.

**MQRD\_HCONN\_ERROR**

(2018, X'7E2') Bağlantı tanıtıcısı geçerli değil.

**MQRD\_HOBJ\_ERROR**

(2019, X'7E3') Nesne tanıtıcısı geçerli değil.

**MQRD\_INT\_ATTR\_COUNT\_ERROR**

(2021, X'7E5') Tamsayı öznitelikleri sayısı geçerli değil.

**MQRD\_INT\_ATTRS\_ARRAY\_ERROR**

(2023, X'7E7') Tamsayı öznitelikleri dizisi geçerli değil.

**MQRD\_NOT\_OPEN\_FOR\_INQUIRE**

(2038, X'7F6') Kuyruk sorgulamak için açık değil.

**MQRD\_OBJECT\_CHANGED**

(2041, X'7F9') Nesne tanımlaması açıldığından bu yana değiştirildi.

**MQRD\_OBJECT\_DAMAGED**

(2101, X'835') Nesne zarar gördü.

**MQRC\_PAGESET\_ERROR**

( 2193, X' 891 ' ) Sayfa kümesi veri kümesine erişilirken hata oluştu.

**MQRC\_Q\_DELETED**

( 2052, X' 804 ' ) Kuyruk silindi.

**MQRC\_Q\_MGR\_NAME\_ERROR**

( 2058, X' 80A ' ) Kuyruk yöneticisi adı geçerli değil ya da bilinmiyor.

**MQRC\_Q\_MGR\_NOT\_AVAILABLE**

( 2059, X' 80B ' ) Bağlantı için kuyruk yöneticisi kullanılamıyor.

**MQRC\_Q\_MGR\_STOPPING**

( 2162, X' 872 ' ) Kuyruk yöneticisi sona erdiriliyor.

**MQRC\_RESOURCE\_PROBLEM**

( 2102, X' 836 ' ) Kullanılabilir sistem kaynakları yetersiz.

**MQRC\_SELECTOR\_COUNT\_ERROR**

( 2065, X' 811 ' ) Seçici sayısı geçerli değil.

**MQRC\_SELECTOR\_ERROR**

( 2067, X' 813 ' ) Öznitelik seçici geçerli değil.

**MQRC\_SELECTOR\_LIMIT\_EXCEEDED**

( 2066, X' 812 ' ) Seçicilerin sayısı çok büyük.

**MQRC\_STORAGE\_NOT\_AVAILABLE**

( 2071, X' 817 ' ) Kullanılabilir bellek yetersiz.

**MQRC\_SUPPRESSED\_BY\_EXIT**

( 2109, X' 83D ' ) Çıkış programı tarafından engellenen çağrı.

**MQRC\_UNEXPECTED\_ERROR**

( 2195, X' 893 ' ) Beklenmeyen hata oluştu.

Bu kodlara ilişkin ayrıntılı bilgi için [İletiler ve neden kodları](#) başlıklı konuya bakın.

**Kullanım notları**

1. Döndürülen değerler, seçilen özniteliklerin bir anlık görünüşüdür. Uygulamanın döndürülen değerler üzerinde işlem yapabilmesi için, özniteliklerin aynı kalması konusunda herhangi bir garanti verilmez.
2. Bir model kuyruğunu açtığınızda, dinamik bir yerel kuyruk yaratılır. Dinamik bir yerel kuyruk, öznitelikleri sorgulamak için model kuyruğunu açsanız bile yaratılır.  
  
Dinamik kuyruğun öznitelikleri, dinamik kuyruk yaratıldığı sırada, model kuyruğunun öznitelikleriyle aynı olur. Daha sonra bu kuyrukdaki MQINQ çağrısını kullanırsanız, kuyruk yöneticisi, model kuyruğunun özniteliklerini değil, dinamik kuyruğun özniteliklerini döndürür. Model kuyruğunun özniteliklerinin dinamik kuyruk tarafından devralındığı ayrıntıları görmek için bkz. [Çizelge 562 sayfa 824](#).
3. Sorgulanmakta olan nesne bir diğer ad kuyruğunsa, MQINQ çağrısı tarafından döndürülen öznitelik değerleri, diğer ad kuyruğunun öznitelikleridir. Diğer ad, diğer adın çözdüğü temel kuyruğun ya da konunun öznitelikleri değildir.
4. Sorgulanmakta olan nesne bir küme kuyruğysa, sorgulanabilen öznitelikler, kuyruğun nasıl açıldığı ile ilgili olarak değişir:
  - Bir sorgu artı bir ya da daha çok giriş, göz atma ya da ayarlama işlemi için bir küme kuyruğu açabilirsiniz. Bunu yapmak için, kümenin başarılı olması için küme kuyruğunun yerel bir eşgörünümü olmalıdır. Bu durumda, sorgulanabilen öznitelikler, yerel kuyruklar için geçerli olan özniteliklerdir.  
  
Giriş, göz atma ya da küme olmaksızın sorgulama işlemi için küme kuyruğu açıksa, yalnızca yerel kuyruklar için geçerli olan ve küme kuyrukları olmayan öznitelikleri sorgulamaya çalışırsanız, çağrı, tamamlanma kodu MQCC\_UYARI ve neden kodu MQC\_SELECTOR\_NOT\_FOR\_TYPE (2068) değerini döndürür.
  - Bağlı kuyruk yöneticisinin temel kuyruk yöneticisi adını geçirirken sorgulamak üzere bir küme kuyruğu açabilirsiniz.

Bunu yapmak için, kümenin başarılı olması için küme kuyruğunun yerel bir eşgörünümü olmalıdır. Temel kuyruk yöneticisi geçirilmezse, yalnızca yerel kuyruklar için geçerli olan ve küme kuyrukları için değil, yalnızca yerel kuyruklar için geçerli olan öznitelikleri sorgulamaya çalışırsanız, çağrı tamamlanma kodu MQCC\_UYARI ve neden kodu MQRC\_SELECTOR\_NOT\_FOR\_TYPE (2068) değerini döndürür.

- Küme kuyruğu tek başına sorgulanmak üzere açılırsa ya da sorgula ve çıktıyla, yalnızca listelenen öznitelikler sorgulanabilir. **QType** özneliği, bu durumda MQQT\_CLUSTER değerini içerir:

- MQCA\_Q\_DESC
- MQCA\_Q\_NAME
- MQIA\_DEF\_BIND
- MQIA\_DEF\_PERSISTENCE
- MQIA\_DEF\_PRIORITY
- MQIA\_INHIBIT\_PUT
- MQIA\_Q\_TYPE

Küme kuyruğunu değişmez bir bağ tanımı olmadan açabilirsiniz. Bunu MQOPEN çağrısında belirtilen MQOO\_BIND\_NOT\_FIXED ile açabilirsiniz. Diğer bir seçenek olarak, MQOO\_BIND\_AS\_Q\_DEF belirtisini belirtin ve kuyruğun **DefBind** özneliğini MQBND\_BIND\_NOT\_FIXED olarak ayarlayın. Değişmez bağ tanımı olmayan bir küme kuyruğunu açarsanız, kuyruk için art arda yapılan MQINQ çağrıları, küme kuyruğunun farklı eşgörünümlerini sorgulayabilir. Ancak, tipik olarak, tüm eşgörünümler için aynı öznitelik değerleri vardır.

- Bir küme için diğer ad kuyruğu nesnesi tanımlanabilir. TARGTYPE ve TARGET küme öznitelikleri olmadığından, diğer ad kuyruğunda bir MQOPEN işlemi gerçekleştiren işlem, diğer adın çözülen nesneden farkında değil.

İlk MQOPEN sırasında, diğer ad kuyruğu, bir kuyruk yöneticisine ve kümedeki bir kuyruğa çözümlenir. Ad çözme işlemi, uzak kuyruk yöneticisinde yeniden gerçekleşir ve diğer ad kuyruğunun TARGTPE 'nin çözüldüğü yerdir.

Diğer ad kuyruğu bir konu diğer adına çözümlerse, diğer ad kuyruğuna gönderilen iletilerin yayınlanması, bu uzak kuyruk yöneticisinde yer alır.

Bkz. [Küme kuyrukları](#)

5. Bir dizi özneliği sorgulamak ve daha sonra, bazılarını MQSET çağrısını kullanarak ayarlamak isteyebilirsiniz. Sorgulamak ve verimli bir şekilde ayarlamak için, ayırıcı dizilerinin başında ayarlanacak öznitelikleri konumlayın. If you do so, the same arrays with reduced counts can be used for MQSET.
6. Uyarı durumlarının birden fazlası ortaya çıkarsa (**CompCode** parametresine bakın), döndürülen neden kodu, aşağıdaki listede yer alan ilk sıradır:
  - a. MQRC\_SELECTOR\_NOT\_FOR\_TYPE
  - b. MQRC\_INT\_ATTR\_COUNT\_TOO\_SMALL
  - c. MQRC\_CHAR\_ATTRS\_TOO\_SHORT
7. Aşağıdaki konu nesne öznitelikleriyle ilgili bilgi içerir:
  - [“Kuyruklara ilişkin öznitelikler” sayfa 821](#)
  - [“Ad listelerine ilişkin öznitelikler” sayfa 854](#)
  - [“Süreç tanımlamalarına ilişkin öznitelikler” sayfa 856](#)
  - [“Kuyruk yöneticisine ilişkin öznitelikler” sayfa 785](#)

## C çağırma

```
MQINQ (Hconn, Hobj, SelectorCount, Selectors, IntAttrCount, IntAttrs,  
CharAttrLength, CharAttrs, &CompCode, &Reason);
```



Değiştirgeleri aşağıdaki gibi bildirin:

```
MQHCONN  Hconn;           /* Connection handle */
MQHOBJ   Hobj;           /* Object handle */
MQLONG   SelectorCount;  /* Count of selectors */
MQLONG   Selectors[n];   /* Array of attribute selectors */
MQLONG   IntAttrCount;   /* Count of integer attributes */
MQLONG   IntAttrs[n];    /* Array of integer attributes */
MQLONG   CharAttrLength; /* Length of character attributes buffer */
MQCHAR   CharAttrs[n];   /* Character attributes */
MQLONG   CompCode;       /* Completion code */
MQLONG   Reason;        /* Reason code qualifying CompCode */
```

## Cobol çağırma

```
CALL 'MQINQ' USING HCONN, HOBJ, SELECTORCOUNT, SELECTORS-TABLE,
                  INTATTRCOUNT, INTATTRS-TABLE, CHARATTRLENGTH,
                  CHARATTRS, COMPCODE, REASON.
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
** Connection handle
01 HCONN          PIC S9(9) BINARY.
** Object handle
01 HOBJ          PIC S9(9) BINARY.
** Count of selectors
01 SELECTORCOUNT PIC S9(9) BINARY.
** Array of attribute selectors
01 SELECTORS-TABLE.
02 SELECTORS     PIC S9(9) BINARY OCCURS n TIMES.
** Count of integer attributes
01 INTATTRCOUNT PIC S9(9) BINARY.
** Array of integer attributes
01 INTATTRS-TABLE.
02 INTATTRS     PIC S9(9) BINARY OCCURS n TIMES.
** Length of character attributes buffer
01 CHARATTRLENGTH PIC S9(9) BINARY.
** Character attributes
01 CHARATTRS     PIC X(n).
** Completion code
01 COMPCODE      PIC S9(9) BINARY.
** Reason code qualifying COMPCODE
01 REASON       PIC S9(9) BINARY.
```

## PL/I çağırımı

```
call MQINQ (Hconn, Hobj, SelectorCount, Selectors, IntAttrCount,
            IntAttrs, CharAttrLength, CharAttrs, CompCode, Reason);
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
dcl Hconn          fixed bin(31); /* Connection handle */
dcl Hobj          fixed bin(31); /* Object handle */
dcl SelectorCount  fixed bin(31); /* Count of selectors */
dcl Selectors(n)   fixed bin(31); /* Array of attribute selectors */
dcl IntAttrCount   fixed bin(31); /* Count of integer attributes */
dcl IntAttrs(n)    fixed bin(31); /* Array of integer attributes */
dcl CharAttrLength fixed bin(31); /* Length of character attributes
buffer */
dcl CharAttrs     char(n);        /* Character attributes */
dcl CompCode      fixed bin(31); /* Completion code */
dcl Reason        fixed bin(31); /* Reason code qualifying
CompCode */
```

## High Level Assembler çağırısı

```
CALL MQINQ, (HCONN, HOBJ, SELECTORCOUNT, SELECTORS, INTATTRCOUNT, X  
INTATTRS, CHARATTRLENGTH, CHARATTRS, COMPCODE, REASON)
```

Değiştirgeleri aşağıdaki gibi bildirin:

HCONN	DS	F	Connection handle
HOBJ	DS	F	Object handle
SELECTORCOUNT	DS	F	Count of selectors
SELECTORS	DS	(n)F	Array of attribute selectors
INTATTRCOUNT	DS	F	Count of integer attributes
INTATTRS	DS	(n)F	Array of integer attributes
CHARATTRLENGTH	DS	F	Length of character attributes buffer
CHARATTRS	DS	CL(n)	Character attributes
COMPCODE	DS	F	Completion code
REASON	DS	F	Reason code qualifying COMPCODE

## Visual Basic çağırısı

```
MQINQ Hconn, Hobj, SelectorCount, Selectors, IntAttrCount, IntAttrs,  
CharAttrLength, CharAttrs, CompCode, Reason
```

Değiştirgeleri aşağıdaki gibi bildirin:

Dim Hconn	As Long	'Connection handle'
Dim Hobj	As Long	'Object handle'
Dim SelectorCount	As Long	'Count of selectors'
Dim Selectors	As Long	'Array of attribute selectors'
Dim IntAttrCount	As Long	'Count of integer attributes'
Dim IntAttrs	As Long	'Array of integer attributes'
Dim CharAttrLength	As Long	'Length of character attributes buffer'
Dim CharAttrs	As String	'Character attributes'
Dim CompCode	As Long	'Completion code'
Dim Reason	As Long	'Reason code qualifying CompCode'

## MQINQMP-Questery ileti özelliği

MQINQMP çağırısı, bir iletinin özellik değerini döndürür.

### Sözdizimi

MQINQMP (*Hconn, Hmsg, InqPropOpts, Ad, PropDesc, Type, ValueLength, Value, DataLength, CompCode, Reason*)

### Parametreler

#### Hconn

Tip: MQHCONN-input

Bu tanıtcı, kuyruk yöneticisiyle olan bağlantıyı gösterir. *Hconn* değeri, **Hmsg** değiştirgesinde belirtilen ileti tanıtcısını yaratmak için kullanılan bağlantı tanıtcısı ile eşleşmelidir.

İleti tanıtcısı MQHC\_UNASSOCIATED\_HCONN kullanılarak yaratıldıysa, ileti tutamacındaki bir özelliği geciktiren iş parçacığında geçerli bir bağlantı oluşturulmalıdır; tersi durumda, çağrı MQRC\_CONNECTION\_BROKEN ile başarısız olur.

#### İtt

Tip: MQHMSG-giriş

Bu, sorgulanacak ileti saptanıdır. Değer önceki bir **MQCRTMH** çağırısı tarafından döndürüldü.

## InqPropOpts

Tip: MQIMPO-giriş/çıkış

Ayrıntılar için [MQIMPO](#) veri tipine bakın.

## Ad

Tip: MQCHARV-giriş/çıkış

Sorgulanacak özelliğin adı.

Bu adı taşıyan bir özellik bulunamazsa, arama işlemi MQRC\_PROPERTY\_NOT\_AVAM neden ile başarısız olur.

Özellik adının sonunda genel arama karakteri yüzde işaretini (%) kullanabilirsiniz. Genel arama karakteri, nokta (.) karakteri de içinde olmak üzere, sıfır ya da daha fazla karakterle eşleşir. Bu, bir uygulamanın birçok özelliğin değerini sorgulamasına olanak sağlar. İlk eşleşen özelliği almak için MQIMPO\_INQ\_FIRST seçeneğiyle MQINQMP ' yi çağırın ve sonraki eşleştirme özelliği almak için MQIMPO\_INQ\_Next seçeneğiyle yeniden çağırın. Daha fazla eşleşen özellik yoksa, çağrı MQRC\_PROPERTY\_NOT\_AVAM ile başarısız olur. InqPropOpts yapısının *ReturnedName* alanı, özelliğin döndürülen adı için bir adres ya da görel konum ile kullanıma hazırlandıysa, bu işlem, eşleştirilen özelliğin adıyla MQINQMP ' den geri dönerek tamamlanır. InqPropOpts yapısındaki *ReturnedName* ' nin *VSBufSize* alanı, döndürülen özellik adının uzunluğundan küçükse, tamamlanma kodu MQCC\_FAILED, MQRC\_PROVANY\_NAME\_TOO\_Büyük nedeniyle başarısız oldu.

Bilinen eşanlamlıları olan özellikler aşağıdaki gibi döndürülür:

1. "mqps." öneğine sahip özellikler IBM MQ özellik adı olarak döndürülür. Örneğin, "MQTopicString", "mqps.Top" yerine döndürülen addır.
2. "jms." öneğine sahip özellikler ya da "mcd." JMS üstbilgi alanı adı olarak döndürülür; örneğin, "JMSExpiration", "jms.Exp" yerine döndürülen addır.
3. "usr." öneğine sahip özellikler bu önek olmadan döndürülür; örneğin, "usr.Color" yerine "Renk" döndürülmesi döndürülür.

Eşanlamlıları olan özellikler yalnızca bir kez döndürülür.

C programlama dilinde, aşağıdaki makro değişkenleri tüm özellikleri ve sonra "usr." başlayan tüm özellikleri sorgulamak için tanımlanır:

## MQPROP\_INQUIRE\_ALL

İletinin tüm özelliklerini sorgulayın.

MQPROP\_INQUENE\_ALL aşağıdaki şekilde kullanılabilir:

```
MQCHARV Name = {MQPROP_INQUIRE_ALL};
```

## MQPROP\_INQUIRE\_ALL\_USR

"usr." başlayan iletinin tüm özellikleri ile ilgili olarak bilgi edinin. Döndürülen ad "usr" olmadan döndürülür. önek.

MQIMP\_INQ\_Next belirtildiyse, ancak önceki çağrıdan sonra ad değiştiyse ya da ilk çağrı bu, MQIMPO\_INQ\_FIRST ise örtük olarak belirtilir.

Özellik adlarının kullanımına ilişkin ek bilgi için [Özellik adları](#) ve [Özellik adı kısıtlamaları](#) başlıklı konuya bakın.

## PropDesc

Tip: MQPD-output

Bu yapı, özellik desteklenmiyorsa neler olacağı, özelliğin ait olduğu ileti bağlamı ve özelliğin hangi iletiye kopyalanması gerektiği de dahil olmak üzere, bir özelliğin özniteliklerini tanımlamak için kullanılır. Bu yapıya ilişkin ayrıntılar için [MQPD](#) başlıklı konuya bakın.

## Tip

Tip: MQHOME-giriş/çıkış

Bu deęiřtirgeyi MQINQMP çağrısından dönuřte, *Deęerveri* tipine ayarlanır. Veri tipi ařaęıdakilerden biri olabilir:

**MQTYPE\_BOOLEAN**

Bir boole.

**MQTYPE\_BYTE\_STRING**

Bayt dizilimi.

**MQTYPE\_INT8**

8 bitlik iřaretli tamsayı.

**MQTYPE\_INT16**

16 bitlik bir iřaretli tamsayı.

**MQTYPE\_INT32**

32 bitlik iřaretli tamsayı.

**MQTYPE\_INT64**

Bir 64 bit imzalı tamsayı.

**MQTYPE\_FLOAT32**

32 bitlik kayan noktalı sayı.

**MQTYPE\_FLOAT64**

64 bitlik kayan noktalı sayı.

**MQTYPE\_STRING**

Bir karakter dizgisi.

**MQTYPE\_NULL**

Özellik var, ancak boş deęere sahip.

Özellik deęerinin veri tipi tanınmadıysa, MQTYPE\_STRING deęeri döndürülür ve deęerin dizgi gösterimi *Deęer* alanına yerleřtirilir. Veri tipinin dizgi gösterimi, *InqPropOpts* parametresinin *TypeString* alanında bulunabilir. MQRC\_PROP\_TYPE\_NOT\_SUPPORTED neden ile bir uyarı tamamlanma kodu döndürülür.

Buna ek olarak, MQIMPO\_CONVERT\_TYPE seçeneęi belirtilirse, özellik deęerinin dönüřtürülmesi istenilir. Özellięin döndürülmesini istedięiniz veri tipini belirtmek için, giriř olarak *Type* (Tip) kullanın. Veri tipi dönüřtürmesine iliřkin ayrıntılar için MQIMPO yapısının MQIMPO\_CONVERT\_TYPE seçeneęinin açıklamasına bakın.

Tip dönüřtürme isteęinde bulunmuyorsanız, giriř sırasında ařaęıdaki deęeri kullanabilirsiniz:

**MQTYPE\_AS\_SET**

Özellięin deęeri, veri tipini dönüřtürmeden döndürülür.

**ValueLength**

Tip: MQHOME-giriři

Deęer alanının bayt cinsinden uzunluęu. Deęerin döndürülmesini gerektirmedięiniz özellikler için sıfır deęerini belirtin. Bu özellikler, bir uygulamanın boş deęerli ya da boş bir dizgi olması için tasarlanmış özellikler olabilir. Ayrıca, MQIMPO\_QUERY\_LENGTH seçeneęi belirtilmiřse, sıfır deęerini de belirtin; bu durumda deęer döndürülmez.

**Deęer**

Tip: MQBYTEx *ValueLength* -çıkıř

Bu, sorgudaki özellik deęerini içermek için kullanılan alandır. Arabellek, döndürülemekte olan deęer için uygun bir sınırla hizalanmalıdır. Bunun yapmaması, daha sonra deęere eriřildięinde bir hatayla sonuçlanabilir.

*ValueLength* özellięi, özellik deęerinin uzunluęundan küçükse, olası deęer *Deęer* ' e tařınır ve çağrı, MQCC\_FAILED tamamlanma kodu ile başarısız olur ve MQRC\_PROPERTY\_VALUE\_TOO\_BIG neden olur.

*Deęer* alanındaki verilerin karakter takımı, InqPropOpts parametresindeki ReturnedCCSID alanı tarafından verilir. The encoding of the data in *Deęer* is given by the ReturnedEncoding field in the InqPropOpts parameter.

C programlama dilinde, parametre bir işaretçi olarak bildirilir; parametrenin herhangi bir tipinin adresi, parametre olarak belirtilebilir.

If the *ValueLength* parameter is zero, *Değer* is not referred to and its value passed by programs written in C or System/390 assembler can be null.

### DataLength

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Bu, *Değer* alanında döndürülen gerçek özellik değerinin bayt cinsinden uzunluğudur.

If *DataLength* is less than the property value length, *DataLength* is still filled in on return from the MQINQMP call. Bu, uygulamanın özellik değerini sığdırmak için gereken arabelleğin büyüklüğünü belirlemesine olanak tanır ve daha sonra, aramayı uygun boyutta bir arabellekle yeniden yayınlayabilir.

Aşağıdaki değerler de döndürülebilmektedir.

Type değiştirgesi MQTYPE\_STRING ya da MQTYPE\_BYTE\_STRING olarak ayarlandıysa:

### MQVL\_EMPTY\_STRING

Özellik var, ancak karakter ya da byte içermiyor.

### CompCode

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Tamamlanma kodu; bu kod aşağıdakilerden biridir:

### MQCC\_OK

İşlem başarıyla tamamlandı.

### MQCC\_UYARI

Uyarı (kısmi tamamlama).

### MQCC\_FAILED

Arama başarısız oldu.

### Neden

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

*CompCode* MQCC\_OK ise:

### MQRC\_NONE

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_UYARI ise:

### MQRC\_PROP\_NAME\_NOT\_CONVERTED

(2492, X'09BC') Döndürülen özellik adı dönüştürülmedi.

### MQRC\_PROP\_VALUE\_NOT\_CONVERTED

(2466, X'09A2') Özellik değeri dönüştürülmedi.

### MQRC\_PROP\_TYPE\_NOT\_SUPPORTED

(2467, X'09A3') Özellik veri tipi desteklenmiyor.

### MQRC\_RFH\_FORMAT\_ERROR

(2421, X'0975 ') Özellikler içeren bir MQRFH2 klasörü ayrıştırılmadı.

*CompCode* MQCC\_FAILED ise:

### MQRC\_ADAPTER\_NOT\_AVAM

(2204, X'089C') Bağdaştırıcı kullanılmıyor.

### MQRC\_ADAPTER\_SERV\_LOAD\_ERROR

(2130, X'0852 ') Bağdaştırıcı hizmeti modülü yüklenemiyor.

### MQRC\_ASID\_MISSMACH

(2157, X'086D') Birincil ve ana ASID değerleri farklı.

### MQRC\_BUFFER\_ERROR

(2004, X'07D4') Değer parametresi geçerli değil.

**MQRC\_BUFFER\_LENGTH\_ERROR**

(2005, X'07D5') Değer uzunluğu parametresi geçerli değil.

**MQRC\_CALL\_IN\_PROGRESS**

(2219, X'08AB') Önceki arama tamamlanmadan önce girilen MQI çağırısı.

**MQRC\_CONNECTION\_BROKEN**

(2009, X'07D9') Kuyruk yöneticisine yönelik bağlantı kaybedildi.

**MQRC\_DATA\_LENGTH\_ERROR**

(2010, X'07DA') Veri uzunluğu parametresi geçerli değil.

**MQRC\_IMPO\_ERROR**

(2464, X'09A0') Sorgula ileti özelliği seçenekleri yapısı geçerli değil.

**MQRC\_HMSG\_ERROR**

(2460, X'099C') İleti tanıtıcısı geçerli değil.

**MQRC\_MSG\_HANDLE\_IN\_USE**

(2499, X'09C3') İleti tanıtıcısı zaten kullanılıyor.

**MQRC\_OPTIONS\_ERROR**

(2046, X'07F8') Seçenekler geçerli değil ya da tutarlı değil.

**MQRC\_PD\_ERROR**

(2482, X'09B2') Özellik tanımlayıcı yapısı geçerli değil.

**MQRC\_PROP\_CONV\_NOT\_SUPPORTED**

(2470, X'09A6') İstenilen veri tipine gerçek olarak dönüştürme desteklenmiyor.

**MQRC\_ÖZELLİĞİ \_**

(2442, X'098A') Özellik adı geçersiz.

**MQRC\_ÖZELLİĞİN\_NAME\_TOO\_**

(2465, X'09A1') Özellik adı, döndürülen ad arabelleği için çok büyük.

**MQRC\_PROPERTY\_NOT\_VAR**

(2471, X'09A7) Özellik yok.

**MQRC\_PROPERTY\_VALUE\_TOO\_BIG**

(2469, X'09A5') Özellik değeri, Değer alanı için çok büyük.

**MQRC\_PROP\_NUMBER\_FORMAT\_ERROR**

(2472, X'09A8') Değer verilerinde sayı biçimi hatası saptandı.

**MQRC\_ÖZELLİĞİ TYPE\_**

(2473, X'09A9') İstenen özellik tipi geçersiz.

**MQRC\_SOURCE\_CCSID\_ERROR**

(2111, X'083F') Özellik adı kodlanmış karakter takımı tanıtıcısı geçerli değil.

**MQRC\_STORAGE\_NOT\_AVAM**

(2071, X'0871 ') Kullanılabilir bellek yetersiz.

**MQRC\_UNEXPECTED\_ERROR**

(2195, X'0893 ') Beklenmeyen bir hata oluştu.

Bu kodlarla ilgili ayrıntılı bilgi için [İletiler ve neden kodları](#) başlıklı konuya bakın.

## C çağırma

```
MQINQMP (Hconn, Hmsg, &InqPropOpts, &Name, &PropDesc, &Type,  
ValueLength, Value, &DataLength, &CompCode, &Reason);
```

Değiştirelileri aşağıdaki gibi bildirin:

```
MQHCONN Hconn; /* Connection handle */  
MQHMSG Hmsg; /* Message handle */  
MQIMPO InqPropOpts; /* Options that control the action of MQINQMP */  
MQCHARV Name; /* Property name */
```

```

MQPD   PropDesc;    /* Property descriptor */
MQLONG Type;        /* Property data type */
MQLONG ValueLength; /* Length in bytes of the Value area */
MQBYTE Value[n];    /* Area to contain the property value */
MQLONG DataLength; /* Length of the property value */
MQLONG CompCode;    /* Completion code */
MQLONG Reason;      /* Reason code qualifying CompCode */

```

## Cobol çağırma

```

CALL 'MQINQMP' USING HCONN, HMSG, INQMSGOPTS, NAME, PROPDESC, TYPE,
VALUELENGTH, VALUE, DATALENGTH, COMPCODE, REASON.

```

Değiştirgeleri aşağıdaki gibi bildirin:

```

** Connection handle
01 HCONN          PIC S9(9) BINARY.
** Message handle
01 HMSG          PIC S9(18) BINARY.
** Options that control the action of MQINQMP
01 INQMSGOPTS.
COPY CMQIMPOV.
** Property name
01 NAME.
COPY CMQCHRUV.
** Property descriptor
01 PROPDESC.
COPY CMQPDV.
** Property data type
01 TYPE          PIC S9(9) BINARY.
** Length in bytes of the VALUE area
01 VALUELENGTH PIC S9(9) BINARY.
** Area to contain the property value
01 VALUE         PIC X(n).
** Length of the property value
01 DATALENGTH PIC S9(9) BINARY.
** Completion code
01 COMPCODE     PIC S9(9) BINARY.
** Reason code qualifying COMPCODE
01 REASON      PIC S9(9) BINARY.

```

## PL/I çağırımı

```

call MQINQMP (Hconn, Hmsg, InqPropOpts, Name, PropDesc, Type,
ValueLength, Value, DataLength, CompCode, Reason);

```

Değiştirgeleri aşağıdaki gibi bildirin:

```

dcl Hconn      fixed bin(31); /* Connection handle */
dcl Hmsg       fixed bin(63); /* Message handle */
dcl InqPropOpts like MQIMPO; /* Options that control the action of MQINQMP */
dcl Name       like MQCHARV; /* Property name */
dcl PropDesc   like MQPD;    /* Property descriptor */
dcl Type       fixed bin (31); /* Property data type */
dcl ValueLength fixed bin (31); /* Length in bytes of the Value area */
dcl Value      char (n);      /* Area to contain the property value */
dcl DataLength fixed bin (31); /* Length of the property value */
dcl CompCode   fixed bin (31); /* Completion code */
dcl Reason     fixed bin (31); /* Reason code qualifying CompCode */

```

## High Level Assembler çağırısı

```

CALL MQINQMP, (HCONN, HMSG, INQMSGOPTS, NAME, PROPDESC, TYPE,
VALUELENGTH, VALUE, DATALENGTH, COMPCODE, REASON)

```

Değiştirgeleri aşağıdaki gibi bildirin:

HCONN	DS	F	Connection handle
HMSG	DS	D	Message handle
INQMSGOPTS	CMQIMPOA	,	Options that control the action of MQINQMP
NAME	CMQCHRVA	,	Property name
PROPDESC	CMQPDA	,	Property descriptor
TYPE	DS	F	Property data type
VALUELENGTH	DS	F	Length in bytes of the VALUE area
VALUE	DS	CL(n)	Area to contain the property value
DATALength	DS	F	Length of the property value
COMPCODE	DS	F	Completion code
REASON	DS	F	Reason code qualifying COMPCODE

## MQMHBUF-İleti tanıtıcısını arabelleğe dönüştür

MQMHBUF çağrısı, bir ileti tanıtıcısını arabellek içine dönüştürür ve MQBUFMH çağrısının tersi olur.

### Sözdizimi

MQMHBUF (*Hconn*, *Hmsg*, *MsgHBufOpts*, *Ad*, *MsgDesc*, *BufferLength*, *Buffer*, *DataLength*, *CompCode*, *Reason*)

### Parametreler

#### Hconn

Tip: MQHCONN-input

Bu tanıtıcı, kuyruk yöneticiyle olan bağlantıyı gösterir. *Hconn* değeri, **Hmsg** değiştirgesinde belirtilen ileti tanıtıcısını yaratmak için kullanılan bağlantı tanıtıcısı ile eşleşmelidir.

İleti tanıtıcısı MQHC\_UNASSOCIATED\_HCONN kullanılarak yaratıldıysa, ileti tanıtıcısını silmeye ilişkin iş parçasısında geçerli bir bağlantı kurulmalıdır. Geçerli bir bağlantı kurulmamışsa, çağrı MQRC\_CONNECTION\_BROKEN ile başarısız olur.

#### İt

Tip: MQHMSG-giriş

Bu, arabelleğin doldurulması gereken ileti tanıtıcısıdır. Değer önceki bir MQCRTMH çağrısı tarafından döndürüldü.

#### MsgHBufOpts

Tip: MQMHBO-input

MQMHBO yapısı, uygulamaların, ileti tanıtıcılarından arabelleklerin nasıl üretildiğini denetleyen seçenekleri belirlemesine olanak sağlar.

Ayrıntılar için bkz. [“MQMHBO-Arabellek seçeneklerine ileti tanıtıcısı” sayfa 471.](#)

#### Ad

Tip: MQCHARV-input

Arabelleğe yerleştirilecek özelliğin ya da özelliklerin adı.

Adla eşleşen herhangi bir özellik bulunamazsa, çağrı MQRC\_PROPERTY\_NOT\_AVAM ile başarısız olur.

Arabelleğe birden çok özellik koymak için genel arama karakteri kullanabilirsiniz. Bunu yapmak için, özellik adının sonundaki genel arama karakteri '%' karakterini kullanın. Bu genel arama karakteri, sıfır ya da daha fazla karakterle eşleşir (!' karakterini kullanın.

C programlama dilinde, aşağıdaki makro değişkenleri, 'usr' olarak başlayan tüm özellikleri ve tüm özellikleri sorgulamak için tanımlanır:

#### MQPROP\_INQUIRE\_ALL

İletinin tüm özelliklerini arabelleğe koy

#### MQPROP\_INQUIRE\_ALL\_USR

İletinin tüm özelliklerini 'usr.' karakterleriyle başlayan arabelleğe alma.



Özellik adlarının kullanımına ilişkin ek bilgi için Özellik adları ve Özellik adı kısıtlamaları başlıklı konuya bakın.

### **MsgDesc**

Tip: MQMD-giriş/çıkış

*MsgDesc* yapısı, arabellek alanının içeriğini açıklar.

Çıkışta, *Encoding*, *CodedCharSetId* ve *Format* alanları, arama tarafından yazıldığı şekilde, arabellek alanındaki verilerin kodlamasını, karakter kümesi tanıtıcısını ve biçimini doğru şekilde tanımlamaya ayarlanır.

Bu yapıdaki veriler, uygulamanın karakter kümesinde ve kodlamasında yer alıyor.

### **BufferLength**

Tip: MQHOME-girişi

*BufferLength*, arabellek alanının bayt cinsinden uzunluğudur.

### **Arabellek**

Tip: MQBYTEExBufferUzunluğu-çıkış

*Buffer*, ileti özelliklerinin içerileceği alanı tanımlar. Arabelleği 4 baytlık bir sınır üzerinde hizalamanız gerekir.

*BufferLength*, özellikleri *Buffer* içinde saklamak için gereken uzunluktan küçükse, MQMHBUF, MQRC\_PROPERTY\_VALUE\_TOO\_BIG ile başarısız olur.

Arama başarısız olsa da arabelleğin içeriği değişebilir.

### **DataLength**

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

*DataLength*, arabelleğindeki döndürülen özelliklerin uzunluğudur (byte olarak). Değer sıfır ise, hiçbir özellik *Name* ' de belirtilen değerle eşleşmedi ve çağrı, MQRC\_PROPERTY\_NOT\_AVAILABLE neden koduyla başarısız olur.

*BufferLength*, belleğindeki özellikleri saklamak için gereken uzunluktan küçükse, MQMHBUF çağrısı MQRC\_PROPERTY\_VALUE\_TOO\_BIG ile başarısız olur, ancak yine de *DataLength* içine bir değer girilir. Bu, uygulamanın özellikleri barındırmak için gereken arabelleğin büyüklüğünü belirlemesine olanak tanır ve daha sonra, gerekli *BufferLength* ile çağrıyı yeniden yayınlayın.

### **CompCode**

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Tamamlanma kodu; bu kod aşağıdakilerden biridir:

#### **MQCC\_OK**

İşlem başarıyla tamamlandı.

#### **MQCC\_FAILED**

Arama başarısız oldu.

### **Neden**

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Neden kodu niteleyici *CompCode*.

*CompCode* MQCC\_OK ise:

#### **MQRC\_NONE**

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_FAILED ise:

#### **MQRC\_ADAPTER\_NOT\_AVAM**

(2204, X'089C') Bağdaştırıcı kullanılamıyor.

#### **MQRC\_ADAPTER\_SERV\_LOAD\_ERROR**

(2130, X'852 ') Bağdaştırıcı hizmeti modülü yüklenemiyor.

**MQRC\_ASID\_MISSMACH**

(2157, X'86D') Birincil ve ana sunucu ASID değerleri farklı.

**MQRC\_MHBO\_ERROR**

(2501, X'095C') Arabellek seçenekleri yapısına ilişkin ileti tanıtıcısı geçerli değil.

**MQRC\_BUFFER\_ERROR**

(2004, X'07D4') Arabellek parametresi geçerli değil.

**MQRC\_BUFFER\_LENGTH\_ERROR**

(2005, X'07D5') Arabellek uzunluğu parametresi geçerli değil.

**MQRC\_CALL\_IN\_PROGRESS**

(2219, X'08AB') Önceki arama tamamlanmadan önce girilen MQI çağırısı.

**MQRC\_CONNECTION\_BROKEN**

(2009, X'07D9') Kuyruk yöneticisine yönelik bağlantı kaybedildi.

**MQRC\_DATA\_LENGTH\_ERROR**

(2010, X'07DA') Veri uzunluğu parametresi geçerli değil.

**MQRC\_HMSG\_ERROR**

(2460, X'099C') İleti tanıtıcısı geçerli değil.

**MQRC\_MD\_ERROR**

(2026, X'07EA') İleti tanımlayıcısı geçerli değil.

**MQRC\_MSG\_HANDLE\_IN\_USE**

(2499, X'09C3') İleti tanıtıcısı zaten kullanılıyor.

**MQRC\_OPTIONS\_ERROR**

(2046, X'07FE') Seçenekler geçerli değil ya da tutarlı değil.

**MQRC\_ÖZELLİĞİ \_**

(2442, X'098A') Özellik adı geçerli değil.

**MQRC\_PROPERTY\_NOT\_VAR**

(2471, X'09A7') Özellik yok.

**MQRC\_PROPERTY\_VALUE\_TOO\_BIG**

(2469, X'09A5') BufferLength değeri, belirtilen özellikleri içermek için çok küçük.

**MQRC\_UNEXPECTED\_ERROR**

(2195, X'893 ') Beklenmeyen bir hata oluştu.

Bu kodlarla ilgili ayrıntılı bilgi için [İletiler ve neden kodları](#) başlıklı konuya bakın.

## C çağırma

```
MQMHBUF (Hconn, Hmsg, &MsgHBufOpts, &Name, &MsgDesc, BufferLength, Buffer,  
&DataLength, &CompCode, &Reason);
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
MQHCONN Hconn;          /* Connection handle */  
MQHMSG  Hmsg;           /* Message handle */  
MQMHBO  MsgHBufOpts;   /* Options that control the action of MQMHBUF */  
MQCHARV Name;          /* Property name */  
MQMD    MsgDesc;       /* Message descriptor */  
MQLONG  BufferLength;   /* Length in bytes of the Buffer area */  
MQBYTE  Buffer[n];      /* Area to contain the properties */  
MQLONG  DataLength;    /* Length of the properties */  
MQLONG  CompCode;      /* Completion code */  
MQLONG  Reason;        /* Reason code qualifying CompCode */
```

## Kullanım notları

MQMHBUF, bir ileti tanıtıcısını arabellek içine dönüştürür.

Bu özelliği, belirli özelliklere erişmek için bir MQGET API çıkışıyla, ileti özelliği API ' lerini kullanarak ve ileti tanıtıcıları yerine MQRFH2 üstbilgilerini kullanmak için tasarlanmış bir uygulamaya geri iletebilirsiniz.

Bu çağrı, ileti özelliklerini bir arabellekten ileti tanıtıcısı olarak ayrıştırmak için kullanabileceğiniz MQBUFMH çağrısının tersidir.

## Cobol çağırma

```
CALL 'MQMHBUF' USING HCONN, HMSG, MSGHBUFOPTS, NAME, MSGDESC,  
BUFFERLENGTH, BUFFER, DATALENGTH, COMPCODE, REASON.
```

Değişirgeleri aşağıdaki gibi bildirin:

```
** Connection handle  
01 HCONN          PIC S9(9) BINARY.  
** Message handle  
01 HMSG          PIC S9(18) BINARY.  
** Options that control the action of MQMHBUF  
01 MSGHBUFOPTS.  
COPY CMQMHBV.  
** Property name  
01 NAME  
COPY CMQCHRVV.  
** Message descriptor  
01 MSGDESC  
COPY CMQMDV.  
** Length in bytes of the Buffer area */  
01 BUFFERLENGTH PIC S9(9) BINARY.  
** Area to contain the properties  
01 BUFFER        PIC X(n).  
** Length of the properties  
01 DATALENGTH  PIC S9(9) BINARY.  
** Completion code  
01 COMPCODE     PIC S9(9) BINARY.  
** Reason code qualifying COMPCODE  
01 REASON       PIC S9(9) BINARY.
```

## PL/I çağırımı

```
call MQMHBUF (Hconn, Hmsg, MsgHBufOpts, Name, MsgDesc, BufferLength, Buffer,  
DataLength, CompCode, Reason);
```

Değişirgeleri aşağıdaki gibi bildirin:

```
dcl Hconn          fixed bin(31); /* Connection handle */  
dcl Hmsg           fixed bin(63); /* Message handle */  
dcl MsgHBufOpts   like MQMHBO; /* Options that control the action of MQMHBUF */  
dcl Name          like MQCHARV; /* Property name */  
dcl MsgDesc       like MQMD; /* Message descriptor */  
dcl BufferLength   fixed bin(31); /* Length in bytes of the Buffer area */  
dcl Buffer         char(n); /* Area to contain the properties */  
dcl DataLength    fixed bin(31); /* Length of the properties */  
dcl CompCode      fixed bin(31); /* Completion code */  
dcl Reason        fixed bin(31); /* Reason code qualifying CompCode */
```

## High Level Assembler çağırısı

```
CALL MQMHBUF, (HCONN, HMSG, MSGHBUFOPTS, NAME, MSGDESC, BUFFERLENGTH,  
BUFFER, DATALENGTH, COMPCODE, REASON)
```

Değişirgeleri aşağıdaki gibi bildirin:

HCONN	DS	F	Connection handle
HMSG	DS	D	Message handle

MSGHBUFOPTS	CMQMHB0A	,	Options that control the action of MQMHBUF
NAME	CMQCHRVA	,	Property name
MSGDESC	CMQMDA	,	Message descriptor
BUFFERLENGTH	DS	F	Length in bytes of the BUFFER area
BUFFER	DS	CL(n)	Area to contain the properties
DATALLENGTH	DS	F	Length of the properties
COMPCODE	DS	F	Completion code
REASON	DS	F	Reason code qualifying COMPCODE

## MQOpen-Open nesnesi

MQOPEN çağırısı bir nesneye erişim oluşturur.

Aşağıdaki nesne tipleri geçerlidir:

- Kuyruk (dağıtım listeleri de içinde olmak üzere)
- Ad Listesi
- Süreç tanımlaması
- Kuyruk yöneticisi
- Konu

## Sözdizimi


MQOPEN (*Hconn*, *ObjDesc*, *Options*, *Hobj*, *CompCode*, *Reason*)

## Parametreler

### Hconn

Tip: MQHCONN-input

Bu tanıtıcı, kuyruk yöneticisiyle olan bağlantıyı gösterir. The value of Hconn was returned by a previous MQCONN or MQCONNX call.

 On z/OS for CICS applications the MQCONN call can be omitted, and the following value specified for *Hconn* :

### MQHC\_DEF\_HCONN

Varsayılan bağlantı tanıtıcısı.

### ObjDesc

Tip: MQOD-giriş/çıkış

Bu yapı, açılacak nesneyi tanımlayan bir yapıdır; ayrıntılar için [“MQOD-Nesne tanımlayıcı” sayfa 473](#) ' e bakın.

**ObjDesc** parametresindeki ObjectName alanı, bir model kuyruğunun adı ise, dinamik bir yerel kuyruk Model kuyruğunun öznitelikleriyle yaratılır; bu, **Options** parametresinde belirlediğiniz seçeneklerle oluşur. MQOPER çağırısı tarafından döndürülen Hobj ile sonraki işlemler, model kuyruğunda değil, yeni dinamik kuyruğun üzerinde gerçekleştirilir. Bu, MQINQ ve MQSET çağırısı için bile geçerlidir.

**ObjDesc** parametresindeki model kuyruğunun adı, yaratılan dinamik kuyruğun adıyla değiştirilir. Dinamik kuyruğun tipi, model kuyruğunun **DefinitionType** özniteliğinin değerine göre belirlenir (bkz. [“Kuyruklara ilişkin öznitelikler” sayfa 821](#) ). Dinamik kuyruklar için geçerli olan kapatma seçeneklerine ilişkin bilgi için, MQCLOSE çağırısının açıklamasına bakın.

## Seçenekler

Tip: MQHOME-girişi

Aşağıdaki seçeneklerden en az birini belirlemelisiniz:

- MQOO\_BROWSE
- MQOO\_INPUT\_ \* (bu öğelerden yalnızca biri)
- MQOO\_SORGULAMA
- MQOO\_OUTPUT

- M<sub>QOO</sub>\_SET
- M<sub>QOO</sub>\_BIND\_\* (bu öğelerden yalnızca biri)

Bu seçeneklerin ayrıntıları için aşağıdaki tabloya bakın; diğer seçenekler zorunlu olarak belirtilebilir. Geçerli olmayan birden çok seçenek belirtmek için, değerleri bir araya ekleyin (aynı değişmezi bir kereden fazla eklemeyin) ya da bit ya da işlemi kullanarak değerleri birleştirin (programlama dili, bit işlemleri destekliyse). Bileşimleri not edilir; diğer tüm birleşimler geçerlidir. Yalnızca, ObjDesc ile belirtilen nesne tipi için geçerli olan seçeneklere izin verilir.

Seçenek	Diğer Ad <sup>1</sup>	Yerel ve Model	Uzak	Yerel Olmayan Küme	Dağıtım listesi	Konu
<u>M<sub>QOO</sub>_INPUT_AS_Q_DEF</u>	Evet	Evet	Hayır	Hayır	Hayır	Hayır
<u>M<sub>QOO</sub>_INPUT_SHARED</u>	Evet	Evet	Hayır	Hayır	Hayır	Hayır
<u>M<sub>QOO</sub>_INPUT_EXCLUSIVE</u>	Evet	Evet	Hayır	Hayır	Hayır	Hayır
<u>M<sub>QOO</sub>_OUTPUT</u>	Evet	Evet	Evet	Evet	Evet	Evet
<u>M<sub>QOO</sub>_BROWSE</u>	Evet	Evet	Hayır	Hayır	Hayır	Hayır
<u>M<sub>QOO</sub>_CO_OP</u>	Evet	Evet	Hayır	Hayır	Hayır	Hayır
<u>M<sub>QOO</sub>_SORGULAMA</u>	Evet	Evet	<u>2</u>	Evet	Hayır	Hayır
<u>M<sub>QOO</sub>_SET</u>	Evet	Evet	<u>2</u>	Hayır	Hayır	Hayır
<u>M<sub>QOO</sub>_BIND_ON_Aç <sup>3</sup></u>	Evet	Evet	Evet	Evet	Evet	Hayır
<u>M<sub>QOO</sub>_BIND_NOT_FIXED <sup>3</sup></u>	Evet	Evet	Evet	Evet	Evet	Hayır
<u>M<sub>QOO</sub>_BIND_ON_GROUP <sup>3</sup></u>	Evet	Evet	Evet	Evet	Evet	Hayır
<u>M<sub>QOO</sub>_BIND_AS_Q_DEF <sup>3</sup></u>	Evet	Evet	Evet	Evet	Evet	Hayır
<u>M<sub>QOO</sub>_SAVE_AL_CONTEXT</u>	Evet	Evet	Hayır	Hayır	Hayır	Hayır
<u>M<sub>QOO</sub>_PASS_IDENTITY_CONTEXT</u>	Evet	Evet	Evet	Evet	Evet	<u>4</u>
<u>M<sub>QOO</sub>_PASS_ALL_CONTEXT</u>	Evet	Evet	Evet	Evet	Evet	Evet
<u>M<sub>QOO</sub>_SET_IDENTITY_CONTEXT</u>	Evet	Evet	Evet	Evet	Evet	<u>4</u>
<u>M<sub>QOO</sub>_SET_AL_CONTEXT</u>	Evet	Evet	Evet	Evet	Evet	Evet
<u>M<sub>QOO</sub>_NO_READ_AHEAD</u>	Evet	Evet	Hayır	Hayır	Hayır	Hayır
<u>M<sub>QOO</sub>_READ_AHEAD</u>	Evet	Evet	Hayır	Hayır	Hayır	Hayır
<u>M<sub>QOO</sub>_READ_AHEAD_AS_Q_DEF</u>	Evet	Evet	Hayır	Hayır	Hayır	Hayır
<u>M<sub>QOO</sub>_ALTERNATE_USER_AUTHORITY</u>	Evet	Evet	Evet	Evet	Evet	Evet
<u>M<sub>QOO</sub>_FAIL_IF_QUIESCING</u>	Evet	Evet	Evet	Evet	Evet	Evet
<u>M<sub>QOO</sub>_RESOLVE_LOCAL_Q</u>	Evet	Evet	Evet	Evet	Hayır	Hayır
<u>M<sub>QOO</sub>_RESOLVE_LOCAL_TOPIC</u>	Hayır	Hayır	Hayır	Hayır	Hayır	Evet
<u>M<sub>QOO</sub>_NO_MULTICAST</u>	Hayır	Hayır	Hayır	Hayır	Hayır	Evet

#### Notlar:

1. Diğer adlara ilişkin seçeneklerin geçerliliği, diğer adın çözdüğü kuyruğa ilişkin seçeneğin geçerliliğine bağlıdır.
2. Bu seçenek yalnızca uzak kuyruğun yerel tanımlaması için geçerlidir.
3. Bu seçenek herhangi bir kuyruk tipi için belirlenebilir; ancak, kuyruk bir küme kuyruğu değilse yoksaılır. Ancak, **DefBind** kuyruk özneliği, diğer ad kuyruğu bir kümede yer almıyorsa bile, temel kuyruğu geçersiz kılar.

4. Bu öznitelikler bir konu ile birlikte kullanılabilir; ancak, herhangi bir aboneye gönderilen bağlam alanları değil, yalnızca alıkonan ileti için ayarlanan bağlamı etkiler.

**Erişim seçenekleri:** Aşağıdaki seçenekler, nesne üzerinde gerçekleştirilebilecek işlemlerin tipini denetler:

#### **MQOO\_INPUT\_AS\_Q\_DEF**

Kuyruk tanımlı varsayılan iletileri kullanarak iletileri almak için kuyruğu açın.

Kuyruk sonraki MQGET çağrısıyla kullanılmak üzere açılmıştır. Erişim tipi, **DefInputOpenOption** kuyruk özneliğinin değerine bağlı olarak paylaşılan ya da dışlayıcı bir tiptir; ayrıntılar için "[Kuyruklara ilişkin öznitelikler](#)" sayfa 821 ' e bakın.

Bu seçenek yalnızca yerel, diğer ad ve model kuyrukları için geçerlidir; uzak kuyruklar, dağıtım listeleri ve kuyruklar olmayan nesnelere için geçerli değildir.

#### **MQOO\_INPUT\_SHARED**

Paylaşılan erişimi olan iletileri almak için kuyruğu açın.

Kuyruk sonraki MQGET çağrısıyla kullanılmak üzere açılmıştır. Kuyruk şu anda bu ya da MQOO\_INPUT\_SHARED ile başka bir uygulama tarafından açıksa başarılı olabilir, ancak kuyruk şu anda MQOO\_INPUT\_EXCLUSIVE ile açıldıysa, MQRC\_OBJECT\_IN\_USE neden koduyla başarısız olursa, çağrı başarılı olabilir.

Bu seçenek yalnızca yerel, diğer ad ve model kuyrukları için geçerlidir; uzak kuyruklar, dağıtım listeleri ve kuyruklar olmayan nesnelere için geçerli değildir.

#### **MQOO\_INPUT\_EXCLUSIVE**

Dışlayıcı erişimi olan iletileri almak için kuyruğu açın.

Kuyruk sonraki MQGET çağrısıyla kullanılmak üzere açılmıştır. Kuyruk şu anda bu ya da herhangi bir tip (MQOO\_INPUT\_SHARED ya da MQOO\_INPUT\_EXCLUSIVE) girişi için başka bir uygulama tarafından açıldıysa, çağrı neden kodu MQRC\_OBJECT\_IN\_USE ile başarısız olur.

Bu seçenek yalnızca yerel, diğer ad ve model kuyrukları için geçerlidir; uzak kuyruklar, dağıtım listeleri ve kuyruklar olmayan nesnelere için geçerli değildir.

#### **MQOO\_OUTPUT**

İletileri koymak için kuyruğu açın ya da iletileri yayınlamak için bir konu ya da konu dizesi.

Kuyruk ya da konu sonraki MQPUT çağrısıyla kullanılmak üzere açılır.

**InhibitPut** kuyruk özneliği MQQA\_PUT\_INHIBITED olarak ayarlansa da, bu seçenek ile bir MQOPEN çağrısı başarılı olabilir (ancak, öznitelik bu değere ayarlandığında sonraki MQPUT çağrıları başarısız olur).

Bu seçenek, dağıtım listeleri ve konular da içinde olmak üzere, tüm kuyruk tipleri için geçerlidir.

Aşağıdaki notlar bu seçenekler için geçerlidir:

- Bu seçeneklerden yalnızca biri belirlenebilir.
- **InhibitGet** kuyruk özneliği MQQA\_GET\_INHIBITED olarak ayarlansa bile, bu seçeneklerden biriyle MQOPER çağrısı başarılı olabilir (ancak, öznitelik bu değere ayarlandığında sonraki MQGET çağrıları başarısız olur).
- Kuyruk paylaşılabilir olarak tanımlandıysa (yani, **Shareability** kuyruk özneliğinin MQQA\_NOT\_SHAREABLE değeri vardır), paylaşılan erişim için kuyruğu açma girişimleri, dışlayıcı erişimle kuyruğu açma girişimi olarak kabul edilir.
- Bu seçeneklerden biriyle bir diğer ad kuyruğu açılırsa, dışlayıcı kullanım sınaması (ya da başka bir uygulamanın dışlayıcı kullanımı olup olmadığı), diğer adın çözümleneceği temel kuyruğa karşıdır.
- These options are not valid if **ObjectQMgrName** is the name of a queue manager alias; this is true even if the value of the **RemoteQMgrName** attribute in the local definition of a remote queue used for queue manager aliasing is the name of the local queue manager.

## MQOO\_BROWSE

İletilere göz atmak için kuyruğu açın.

Kuyruk, sonraki MQGET çağrılarıyla kullanılmak üzere aşağıdaki seçeneklerden biriyle kullanılmak üzere açılır:

- MQGMO\_BROWSE\_FIRST
- MQGMO\_BROWSE\_NEXT
- MQGMO\_BROWSE\_MSG\_UNDER\_CURSOR

Kuyruk şu an MQOO\_INPUT\_EXCLUSIVE için açık olsa bile bu izin verilir. MQOO\_BROWSE seçeneğiyle bir MQOPEN çağrısı bir göz atma imleci oluşturur ve bunu, kuyruğun ilk iletişiyle mantıksal olarak önce mantıksal olarak konumlandırır; ek bilgi için [MQGMO-Options field](#) konusuna bakın.

Bu seçenek yalnızca yerel, diğer ad ve model kuyrukları için geçerlidir; uzak kuyruklar, dağıtım listeleri ve kuyruklar olmayan nesnelere için geçerli değildir. It is also not valid if `ObjectQMgrName` is the name of a queue manager alias; this is true even if the value of the **RemoteQMgrName** attribute in the local definition of a remote queue used for queue manager aliasing is the name of the local queue manager.

## MQOO\_CO\_OP

Çekme noktalarının işbirliği üyesi olarak aç.

Bu seçenek yalnızca MQOO\_BROWSE seçeneği ile geçerlidir. MQOO\_BROWSE olmadan belirtilirse, MQRC\_OPTIONS\_ERROR ile MQOPEN işlevi döndürülür.

Döndürülen tanıtıcı, aşağıdaki seçeneklerden biriyle, sonraki MQGET çağrılarına ilişkin bir iş birliğine ilişkin tanıtıcı takımının bir üyesi olarak kabul edilir:

- MQGMO\_MARK\_BROWSE\_CO\_OP
- MQGMO\_UNMARK\_BROWSE\_MSG
- MQGMO\_UNMARK\_BROWSE\_CO\_OP

Bu seçenek yalnızca yerel, diğer ad ve model kuyrukları için geçerlidir; uzak kuyruklar, dağıtım listeleri ve kuyruklar olmayan nesnelere için geçerli değildir.

## MQOO\_SORGULAMA

Öznitelikleri sorgulamak için nesneyi açın.

İzleyen MQINQ çağrılarıyla kullanılmak üzere kuyruk, ad listesi, süreç tanımlaması ya da kuyruk yöneticisi açılmıştır.

Bu seçenek, dağıtım listesi dışındaki tüm nesne tipleri için geçerlidir. Bir kuyruk yöneticisi diğer adının adı `ObjectQMgrName` ise, bu değer geçerli değildir; bu, kuyruk yöneticisi aliting için kullanılan uzak bir kuyruğun yerel tanımlamasındaki **RemoteQMgrName** özniteliğinin değeri yerel kuyruk yöneticisinin adıdır.

## MQOO\_SET

Öznitelikleri ayarlamak için kuyruğu açın.

Kuyruk, sonraki MQSET çağrılarıyla kullanılmak üzere açılmıştır.

Bu seçenek, dağıtım listesi dışındaki tüm kuyruk tipleri için geçerlidir. It is not valid if `ObjectQMgrName` is the name of a local definition of a remote queue; this is true even if the value of the **RemoteQMgrName** attribute in the local definition of a remote queue used for queue manager aliasing is the name of the local queue manager.

**Bağ tanımlama seçenekleri:** Açılmakta olan nesne bir küme kuyruğu olduğunda, aşağıdaki seçenekler geçerli olur; bu seçenekler kuyruk tanıtıcısı bağ tanımını, küme kuyruğunun bir örneğine denetler:

## **MQOO\_BIND\_ON\_AÇIK**

Kuyruk açıldığı sırada, yerel kuyruk yöneticisi kuyruk tanıtıcısını hedef kuyruğun bir eşgörünümüdür. Sonuç olarak, bu tanıtıcıyı kullanan tüm iletiler hedef kuyruğun aynı örneğine ve aynı rota tarafından gönderilir.

Bu seçenek yalnızca kuyruklar için geçerlidir ve yalnızca küme kuyruklarını etkiler. Bir küme kuyruğu olmayan bir kuyruk için belirtilirse, bu seçenek yoksayılr.

## **MQOO\_BIND\_NOT\_FIXED**

Bu, yerel kuyruk yöneticisinin kuyruk tanıtıcısını hedef kuyruğun bir örneğine bağlamasını durdurur. Sonuç olarak, bu tanıtıcıyı kullanan ardışık MQPUT çağrıları, iletileri hedef kuyruğun farklı eşgörünümüne ya da aynı yönetim ortamına ancak farklı rotalara göndermesini sağlar. Ayrıca, seçilen yönetim ortamının yerel kuyruk yöneticisi tarafından, uzak kuyruk yöneticisi tarafından ya da ağ koşullarına göre bir ileti kanalı aracısı (MCA) tarafından değiştirilmesini de sağlar.

**Not:** Bir işlemin tamamlanması için bir dizi iletinin değiştirilmesi gereken istemci ve sunucu uygulamaları MQOO\_BIND\_NOT\_FIXED (ya da DefBind MQBND\_BIND\_NOT\_FIXED değeri olduğunda MQOO\_BIND\_AS\_Q\_DEF) kullanılmamalıdır; serideki sonraki iletiler, sunucu uygulamasının farklı örneklerine gönderilebilir.

Bir küme kuyruğu için MQOO\_BROWSE ya da MQOO\_INPUT\_\* seçeneklerinden biri belirlenirse, kuyruk yöneticisi, küme kuyruğunun yerel yönetim ortamını seçmekte zorlanır. Sonuç olarak, MQOO\_BIND\_NOT\_FIXED belirtildiyse bile, kuyruk tanıtıcısı bağ tanımı düzeltilir.

MQOO\_SORGULAMA MQOO\_BIND\_NOT\_FIXED ile belirtilirse, bu tanıtıcı kullanılarak yapılan ardışık MQINQ çağrıları küme kuyruğunun farklı eşgörünümünü sorgulayabilir, ancak tipik olarak tüm eşgörünüm aynı öznitelik değerlerine sahip olur.

MQOO\_BIND\_NOT\_FIXED yalnızca kuyruklar için geçerlidir ve yalnızca küme kuyruklarını etkiler. Bir küme kuyruğu olmayan bir kuyruk için belirtilirse, bu seçenek yoksayılr.

## **MQOO\_BIND\_ON\_GROUP**

Bir uygulamanın, bir ileti grubunun tümünün aynı hedef eşgörünümüne ayrılmasını istemesine olanak tanır.

Bu seçenek yalnızca kuyruklar için geçerlidir ve yalnızca küme kuyruklarını etkiler. Bir küme kuyruğu olmayan bir kuyruk için belirtilirse, bu seçenek yoksayılr.

## **MQOO\_BIND\_AS\_Q\_DEF**

Yerel kuyruk yöneticisi, kuyruk tanıtıcısını **DefBind** kuyruk özniteliğe göre tanımlanan şekilde bağlar. Bu özniteliğin değeri, MQBND\_BIND\_ON\_OPEL, MQBND\_BIND\_NOT\_FIXED ya da MQBND\_BIND\_ON\_GROUP olarak değerlendirilmektedir.

MQOO\_BIND\_ON\_RESH, MQOO\_BIND\_NOT\_FIXED ya da MQOO\_BIND\_ON\_GROUP belirtilmediği için, MQOO\_BIND\_AS\_Q\_DEF varsayılan değerdir.

MQOO\_BIND\_AS\_Q\_DEF aids programı belgeleri. Bu seçeneğin diğer iki bağ tanımlama seçeneğinden biriyle birlikte kullanılması amaçlanmaz, ancak değeri sıfır olduğu için bu tür bir kullanım saptanamaz.

**Bağlam seçenekleri:** Aşağıdaki seçenekler ileti bağlamının işlenmesini denetler:

## **MQOO\_SAVE\_ALL\_CONTEXT**

Bağlam bilgileri bu kuyruk tanıtıcısı ile ilişkilendirildi. Bu bilgi, bu tanıtıcı kullanılarak alınan herhangi bir iletinin bağlamından ayarlanır. İleti bağlamına ilişkin daha fazla bilgi için bkz. [İleti bağlamı](#) ve [Bağlam bilgilerini denetleme](#).

Bu bağlam bilgisi, MQPUT ya da MQPUT1 çağrıları kullanan bir kuyruğa aktarılabilecek bir iletiye aktarılabilir. “MQPMO-İleti koyma seçenekleri” sayfa 492’inde açıklanan MQPMO\_PASS\_IDENTITY\_CONTEXT ve MQPMO\_PASS\_ALL\_CONTEXT seçeneklerine bakın.

Bir ileti başarıyla alınıncaya kadar, bir kuyrukta belirtilen bir iletiye bağlam geçirilemez.

MQGMO\_BROWSE\_\* göz atma seçeneklerinden biri kullanılarak alınan bir ileti, bağlam bilgilerinin saklanmasına neden olmaz (**MsgDesc** parametresindeki bağlam alanları göz atma işleminden sonra ayarlansa da).



Bu seçenek yalnızca yerel, diğer ad ve model kuyrukları için geçerlidir; uzak kuyruklar, dağıtım listeleri ve kuyruklar olmayan nesnelere için geçerli değildir. MQOO\_INPUT\_\* seçeneklerinden biri belirtilmelidir.

#### **MQOO\_PASS\_IDENTITY\_CONTEXT**

Bu, bir kuyruğa ilişkin bir ileti konduğunda **PutMsgOpts** parametresinde MQPMO\_PASS\_IDENTITY\_CONTEXT seçeneğinin belirtilmesine olanak sağlar; bu, MQOO\_SAVE\_ALL\_CONTEXT seçeneğiyle açılan bir giriş kuyruğundan kimlik bağlamı bilgilerini verir. İleti bağlamına ilişkin daha fazla bilgi için bkz. [İleti bağlamı](#) ve [Bağlam bilgilerini denetleme](#).

MQOO\_OUTPUT seçeneği belirtilmeli.

Bu seçenek, dağıtım listeleri de içinde olmak üzere, tüm kuyruk tipleri için geçerlidir.

#### **MQOO\_PASS\_ALL\_CONTEXT**

This allows the MQPMO\_PASS\_ALL\_CONTEXT option to be specified in the **PutMsgOpts** parameter when a message is put on a queue; this gives the message the identity and origin context information from an input queue that was opened with the MQOO\_SAVE\_ALL\_CONTEXT option. İleti bağlamına ilişkin daha fazla bilgi için bkz. [İleti bağlamı](#) ve [Bağlam bilgilerini denetleme](#).

Bu seçenek, MQOO\_PASS\_IDENTITY\_CONTEXT ögesini belirtir; bu nedenle, belirtilmemesi gerekir. MQOO\_OUTPUT seçeneği belirtilmeli.

Bu seçenek, dağıtım listeleri de içinde olmak üzere, tüm kuyruk tipleri için geçerlidir.

#### **MQOO\_SET\_IDENTITY\_CONTEXT**

Bu, bir kuyruğa ileti konduğunda **PutMsgOpts** parametresinde MQPMO\_SET\_IDENTITY\_CONTEXT seçeneğinin belirtilmesine izin verir; bu ileti, MQPUT ya da MQPUT1 çağrısında belirtilen **MsgDesc** değıştirgesinde bulunan kimlik bağlamı bilgilerini verir. İleti bağlamına ilişkin daha fazla bilgi için bkz. [İleti bağlamı](#) ve [Bağlam bilgilerini denetleme](#).

Bu seçenek, MQOO\_PASS\_IDENTITY\_CONTEXT ögesini belirtir; bu nedenle, belirtilmemesi gerekir. MQOO\_OUTPUT seçeneği belirtilmeli.

Bu seçenek, dağıtım listeleri de içinde olmak üzere, tüm kuyruk tipleri için geçerlidir.

#### **MQOO\_SET\_ALL\_CONTEXT**

This allows the MQPMO\_SET\_ALL\_CONTEXT option to be specified in the **PutMsgOpts** parameter when a message is put on a queue; this gives the message the identity and origin context information contained in the **MsgDesc** parameter specified on the MQPUT or MQPUT1 call. İleti bağlamına ilişkin daha fazla bilgi için bkz. [İleti bağlamı](#) ve [Bağlam bilgilerini denetleme](#).

Bu seçenek, bu nedenle belirtilmemesi gereken aşağıdaki seçenekleri belirtir:

- MQOO\_PASS\_IDENTITY\_CONTEXT
- MQOO\_PASS\_ALL\_CONTEXT
- MQOO\_SET\_IDENTITY\_CONTEXT

MQOO\_OUTPUT seçeneği belirtilmeli.

Bu seçenek, dağıtım listeleri de içinde olmak üzere, tüm kuyruk tipleri için geçerlidir.

#### **önden okuma seçenekleri:**

MQOO\_READ\_ÖNCEDEN ile MQOPEN ' i çağırdığınızda, IBM MQ istemcisi belirli koşullar karşılandığında yalnızca önden okuma seçeneğini etkinleştirir. Bu koşullar şunlardır:

- Hem istemci, hem de uzak kuyruk yöneticisi IBM WebSphere MQ 7.0 ya da sonraki bir yerde olmalıdır.
- İstemci uygulaması derlenmeli ve iş parçacıklı IBM MQ MQI istemcisi kitaplıklarına bağlanmalıdır.
- İstemci kanalı TCP/IP protokolünü kullanıyor olmalıdır
- Kanal, istemci ve sunucu kanalı tanımlarında sıfır dışında bir SharingConversations (SHARECNV) ayarı içermelidir.

Aşağıdaki seçenekler, bir uygulama tarafından istekte bulunmadan önce istemciye kalıcı olmayan iletilerin gönderilip gönderilmeyeceğini denetler. Öndeki okuma seçenekleri için aşağıdaki notlar geçerlidir:

- Bu seçeneklerden yalnızca biri belirlenebilir.
- Bu seçenekler yalnızca yerel, diğer ad ve model kuyrukları için geçerlidir. Bunlar, uzak kuyruklar, dağıtım listeleri, konular ya da kuyruk yöneticileri için geçerli değildir.
- Bu seçenekler yalnızca MQOO\_BROWSE, MQOO\_INPUT\_SHARED ve MQOO\_INPUT\_EXCLUSIVE değeri de belirtildiyse geçerlidir; ancak, bu seçeneklerin MQOO\_SORGULA ya da MQOO\_SET ile belirtilmesine ilişkin bir hata yoktur.
- Uygulama bir IBM MQ istemcisi olarak çalışmazsa, bu seçenekler yoksayılır.

#### **MQOO\_NO\_READ\_AHEAD**

Kalıcı olmayan iletiler, bir uygulama tarafından istekte bulunmadan önce istemciyi göndermez.

#### **MQOO\_READ\_AHEAD**

Kalıcı olmayan iletiler, bir uygulama tarafından istekte bulunmadan önce istemciye gönderilir.

#### **MQOO\_READ\_AHEAD\_AS\_Q\_DEF**

İleriye okuma davranışı, açılmakta olan kuyruğun varsayılan ileriye okuma özniemesiyle belirlenir. Bu varsayılan değerdir.

**Diğer seçenekler:** Aşağıdaki seçenekler denetim yetkisi denetimini, kuyruk yöneticisi susturulduğunda ne olur, yerel kuyruk adının çözülmesini ve çok sayıda yayını mı çözeceğini denetler:


#### **MQOO\_ALTERNATE\_USER\_AUTHORITY**

**ObjDesc** parametresindeki *AlternateUserId* alanı, bu MQOPEN çağrısını doğrulamak için kullanılacak bir kullanıcı kimliği içerir. Bu çağrı yalnızca, uygulamanın çalışmakta olduğu kullanıcı kimliğinin bu işlemi gerçekleştirme yetkisine sahip olup olmadığına bakılmaksızın, bu *AlternateUserId* nesnesi belirtilen erişim seçenekleriyle nesneyi açma yetkisine sahip olduğunda başarılı olabilir. Ancak, her zaman uygulamanın çalışmakta olduğu kullanıcı kimliği ile karşılaştırılan bağlam seçenekleri için bu seçenek geçerli değildir.

Bu seçenek, tüm nesne tipleri için geçerlidir.

#### **MQOO\_FAIL\_IF QUIESCING**

Kuyruk yöneticisi susturma durumundaysa, MQOPEN çağrısı başarısız olur.

 On z/OS, for a CICS or IMS application, this option also forces the MQOPEN call to fail if the connection is in quiescing state.

Bu seçenek, tüm nesne tipleri için geçerlidir.

İstemci kanallarına ilişkin bilgi için bkz. [IBM MQ MQI clients' a genel bakış](#).

#### **MQOO\_RESOLVE\_LOCAL\_Q**

Açılan MQOD yapısındaki ResolvedQName ögesini, açılan yerel kuyruğun adıyla doldurun. Benzer şekilde, ResolvedQMGrAdı yerel kuyruk barındırma yerel kuyruk yöneticisinin adıyla doldurulur. MQOD yapısı Sürüm 3 'ten küçükse, hiçbir hata döndürülmemesiyle MQOO\_RESOLVE\_LOCAL\_Q yoksayılır.

Yerel kuyruk her zaman yerel, diğer ad ya da model kuyruğu açıldığında döndürülür; ancak, örneğin, uzak kuyruk ya da yerel olmayan bir küme kuyruğu MQOO\_RESOLVE\_LOCAL\_Q seçeneği olmadan açıldığında; ResolvedQName ve ResolvedQMGrAdı, uzak kuyruk tanımlamasında bulunan RemoteQName ve RemoteQMgradı ile ya da seçilen uzak küme kuyruğunda benzer şekilde bulunur.

Açılırken MQOO\_RESOLVE\_LOCAL\_Q belirtilirse, örneğin, uzak bir kuyruk ResolvedQName , iletilerin yerleştirilecek iletim kuyruğudur. ResolvedQMGrAdı, iletim kuyruğunu bulunduran yerel kuyruk yöneticisinin adıyla doldurulur.

Bir kuyrukta göz atma, giriş ya da çıkış için yetkiniz varsa, MQOPEN çağrısında bu işareti belirtmek için gerekli yetkiniz olmalıdır. Özel bir yetki gerekli değildir.

Bu seçenek yalnızca kuyruklar ve kuyruk yöneticileri için geçerlidir.

#### **MQOO\_RESOLVE\_LOCAL\_TOPIC**

MQOD yapısındaki ResolvedQName 'ı, açılan denetim konularının adıyla doldurun.

#### **MQOO\_NO\_MULTICAST**

Yayın iletileri çoklu yayın kullanılarak gönderilmez.

Bu seçenek yalnızca MQOO\_OUTPUT seçeneği ile geçerlidir. MQOO\_OUTPUT olmadan belirtilirse, MQRC\_OPTIONS\_ERROR ile MQOPEN işlevi döndürülür.

Bu seçenek yalnızca bir konu için geçerlidir.

#### **Hobj**

Tip: MQHOBJ-çıkış

Bu tanıtıcı, nesne için kurulmuş olan erişimi gösterir. Bu, nesne üzerinde çalışan sonraki IBM MQ çağrılarında belirtilmelidir. MQCLOSE çağrısı yayınlandığında ya da tutamaç işleminin kapsamını tanımlayan işlem birimi sonlandığında geçerli olmak üzere sona erer.

Döndürülen nesne tanıtıcısı kapsamı, çağrıda belirtilen bağlantı tanıtıcısı kapsamının kapsamı ile aynı. Tutamaç kapsamına ilişkin bilgi için [MQCONN-Hconn değiştirgesi](#) başlıklı konuya bakın.

#### **CompCode**

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Tamamlanma kodu; bu kod aşağıdakilerden biridir:

#### **MQCC\_OK**

İşlem başarıyla tamamlandı.

#### **MQCC\_UYARI**

Uyarı (kısmi tamamlama).

#### **MQCC\_FAILED**

Arama başarısız oldu.

#### **Neden**

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Neden kodu niteleyici *CompCode*.

*CompCode* MQCC\_OK ise:

#### **MQRC\_NONE**

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_UYARI ise:

#### **MQRC\_MULTIPLE\_REASONS**

(2136, X'858 ') Birden çok neden kodu döndürüldü.

*CompCode* MQCC\_FAILED ise:

#### **MQRC\_ADAPTER\_NOT\_AVAM**

(2204, X'89C') Bağdaştırıcı kullanılmıyor.

#### **MQRC\_ADAPTER\_SERV\_LOAD\_ERROR**

(2130, X'852 ') Bağdaştırıcı hizmeti modülü yüklenemiyor.

#### **MQRC\_ALIAS\_BASE\_Q\_TYPE\_ERROR**

(2001, X'7D1') Diğer ad taban kuyruğu geçerli bir tip değil.

#### **MQRC\_API\_EXIT\_ERROR**

(2374, X' 946 ') API çıkışı başarısız oldu.

#### **MQRC\_API\_EXIT\_LOAD\_ERROR**

(2183, X'887 ') API çıkışı yüklenemiyor.

#### **MQRC\_ASID\_MISSMACH**

(2157, X'86D') Birincil ve ana sunucu ASID değerleri farklı.

**MQRC\_CALL\_IN\_PROGRESS**

(2219, X'8AB') Önceki arama tamamlanmadan önce girilen MQI çağrısı.

**MQRC\_CF\_NOT\_VAR**

(2345, X' 929 ') Coupling tesisi kullanılamıyor.

**MQRC\_CF\_STRUC\_AUTH\_FAILED**

(2348, X'92C') Coupling-tesis yapısı yetki denetimi başarısız oldu.

**MQRC\_CF\_STRUC\_ERROR**

(2349, X'92D') Coupling-tesis yapısı geçerli değil.

**MQRC\_CF\_STRUC\_FAILED**

(2373, X' 945 ') Coupling-tesis yapısı başarısız oldu.

**MQRC\_CF\_STRUC\_IN\_USE**

(2346, X'92A') Coupling-tesis yapısı kullanımda.

**MQRC\_CF\_STRUC\_LIST\_HDR\_IN\_USE**

(2347, X'92B') Coupling-tesis yapısı listesi-üstbilgi kullanımda.

**MQRC\_CICS\_WAIT\_FAILED**

(2140, X'85C') Bekleme isteği CICStarafından reddedildi.

**MQRC\_CLUSTER\_EXIT\_ERROR**

(2266, X'8DA') Küme iş yükü çıkışı başarısız oldu.

**MQRC\_CLUSTER\_PUT\_INHIMATED**

(2268, X'8DC') Kümede bulunan tüm kuyruklar için çağrı engelini yerleştirin.

**MQRC\_CLUSTER\_RESOLUTION\_ERROR**

(2189, X'88D') Küme adının çözülmesi başarısız oldu.

**MQRC\_CLUSTER\_KAYNAĞI\_HATASI**

(2269, X'8DD') Küme kaynağı hatası.

**MQRC\_CONNECTION\_BROKEN**

(2009, X'7D9') Kuyruk yöneticisine yönelik bağlantı kaybedildi.

**MQRC\_CONNECTION\_NOT\_YETKILI**

(2217, X'8A9') Bağlantı için yetkili değil.

**MQRC\_CONNECTION\_QUIESCING**

(2202, X'89A') Bağlantı susturulmuş durumda.

**MQRC\_CONNECTION\_DURATION**

(2203, X'89B') Bağlantı sona erdiriliyor.

**MQRC\_DB2\_NOT\_AVAILABLE**

(2342, X' 926 ') Db2 altsistemi kullanılamıyor.

**MQRC\_DEF\_XMIT\_Q\_TYPE\_ERROR**

(2198, X'896 ') Varsayılan iletim kuyruğu yerel değil.

**MQRC\_DEF\_XMIT\_Q\_USAGE\_ERROR**

(2199, X'897 ') Varsayılan iletim kuyruğu kullanım hatası.

**MQRC\_DYNAMIC\_Q\_NAME\_ERROR**

(2011, X'7DB') Dinamik kuyruk adı geçerli değil.

**MQRC\_HANDLE\_NOT\_VAR**

(2017, X'7E1') Kullanılabilir başka tanıtıcı yok.

**MQRC\_HCONN\_ERROR**

(2018, X'7E2') Bağlantı tanıtıcısı geçerli değil.

**MQRC\_HOBJ\_ERROR**

(2019, X'7E3') Nesne tanıtıcısı geçerli değil.

**MQRC\_MULTIPLE\_REASONS**

(2136, X'858 ') Birden çok neden kodu döndürüldü.

**MQRC\_NAME\_IN\_USE**

(2201, X'899 ') Ad kullanımda.

**MQRC\_NAME\_NOT\_VALID\_FOR\_TYPE**  
(2194, X'892 ') Nesne adı nesne tipi için geçerli değil.

**MQRC\_NOT\_YETKILI**  
(2035, X'7F3') Erişim için yetkili değil.

**MQRC\_OBJECT\_ALREADY\_EXISTS**  
(2100, X'834 ') Nesne var.

**MQRC\_OBJECT\_ZARAR**  
(2101, X'835 ') Nesne zarar gördü.

**MQRC\_OBJECT\_IN\_USE**  
(2042, X'7FA') Nesnesi çakışan seçeneklerle zaten açık.

**MQRC\_OBJECT\_LEVEL\_UYUMSUZ**  
(2360, X' 938 ') Nesne düzeyi uyumlu değil.

**MQRC\_OBJECT\_NAME\_ERROR**  
(2152, X'868 ') Nesne adı geçerli değil.

**MQRC\_OBJECT\_NOT\_UNIQUE**  
(2343, X' 927 ') Nesne benzersiz değil.

**MQRC\_OBJECT\_Q\_MGR\_NAME\_ERROR**  
(2153, X'869 ') Nesne kuyruğu yöneticisi adı geçerli değil.

**MQRC\_OBJECT\_RECORDS\_ERROR**  
(2155, X'86B') Nesne kayıtları geçerli değil.

**MQRC\_OBJECT\_STRING\_ERROR**  
(2441, X'0989 ') Objectstring alanı geçerli değil

**MQRC\_OBJECT\_TYPE\_ERROR**  
(2043, X'7FB') Nesne tipi geçerli değil.

**MQRC\_OD\_ERROR**  
(2044, X'7FC') Nesne tanımlayıcı yapısı geçerli değil.

**MQRC\_OPTION\_NOT\_VALID\_FOR\_TYPE**  
(2045, X'7FD') Seçenek, nesne tipi için geçerli değil.

**MQRC\_OPTIONS\_ERROR**  
(2046, X'7FE') Seçenekler geçerli değil ya da tutarlı değil.

**MQRC\_PAGEES\_HATASI**  
(2193, X'891 ') Sayfa kümesi veri kümesine erişilirken hata oluştu.

**MQRC\_PAGEET\_FULL**  
(2192, X'890 ') Dış depolama ortamı dolu.

**MQRC\_Q\_DELETED**  
(2052, X'804 ') Kuyruk silinmiştir.

**MQRC\_Q\_MGR\_NAME\_ERROR**  
(2058, X'80A') Kuyruk yöneticisi adı geçerli değil ya da bilinmiyor.

**MQRC\_Q\_MGR\_NOT\_VAR**  
(2059, X'80B') Kuyruk yöneticisi bağlantı için kullanılabilir değil.

**MQRC\_Q\_MGR QUIESCING**  
(2161, X'871 ') Kuyruk yöneticisi susturuyor.

**MQRC\_Q\_MGR\_STOPPING**  
(2162, X'872 ') Kuyruk yöneticisi sona erdiriliyor.

**MQRC\_Q\_TYPE\_ERROR**  
(2057, X'809 ') Kuyruk tipi geçerli değil.

**MQRC\_RECS\_PRESENT\_ERROR**  
(2154, X'86A') Kayıt sayısı geçerli değil.

**MQRC\_REMOTE\_Q\_NAME\_ERROR**  
(2184, X'888 ') Uzak kuyruk adı geçerli değil.

**MQRC\_RESOURCE\_SORUNU**

(2102, X'836 ') Yetersiz sistem kaynakları var.

**MQRC\_RESPONSE\_RECORDS\_ERROR**

(2156, X'86C') Yanıt kayıtları geçerli değil.

**MQRC\_SECURITY\_ERROR**

(2063, X'80F') Güvenlik hatası oluştu.

**MQRC\_SELECTOR\_SYNTAX\_ERROR**

2459 (X'099B') Bir MQOP, MQPUT1 ya da MQSUB çağrısı yayınlandı, ancak sözdizimi hatası içeren bir seçim dizgisi belirtildi.

**MQRC\_STOPPED\_BY\_CLUSTER\_EXIT**

(2188, X'88C') Küme iş yükü çıkışı tarafından çağrılan çağrı reddedildi.

**MQRC\_STORAGE\_MEDIUM\_FULL**

(2192, X'890 ') Dış depolama ortamı dolu.

**MQRC\_STORAGE\_NOT\_AVAM**

(2071, X'817 ') Kullanılabilir bellek yetersiz.

**MQRC\_SUPPRESSED\_BY\_EXIT**

(2109, X'83D') Çıkış programı tarafından çağrı engellendi.

**MQRC\_UNEXPECTED\_ERROR**

(2195, X'893 ') Beklenmeyen bir hata oluştu.

**MQRC\_UNKNOWN\_ALIAS\_BASE\_Q**

(2082, X'822 ') Bilinmeyen diğer ad temel kuyruğu.

**MQRC\_UNKNOWN\_DEF\_XMIT\_Q**

(2197, X'895 ') Bilinmeyen varsayılan iletim kuyruğu.

**MQRC\_UNKNOWN\_OBJECT\_NAME**

(2085, X'825 ') Bilinmeyen nesne adı.

**MQRC\_UNKNOWN\_OBJECT\_Q\_MGR**

(2086, X'826 ') Bilinmeyen nesne kuyruk yöneticisi.

**MQRC\_UNKNOWN\_REMOTE\_Q\_MGR**

(2087, X'827 ') Bilinmeyen uzak kuyruk yöneticisi.

**MQRC\_UNKNOWN\_XMIT\_Q**

(2196, X'894 ') Bilinmeyen iletim kuyruğu.

**MQRC\_HATA\_CF\_DÜZEYI**

(2366, X'93E') Coupling-tesis yapısı yanlış düzeyde.

**MQRC\_XMIT\_Q\_TYPE\_ERROR**

(2091, X'82B') İletim kuyruğu yerel değil.

**MQRC\_XMIT\_Q\_USAGE\_ERROR**

(2092, X'82C') İletim kuyruğu yanlış kullanıma sahip.

Bu kodlarla ilgili ayrıntılı bilgi için [İletiler ve neden kodları](#) başlıklı konuya bakın.

## Genel kullanım notları

1. Açılan nesne aşağıdakilerden biridir:

- Kuyruğun bulunduğu yer:
  - İletileri alma ya da göz atma (MQGET çağrısını kullanarak)
  - İletileri koy (MQPUT çağrısını kullanarak)
  - Kuyruğun özniteliklerine ilişkin olarak sorgulama (MQINQ çağrısını kullanarak)
  - Kuyruğun özniteliklerini ayarlar (MQSET çağrısını kullanarak)

Kuyruk adı bir model kuyruğusa, dinamik bir yerel kuyruk yaratılır. [“MQOpen-Open nesnesi” sayfa 720](#) içinde açıklanan **ObjDesc** parametresine bakın.

Dağıtım listesi, kuyrukların listesini içeren özel bir kuyruk nesnesidir. İleti koymak için açılabilir, ancak ileti almak ya da bunlara göz atmak ya da öznitelikleri sorgulamak ya da ayarlamak için açılabilir. Ek ayrıntılar için kullanım notunun 8 'e bakın.

QSGDISP (GROUP) sahip olan bir kuyruk, MQOPEN ya da MQPUT1 çağrılarıyla kullanılmayacak özel bir kuyruk tanımlaması tipidir.

- Listedeki kuyrukların adlarını sorgulamak için kullanılan bir ad listesi (MQINQ çağrısını kullanarak).
  - Süreç öznitelikleriyle ilgili olarak (MQINQ çağrısını kullanarak) sorgulanacak bir süreç tanımlaması.
  - Kuyruk yöneticisi, yerel kuyruk yöneticisinin özniteliklerini sorgulamak için (MQINQ çağrısını kullanarak).
  - Bir iletiyi yayınlamak için bir konu (MQPUT çağrısını kullanarak)
2. Bir uygulama aynı nesneyi bir kereden fazla açabilir. Her bir açık için farklı bir nesne tanıtıcısı döndürülür. Döndürülen her bir tanıtıcı, ilgili açık olan işlevlerin gerçekleştirildiği işlevler için kullanılabilir.
3. Açılmakta olan nesne, bir küme kuyruğundan başka bir kuyruksa, yerel kuyruk yöneticisi içindeki tüm ad çözümlemesi, MQOPER çağrısının saatinde gerçekleşir. Bu işlem aşağıdakileri içerebilir:
- Uzak bir kuyruğun yerel tanımlamasını uzak kuyruk yöneticisinin adına ve kuyruğun uzak kuyruk yöneticisinde bulunduğu ada göre çözme
  - Uzak kuyruk yöneticisi adının yerel iletim kuyruğu adına çözülmesi
  - **z/OS** Yalnızca z/OS üzerinde, uzak kuyruk yöneticisi adının, IGQ aracısı tarafından kullanılan paylaşılan iletim kuyruğu adına çözülmesi (yalnızca yerel ve uzak kuyruk yöneticileri aynı kuyruk paylaşım grubuna aitse uygulanır).
  - Diğer ad çözümlemesi, bir temel kuyruk ya da bir konu nesnesiyle ilgili olarak.

Ancak, tanıtıcı için sonraki MQINQ ya da MQSET çağrılarının yalnızca açılmış olan ada ve ad çözme işleminden sonra ortaya çıkan nesne için değil, bu adla ilişkilendirilmesinin farkında olun. Örneğin, açılan nesne bir diğer ad ise, MQINQ çağrısının döndürdüğü öznitelikler, diğer adın öznitelikleridir; temel kuyruğun öznitelikleri ya da diğer adın çözümleyicilerinin bulunduğu bir konu nesnesidir.

Açılmakta olan nesne bir küme kuyruğuna, ad çözümlemesi, MQOPEN çağrısı sırasında ortaya çıkabilir ya da daha sonra ertelenebilir. Çözülme işleminin gerçekleştirildiği nokta, MQOPEN çağrısında belirtilen MQOO\_BIND\_\* seçenekleri tarafından denetlenir.

- MQOO\_BIND\_ON\_AÇIK
- MQOO\_BIND\_NOT\_FIXED
- MQOO\_BIND\_AS\_Q\_DEF
- MQOO\_BIND\_ON\_GROUP

Küme kuyruklarına ilişkin ad çözümlemesiyle ilgili ek bilgi için [Ad çözünürlüğü](#) başlıklı konuya bakın.

4. MQOO\_BROWSE seçeneğiyle MQOPEN çağrısı, nesne tanıtıcısını ve göz atma seçeneklerinden birini belirten MQGET çağrılarını kullanarak kullanılmak üzere bir göz atma imleçle oluşturur. Bu, kuyruğun içeriği değiştirilmeden taranmasını sağlar. Gözetme yoluyla bulunan bir ileti, MQGMO\_MSG\_UNDER\_CURSOR seçeneği kullanılarak kuyruktan kaldırılabilir.

Aynı kuyruk için birden çok MQOPEN isteği yayınlamak, tek bir uygulama için birden çok imleçle etkin durumda olabilen geçici çizelge sayısı kullanılabilir.

5. Tetikleme izleme programı tarafından başlatılan uygulamalar, uygulama başlatıldığında uygulamayla ilişkili olan kuyruğun adını iletir. Kuyruğu açmak için bu kuyruk adı **ObjDesc** değiştirilmesinde belirtilebilir. Ek ayrıntılar için "[MQTMC2 -Tetikleyici ileti 2 \(karakter biçimi\)](#)" sayfa 596 ' e bakın.

## Önden okuma seçenekleri

MQOO\_READ\_ÖNCEDEN ile MQOPEN ' i çağırdığınızda, IBM MQ istemcisi belirli koşullar karşılandığında yalnızca önden okuma seçeneğini etkinleştirir. Bu koşullar şunlardır:

- Hem istemci, hem de uzak kuyruk yöneticisi IBM WebSphere MQ 7.0 ya da sonraki bir yerde olmalıdır.
- İstemci uygulaması derlenmeli ve iş parçacıklı IBM MQ MQI istemcisi kitaplıklarına bağlanmalıdır.
- İstemci kanalı TCP/IP protokolünü kullanıyor olmalıdır
- Kanal, istemci ve sunucu kanalı tanımlarında sıfır dışında bir SharingConversations (SHARECNV) ayarı içermelidir.

Aşağıdaki notlarda, önden okuma seçeneklerinin kullanılması için geçerlidir.

1. İleriye okuma seçenekleri yalnızca, MQOO\_BROWSE, MQOO\_INPUT\_SHARED ve MQOO\_INPUT\_EXCLUSIVE seçeneklerinin yalnızca biri ve yalnızca bir tanesi de belirtildiğinde geçerlidir. MQOO\_SORGULAMASI YA DA MQOO\_SET seçenekleriyle önkuma seçenekleri belirtilirse bir hata yayınlanmaz.
2. İlk MQGET çağrısında kullanılan seçenekler, ileriye okuma ile kullanım için desteklenmiyorsa, önceden okuma seçeneği etkinleştirilmez. Ayrıca, önceden okunmayı desteklemeyen bir kuyruk yöneticisine istemci bağlandığında, okuma öncesinde de okuma geçersiz kılır.
3. Uygulama bir IBM MQ istemcisi olarak çalışmıyorsa, ileriye doğru okuma seçenekleri yoksayılr.

## Küme kuyrukları

Aşağıdaki notlar küme kuyruklarının kullanımı için geçerlidir.

1. Bir küme kuyruğu ilk kez açıldığında ve yerel kuyruk yöneticisi tam bir havuz kuyruğu yöneticisi değilse, yerel kuyruk yöneticisi, tam havuz kuyruk yöneticisinden küme kuyruğuna ilişkin bilgileri alır. Ağ meşgul olduğunda, yerel kuyruk yöneticisinin gerekli bilgileri havuz kuyruğu yöneticisinden alması birkaç saniye sürebilir. Sonuç olarak, MQOPER çağrısını yayınlayan uygulamanın, denetim MQOPEN çağrısından döndürmeden önce en çok 10 saniye beklemesi gerekebilir. Yerel kuyruk yöneticisi, bu süre içinde küme kuyrukla ilgili gerekli bilgileri alamazsa, çağrı neden kodu MQRC\_CLUSTER\_RESOLUTION\_ERROR ile başarısız olur.
2. Bir küme kuyruğu açıldığında ve kümede kuyruğun birden çok eşgörünümü varsa, açılan eşgörünüm, MQOPEN çağrısında belirtilen seçeneklere bağlıdır:
  - Belirtilen seçenekler aşağıdakilerden birini içerirse:
    - MQOO\_BROWSE
    - MQOO\_INPUT\_AS\_Q\_DEF
    - MQOO\_INPUT\_EXCLUSIVE
    - MQOO\_INPUT\_SHARED
    - MQOO\_SET

Açılan küme kuyruğunun yönetim ortamı yerel yönetim ortamı olmalıdır. Kuyruğun yerel bir eşgörünümü yoksa, MQOPEN çağrısı başarısız olur.
  - Belirtilen seçenekler, daha önce açıklanan seçeneklerden hiçbiri içermiyorsa, aşağıdakilerden birini ya da her ikisini de ekleyin:
    - MQOO\_SORGULAMA
    - MQOO\_OUTPUT

Açılan yönetim ortamı, varsa, yerel yönetim ortamı ve tersi durumda (CLWLUSEQ varsayılanları kullanılıyorsa) yerel yönetim ortağıdır. Ancak, kuyruk yöneticisi tarafından seçilen yönetim ortamı bir küme iş yükü çıkışıyla (varsa) değiştirilebilir.
3. Kuyruk için bir abonelik varsa, ancak bu abonelik tam bir havuz tarafından onaylanmadıysa, nesne kümede yok ve çağrı, MQRC\_OBJECT\_NAME neden koduyla başarısız olur.

Küme kuyruklarıyla ilgili ek bilgi için [Küme kuyrukları](#) başlıklı konuya bakın.

## Dağıtım listeleri

Aşağıdaki notlar dağıtım listelerinin kullanımı için geçerlidir.



Dağıtım listeleri aşağıdaki ortamlarda desteklenir:

-  AIX
-  IBM i
-  Linux
-  Windows

ve bu sistemlere bağlı IBM MQ MQI clients için.

1. Bir dağıtım listesi açılırken, MQOD yapısındaki alanların aşağıdaki gibi ayarlanması gerekir:

- Version , MQOD\_VERSION\_2 ya da daha büyük bir değer olmalıdır.
- ObjectType , MQOT\_Q olmalıdır.
- ObjectName boş bırakılmalı ya da boş değerli dizgi olmalıdır.
- ObjectQMgrName boş bırakılmalı ya da boş değerli dizgi olmalıdır.
- RecsPresent sıfırdan büyük olmalıdır.
- ObjectRecOffset ve ObjectRecPtr ' dan biri sıfır ve diğer sıfır dışında bir değer olmalıdır.
- ResponseRecOffset ve ResponseRecPtr dışında birden çok kişi sıfır olamaz.
- ObjectRecOffset ya da ObjectRecPtr tarafından adreslenen RecsPresent nesne kayıtları olmalıdır. Nesne kayıtlarının açılması için hedef kuyrukların adlarına ayarlanması gerekir.
- If one of ResponseRecOffset and ResponseRecPtr is nonzero, there must be RecsPresent response records present. Arama tamamlanmışsa kuyruk yöneticisi tarafından ayarlanır. Neden kodu MQRC\_MULTIPLE\_REASONS ile tamamlanır.

Bir version-2 MQOD, RecsPresent ' un sıfır olduğunu doğrularak, dağıtım listesinde olmayan tek bir kuyruğu açmak için de kullanılabilir.

2. **Options** değiştirilmesinde yalnızca aşağıdaki açık seçenekler geçerlidir:

- MQOO\_OUTPUT
- MQOO\_PASS\_ \* \_CONTEXT
- MQOO\_SET\_ \* \_CONTEXT
- MQOO\_ALTERNATE\_USER\_AUTHORITY
- MQOO\_FAIL\_IF\_QUIESCING

3. Dağıtım listesindeki hedef kuyruklar yerel, diğer ad ya da uzak kuyruklar olabilir, ancak bunlar model kuyrukları olamaz. Bir model kuyruğu belirtilirse, bu kuyruk açılmazsa, neden kodu MQRC\_Q\_TYPE\_ERROR olur. Ancak bu, listedeki diğer kuyrukların başarıyla açılmamasını engellememektedir.

4. Tamamlanma kodu ve neden kodu değiştirilmesinde aşağıdaki gibi ayarlanır:

- Dağıtım listesindeki kuyruklara ilişkin açık işlemler başarılı olursa ya da aynı şekilde başarısız olursa, tamamlanma kodu ve neden kodu değiştirilmesinde ortak sonucu tanımlamaya ayarlanır. Bu durumda, MQRR yanıt kayıtları (uygulama tarafından sağlandıysa) belirlenmez.

Örneğin, her açık başarılı başarılı olursa, tamamlanma kodu MQCC\_OK olarak ayarlanır ve neden kodu MQRC\_NONE; olarak ayarlanır; her bir açma başarısız olursa, kuyrukların hiçbiri var olmadığından, parametreler MQCC\_FAILED ve MQRC\_UNKNOWN\_OBJECT\_NAME olarak ayarlanır.

- Dağıtım listesindeki kuyruklara ilişkin açık işlemler başarısız olamazsa ya da aynı şekilde başarısız olursa:
  - En az bir açma başarılı olursa, tamamlanma kodu parametresi MQCC\_UYARI değerine ayarlanır ve tümü başarısız olursa MQCC\_FAILED değerine ayarlanır.
  - Neden kodu parametresi MQRC\_MULTIPLE\_REASONS olarak ayarlanır.

- Yanıt kayıtları (uygulama tarafından sağlandıysa), dağıtım listesindeki kuyruklara ilişkin tek tek tamamlama kodlarına ve neden kodlarına ayarlanmışsa.
5. When a distribution list has been opened successfully, the handle Hobj returned by the call can be used on subsequent MQPUT calls to put messages to queues in the distribution list, and on an MQCLOSE call to relinquish access to the distribution list. Dağıtım listesi için geçerli olan tek kapatma seçeneği MQCO\_NONE olur.  
MQPUT1 çağrısı, bir dağıtım listesine ileti koymak için de kullanılabilir; listedeki kuyrukları tanımlayan MQOD yapısı, o çağrıdaki bir parametre olarak belirtilir.
  6. Uygulamanın izin verilen tanıtıcı sayısı üst sınırını aştığını (**MaxHandles** kuyruk yöneticisi özniteliğe bakın) kontrol ederken, dağıtım listesindeki her başarıyla açılan hedef ayrı bir tanıtıcı olarak sayılır. Dağıtım listesindeki hedeflerin ikisi ya da daha çoğu aynı fiziksel kuyruğa çözüldüğünde bile bu durum geçerlidir. Bir dağıtım listesi için MQOPEN ya da MQPUT1 çağrısı, uygulama tarafından kullanılan tanıtıcı sayısının MaxHandles' u aşmasına neden olursa, çağrı neden kodu MQRC\_HANDLE\_NOT\_AVAILABLE koduyla başarısız olur.
  7. Açılan her hedef, **OpenOutputCount** özniteliğinin değerine göre bir artırılarak bir değer artırılır. Dağıtım listesindeki hedeflerden iki ya da daha fazlası aynı fiziksel kuyruğa çözülsün, o kuyruk, dağıtım listesindeki hedef sayısı tarafından bu kuyruğa uygun olarak artırılan **OpenOutputCount** özniteliğine sahip olur.
  8. Bir tanıtıcıyı geçersiz hale getirmesine neden olacak kuyruk tanımlamalarında yapılan herhangi bir değişiklik, kuyrukların tek tek açılmasına neden oldu (örneğin, çözüm yolundaki bir değişiklik), dağıtım listesi tutamaçlarının geçersiz olmasına neden olmaz. Ancak, sonraki bir MQPUT çağrısında dağıtım listesi tanıtıcısı kullanıldığında, bu kuyruğun başarısızlığa neden olması da sonuçlanabilir.
  9. Dağıtım listesi yalnızca bir hedef içerebilir.

## Uzak kuyruklar

Aşağıdaki notlar uzak kuyrukların kullanılması için geçerlidir.

Uzak bir kuyruk, bu çağrıya ilişkin **ObjDesc** parametresindeki iki yöntemden birinde belirtilebilir.

- **ObjectName** için, uzak kuyruğun yerel tanımlamasının adını belirtilerek. Bu durumda, **ObjectQMgrName** yerel kuyruk yöneticisine başvurur ve boşluk olarak ya da (C programlama dilinde) boş değerli bir dizgi olarak belirtilebilir.

Yerel kuyruk yöneticisi tarafından gerçekleştirilen güvenlik doğrulaması, kullanıcının uzak kuyruğun yerel tanımlamasını açma yetkisine sahip olduğunu doğrular.

- **ObjectName** için, uzak kuyruk yöneticisi tarafından bilindiği şekilde uzak kuyruğun adını belirterek. Bu durumda, **ObjectQMgrName** uzak kuyruk yöneticisinin adıdır.

Yerel kuyruk yöneticisi tarafından gerçekleştirilen güvenlik doğrulaması, kullanıcının ad çözme işleminden kaynaklanan iletim kuyruğuna ileti gönderme yetkisine sahip olduğunu doğrular.

Her iki durumda da:

- Kullanıcının kuyruğa ileti koyma yetkisi olup olmadığını denetlemek için, yerel kuyruk yöneticisi tarafından uzak kuyruk yöneticisine hiçbir ileti gönderilmez.
- Uzak kuyruk yöneticisinde bir ileti geldiğinde, iletiyi oluşturan kullanıcının yetkisi olmadığı için uzak kuyruk yöneticisi bunu reddedebilir.

Ek bilgi için [“MQOD-Nesne tanımlayıcı” sayfa 473](#) içinde açıklanan **ObjectName** ve **ObjectQMgrName** alanlarına bakın.

## Nesneler

### Güvenlik

Aşağıdaki notlarda, MQOPEN kullanılmasının güvenlik yönleriyle ilişkilendirilmiş.

Kuyruk yöneticisi, uygulamanın çalışmakta olduğu kullanıcı kimliğinin erişim izni verilmeden önce uygun yetki düzeyine sahip olduğunu doğrulamak için, bir MQOPEN çağrısı yayınlandığında güvenlik denetimleri gerçekleştirir. Yetki denetimi, bir ad çözüldükten sonra, açılmakta olan nesnenin adı ya da adlarında değil, açılmakta olan nesnenin adı üzerinde yapılır.

Açılmakta olan nesne bir konu nesnesini işaret eden bir diğer ad kuyruğuna, kuyruk yöneticisi, konu nesnesinin doğrudan kullanıldığı gibi, konu için bir güvenlik denetimi gerçekleştirilmeden önce, diğer ad kuyruğu adında bir güvenlik denetimi gerçekleştirir.

Açılmakta olan nesne bir konu nesnesiyse, `ObjectName` ile ya da `ObjectString` kullanılarak (basingile ya da baslayan `ObjectName` ile), kuyruk yöneticisi, sonuçtaki konu dizesini kullanarak, `ObjectName`'ta belirtilen konu nesnesinden alınan ve `ObjectString`'ta sağlanan ve gerekiyorsa, güvenlik denetimini gerçekleştirmek için konu ağacındaki en yakın konu nesnesini bulmak için, sonuçtaki konu dizesini kullanarak güvenlik denetimini gerçekleştirir. Bu, `ObjectName`'inde belirtilen konu nesnesiyle aynı olmayabilir.

Açılmakta olan nesne bir model kuyruğuyse, kuyruk yöneticisi hem model kuyruğunun adı, hem de yaratılan dinamik kuyruğun adı için bir tam güvenlik denetimi gerçekleştirir. Sonuçta ortaya çıkan dinamik kuyruk açık bir şekilde açılırsa, dinamik kuyruk adına göre daha fazla kaynak güvenliği denetimi gerçekleştirilir.

**z/OS** z/OS üzerinde, kuyruk yöneticisi yalnızca güvenlik etkinleştirilmişse güvenlik denetimlerini gerçekleştirir. Güvenlik denetimiyle ilgili daha fazla bilgi için bkz. [z/OS üzerinde güvenliğin ayarlanması](#).

## Öznitelikler

Aşağıdaki notlarda özniteliklerle ilgili bilgiler yer aldı.

Bir nesnenin öznitelikleri, bir uygulama için açık olduğunda, nesnenin öznitelikleri değişebilir. Birçok durumda, uygulama bunu fark etmez; ancak, bazı öznitelikler için kuyruk yöneticisi tanıtıcıyı artık geçerli değil olarak işaretler. Bu öznitelikler şunlardır:

- Nesnenin ad çözümlemesini etkileyen herhangi bir öznitelik. Bu, kullanılan açık seçeneklerden bağımsız olarak geçerlidir ve aşağıdakileri içerir:
  - Açık olan bir diğer ad kuyruğunun **BaseQName** özniteliğine yapılan bir değişiklik.
  - Açık olan bir diğer ad kuyruğunun **TargetType** özniteliğine yapılan bir değişiklik.
  - A change to the **RemoteQName** or **RemoteQMgrName** queue attributes, for any handle that is open for this queue, or for a queue that resolves through this definition as a queue manager alias.
  - Uzak kuyruğun şu anda açık olan bir uzak kuyruğun farklı bir iletim kuyruğuna çözümlenmesine neden olan ya da çözülemekte başarısız olan herhangi bir değişiklik başarısızlıkla sonuçlansa. Örneğin, şunları içerebilir:
    - Tanımın bir kuyruk için mi, yoksa bir kuyruk yöneticisi diğer adı için mi kullanılmakta olduğunu, uzak kuyruğun yerel tanımlamasının **XmitQName** özniteliğine ilişkin bir değişiklik.
    - **z/OS** Yalnızca z/OS üzerinde, **IntraGroupqueuing** kuyruk yöneticisi özniteliğinin değerine ilişkin bir değişiklik ya da paylaşılan iletim kuyruğunun tanımlamasındaki bir değişiklik (SYSTEM.QSG.TRANSMIT.QUEUE) IGQ aracı tarafından kullanılır.

Bu konuda bir kural dışı durum vardır: yeni bir iletim kuyruğu yaratılması. Bu kuyruğa çözülmüş olan bir tanıtıcı, tanıtıcı açıldığı zaman, ancak varsayılan iletim kuyruğuna çözüldüğünde, geçersiz kılınmadığında bu kuyruğa ilişkin olarak çözülmüş olabilir.

- **DefXmitQName** kuyruk yöneticisi özniteliğine ilişkin bir değişiklik. Bu durumda, önceden adlandırılan kuyruğa (yalnızca varsayılan iletim kuyruğu olduğu için) çözülmüş olan tüm açık tutamaçlar geçersiz olarak işaretlenir. Bu kuyruğun diğer nedenleri için çözülen işler etkilenmez.
- Bu kuyruk için şu anda MQOO\_INPUT\_SHARED erişimi sağlayan iki ya da daha çok tanıtıcı varsa ya da bu kuyruğa çözülen bir kuyruk için, **Shareability** kuyruk özniteliği. Bu durumda, bu kuyruk için açık olan tüm tutamaçlar ya da bu kuyruğa çözülen bir kuyruk için, açık seçeneklerden bağımsız olarak, geçersiz olarak işaretlenir.



z/OS üzerinde, daha önce tanımlanan tutamaçlar, şu anda bir ya da daha çok tanıtıcı MQOO\_INPUT\_SHARED ya da MQOO\_INPUT\_EXCLUSIVE erişimini sağlıyorsa, geçersiz olarak işaretlenir.

- The **Usage** queue attribute, for all handles that are open for this queue, or for a queue that resolves to this queue, regardless of the open options.

Bir tanıtıcı geçersiz olarak imlenirse, bu tanıtıcıyı kullanarak sonraki tüm çağrılar (MQCLOSE dışında) başarısız olur; neden kodu MQRC\_OBJECT\_CHANGED neden kodu. Uygulama bir MQCLOSE çağrısı yayınlamalıdır (özgün tanıtıcıyı kullanarak) ve sonra kuyruğu yeniden açmalıdır. Uygulama mantığının gerektirdiği şekilde, önceki başarılı aramalardan eski tanıtıcı ile ilgili kesinleştirilmemiş güncellemeler kesinleştirilebilir ya da yedeklenebilirler.

Bir öznitelik değiştirilmesinin gerçekleşmesine neden olursa, çağrı için özel bir zorlama sürümü kullanın.

## C çağırma

```
MQOPEN (Hconn, &ObjDesc, Options, &Hobj, &CompCode,  
&Reason);
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
MQHCONN Hconn; /* Connection handle */  
MQOD ObjDesc; /* Object descriptor */  
MQLONG Options; /* Options that control the action of MQOPEN */  
MQHOBJ Hobj; /* Object handle */  
MQLONG CompCode; /* Completion code */  
MQLONG Reason; /* Reason code qualifying CompCode */
```

## Cobol çağırma

```
CALL 'MQOPEN' USING HCONN, OBJDESC, OPTIONS, HOBJ, COMPCODE, REASON
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
** Connection handle  
01 HCONN PIC S9(9) BINARY.  
** Object descriptor  
01 OBJDESC.  
COPY CMQODV.  
** Options that control the action of MQOPEN  
01 OPTIONS PIC S9(9) BINARY.  
** Object handle  
01 HOBJ PIC S9(9) BINARY.  
** Completion code  
01 COMPCODE PIC S9(9) BINARY.  
** Reason code qualifying COMPCODE  
01 REASON PIC S9(9) BINARY.
```

## PL/I çağırımı

```
call MQOPEN (Hconn, ObjDesc, Options, Hobj, CompCode, Reason);
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
dcl Hconn fixed bin(31); /* Connection handle */  
dcl ObjDesc like MQOD; /* Object descriptor */  
dcl Options fixed bin(31); /* Options that control the action of  
MQOPEN */  
dcl Hobj fixed bin(31); /* Object handle */  
dcl CompCode fixed bin(31); /* Completion code */  
dcl Reason fixed bin(31); /* Reason code qualifying CompCode */
```

## High Level Assembler çağırısı

```
CALL MQOPEN, (HCONN, OBJDESC, OPTIONS, HOBJ, COMPCODE, REASON)
```

Değiştirgeleri aşağıdaki gibi bildirin:

HCONN	DS	F	Connection handle
OBJDESC	CMQODA	,	Object descriptor
OPTIONS	DS	F	Options that control the action of MQOPEN
HOBJ	DS	F	Object handle
COMPCODE	DS	F	Completion code
REASON	DS	F	Reason code qualifying COMPCODE

## Visual Basic çağırısı

Windows

```
MQOPEN Hconn, ObjDesc, Options, Hobj, CompCode, Reason
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
Dim Hconn As Long 'Connection handle'  
Dim ObjDesc As MQOD 'Object descriptor'  
Dim Options As Long 'Options that control the action of MQOPEN'  
Dim Hobj As Long 'Object handle'  
Dim CompCode As Long 'Completion code'  
Dim Reason As Long 'Reason code qualifying CompCode'
```

## MQPUT-İleti koy

MQPUT çağırısı, bir kuyruğa ya da dağıtım listesine ya da bir konuya ileti koyar. Kuyruk, dağıtım listesi ya da konu zaten açık olmalıdır.

## Sözdizimi

MQPUT (*Hconn, Hobj, MsgDesc, PutMsgOpts, BufferLength, Buffer, CompCode, Reason*)

## Parametreler

### Hconn

Tip: MQHCONN-input

Bu tanıtcı, kuyruk yöneticisiyle olan bağlantıyı gösterir. The value of Hconn was returned by a previous MQCONN or MQCONNX call.

**z/OS** On z/OS for CICS applications the MQCONN call can be omitted, and the following value specified for *Hconn* :

### MQHC\_DEF\_HCONN

Varsayılan bağlantı tanıtcısı.

### Hobj

Tip: MQHOBJ-input

Bu tanıtcı, iletinin eklendiği kuyruğu ya da iletinin yayınlandığı konuyu gösterir. Hobj değeri, MQOO\_OUTPUT seçeneğini belirten önceki bir MQOPEN çağırısıyla döndürüldü.

### MsgDesc

Tip: MQMD-giriş/çıkış

Bu yapı, gönderilmekte olan iletinin özniteliklerini tanımlar ve koyma isteği tamamlandıktan sonra iletiyle ilgili bilgileri alır. Ayrıntılar için bkz. [“MQMD-İleti tanımlayıcı” sayfa 416.](#)

Uygulama bir version-1 MQMD sağlıyorsa, ileti verilerinin başında, version-2 MQMD içinde var olan alanlara ilişkin değerleri belirtmek için bir MQMDE yapısı eklenmiş olabilir, ancak version-1 bu şekilde bir MQMD yapısı eklenmeyebilir. MQMD 'deki *Biçim* alanı, MQMDE' nin var olduğunu göstermek için MQFM\_MD\_UNUTPT olarak ayarlanmalıdır. Daha ayrıntılı bilgi için bkz. [“MQMDE-İleti tanımlayıcı uzantısı” sayfa 464](#) .

MQPMO yapısının OriginalMsgHandle ya da NewMsgHandle alanlarında geçerli bir ileti tanıtıcısı sağlandıysa, uygulamanın bir MQMD yapısı sağlaması gerekmez. Bu alanlardan birinde hiçbir şey sağlanmıyorsa, iletinin tanımlayıcı, ileti tanıtıcılarıyla ilişkili tanımlayıcıdan alınır.

API çıkışlarını kullanırsanız ya da kullanmayı planlıyorsanız, bir MQMD yapısını açık bir şekilde sağlamanızı ve ileti tutamaçlarıyla ilişkili ileti tanımlayıcılarını kullanmayamanızı öneririz. Bunun nedeni, MQPUT ya da MQPUT1 çağrısıyla ilişkilendirilmiş API çıkışının, kuyruk yöneticisi tarafından MQPUT ya da MQPUT1 isteğini tamamlamak için hangi MQMD değerlerinin kullanılacağını kesinlemede şekilde ifade eder.

### **PutMsg(PutMsg)**

Tip: MQPMO-giriş/çıkış

Ayrıntılar için bkz. [“MQPMO-İleti koyma seçenekleri” sayfa 492](#).

### **BufferLength**

Tip: MQHOME-girişi

The length of the message in Buffer. Sıfır geçerli ve iletinin herhangi bir uygulama verisi içermediği anlamına gelir. BufferLength için üst sınır çeşitli etmenlere bağlıdır:

- Hedef yerel bir kuyruksa ya da yerel bir kuyruğa çözüyorsa, üst sınır aşağıdakilerden biri olup olmadığına bağlıdır:
  - Yerel kuyruk yöneticisi kesimlere ayırma özelliğini destekler.
  - Gönderme uygulaması, kuyruk yöneticisinin iletiyi bölümlemesine izin veren işareti belirtir. Bu işaret, MQMF\_SEGMENTATION\_ALLOWLU işaretidir ve version-2 MQMD ' de ya da version-1 MQMD ile kullanılan bir MQMDE içinde belirtilebilir.

Bu koşulların her ikisi de karşılanırsa, BufferLength , MQMD ' deki Offset alanının değeri ile 999 999 999 'u geçemez. Koyulabilecek en uzun mantıksal ileti, bu nedenle 999 999 bayttır ( Offset sıfır olduğunda). Ancak, uygulamanın çalıştırıldığı işletim sistemi ya da ortam tarafından uygulanan kaynak kısıtlamaları daha düşük bir sınırla sonuçlanabilir.

Önceki koşullardan biri ya da her ikisi karşılanmazsa, BufferLength , kuyruğun **MaxMsgLength** özneliğinin ve kuyruk yöneticisinin **MaxMsgLength** özneliğinin daha küçük değerini geçemez.

- Hedef, uzak bir kuyruksa ya da uzak bir kuyruğa çözülsün, yerel kuyruklara ilişkin koşullar geçerli olur, ancak hedef kuyruğa ulaşmak için iletinin geçmesi gereken her kuyruk yöneticisinde; özellikle:
  1. Yerel kuyruk yöneticisinde, iletiyi geçici olarak saklamak için kullanılan yerel iletim kuyruğu.
  2. Ara iletim kuyrukları (varsa), yerel ve hedef kuyruk yöneticileri arasındaki rotadaki kuyruk yöneticilerindeki iletiyi saklamak için kullanılır.
  3. Hedef kuyruk yöneticisinde hedef kuyruk

Bu nedenle, en uzun ileti bu kuyruklar ve kuyruk yöneticilerinin en kısıtlayıcı tarafından yönetilir.

İleti bir iletim kuyruğunda olduğunda, ek bilgiler ileti verileriyle birlikte bulunur ve bu işlem, yürütülebilecek uygulama verileri miktarını azaltır. Bu durumda, BufferLength ile ilgili sınır belirlenirken, iletim kuyruklarının MaxMsgLength değerlerinden MQ\_MSG\_HEADER\_LENGTH byte 'ı alçaltmak gerekir.

**Not:** İleti konduğunda, yalnızca 1. koşula uymamanın durumu zamanuyumlu olarak tanımlanabilir (neden kodu MQRC\_MSG\_TOO\_BIG\_FOR\_Q ya da MQRC\_MSG\_TOO\_BIG\_FOR\_Q\_MGR). 2 ya da 3 numaralı koşullar karşılanmazsa, ileti bir ara kuyruk yöneticisinde ya da hedef kuyruk yöneticisinde bir ölü-mektup (teslim edilemeyen ileti) kuyruğuna yeniden yönlendirilir. Bu gerçekleşirse, gönderici tarafından istendiye bir rapor iletisi oluşturulur.

## Arabellek

Tip: MQBYTEExBufferUzunluđu-giriş

Bu, gönderilecek uygulama verilerini içeren bir arabelleğidir. Arabellek, iletteki verilerin niteine uygun bir sınırla hizalanmalıdır. 4 baytlık hizalama çođu ileti için uygundur ( IBM MQ üstbilgi yapılarını içeren iletiler de içinde olmak üzere), ancak bazı iletiler daha sıkı hizalamada gerektirebilir. Örneğin, 64 bitlik bir ikili tamsayı içeren bir ileti 8 byte 'lık hizalama gerektirebilir.

Buffer , karakter ya da sayısal veri içeriyorsa, **MsgDesc** parametresindeki CodedCharSetId ve Encoding alanlarını, verilere uygun değerlere ayarlayın; bu işlem, iletinin alıcısını, verileri (gerekliyse), alıcı tarafından kullanılan karakter takımı ve kodlamaya dönüştürmesini sağlar.

**Not:** MQPUT çağrısındaki diğer tüm parametreler, yerel kuyruk yöneticisinin karakter kümesinde ve kodlamasında ( **CodedCharSetId** kuyruk yöneticisi özniteliđi ve MQENC\_NATIVE ile verilir) olmalıdır.

C programlama dilinde, parametre bir işaretçi olarak bildirilir; parametrenin herhangi bir tipinin adresi, parametre olarak belirtilebilir.

**BufferLength** parametresi sıfır ise, Buffer bu durumda gönderilmez; bu durumda, C ya da System/390 çevirici programlarında yazılan programlarla geçirilen parametre adresi boş değerli olabilir.

## CompCode

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Tamamlanma kodu; bu kod aşağıdakilerden biridir:

### MQCC\_OK

İşlem başarıyla tamamlandı.

### MQCC\_UYARI

Uyarı (kısmi tamamlama).

### MQCC\_FAILED

Arama başarısız oldu.

## Neden

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Neden kodu niteleyici CompCode.

CompCode MQCC\_OK ise:

### MQRC\_NONE

(0, X'000 ') Raporlamak için bir neden yok.

CompCode MQCC\_UYARI ise:

### MQRC\_INCOMPLE\_GROUP

(2241, X'8C1') İleti grubu tamamlanmadı.

### MQRC\_INCOMPLE\_MSG

(2242, X'8C2') Mantıksal ileti tamamlanmadı.

### MQRC\_INTUTARLMENT\_PERSISTENCE

(2185, X'889 ') Tutarsız kalıcılık belirtimi.

### MQRC\_INCONSISTENT\_UOW

(2244, X'8C5') Tutarsız iş birliđi-iş belirtimi.

### MQRC\_MULTIPLE\_REASONS

(2136, X'858 ') Birden çok neden kodu döndürüldü.

### MQRC\_PRIORITY\_EXCEEDS\_MAXIMUM

(2049, X'801 ') İleti Önceliđi desteklenen üst sınır değerini aşiyor.

### MQRC\_UNKNOWN\_REPORT\_OPTION

(2104, X'838 ') İleti tanımlayıcısında rapor seçenek (ler) tanınmıyor.

CompCode MQCC\_FAILED ise:

**MQR\_ADAPTER\_NOT\_AVAM**

(2204, X'89C') Baędařtırıcı kullanılmıyor.

**MQR\_ADAPTER\_SERV\_LOAD\_ERROR**

(2130, X'852 ') Baędařtırıcı hizmeti modülü yüklenemiyor.

**MQR\_ALIAS\_TARGTYPE\_CHANGED**

(2480, X'09B0') Abonelik hedef tipi, kuyruktan konu nesnesine çevrim yaptı.

**MQR\_API\_EXIT\_ERROR**

(2374, X' 946 ') API çıkışı başarısız oldu.

**MQR\_API\_EXIT\_LOAD\_ERROR**

(2183, X'887 ') API çıkışı yüklenemiyor.

**MQR\_ASID\_MISSMACH**

(2157, X'86D') Birincil ve ana sunucu ASID deęerleri farklı.

**MQR\_BACKED\_OUT**

(2003, X'7D3') İş birimi yedeklendi.

**MQR\_BUFFER\_ERROR**

(2004, X'7D4') Arabellek parametresi geçerli deęil.

**MQR\_BUFFER\_LENGTH\_ERROR**

(2005, X'7D5') Arabellek uzunluęu parametresi geçerli deęil.

**MQR\_CALL\_IN\_PROGRESS**

(2219, X'8AB') Önceki arama tamamlanmadan önce girilen MQI çağrısı.

**MQR\_CALL\_INTERRUPT**

(2549, X'9F5') MQPUT ya da MQCMIT kesintiye uğradı ve yeniden baęlanma işlemi kesin bir sonucu yeniden oluşturamıyor.

**MQR\_CF\_NOT\_VAR**

(2345, X' 929 ') Coupling tesisi kullanılmıyor.

**MQR\_CF\_STRUC\_FAILED**

(2373, X' 945 ') Coupling-tesis yapısı başarısız oldu.

**MQR\_CF\_STRUC\_IN\_USE**

(2346, X'92A') Coupling-tesis yapısı kullanımda.

**MQR\_CFGR\_ERROR**

(2416, X' 970 ') İleti verilerinde PCF grup deęiřtirgesi yapısı MQCFGR geçerli deęil.

**MQR\_CFH\_ERROR**

(2235, X'8BB') PCF üstbilgi yapısı geçerli deęil.

**MQR\_CFIF\_ERROR**

(2414, X'96E') İleti verilerinde PCF tamsayı süzgeç parametresi yapısı geçerli deęil.

**MQR\_CFIL\_ERROR**

(2236, X'8BC') PCF tamsayı listesi parametre yapısı ya da PCIF\*64 tamsayı listesi parametre yapısı geçerli deęil.

**MQR\_CFIN\_ERROR**

(2237, X'8BD') PCF tamsayı deęiřtirgesi yapısı ya da PCIF\*64 tamsayı deęiřtirgesi yapısı geçerli deęil.

**MQR\_CFSF\_ERROR**

(2415, X'96F') İleti verilerinde PCF dizgi süzgeci deęiřtirgesi yapısı geçerli deęil.

**MQR\_CFSL\_ERROR**

(2238, X'8BE') PCF dizgi listesi deęiřtirgesi yapısı geçerli deęil.

**MQR\_CFST\_ERROR**

(2239, X'8BF') PCF dizgi deęiřtirgesi yapısı geçerli deęil.

**MQR\_CICS\_WAIT\_FAILED**

(2140, X'85C') Bekleme isteęi CICStarafından reddedildi.



**MQRC\_CLUSTER\_EXIT\_ERROR**  
(2266, X'8DA') Küme iş yükü çıkışı başarısız oldu.

**MQRC\_CLUSTER\_RESOLUTION\_ERROR**  
(2189, X'88D') Küme adının çözülmesi başarısız oldu.

**MQRC\_CLUSTER\_KAYNAĞI\_HATASI**  
(2269, X'8DD') Küme kaynağı hatası.

**MQRC\_COD\_NOT\_VALID\_FOR\_XCF\_Q**  
(2106, X'83A') COD rapor seçeneği, XCF kuyruğu için geçerli değil.

**MQRC\_CONNECTION\_BROKEN**  
(2009, X'7D9') Kuyruk yöneticisine yönelik bağlantı kaybedildi.

**MQRC\_CONNECTION\_NOT\_YETKILI**  
(2217, X'8A9') Bağlantı için yetkili değil.

**MQRC\_CONNECTION\_QUIESCING**  
(2202, X'89A') Bağlantı susturulmuş durumda.

**MQRC\_CONNECTION\_DURATION**  
(2203, X'89B') Bağlantı sona erdiriliyor.

**MQRC\_İÇERİĞİ**  
2554 (X'09FA') İleti içeriği, iletinin genişletilmiş bir ileti seçiciyle bir aboneye teslim edilmesi gerekip gerekmediğini belirlemek için ayrıştırılmadı.

**MQRC\_CONTEXT\_HANDLE\_ERROR**  
(2097, X'831 ') Bu işleme gönderme yapılan kuyruk tanıtıcısı saklama bağlamı değil.

**MQRC\_CONTEXT\_NOT\_AVAILABLE**  
(2098, X'832 ') Kuyruk tanıtıcısı gönderme yapılan bir bağlam kullanılamaz.

**MQRC\_DATA\_LENGTH\_ERROR**  
(2010, X'7DA') Veri uzunluğu parametresi geçerli değil.

**MQRC\_DH\_HATASI**  
(2135, X'857 ') Dağıtım üstbilgisi yapısı geçerli değil.

**MQRC\_DLH\_ERROR**  
(2141, X'85D') Ölü harf üstbilgisi yapısı geçerli değil.

**MQRC\_EPH\_ERROR**  
(2420, X' 974 ') Embedded PCF yapısı geçerli değil.

**MQRC\_EXPIRY\_ERROR**  
(2013, X'7DD') Süre bitim zamanı geçerli değil.

**MQRC\_FEEDBACK\_ERROR**  
(2014, X'7DE') Geribildirim kodu geçerli değil.

**MQRC\_GLOBAL\_UOW\_CONFLICTIONS**  
(2351, X'92F') Genel iş çakışmaları birimleri.

**MQRC\_GROUP\_ID\_ERROR**  
(2258, X'8D2') Grup tanıtıcısı geçerli değil.

**MQRC\_HANDLE\_IN\_USE\_FOR\_UOW**  
(2353, X' 931 ') Genel iş birimi için kullanılan tanıtıcı.

**MQRC\_HCONN\_ERROR**  
(2018, X'7E2') Bağlantı tanıtıcısı geçerli değil.

**MQRC\_HEADER\_ERROR**  
(2142, X'85E') MQ üstbilgi yapısı geçerli değil.

**MQRC\_HOBJ\_ERROR**  
(2019, X'7E3') Nesne tanıtıcısı geçerli değil.

**MQRC\_IIH\_ERROR**  
(2148, X'864 ') IMS bilgi üstbilgisi yapısı geçerli değil.

**MQRC\_INCOMPLE\_GROUP**  
(2241, X'8C1') İleti grubu tamamlanmadı.

**MQRC\_INCOMPLE\_MSG**  
(2242, X'8C2') Mantıksal ileti tamamlanmadı.

**MQRC\_INTUTARLMENT\_PERSISTENCE**  
(2185, X'889 ') Tutarsız kalıcılık belirtimi.

**MQRC\_INCONSISTENT\_UOW**  
(2244, X'8C5') Tutarsız iş birliği-iş belirtimi.

**MQRC\_LOCAL\_UOW\_CONFLICTID**  
(2352, X' 930 ') Genel iş birimi yerel iş birimi ile çakışıyor.

**MQRC\_MD\_ERROR**  
(2026, X'7EA') İleti tanımlayıcısı geçerli değil.

**MQRC\_MDE\_HATASI**  
(2248, X'8C8') İleti tanımlayıcısı uzantısı geçersiz.

**MQRC\_MISSING\_REPLY\_TO\_Q**  
(2027, X'7EB') Eksik yanıt kuyruğu ya da MQPMO\_SUPPRESS\_REPLYTO kullanıldı

**MQRC\_MISSING\_WIH**  
(2332, X'91C') İleti verileri MQWIH ile başlamaz.

**MQRC\_MSG\_FLAGS\_ERROR**  
(2249, X'8C9') İleti işaretleri geçerli değil.

**MQRC\_MSG\_SEQ\_NUMBER\_ERROR**  
(2250, X'8CA') İleti sıra numarası geçerli değil.

**MQRC\_MSG\_TOO\_BIG\_FOR\_Q**  
(2030, X'7EE') İleti uzunluğu, kuyruk için üst sınırdan büyük.

**MQRC\_MSG\_TOO\_BIG\_FOR\_Q\_MGR**  
(2031, X'7EF') İleti uzunluğu kuyruk yöneticisi için üst sınırdan büyük.

**MQRC\_MSG\_TYPE\_ERROR**  
(2029, X'7ED') İleti tanımlayıcısında ileti tipi geçerli değil.

**MQRC\_MULTIPLE\_REASONS**  
(2136, X'858 ') Birden çok neden kodu döndürüldü.

**MQRC\_NO\_DESTINATIONS\_AVALABILIR**  
(2270, X'8DE') Kullanılabilir hedef kuyruğu yok.

**MQRC\_NOT\_OPEN\_FOR\_OUTPUT**  
(2039, X'7F7') Kuyruk çıkış için açık değil.

**MQRC\_NOT\_OPEN\_FOR\_PASS\_ALL**  
(2093, X'82D') Kuyruğu, tüm bağlamın geçirilmek üzere açık değil.

**MQRC\_NOT\_OPEN\_FOR\_PASS\_IDENT**  
(2094, X'82E') Kuyruk, geçiş kimliği bağlamı için açık değil.

**MQRC\_NOT\_OPEN\_FOR\_SET\_ALL**  
(2095, X'82F') Tüm bağlamın ayarlanması için kuyruk açık değil.

**MQRC\_NOT\_OPEN\_FOR\_SET\_IDENT**  
(2096, X'830 ') Kuyruk tanıtıcısı bağlamı için kuyruk açık değil.

**MQRC\_OBJECT\_CHANGED**  
(2041, X'7F9') Nesne tanımlaması açıldığından beri değiştirildi.

**MQRC\_OBJECT\_ZARAR**  
(2101, X'835 ') Nesne zarar gördü.

**MQRC\_OFFSET\_ERROR**  
(2251, X'8CB') İleti bölümü görelî konumu geçerli değil.

**MQRC\_OPEN\_FAILED**  
(2137, X'859 ') Nesne başarıyla açılmadı.

**MQRC\_OPTIONS\_ERROR**

(2046, X'7FE') Seçenekler geçerli değil ya da tutarlı değil.

**MQRC\_ORIGINAL\_LENGTH\_ERROR**

(2252, X'8CC') Özgün uzunluk geçerli değil.

**MQRC\_PAGEES\_HATASI**

(2193, X'891 ') Sayfa kümesi veri kümesine erişilirken hata oluştu.

**MQRC\_PAGEET\_FULL**

(2192, X'890 ') Dış depolama ortamı dolu.

**MQRC\_PCF\_ERROR**

(2149, X'865 ') PCF yapıları geçerli değil.

**MQRC\_PERSISTENCE\_ERROR**

(2047, X'7FF') Persistence geçerli değil.

**MQRC\_PERSISTENT\_NOT\_ALLOWED**

(2048, X'800 ') Kuyruk kalıcı iletileri desteklemiyor.

**MQRC\_PMO\_ERROR**

(2173, X'87D') Koyma-ileti seçenekleri yapısı geçerli değil.

**MQRC\_PMO\_RECORD\_FLAGS\_ERROR**

(2158, X'86E') İleti kaydı işaretlerinin konması geçerli değil.

**MQRC\_PRIORITY\_ERROR**

(2050, X'802 ') İleti önceliği geçerli değil.

**MQRC\_PUBLICATION\_FAILURE**

(2502, X'9C6') Yayın, abonelerin hiçbirine teslim edilemedi.

**MQRC\_PUT\_INHIBITED**

(2051, X'803 ') Bu kuyruğun çözümlendiği kuyruk için, kuyruk için ya da konu için arama engellenmiş olarak çağrılar girin.

**MQRC\_PUT\_MSG\_RECORDS\_ERROR**

(2159, X'86F') İleti içeren ileti kayıtları geçerli değil.

**MQRC\_PUT\_NOT\_ALICILI**

(2479, X'09AF') Yayın saklanamadı

**MQRC\_Q\_DELETED**

(2052, X'804 ') Kuyruk silinmiştir.

**MQRC\_Q\_FULL**

(2053, X'805 ') Kuyruk, ileti sayısı üst sınırını zaten içeriyor.

**MQRC\_Q\_MGR\_NAME\_ERROR**

(2058, X'80A') Kuyruk yöneticisi adı geçerli değil ya da bilinmiyor.

**MQRC\_Q\_MGR\_NOT\_VAR**

(2059, X'80B') Kuyruk yöneticisi bağlantı için kullanılabilir değil.

**MQRC\_Q\_MGR QUIESCING**

(2161, X'871 ') Kuyruk yöneticisi susturuyor.

**MQRC\_Q\_MGR\_STOPPING**

(2162, X'872 ') Kuyruk yöneticisi sona erdiriliyor.

**MQRC\_Q\_SPACE\_NOT\_VAR**

(2056, X'808 ') Kuyruğun diskte kullanılabilir alan yok.

**MQRC\_RECONNECT\_FAILED**

(2548, X'9F4') Yeniden bağlandıktan sonra, yeniden bağlanabilir bağlantı için tutamaçlar geri döndürülürken bir hata oluştu.

**MQRC\_RECS\_PRESENT\_ERROR**

(2154, X'86A') Kayıt sayısı geçerli değil.

**MQRC\_REPORT\_OPTIONS\_ERROR**

(2061, X'80D') İleti tanımlayıcısındaki rapor seçenekleri geçerli değil.

**MQRC\_RESOURCE\_SORUNU**

(2102, X'836 ') Yetersiz sistem kaynakları var.

**MQRC\_RESPONSE\_RECORDS\_ERROR**

(2156, X'86C') Yanıt kayıtları geçerli değil.

**MQRC\_RFH\_ERROR**

(2334, X'91E') MQRFH ya da MQRFH2 yapısı geçerli değil.

**MQRC\_RMH\_ERROR**

(2220, X'8AC') Başvuru iletisi üstbilgi yapısı geçerli değil.

**MQRC\_SEGMENT\_LENGTH\_ZERO**

(2253, X'8CD') İleti kesiminde veri uzunluğu sıfır.

**MQRC\_SEGMENTS\_NOT\_SUPPORTED**

(2365, X'93D') Bölümleri desteklenmiyor.

**MQRC\_SELECTION\_NOT\_AVAM**

2551 (X'09F7') Yayın için olası bir abone var, ancak kuyruk yöneticisi yayının aboneye gönderilip gönderilmeyeceğini denetleyemiyor.

**MQRC\_STOPPED\_BY\_CLUSTER\_EXIT**

(2188, X'88C') Küme iş yükü çıkışı tarafından çağrılan çağrı reddedildi.

**MQRC\_STORAGE\_CLASS\_ERROR**

(2105, X'839 ') Depolama sınıfı hatası.

**MQRC\_STORAGE\_MEDIUM\_FULL**

(2192, X'890 ') Dış depolama ortamı dolu.

**MQRC\_STORAGE\_NOT\_AVAM**

(2071, X'817 ') Kullanılabilir bellek yetersiz.

**MQRC\_SUPPRESSED\_BY\_EXIT**

(2109, X'83D') Çıkış programı tarafından çağrı engellendi.

**MQRC\_SYNCPOINT\_LIMIT\_UVARD**

(2024, X'7E8') Yürürlükteki iş birimi içinde başka ileti işlenemez.

**MQRC\_SYNCPOINT\_NOT\_AVAM**

(2072, X'818 ') Syncpoint desteği yok.

**MQRC\_TM\_HATASI**

(2265, X'8D9') Tetikleyici ileti yapısı geçerli değil.

**MQRC\_TMC\_ERROR**

(2191, X'88F') Karakter tetikleyicisi ileti yapısı geçerli değil.

**MQRC\_UNEXPECTED\_ERROR**

(2195, X'893 ') Beklenmeyen bir hata oluştu.

**MQRC\_UOWENLISTMENT\_ERROR**

(2354, X' 932 ') Genel iş birimi içindeki bilgiler başarısız oldu.

**MQRC\_UOW\_MIX\_NOT\_SUPPORTANT**

(2355, X' 933 ') İşlerin birim içi çağrıları karışımı desteklenmiyor.

**MQRC\_UOW\_NOT\_VAR**

(2255, X'8CF') Kuyruk yöneticisinin kullanması için kullanılacak iş birimi.

**MQRC\_WIH\_ERROR**

(2333, X'91D') MQWIH yapısı geçerli değil.

**MQRC\_WRONG\_MD\_VERSION**

(2257, X'8D1') MQMD' nin yanlış sürümü sağlandı.

**MQRC\_XQH\_ERROR**

(2260, X'8D4') İletim kuyruğu üstbilgi yapısı geçerli değil.

Bu kodlarla ilgili ayrıntılı bilgi için [İletiler ve neden kodları](#) başlıklı konuya bakın.

## Konu Kullanım notları

1. Konuların kullanımı için aşağıdaki notlar geçerlidir:

a. Bir konuya ilişkin iletileri yayınlamak için MQPUT kullanılırken, bir ya da daha çok abonenin abone kuyrukları (örneğin, dolu olduğu gibi) ile ilgili bir sorun nedeniyle yayın verilemez, MQPUT çağrısına döndürülen neden kodu ve teslim davranışı KONU üzerindeki PMSGDLV ya da NPMSGDLV özniteliklerinin ayarına bağlıdır. Bir yayının MQRO\_DEAD\_LETTER\_Q belirtildiğinde ya da MQRO\_DISCARD\_MSG belirtildiğinde iletinin atılması, iletinin başarılı bir şekilde teslim edilmesi olarak dikkate alındığında, bir yayının teslimi için not teslim edilir. Yayınların hiçbiri teslim edilmezse, MQPUT, MQRC\_PUBLICATION\_FAILURE ile geri döner. Bu durum aşağıdaki durumlarda oluşabilir:

- Bir ileti, PMSGDLV ya da NPMSGDLV (iletinin kalıcısına bağlı olarak) ALL ve herhangi bir aboneliğe (dayanıklı ya da değil) ilişkin olarak, yayını alamayacak bir kuyruğa sahip bir KONU için yayınlanır.
- Bir ileti, PMSGDLV ya da NPMSGDLV (iletinin kalıcısına bağlı olarak) değeri ALLUR ile bir TOPIC 'e yayınlanır ve kalıcı bir abonelik, yayını alamayacak bir kuyruğa sahiptir.

Yayınlara bazı abonelere aşağıdaki durumlarda teslim edilememesine rağmen, MQPUT MQRC\_NONE ile geri dönebilir:

- Bir ileti, PMSGDLV ya da NPMSGDLV (iletinin kalıcısına bağlı olarak) ALLAVAIL değerine ayarlanmış ve herhangi bir abonelik, dayanıklı ya da değil, yayını alamayacak bir kuyruğa sahip bir KONU için yayınlanır.
- Bir ileti, PMSGDLV ya da NPMSGDLV (iletinin kalıcısına bağlı olarak) değeri ALLUR ile bir TOPIC 'e yayınlanır ve kalıcı olmayan bir aboneliğe, yayını alamayan bir kuyruk vardır.

Yayın iletileri doğru abone kuyruğuna teslim edilemediğinde, ölü-harf kuyruğunun kullanılıp kullanılmayacağını belirlemek için USEDLO konu özneliğini kullanabilirsiniz. USEDLO kullanımına ilişkin ek bilgi için [DEFINE TOPIC](#) başlıklı konuya bakın.

b. Bu konuya abone olmayan bir abone yoksa, yayınlanan ileti herhangi bir kuyruğa gönderilmez ve atılır. İletinin kalıcı mı, yoksa kalıcı olmayan mı olduğu, yoksa sınırsız süre bitimi mi olduğu ya da bir süre bitimi mi olduğu önemli değildir; abone yoksa yine de atılır. Bu durumda, ileti alıkonacaksa, bu durumda, herhangi bir abonenin kuyruklarına gönderilmemiş olsa da, yeni aboneliklere ya da MQSUBRQ kullanılarak alıkonan yayınlar için tutulan yayınlara teslim edilecek herhangi bir aboneye teslim edilmeleri için bu iletinin saklanmasına ilişkin bir sorun yoktur.

## MQPUT ve MQPUT1

İletileri bir kuyruğa koymak için hem MQPUT hem de MQPUT1 çağrılarını kullanabilirsiniz; bu çağrılar, koşullara bağlı olarak değişir.

- Aynı kuyruğa birden çok ileti yerleştirmek için MQPUT çağrısını kullanın.

Önce MQOO\_OUTPUT seçeneğini belirten bir MQOPEN çağrısı yayınlandı, ardından kuyruğa ileti eklemek için bir ya da daha çok MQPUT isteği sunuldu; son olarak, kuyruk bir MQCLOSE çağrısıyla kapatıldı. Bu, MQPUT1 çağrısının yinelenen kullanımından daha iyi başarımlar sağlar.

- Bir kuyruğa yalnızca bir ileti koymak için MQPUT1 çağrısını kullanın.

Bu çağrı, MQOPER, MQPUT ve MQCLOSE çağrılarını tek bir çağrıya sarkıyor ve verilmesi gereken çağrıların sayısını en aza indiriyor.

## Hedef Kuyruklar

Hedef kuyrukların kullanımı için aşağıdaki notlar geçerlidir:

1. Bir uygulama, ileti grupları kullanılmadan aynı kuyruğa ileti dizisi koyarsa, Ayrıntılı bilgi yerine getirilirse, bu iletilerin sırası korunur. Bazı koşullar hem yerel, hem de uzak hedef kuyruklar için geçerlidir; diğer koşullar yalnızca uzak hedef kuyruklar için geçerlidir.

### Yerel ve uzak hedef kuyruklar için geçerli koşullar

- MQPUT çağrılarının tümü aynı iş birimi içinde ya da hiçbiri iş birimi içinde değil.

İletiler tek bir iş birimi içinde belirli bir kuyruğa konduğunda, diğer uygulamalardan gelen iletilerin kuyruklardaki ileti dizisiyle birlikte spermlenmiş olabileceğini unutmayın.

- Tüm MQPUT çağrıları, aynı nesne tanıtıcısı *Hobj* kullanılarak yapılır.

Bazı ortamlarda, çağrılar aynı uygulamadan yapılmışsa, farklı nesne tanıtıcıları kullanıldığında ileti sırası da korunur. *Aynı uygulama* ' in anlamı, ortam tarafından belirlenir:

- **z/OS** z/OS üzerinde uygulama:
  - CICS için, CICS görevi
  - IMS için görev
  - z/OS toplu işi için, görev
- **IBM i** IBM üzerinde, uygulama işidir.
- **ALW** AIX, Linux, and Windows üzerinde uygulama iş parçacığıdır.

- İletilerin hepsi aynı önceliğe sahip.
- The messages are not put to a cluster queue with MQOO\_BIND\_NOT\_FIXED specified (or with MQOO\_BIND\_AS\_Q\_DEF in effect when the DefBind queue attribute has the value MQBND\_BIND\_NOT\_FIXED).

### Uzak hedef kuyrukları için geçerli olan ek koşullar

- Kuyruk yöneticisinden hedef kuyruk yöneticisine yalnızca bir yol var.

Sıradaki bazı iletiler farklı bir yola girerse (örneğin, yeniden yapılandırma, trafik dengeleme ya da ileti büyüklüğüne dayalı yol seçimi nedeniyle), hedef kuyruk yöneticisinde iletilerin sırası garanti edilemez.

- İletiler, gönderme, ara düzey ya da hedef kuyruk yöneticisinde geçici olarak ölü harf kuyruklarına yerleştirilmedi.

İletilerden biri ya da daha fazlası geçici olarak bir ölü-mektup kuyruğuna konursa (örneğin, bir iletim kuyruğu ya da hedef kuyruğu geçici olarak dolu olduğu için), iletiler hedef kuyruğa sıra dışında gelebilir.

- İletiler, kalıcı ya da kalıcı olmayan tüm iletiler.

If a channel on the route between the sending and destination queue managers has its **NonPersistentMsgSpeed** attribute set to MQNPM\_FAST, nonpersistent messages can jump ahead of persistent messages, resulting in the order of persistent messages relative to nonpersistent messages not being preserved. Ancak, birbiriyle görece kalıcı iletilerin sırası ve birbiriyle görece kalıcı olmayan iletilerin sırası korunur.

Bu koşullar yerine getirilmezse, ileti sırasını korumak için ileti gruplarını kullanabilirsiniz; ancak, bu işlem hem gönderme hem de alma uygulamalarını, ileti gruplama desteğini kullanmak için gerektirir. İleti gruplarına ilişkin ek bilgi için aşağıdaki başlıklara bakın:

- [MQMD- MsgFlags alanı](#)
- [MQPMO\\_LOGICAL\\_ORDER](#)
- [MQGMO\\_LOGICAL\\_ORDER](#)

### dağıtım listeleri

Aşağıdaki notlar dağıtım listelerinin kullanımı için geçerlidir.

Dağıtım listeleri aşağıdaki ortamlarda desteklenir:

- **AIX** AIX
- **IBM i** IBM i

-  Linux
-  Windows

ve bu sistemlere bağı IBM MQ MQI clients için.

1. Bir version-1 ya da bir version-2 MQPMO kullanarak dağıtım listesine ileti koyabilirsiniz. Bir version-1 MQPMO (ya da RecsPresent ile sıfıra eşit bir version-2 MQPMO kullanıyorsanız), uygulama herhangi bir ileti kaydı ya da yanıt kaydı sağlayamaz. İleti, dağıtım listesindeki bazı kuyruklara başarılı bir şekilde gönderildiyse ve diğerleri tarafından başarılı bir şekilde gönderildiyse, hatalarla karşılaşan kuyrukları tanımlayamazsınız.

Uygulama, ileti kayıtları ya da yanıt kayıtları sağlıyorsa, Version alanını MQPMO\_VERSION\_2olarak ayarlayın.

You can also use a version-2 MQPMO to send messages to a single queue that is not in a distribution list, by ensuring that RecsPresent is zero.

2. Tamamlanma kodu ve neden kodu değıştirgeleri aşağıdaki gibi ayarlanır:

- Dağıtım listesindeki kuyruklara başarılı olan ya da aynı şekilde başarısız olursa, tamamlanma kodu ve neden kodu değıştirgeleri ortak sonucu tanımlamaya ayarlanır. Bu durumda, MQRR yanıt kayıtları (uygulama tarafından sağlandıysa) belirlenmez.

Örneğin, her put başarılı olursa, tamamlanma kodu ve neden kodu MQCC\_OK ve MQRC\_NONE; olarak ayarlanır; her put başarısız olursa, tüm kuyruklar engellenirse, parametreler MQCC\_FAILED ve MQRC\_PUT\_INHIMATED olarak ayarlanır.

- Dağıtım listesindeki kuyruklara koyma işlemi başarılı olamazsa ya da aynı şekilde başarısız olursa:

- En az bir put başarılı olursa ve tümü başarısız olursa, tamamlanma kodu parametresi MQCC\_UYARI değerine ayarlanır ve MQCC\_FAILED değerine ayarlanır.
- Neden kodu parametresi MQRC\_MULTIPLE\_REASONS olarak ayarlandı.
- Yanıt kayıtları (uygulama tarafından sağlandıysa), dağıtım listesindeki kuyruklara ilişkin tek tek tamamlama kodlarına ve neden kodlarına ayarlanmışsa.

Hedef için açık olan hedef başarısız olduğu için bir hedefe koyma başarısız olursa, yanıt kaydındaki alanlar MQCC\_FAILED ve MQRC\_OPEN\_FAILED; bu hedef InvalidDestCount' ta yer alır.

3. Dağıtım listesindeki bir hedef yerel bir kuyruğa çözümlerse, ileti o kuyruğa normal biçimde yerleştirilir (yani, dağıtım listesi iletisi olarak değil). Birden çok hedef aynı yerel kuyruğa giderilirse, kuyruğa her hedef için bir ileti yerleştirilir.

Dağıtım listesindeki bir hedef uzak bir kuyruğa giderilirse, uygun iletim kuyruğuna bir ileti yerleştirilir. Birden çok hedef, aynı iletim kuyruğuna çözüldüğü yerlerde, bu hedefleri içeren tek bir dağıtım listesi iletisi, uygulama tarafından sağlanan varış noktaları listesinde bitişik olmasa da iletim kuyruğuna yerleştirilebilir. Ancak, bu işlem yalnızca iletim kuyruğu dağıtım listesi iletilerini destekliyorsa yapılabilir ([DistLists](#) başlıklı konuya bakın).

İletim kuyruğu dağıtım listelerini desteklemiyorsa, iletinin olağan biçimde bir kopyası, iletim kuyruğunu kullanan her hedef için iletim kuyruğuna yerleştirilir.

Uygulama iletisi verilerini içeren bir dağıtım listesi iletim kuyruğu için çok büyükse, dağıtım listesi iletisi daha az sayıda hedef içeren daha küçük dağıtım listesi iletilerine bölünmesi gerekir. Uygulama iletisi verileri yalnızca kuyruğa sığarsa, dağıtım listesi iletileri hiç kullanılamaz ve kuyruk yöneticisi, bu iletim kuyruğunu kullanan her hedef için iletinin olağan biçimde bir kopyasını oluşturur.

Farklı hedeflerde farklı ileti önceliği ya da ileti kalıcılığı varsa (uygulama MQPRI\_PRIORITY\_AS\_Q\_DEF ya da MQPER\_PERSISTENCE\_AS\_Q\_DEF 'i belirttiğinde bu durum oluşabilir), iletiler aynı dağıtım listesi iletisinde tutulmaz. Bunun yerine, kuyruk yöneticisi, farklı öncelik ve kalıcılık değerlerini barındırmak için gereken sayıda dağıtım listesi iletisi oluşturur.

4. Dağıtım listesine bir kont aşağıdakiyle sonuçlanabilir:

- Tek bir dağıtım listesi iletisi ya da

- Daha küçük dağıtım listesi iletileri ya da
- Dağıtım listesi iletilerinin ve normal iletilerin bir karışımı ya da
- Yalnızca normal iletiler.

Yukarıdaki durum aşağıdakilerden hangisinin gerçekleştirilmediğine bağlıdır:

- Listedeki hedefler yerel, uzak ya da bir karışımdır.
- Hedefler aynı ileti önceliğine ve ileti kalıcılarına sahiptir.
- İletim kuyrukları, dağıtım listesi iletilerini tutabilirler.
- İletim kuyruklarının üst sınır ileti uzunluğu, dağıtım listesi formundaki iletiyi sığdırabilmek için yeterli büyüklükte.

Ancak, yukarıdaki durumda ne olursa olsun, sonuçta ortaya çıkan her *fiziksel* ileti (yani, her normal ileti ya da dağıtım listesi iletilerinin ortaya konması), aşağıdaki durumlarda yalnızca *bir* ileti olarak sayılır:

- Uygulamanın bir iş birimindeki izin verilen ileti sayısı üst sınırını aştığını kontrol etmek (**MaxUncommittedMsgs** kuyruk yöneticisi özneliğine bakın).
- Tetikleme koşullarının yerine getirilip karşılanmadığı denetleniyor.
- Kuyruk derinliklerinin artırılması ve kuyrukların kuyruk derinliği üst sınırının aşılıp aşılmayacağını kontrol etmek.

5. Bir tanıtıcıyı geçersiz hale getirmesine neden olacak kuyruk tanımlamalarında yapılan herhangi bir değişiklik, kuyrukların tek tek açılmasına neden oldu (örneğin, çözüm yolundaki bir değişiklik), dağıtım listesi tutamaçlarının geçersiz olmasına neden olmaz. Ancak, sonraki bir MQPUT çağrısında dağıtım listesi tanıtıcısı kullanıldığında, bu kuyruğun başarısızlığa neden olması da sonuçlanabilir.

## Üst Bilgiler

Bir ileti, uygulama iletileri verilerinin başına bir ya da daha çok IBM MQ üstbilgi yapısıyla konursa, kuyruk yöneticisi, geçerli olduğunu doğrulamak için üstbilgi yapılarında bazı denetimleri gerçekleştirir. Kuyruk yöneticisi bir hata saptarsa, arama uygun bir neden kodunda başarısız olur. Gerçekleştirilen denetimler, şu anda bulunan belirli yapılara göre değişiklik gösterir:

- Çekler yalnızca, MQPUT ya da MQPUT1 çağrısında bir version-2 ya da üstü bir MQMD kullanılırsa gerçekleştirilir. Checks are not performed if a version-1 MQMD is used, even if an MQMDE is present at the start of the message data.
- Yerel kuyruk yöneticisi tarafından desteklenmeyen yapılar ve iletteki ilk MQDLH ' nin ardından yapılarda geçerlilik denetimi yapılmaz.
- MQDH ve MQMDE yapıları, kuyruk yöneticisi tarafından tamamen doğrulanır.
- Diğer yapılar, kuyruk yöneticisi tarafından kısmen doğrulanır (tüm alanlar işaretlenmez).

Kuyruk yöneticisi tarafından gerçekleştirilen genel denetimler aşağıdakileri içerir:

- **StrucId** alanının geçerli olması gerekir.
- **Version** alanının geçerli olması gerekir.
- **StrucLength** alanı, yapıyı içerecek kadar büyük bir değer belirtmeli ve yapının bir parçası olan değişken uzunluklu verileri içermelidir.
- **CodedCharSetId** alanı sıfır ya da geçerli olmayan bir eksi değer (MQCCSI\_VARSAY, MQCCSI\_EMBEDDED, MQCCSI\_Q\_MGR ve MQCCSI\_UNDEFINED, çoğu IBM MQ üstbilgi yapısında geçerli değildir) olmalıdır.
- Çağrıya ilişkin **BufferLength** parametresi, yapıyı içerecek kadar büyük bir değer belirtmelidir (yapı, iletinin sonuna kadar uzatılamamalıdır).

Yapılara ilişkin genel denetlemelere ek olarak, aşağıdaki koşullar yerine getirilmelidir:

- Bir PCF iletisinde yapıların uzunluklarının toplamı, MQPUT ya da MQPUT1 çağrısında **BufferLength** parametresiyle belirtilen uzunluğa eşit olmalıdır. Bir PCF iletisi, MQFMT\_ADMIN, MQFMT\_EVENT ya da MQFMT\_PCF biçim adına sahip bir iletidir.



- Kesilmiş yapıların izin verildiği durumlar dışında, bir IBM MQ yapısının kesilmemesi gerekir:
  - Rapor iletileri olan iletiler.
  - PCF iletileri.
  - MQDLH yapısı içeren iletiler. (İlk MQDLH 'nin ardından yapılar kesilerek kesilebilir; MQDLH' dan önceki yapılar olamaz.)
- Bir IBM MQ yapısı, iki ya da daha fazla kesim üzerinden bölünmemelidir; yapı tamamen tek bir kesim içinde bulunmalıdır.

## Arabellek

Visual Basic programlama dili için, aşağıdaki noktalar geçerlidir:

- **Buffer** parametresinin boyutu, **BufferLength** parametresi tarafından belirtilen uzunluktan küçükse, çağrı neden kodu MQRC\_BUFFER\_LENGTH\_ERROR neden ile başarısız olur.
- **Buffer** parametresi, Stringtipinde olduğu için bildirilir. If the data to be placed on the queue is not of type String, use theMQPUT yerine MQPUTAny çağrısı.

MQPUTAny çağrısı, MQPUT çağrıyla aynı parametrelere sahiptir; ancak, **Buffer** değıştirgesi Anytipinde olduğu için, kuyrukta herhangi bir veri tipinin yerleřtirilmesine olanak tanımaktadır. Ancak bu, Buffer 'in büyüklük olarak en az BufferLength byte olduğundan emin olmak için denetleyemediđi anlamına gelir.

## C çağırma

```
MQPUT (Hconn, Hobj, &MsgDesc, &PutMsgOpts, BufferLength, Buffer,
      &CompCode, &Reason);
```

Deđiřtirgeleri aşağıdaki gibi bildirin:

```
MQHCONN  Hconn;           /* Connection handle */
MQHOBJ   Hobj;           /* Object handle */
MQMD     MsgDesc;       /* Message descriptor */
MQPMO    PutMsgOpts;    /* Options that control the action of MQPUT */
MQLONG   BufferLength;   /* Length of the message in Buffer */
MQBYTE   Buffer[n];     /* Message data */
MQLONG   CompCode;     /* Completion code */
MQLONG   Reason;       /* Reason code qualifying CompCode */
```

## Cobol çağırma

```
CALL 'MQPUT' USING HCONN, HOBJ, MSGDESC, PUTMSGOPTS, BUFFERLENGTH,
                  BUFFER, COMPCODE, REASON.
```

Deđiřtirgeleri aşağıdaki gibi bildirin:

```
** Connection handle
01 HCONN          PIC S9(9) BINARY.
** Object handle
01 HOBJ          PIC S9(9) BINARY.
** Message descriptor
01 MSGDESC.
   COPY CMQMDV.
** Options that control the action of MQPUT
01 PUTMSGOPTS.
   COPY CMQPMOV.
** Length of the message in BUFFER
01 BUFFERLENGTH PIC S9(9) BINARY.
** Message data
01 BUFFER        PIC X(n).
** Completion code
01 COMPCODE      PIC S9(9) BINARY.
```

```
** Reason code qualifying COMPCODE
01 REASON          PIC S9(9) BINARY.
```

## PL/I çağırımı

```
call MQPUT (Hconn, Hobj, MsgDesc, PutMsgOpts, BufferLength, Buffer,
            CompCode, Reason);
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
dcl Hconn          fixed bin(31); /* Connection handle */
dcl Hobj           fixed bin(31); /* Object handle */
dcl MsgDesc        like MQMD;     /* Message descriptor */
dcl PutMsgOpts     like MQPMO;    /* Options that control the action of
MQPUT */
dcl BufferLength    fixed bin(31); /* Length of the message in Buffer */
dcl Buffer          char(n);       /* Message data */
dcl CompCode       fixed bin(31); /* Completion code */
dcl Reason         fixed bin(31); /* Reason code qualifying CompCode */
```

## High Level Assembler çağırısı

```
CALL MQPUT, (HCONN, HOBJ, MSGDESC, PUTMSGOPTS, BUFFERLENGTH, X
            BUFFER, COMPCODE, REASON)
```

Değiştirgeleri aşağıdaki gibi bildirin:

HCONN	DS	F	Connection handle
HOBJ	DS	F	Object handle
MSGDESC	CMQMDA	,	Message descriptor
PUTMSGOPTS	CMQPMOA	,	Options that control the action of MQPUT
BUFFERLENGTH	DS	F	Length of the message in BUFFER
BUFFER	DS	CL(n)	Message data
COMPCODE	DS	F	Completion code
REASON	DS	F	Reason code qualifying COMPCODE

## Visual Basic çağırısı

Windows

```
MQPUT Hconn, Hobj, MsgDesc, PutMsgOpts, BufferLength, Buffer, CompCode,
      Reason
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
Dim Hconn          As Long 'Connection handle'
Dim Hobj           As Long 'Object handle'
Dim MsgDesc        As MQMD 'Message descriptor'
Dim PutMsgOpts     As MQPMO 'Options that control the action of MQPUT'
Dim BufferLength    As Long 'Length of the message in Buffer'
Dim Buffer          As String 'Message data'
Dim CompCode       As Long 'Completion code'
Dim Reason         As Long 'Reason code qualifying CompCode'
```

## MQPUT1 -Bir ileti girin

MQPUT1 çağırısı, bir kuyruğa ya da dağıtım listesine ya da bir konuya ilişkin bir ileti koyar.

Kuyruk, dağıtım listesi ya da konunun açık olması gerekmez.

## Sözdizimi


MQPUT1 (*Hconn*, *ObjDesc*, *MsgDesc*, *PutMsgOpts*, *BufferLength*, *Buffer*, *CompCode*, *Reason*)

## Parametreler

### Hconn

Tip: MQHCONN-input

Bu tanıttıcı, kuyruk yöneticisiyle olan bağlantıyı gösterir. The value of *Hconn* was returned by a previous MQCONN or MQCONNX call.

 On z/OS for CICS applications the MQCONN call can be omitted, and the following value specified for *Hconn* :

### MQHC\_DEF\_HCONN

Varsayılan bağlantı tanıttıcısı.

### ObjDesc

Tip: MQOD-giriş/çıkış

Bu yapı, iletinin eklendiği kuyruğu ya da iletinin yayınlandığı konuyu tanımlayan bir yapıdır. Ayrıntılar için bkz. [“MQOD-Nesne tanımlayıcı” sayfa 473.](#)

Yapı bir kuyruksa, çıkış için kuyruğu açma yetkisine sahip olmalıdır. Kuyruk bir model kuyruğu olmamalıdır.

### MsgDesc

Tip: MQMD-giriş/çıkış

Bu yapı, gönderilmekte olan iletinin özniteliklerini tanımlar ve koyma isteği tamamlandıktan sonra geri bildirim bilgilerini alır. Ayrıntılar için bkz. [“MQMD-İleti tanımlayıcı” sayfa 416.](#)

Uygulama bir version-1 MQMD sağlıyorsa, ileti verilerinin başında, version-2 MQMD içinde var olan alanlara ilişkin değerleri belirtmek için bir MQMDE yapısı eklenmiş olabilir, ancak version-1 bu şekilde bir MQMD yapısı eklenmeyebilir. MQMD 'deki Format alanını MQMDE' nin var olduğunu göstermek için MQFM\_md\_extension değerine ayarlayın. Daha ayrıntılı bilgi için bkz. [“MQMDE-İleti tanımlayıcı uzantısı” sayfa 464 .](#)

MQGMO yapısının MsgHandle alanında ya da MQPMO yapısının OriginalMsgHandle ya da NewMsgHandle alanlarında geçerli bir ileti tanıttıcısı sağlanırsa, uygulamanın bir MQMD yapısı sağlaması gerekmez. Bu alanlardan birinde hiçbir şey sağlanmıyorsa, iletinin tanımlayıcı, ileti tanıttıcılarıyla ilişkili tanımlayıcıdan alınır.

### PutMsg(PutMsg)

Tip: MQPMO-giriş/çıkış

Ayrıntılar için bkz. [“MQPMO-İleti koyma seçenekleri” sayfa 492.](#)

### BufferLength

Tip: MQHOME-giriş

The length of the message in *Buffer*. Sıfır geçerli ve iletinin herhangi bir uygulama verisi içermediği anlamına gelir. Üst sınır çeşitli etkenlere bağlıdır; **BufferLength** parametresinin açıklaması için bkz. [“MQPUT-İleti koy” sayfa 737 .](#)

### Arabellek

Tip: MQBYTEExBufferUzunluğu-giriş

Bu, gönderilecek uygulama iletisi verilerini içeren bir arabelleğidir. Arabelleği, iletteki verilerin niteine uygun bir sınırla hizalayın. 4 baytlık hizalama çoğu ileti için uygundur ( IBM MQ üstbilgi yapılarını içeren iletiler de içinde olmak üzere), ancak bazı iletiler daha sıkı hizalamada gerektirebilir. Örneğin, 64 bitlik bir ikili tamsayı içeren bir ileti 8 byte 'lık hizalama gerektirebilir.

Buffer , karakter ya da sayısal veri içeriyorsa, **MsgDesc** parametresindeki CodedCharSetId ve Encoding alanlarını, verilere uygun değerlere ayarlayın; bu işlem, iletinin alıcısını, verileri (gerekliyse), alıcı tarafından kullanılan karakter takımı ve kodlamaya dönüştürmesini sağlar.

**Not:** MQPUT1 çağrısındaki diğer tüm parametreler, yerel kuyruk yöneticisinin karakter kümesinde ve kodlamasında olmalıdır ( **CodedCharSetId** kuyruk yöneticisi özneliği ve MQENC\_NATIVE ile verilir).

C programlama dilinde, parametre bir işaretçi olarak bildirilir; parametrenin herhangi bir tipinin adresi, parametre olarak belirtilebilir.

**BufferLength** parametresi sıfır ise, Buffer bu durumda gönderilmez; bu durumda, C ya da System/390 çevirici programlarında yazılan programlarla geçirilen parametre adresi boş değerli olabilir.

### CompCode

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Tamamlanma kodu; bu kod aşağıdakilerden biridir:

#### **MQCC\_OK**

İşlem başarıyla tamamlandı.

#### **MQCC\_UYARI**

Uyarı (kısmi tamamlama).

#### **MQCC\_FAILED**

Arama başarısız oldu.

### Neden

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Neden kodu niteleyici CompCode.

CompCode MQCC\_OK ise:

#### **MQRC\_NONE**

(0, X'000 ') Raporlamak için bir neden yok.

CompCode MQCC\_UYARI ise:

#### **MQRC\_MULTIPLE\_REASONS**

(2136, X'858 ') Birden çok neden kodu döndürüldü.

#### **MQRC\_INCOMPLE\_GROUP**

(2241, X'8C1') İleti grubu tamamlanmadı.

#### **MQRC\_INCOMPLE\_MSG**

(2242, X'8C2') Mantıksal ileti tamamlanmadı.

#### **MQRC\_PRIORITY\_EXCEEDS\_MAXIMUM**

(2049, X'801 ') İleti Önceliği desteklenen üst sınır değerini aşıyor.

#### **MQRC\_UNKNOWN\_REPORT\_OPTION**

(2104, X'838 ') İleti tanımlayıcısındaki rapor seçenekleri tanınmadı.

CompCode MQCC\_FAILED ise:

#### **MQRC\_ADAPTER\_NOT\_AVAM**

(2204, X'89C') Bağdaştırıcı kullanılmıyor.

#### **MQRC\_ADAPTER\_SERV\_LOAD\_ERROR**

(2130, X'852 ') Bağdaştırıcı hizmeti modülü yüklenemiyor.

#### **MQRC\_ALIAS\_BASE\_Q\_TYPE\_ERROR**

(2001, X'7D1') Diğer ad taban kuyruğu geçerli bir tip değil.

#### **MQRC\_API\_EXIT\_ERROR**

(2374, X' 946 ') API çıkışı başarısız oldu.

#### **MQRC\_API\_EXIT\_LOAD\_ERROR**

(2183, X'887 ') API çıkışı yüklenemiyor.

**MQRC\_ASID\_MISSMACH**

(2157, X'86D') Birincil ve ana sunucu ASID deęerleri farklı.

**MQRC\_BACKED\_OUT**

(2003, X'7D3') İş birimi yedeklendi.

**MQRC\_BUFFER\_ERROR**

(2004, X'7D4') Arabellek parametresi geçerli deęil.

**MQRC\_BUFFER\_LENGTH\_ERROR**

(2005, X'7D5') Arabellek uzunluğu parametresi geçerli deęil.

**MQRC\_CALL\_IN\_PROGRESS**

(2219, X'8AB') Önceki arama tamamlanmadan önce girilen MQI çağırısı.

**MQRC\_CF\_NOT\_VAR**

(2345, X' 929 ') bağlaşım olanağı yok.

**MQRC\_CF\_STRUC\_AUTH\_FAILED**

(2348, X'92C') Coupling-tesis yapısı yetki denetimi başarısız oldu.

**MQRC\_CF\_STRUC\_ERROR**

(2349, X'92D') Coupling-tesis yapısı geçerli deęil.

**MQRC\_CF\_STRUC\_FAILED**

(2373, X' 945 ') Coupling-tesis yapısı başarısız oldu.

**MQRC\_CF\_STRUC\_IN\_USE**

(2346, X'92A') Coupling-tesis yapısı kullanımda.

**MQRC\_CF\_STRUC\_LIST\_HDR\_IN\_USE**

(2347, X'92B') Coupling-tesis yapısı listesi-üstbilgi kullanımda.

**MQRC\_CFGR\_ERROR**

(2416, X' 970 ') İleti verilerinde PCF grup deęiştirgesi yapısı MQCFGFR geçerli deęil.

**MQRC\_CFH\_ERROR**

(2235, X'8BB') PCF üstbilgi yapısı geçerli deęil.

**MQRC\_CFIF\_ERROR**

(2414, X'96E') İleti verilerinde PCF tamsayı süzgeç parametresi yapısı geçerli deęil.

**MQRC\_CFIL\_ERROR**

(2236, X'8BC') PCF tamsayı listesi parametre yapısı ya da PCIF\*64 tamsayı listesi parametre yapısı geçerli deęil.

**MQRC\_CFIN\_ERROR**

(2237, X'8BD') PCF tamsayı deęiştirgesi yapısı ya da PCIF\*64 tamsayı deęiştirgesi yapısı geçerli deęil.

**MQRC\_CFSF\_ERROR**

(2415, X'96F') İleti verilerinde PCF dizgi süzgeci deęiştirgesi yapısı geçerli deęil.

**MQRC\_CFSL\_ERROR**

(2238, X'8BE') PCF dizgi listesi deęiştirgesi yapısı geçerli deęil.

**MQRC\_CFST\_ERROR**

(2239, X'8BF') PCF dizgi deęiştirgesi yapısı geçerli deęil.

**MQRC\_CICS\_WAIT\_FAILED**

(2140, X'85C') Bekleme isteęi CICStarafından reddedildi.

**MQRC\_CLUSTER\_EXIT\_ERROR**

(2266, X'8DA') Küme iş yükü çıkışı başarısız oldu.

**MQRC\_CLUSTER\_RESOLUTION\_ERROR**

(2189, X'88D') Küme adının çözülmesi başarısız oldu.

**MQRC\_CLUSTER\_KAYNAęI\_HATASI**

(2269, X'8DD') Küme kaynaęı hatası.

**MQRC\_COD\_NOT\_VALID\_FOR\_XCF\_Q**

(2106, X'83A') COD rapor seçeneęi, XCF kuyruęu için geçerli deęil.

**MQRC\_CONNECTION\_BROKEN**

(2009, X'7D9') Kuyruk yöneticisine yönelik bağlantı kaybedildi.

**MQRC\_CONNECTION\_NOT\_YETKILI**

(2217, X'8A9') Bağlantı için yetkili değil.

**MQRC\_CONNECTION\_QUIESCING**

(2202, X'89A') Bağlantı susturulmuş durumda.

**MQRC\_CONNECTION\_DURATION**

(2203, X'89B') Bağlantı sona erdiriliyor.

**MQRC\_İÇERİĞİ**

2554 (X'09FA') İleti içeriği, iletinin genişletilmiş bir ileti seçiciyle bir aboneye teslim edilip edilemeyeceğini belirlemek için ayrıştırılmadı.

**MQRC\_CONTEXT\_HANDLE\_ERROR**

(2097, X'831 ') Bu işleme gönderme yapılan kuyruk tanıtıcısı saklama bağlamı değil.

**MQRC\_CONTEXT\_NOT\_AVAILABLE**

(2098, X'832 ') Kuyruk tanıtıcısı gönderme yapılan bir bağlam kullanılamaz.

**MQRC\_DATA\_LENGTH\_ERROR**

(2010, X'7DA') Veri uzunluğu parametresi geçerli değil.

**MQRC\_DB2\_NOT\_AVAILABLE**

(2342, X' 926 ') Db2 altsistemi kullanılmıyor.

**MQRC\_DEF\_XMIT\_Q\_TYPE\_ERROR**

(2198, X'896 ') Varsayılan iletim kuyruğu yerel değil.

**MQRC\_DEF\_XMIT\_Q\_USAGE\_ERROR**

(2199, X'897 ') Varsayılan iletim kuyruğu kullanım hatası.

**MQRC\_DH\_HATASI**

(2135, X'857 ') Dağıtım üstbilgisi yapısı geçerli değil.

**MQRC\_DLH\_ERROR**

(2141, X'85D') Ölü harf üstbilgisi yapısı geçerli değil.

**MQRC\_EPH\_ERROR**

(2420, X' 974 ') Embedded PCF yapısı geçerli değil.

**MQRC\_EXPIRY\_ERROR**

(2013, X'7DD') Süre bitim zamanı geçerli değil.

**MQRC\_FEEDBACK\_ERROR**

(2014, X'7DE') Geribildirim kodu geçerli değil.

**MQRC\_GLOBAL\_UOW\_CONFLICTIONS**

(2351, X'92F') Genel iş çakışmaları birimleri.

**MQRC\_GROUP\_ID\_ERROR**

(2258, X'8D2') Grup tanıtıcısı geçerli değil.

**MQRC\_ANDLE\_IN\_USE\_FOR\_UOW**

(2353, X' 931 ') Genel iş birimi için kullanılan tanıtıcı.

**MQRC\_HANDLE\_NOT\_VAR**

(2017, X'7E1') Kullanılabilir başka tanıtıcı yok.

**MQRC\_HCONN\_ERROR**

(2018, X'7E2') Bağlantı tanıtıcısı geçerli değil.

**MQRC\_HEADER\_ERROR**

(2142, X'85E') IBM MQ üstbilgi yapısı geçerli değil.

**MQRC\_IIH\_ERROR**

(2148, X'864 ') IMS bilgi üstbilgisi yapısı geçerli değil.

**MQRC\_LOCAL\_UOW\_CONFLICTID**

(2352, X' 930 ') Genel iş birimi yerel iş birimi ile çakışıyor.

**MQRC\_MD\_ERROR**

(2026, X'7EA') İleti tanımlayıcısı geçerli değil.

**MQRC\_MDE\_HATASI**

(2248, X'8C8') İleti tanımlayıcısı uzantısı geçersiz.

**MQRC\_MISSING\_REPLY\_TO\_Q**

(2027, X'7EB') Yanıtın gönderileceği kuyruk.

**MQRC\_MISSING\_WIH**

(2332, X'91C') İleti verileri MQWIH ile başlamaz.

**MQRC\_MSG\_FLAGS\_ERROR**

(2249, X'8C9') İleti işaretleri geçerli değil.

**MQRC\_MSG\_SEQ\_NUMBER\_ERROR**

(2250, X'8CA') İleti sıra numarası geçerli değil.

**MQRC\_MSG\_TOO\_BIG\_FOR\_Q**

(2030, X'7EE') İleti uzunluğu, kuyruk için üst sınırdan büyük.

**MQRC\_MSG\_TOO\_BIG\_FOR\_Q\_MGR**

(2031, X'7EF') İleti uzunluğu kuyruk yöneticisi için üst sınırdan büyük.

**MQRC\_MSG\_TYPE\_ERROR**

(2029, X'7ED') İleti tanımlayıcısında ileti tipi geçerli değil.

**MQRC\_MULTIPLE\_REASONS**

(2136, X'858 ') Birden çok neden kodu döndürüldü.

**MQRC\_NO\_DESTINATIONS\_AVAILABLE**

(2270, X'8DE') Kullanılabilir hedef kuyruğu yok.

**MQRC\_NOT\_YETKILI**

(2035, X'7F3') Erişim için yetkili değil.

**MQRC\_OBJECT\_ZARAR**

(2101, X'835 ') Nesne zarar gördü.

**MQRC\_OBJECT\_IN\_USE**

(2042, X'7FA') Nesnesi çakışan seçeneklerle zaten açık.

**MQRC\_OBJECT\_LEVEL\_UYUMSUZ**

(2360, X' 938 ') Nesne düzeyi uyumlu değil.

**MQRC\_OBJECT\_NAME\_ERROR**

(2152, X'868 ') Nesne adı geçerli değil.

**MQRC\_OBJECT\_NOT\_UNIQUE**

(2343, X' 927 ') Nesne benzersiz değil.

**MQRC\_OBJECT\_Q\_MGR\_NAME\_ERROR**

(2153, X'869 ') Nesne kuyruğu yöneticisi adı geçerli değil.

**MQRC\_OBJECT\_RECORDS\_ERROR**

(2155, X'86B') Nesne kayıtları geçerli değil.

**MQRC\_OBJECT\_TYPE\_ERROR**

(2043, X'7FB') Nesne tipi geçerli değil.

**MQRC\_OD\_ERROR**

(2044, X'7FC') Nesne tanımlayıcı yapısı geçerli değil.

**MQRC\_OFFSET\_ERROR**

(2251, X'8CB') İleti bölümü görelî konumu geçerli değil.

**MQRC\_OPTIONS\_ERROR**

(2046, X'7FE') Seçenekler geçerli değil ya da tutarlı değil.

**MQRC\_ORIGINAL\_LENGTH\_ERROR**

(2252, X'8CC') Özgün uzunluk geçerli değil.

**MQRC\_PAGEES\_HATASI**

(2193, X'891 ') Sayfa kümesi veri kümesine erişilirken hata oluştu.

**MQRC\_PAGEET\_FULL**

(2192, X'890 ') Dış depolama ortamı dolu.

**MQRC\_PCF\_ERROR**

(2149, X'865 ') PCF yapıları geçerli değil.

**MQRC\_PERSISTENCE\_ERROR**

(2047, X'7FF') Persistence geçerli değil.

**MQRC\_PERSISTENT\_NOT\_ALLOWED**

(2048, X'800 ') Kuyruk kalıcı iletileri desteklemiyor.

**MQRC\_PMO\_ERROR**

(2173, X'87D') Koyma-ileti seçenekleri yapısı geçerli değil.

**MQRC\_PMO\_RECORD\_FLAGS\_ERROR**

(2158, X'86E') İleti kaydı işaretlerinin konması geçerli değil.

**MQRC\_PRIORITY\_ERROR**

(2050, X'802 ') İleti önceliği geçerli değil.

**MQRC\_PUBLICATION\_FAILURE**

(2502, X'9C6') Yayın, abonelerin hiçbirine teslim edilemedi.

**MQRC\_PUT\_INHIBITED**

(2051, X'803 ') Kuyruk araması engellenmiş olarak çağrılıyor.

**MQRC\_PUT\_MSG\_RECORDS\_ERROR**

(2159, X'86F') İleti içeren ileti kayıtları geçerli değil.

**MQRC\_Q\_DELETED**

(2052, X'804 ') Kuyruk silinmiştir.

**MQRC\_Q\_FULL**

(2053, X'805 ') Kuyruk, ileti sayısı üst sınırını zaten içeriyor.

**MQRC\_Q\_MGR\_NAME\_ERROR**

(2058, X'80A') Kuyruk yöneticisi adı geçerli değil ya da bilinmiyor.

**MQRC\_Q\_MGR\_NOT\_VAR**

(2059, X'80B') Kuyruk yöneticisi bağlantı için kullanılabilir değil.

**MQRC\_Q\_MGR QUIESCING**

(2161, X'871 ') Kuyruk yöneticisi susturuyor.

**MQRC\_Q\_MGR\_STOPPING**

(2162, X'872 ') Kuyruk yöneticisi sona erdiriliyor.

**MQRC\_Q\_SPACE\_NOT\_VAR**

(2056, X'808 ') Kuyruğun diskte kullanılabilir alan yok.

**MQRC\_Q\_TYPE\_ERROR**

(2057, X'809 ') Kuyruk tipi geçerli değil.

**MQRC\_RECS\_PRESENT\_ERROR**

(2154, X'86A') Kayıt sayısı geçerli değil.

**MQRC\_REMOTE\_Q\_NAME\_ERROR**

(2184, X'888 ') Uzak kuyruk adı geçerli değil.

**MQRC\_REPORT\_OPTIONS\_ERROR**

(2061, X'80D') İleti tanımlayıcısındaki rapor seçenekleri geçerli değil.

**MQRC\_RESOURCE\_SORUNU**

(2102, X'836 ') Yetersiz sistem kaynakları var.

**MQRC\_RESPONSE\_RECORDS\_ERROR**

(2156, X'86C') Yanıt kayıtları geçerli değil.

**MQRC\_RFH\_ERROR**

(2334, X'91E') MQRFH ya da MQRFH2 yapısı geçerli değil.

**MQRC\_RMH\_ERROR**

(2220, X'8AC') Başvuru iletisi üstbilgi yapısı geçerli değil.



**MQRC\_SECURITY\_ERROR**

(2063, X'80F') Güvenlik hatası oluştu.

**MQRC\_SEGMENT\_LENGTH\_ZERO**

(2253, X'8CD') İleti kesiminde veri uzunluğu sıfır.

**MQRC\_SELECTION\_NOT\_AVAM**

2551 (X'09F7') Yayın için olası bir abone var, ancak kuyruk yöneticisi yayının aboneye gönderilip gönderilmeyeceğini denetleyemiyor.

**MQRC\_STOPPED\_BY\_CLUSTER\_EXIT**

(2188, X'88C') Küme iş yükü çıkışı tarafından çağrılan çağrı reddedildi.

**MQRC\_STORAGE\_CLASS\_ERROR**

(2105, X'839 ') Depolama sınıfı hatası.

**MQRC\_STORAGE\_MEDIUM\_FULL**

(2192, X'890 ') Dış depolama ortamı dolu.

**MQRC\_STORAGE\_NOT\_AVAM**

(2071, X'817 ') Kullanılabilir bellek yetersiz.

**MQRC\_SUPPRESSED\_BY\_EXIT**

(2109, X'83D') Çıkış programı tarafından çağrı engellendi.

**MQRC\_SYNCPOINT\_LIMIT\_UVARD**

(2024, X'7E8') Yürürlükteki iş birimi içinde başka ileti işlenemez.

**MQRC\_SYNCPOINT\_NOT\_AVAM**

(2072, X'818 ') Syncpoint desteği yok.

**MQRC\_TM\_HATASI**

(2265, X'8D9') Tetikleyici ileti yapısı geçerli değil.

**MQRC\_TMC\_ERROR**

(2191, X'88F') Karakter tetikleyicisi ileti yapısı geçerli değil.

**MQRC\_UNEXPECTED\_ERROR**

(2195, X'893 ') Beklenmeyen bir hata oluştu.

**MQRC\_UNKNOWN\_ALIAS\_BASE\_Q**

(2082, X'822 ') Bilinmeyen diğer ad temel kuyruğu.

**MQRC\_UNKNOWN\_DEF\_XMIT\_Q**

(2197, X'895 ') Bilinmeyen varsayılan iletim kuyruğu.

**MQRC\_UNKNOWN\_OBJECT\_NAME**

(2085, X'825 ') Bilinmeyen nesne adı.

**MQRC\_UNKNOWN\_OBJECT\_Q\_MGR**

(2086, X'826 ') Bilinmeyen nesne kuyruk yöneticisi.

**MQRC\_UNKNOWN\_REMOTE\_Q\_MGR**

(2087, X'827 ') Bilinmeyen uzak kuyruk yöneticisi.

**MQRC\_UNKNOWN\_XMIT\_Q**

(2196, X'894 ') Bilinmeyen iletim kuyruğu.

**MQRC\_UOWENLISTMENT\_ERROR**

(2354, X' 932 ') Genel iş birimi içindeki bilgiler başarısız oldu.

**MQRC\_UOW\_MIX\_NOT\_SUPPORTANT**

(2355, X' 933 ') İşlerin birim içi çağrıları karışımı desteklenmiyor.

**MQRC\_UOW\_NOT\_VAR**

(2255, X'8CF') Kuyruk yöneticisinin kullanması için kullanılacak iş birimi.

**MQRC\_WIH\_ERROR**

(2333, X'91D') MQWIH yapısı geçerli değil.

**MQRC\_HATA\_CF\_DÜZEYI**

(2366, X'93E') Coupling-tesis yapısı yanlış düzeyde.

### **MQRC\_WRONG\_MD\_VERSION**

(2257, X'8D1') MQMD' nin yanlış sürümü sağlandı.

### **MQRC\_XMIT\_Q\_TYPE\_ERROR**

(2091, X'82B') İletim kuyruğu yerel değil.

### **MQRC\_XMIT\_Q\_USAGE\_ERROR**

(2092, X'82C') İletim kuyruğu yanlış kullanıma sahip.

### **MQRC\_XQH\_ERROR**

(2260, X'8D4') İletim kuyruğu üstbilgi yapısı geçerli değil.

Bu kodlarla ilgili ayrıntılı bilgi için [İletiler ve neden kodları](#) başlıklı konuya bakın.

## **Kullanım notları**

1. İletileri bir kuyruğa koymak için hem MQPUT hem de MQPUT1 çağrılarını kullanılabilir; bu çağrılar, koşullara bağlı olarak değişir:
  - Aynı kuyruğa birden çok ileti yerleştirmek için MQPUT çağrısını kullanın.  
Önce MQOO\_OUTPUT seçeneğini belirten bir MQOPEN çağrısı yayınlandı, ardından kuyruğa ileti eklemek için bir ya da daha çok MQPUT isteği sunuldu; son olarak, kuyruk bir MQCLOSE çağrısıyla kapatıldı. Bu, MQPUT1 çağrısının yinelenen kullanımından daha iyi başarımlar sağlar.
  - Bir kuyruğa yalnızca *bir* ileti yerleştirmek için MQPUT1 çağrısını kullanın.  
Bu çağrı, MQOPER, MQPUT ve MQCLOSE çağrılarını tek bir çağrıya sarkıyor ve verilmesi gereken çağrılarının sayısını en aza indiriyor.
2. Bir uygulama, ileti grupları kullanılmadan aynı kuyruğa ileti dizisi koyarsa, Belirli koşullar karşılanırsa, bu iletilerin sırası korunur. Ancak, çoğu ortamda MQPUT1 çağrısı bu koşulları karşılamıyor ve ileti düzenini korumuyor. MQPUT çağrısı bu ortamlarda kullanılmalı. Ayrıntılar için bkz. [MQPUT kullanım notları](#).
3. İletileri dağıtım listelerine yerleştirmek için MQPUT1 çağrısı kullanılabilir. Bu konuda genel bilgi edinmek için, MQOPER ve MQPUT çağrılarına ilişkin kullanım notlarına bakın.

Dağıtım listeleri aşağıdaki ortamlarda desteklenir:

-  AIX
-  IBM i
-  Linux
-  Windows

ve bu sistemlere bağlı IBM MQ istemcileri için geçerli olur.

MQPUT1 çağrısı kullanılırken aşağıdaki farklar geçerlidir:

- a. Uygulama MQRR yanıt kayıtları sağlıyorsa, bunlar MQOD yapısı kullanılarak sağlanmalıdır; MQPMO yapısı kullanılarak sağlanamazlar.
- b. The reason code MQRC\_OPEN\_FAILED is never returned by MQPUT1 in the response records; if a queue fails to open, the response record for that queue contains the reason code resulting from the open operation.

Bir kuyruğa ilişkin açık bir işlem MQCC\_UYARI tamamlanma koduyla başarılı olursa, o kuyruğa ilişkin yanıt kaydındaki tamamlanma kodu ve neden kodu, koyma işleminden kaynaklanan tamamlanma ve neden kodlarıyla değiştirilir.

MQOPED ve MQPUT çağrılarında olduğu gibi, kuyruk yöneticisi yanıt kayıtlarını (sağlandıysa) yalnızca, çağrının sonucu dağıtım listesindeki tüm kuyruklar için aynı değilse ayarlar; bu, MQRC\_MULTIPLE\_REASONS neden koduyla belirtilen çağrıyla belirtilir.

4. MQPUT1 çağrısı, bir iletiyi küme kuyruğuna yerleştirmek için kullanılırsa, MQOPED çağrısında MQOO\_BIND\_NOT\_FIXED değeri belirlenmiş gibi davranır.

5. Bir ileti, uygulama iletileri verilerinin başına bir ya da daha çok IBM MQ üstbilgi yapısıyla konursa, kuyruk yöneticisi, geçerli olduğunu doğrulamak için üstbilgi yapılarında bazı denetimleri gerçekleştirir. Bu konuya ilişkin ek bilgi için, MQPUT çağrısına ilişkin kullanım notlarına bakın.
6. Uyarı durumlarının birden fazlası ortaya çıkarsa (**CompCode** parametresine bakın), döndürülen neden kodu, aşağıdaki listede yer alan ilk sıradır:
  - a. MQRC\_MULTIPLE\_REASONS
  - b. MQRC\_INCOMPLE\_MSG
  - c. MQRC\_INCOMPLE\_GROUP
  - d. MQRC\_PRIORITY\_EXCEEDS\_MAXIMUM ya da MQRC\_UNKNOWN\_REPORT\_OPTION
7. Visual Basic programlama dili için, aşağıdaki noktalar geçerlidir:
  - **Buffer** parametresinin boyutu, **BufferLength** parametresi tarafından belirtilen uzunluktan küçükse, çağrı neden kodu MQRC\_BUFFER\_LENGTH\_ERROR neden ile başarısız olur.
  - **Buffer** parametresi, Stringtipinde olduğu için bildirilir. If the data to be placed on the queue is not of type String, use theMQPUT1Any çağrısı MQPUT1yerine çağrılıyor.

MQPUT1Any çağrısı, MQPUT1 çağrıyla aynı parametrelere sahiptir; ancak, **Buffer** parametresinin Anytipi olarak bildirilmesi, kuyrukta herhangi bir veri tipinin yerleştirilmesine olanak sağlar. Ancak bu, Buffer ' in büyüklük olarak en az BufferLength byte olduğundan emin olmak için denetleyemediği anlamına gelir.
8. When an MQPUT1 call is issued with MQPMO\_SYNCPOINT, the default behavior changes, so that the put operation is completed asynchronously. Bu, MQOD ve MQMD yapılarındaki bazı alanlara dayanan, ancak şimdi tanımsız değerler içeren bazı uygulamaların davranışlarında değişikliğe neden olabilir. Bir uygulama, koyma işleminin zamanuyumlu olarak gerçekleştirildiğinden ve tüm uygun alan değerlerinin tamamlandığından emin olmak için MQPMO\_SYNC\_RESPONSE değerini belirleyebilir.

## C çağırma

```
MQPUT1 (Hconn, &ObjDesc, &MsgDesc, &PutMsgOpts,
        BufferLength, Buffer, &CompCode, &Reason);
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
MQHCONN  Hconn;           /* Connection handle */
MQOD     ObjDesc;        /* Object descriptor */
MQMD     MsgDesc;        /* Message descriptor */
MQPMO    PutMsgOpts;     /* Options that control the action of MQPUT1 */
MQLONG   BufferLength;    /* Length of the message in Buffer */
MQBYTE   Buffer[n];      /* Message data */
MQLONG   CompCode;       /* Completion code */
MQLONG   Reason;         /* Reason code qualifying CompCode */
```

## Cobol çağırma

```
CALL 'MQPUT1' USING HCONN, OBJDESC, MSGDESC, PUTMSGOPTS,
                  BUFFERLENGTH, BUFFER, COMPCODE, REASON.
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
** Connection handle
01 HCONN          PIC S9(9) BINARY.
** Object descriptor
01 OBJDESC.
   COPY CMQODV.
** Message descriptor
01 MSGDESC.
   COPY CMQMDV.
** Options that control the action of MQPUT1
```

```

01 PUTMSGOPTS.
   COPY CMQPMOV.
** Length of the message in BUFFER
01 BUFFERLENGTH PIC S9(9) BINARY.
** Message data
01 BUFFER        PIC X(n).
** Completion code
01 COMPCODE      PIC S9(9) BINARY.
** Reason code qualifying COMPCODE
01 REASON        PIC S9(9) BINARY.

```

## PL/I çağırımı

```

call MQPUT1 (Hconn, ObjDesc, MsgDesc, PutMsgOpts, BufferLength, Buffer,
            CompCode, Reason);

```

Değiştirgeleri aşağıdaki gibi bildirin:

```

dcl Hconn          fixed bin(31); /* Connection handle */
dcl ObjDesc        like MQOD;    /* Object descriptor */
dcl MsgDesc        like MQMD;    /* Message descriptor */
dcl PutMsgOpts     like MQPMO;   /* Options that control the action of
MQPUT1 */
dcl BufferLength    fixed bin(31); /* Length of the message in Buffer */
dcl Buffer          char(n);      /* Message data */
dcl CompCode       fixed bin(31); /* Completion code */
dcl Reason         fixed bin(31); /* Reason code qualifying CompCode */

```

## High Level Assembler çağırısı

```

CALL MQPUT1, (HCONN,OBJDESC,MSGDESC,PUTMSGOPTS,BUFFERLENGTH, X
            BUFFER,COMPCODE,REASON)

```

Değiştirgeleri aşağıdaki gibi bildirin:

HCONN	DS	F	Connection handle
OBJDESC	CMQODA	,	Object descriptor
MSGDESC	CMQMDA	,	Message descriptor
PUTMSGOPTS	CMQPMOA	,	Options that control the action of MQPUT1
BUFFERLENGTH	DS	F	Length of the message in BUFFER
BUFFER	DS	CL(n)	Message data
COMPCODE	DS	F	Completion code
REASON	DS	F	Reason code qualifying COMPCODE

## Visual Basic çağırısı



```

MQPUT1 Hconn, ObjDesc, MsgDesc, PutMsgOpts, BufferLength, Buffer,
      CompCode, Reason

```

Değiştirgeleri aşağıdaki gibi bildirin:

```

Dim Hconn          As Long   'Connection handle'
Dim ObjDesc        As MQOD   'Object descriptor'
Dim MsgDesc        As MQMD   'Message descriptor'
Dim PutMsgOpts     As MQPMO  'Options that control the action of MQPUT1'
Dim BufferLength    As Long   'Length of the message in Buffer'
Dim Buffer          As String  'Message data'
Dim CompCode       As Long   'Completion code'
Dim Reason         As Long   'Reason code qualifying CompCode'

```

## MQSET-Nesne özniteliklerinin ayarlanması

Bir tanıtıcı tarafından temsil edilen bir nesnenin özniteliklerini değiştirmek için MQSET çağrısını kullanın. Nesne bir kuyruk olmalıdır.

### Sözdizimi


MQSET (*Hconn*, *Hobj*, *SelectorCount*, *Selectors*, *IntAttrCount*, *IntAttrs*, *CharAttrLength*, *CharAttrs*, *Compcode*, *Reason*)

### Parametreler

#### Hconn

Tip: MQHCONN-input

Bu tanıtıcı, kuyruk yöneticisiyle olan bağlantıyı gösterir. The value of Hconn was returned by a previous MQCONN or MQCONNX call.

 On z/OS for CICS applications the MQCONN call can be omitted, and the following value specified for *Hconn* :

#### MQHC\_DEF\_HCONN

Varsayılan bağlantı tanıtıcısı.

#### Hobj

Tip: MQHOBJ-input

Bu tanıtıcı, ayarlanacak özniteliklere sahip kuyruk nesnesini gösterir. Tanıtıcı, MQOO\_SET seçeneğini belirten önceki bir MQOPER çağrısı tarafından döndürüldü.

#### SelectorCount

Tip: MQHOME-girişi

Bu, *Selectors* dizisinde sağlanan seçicilerin sayısıdır. Bu, ayarlanacak özniteliklerin sayısıdır. Sıfır geçerli bir değerdir. İzin verilen sayı üst sınırı 256 'tır.

#### Seçiciler

Tip: MQLONGxSelectorCount-input

Bu, **SelectorCount** öznitelik seçicilerinden oluşan bir dizidir; her seçici, bir özniteliği (tamsayı ya da karakter) ayarlanacak bir değerle tanımlar.

Her seçici, *Hobj* ' in temsil ettiği kuyruk türü için geçerli olmalıdır. Daha sonra listelendiği gibi, yalnızca belirli MQIA\_\* ve MQCA\_\* değerlerine izin verilir.

Seçiciler herhangi bir sırada belirtilebilir. Attribute values that correspond to integer attribute selectors (MQIA\_\* selectors) must be specified in *IntAttrs* in the same order in which these selectors occur in *Selectors*. Karakter özniteliği seçicilere karşılık gelen öznitelik değerleri (MQCA\_\* seçiciler), bu seçicilerin olduğu sırada *CharAttrs* içinde belirtilmelidir. MQIA\_\* seçicileri MQCA\_\* seçicileriyle birlikte serpiştirilebilir; her tip içindeki yalnızca göreceli sıralama önemlidir.

Aynı seçiciyi birden çok kez belirtebilirsiniz; varsa, belirli bir seçici için belirtilen son değer, yürürlüğe girecektir.

#### Not:

1. Tamsayı ve karakter özniteliği seçicileri farklı iki aralık içinde ayrılır; MQIA\_\* seçicileri MQIA\_FIRST içinde MQIA\_LAST ve MQCA\_FIRST aralığında MQCA\_SON ile MQCA\_\* seçicileri bulunur.

Her bir aralık için, MQIA\_LAST\_ALD değişmezleri ve MQCA\_LAST\_USED, kuyruk yöneticisinin kabul ettiği en yüksek değeri tanımlar.

2. Tüm MQIA\_\* seçicileri önce gerçekleşirse, aynı öge numaraları *Selectors* ve *IntAttrs* dizilerinde karşılık gelen öğeleri adreslemek için kullanılabilir.

3. **SelectorCount** parametresi sıfır ise, **Selectors** bu durumda gönderilmez; bu durumda, C ya da System/390 çevirici programlarında yazılan programlarla geçirilen parametre adresi boş değerli olabilir.

Ayarlanabilen öznitelikler aşağıdaki çizelgede listelenir. Bu çağrıyı kullanarak başka hiçbir öznitelik ayarlanmaz. MQCA\_\* özniteliği seçicileri için, CharAttrrs içinde gerekli olan dizinin bayt cinsinden uzunluğunu tanımlayan sabit parantez içinde sağlanır.

Çizelge 555. Kuyruklar için MQSET öznitelik seçicileri		
Seçici	Tanım	Not
MQCA_TRIGGER_DATA	Tetikleme verileri (MQ_TRIGGER_DATA_LENGTH).	
MQIA_DIST_LISTS	Dağıtım listesi desteği.	1
MQIA_INHIBIT_GET	Alma işlemlerine izin verilip verilmediği.	
MQIA_INHIBIT_PUT	Koyma işlemlerine izin verilip verilmediği.	
MQIA_TRIGGER_CONTROL	Tetik kontrolü.	
MQIA_TRIGGER_DEPTH	Tetik derinliği.	
MQIA_TRIGGER_MSG_PRIORITY	Tetikleyiciler için eşik iletisi önceliği.	
MQIA_TRIGGER_TYPE	Tetikleyici tipi.	

**Not:**

1. Yalnızca şu altyapılarda desteklenir:

-  AIX
-  IBM i
-  Linux
-  Windows

ve bu sistemlere bağlı IBM MQ MQI clients için.

**IntAttrSayı**

Tip: MQHOME-girişi

Bu, IntAttrrs dizindeki öğelerin sayısıdır ve **Selectors** parametresindeki MQIA\_\* seçicilerinin en az sayıda olması gerekir. Sıfır (sıfır) değeri yoksa, geçerli bir değerdir.

**IntAttrrs**

Tip: MQLONGxIntAttrCount -giriş

Bu, IntAttrCount tamsayı öznitelik değerlerinden oluşan bir dizidir. Bu öznitelik değerleri, Selectors dizisindeki MQIA\_\* seçicileriyle aynı sırada olmalıdır.

**IntAttrCount** ya da **SelectorCount** parametresi sıfır ise, IntAttrrs bu durumda gönderilmez; bu durumda, C ya da System/390 çevirici programlarında yazılan programlarla geçirilen parametre adresi boş değerli olabilir.

**CharAttrUzunluğu**

Tip: MQHOME-girişi

Bu, **CharAttrrs** parametresinin bayt cinsinden uzunluğudur ve en az Selectors dizisinde belirtilen karakter özniteliklerinin uzunluklarının toplamını olmalıdır. Selectors içinde MQCA\_\* seçicileri yoksa, sıfır geçerli bir değerdir.

**CharAttrrs**

Tip: MQCHAR x CharAttrUzunluk-giriş

Bu, karakter öznitelik değerlerini içeren arabelleğidir ve birlikte bitleştirilir. Arabellek uzunluğu **CharAttrLength** parametresiyle verilir.

Karakter öznitelikleri, **Selectors** dizesindeki **MQCA\_\*** seçicileriyle aynı sırada belirtilmelidir. Her bir karakter özniteliğinin uzunluğu sabittir (bkz. **Seçiciler**). Bir öznitelik için ayarlanacak değer, özniteliğin tanımlı uzunluğundan daha az boşluk içermeyen karakter içeriyorsa, öznitelik değerinin, özniteliğin tanımlı uzunlukla eşleşmesini sağlamak için **CharAttrs** içindeki değeri boşluklarla sağa doğru ayarlayın.

**CharAttrLength** ya da **SelectorCount** parametresi sıfır ise, **CharAttrs** bu durumda gönderilmez; bu durumda, C ya da System/390 çevirici programlarında yazılan programlarla geçirilen parametre adresi boş değerli olabilir.

### **CompCode**

Tip: **MQXX\_ENCODE\_CASE\_ONE** long-output

Tamamlanma kodu; bu kod aşağıdakilerden biridir:

#### **MQCC\_OK**

İşlem başarıyla tamamlandı.

#### **MQCC\_FAILED**

Arama başarısız oldu.

### **Neden**

Tip: **MQXX\_ENCODE\_CASE\_ONE** long-output

Neden kodu niteleyici **CompCode**.

*CompCode* **MQCC\_OK** ise:

#### **MQRC\_NONE**

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* **MQCC\_FAILED** ise:

#### **MQRC\_ADAPTER\_NOT\_AVAM**

(2204, X'89C') Bağdaştırıcı kullanılamıyor.

#### **MQRC\_ADAPTER\_SERV\_LOAD\_ERROR**

(2130, X'852 ') Bağdaştırıcı hizmeti modülü yüklenemiyor.

#### **MQRC\_API\_EXIT\_ERROR**

(2374, X' 946 ') API çıkışı başarısız oldu.

#### **MQRC\_API\_EXIT\_LOAD\_ERROR**

(2183, X'887 ') API çıkışı yüklenemiyor.

#### **MQRC\_ASID\_MISSMACH**

(2157, X'86D') Birincil ve ana sunucu ASID değerleri farklı.

#### **MQRC\_CALL\_IN\_PROGRESS**

(2219, X'8AB') Önceki arama tamamlanmadan önce girilen MQI çağrısı.

#### **MQRC\_CF\_NOT\_VAR**

(2345, X' 929 ') Coupling tesisi kullanılamıyor.

#### **MQRC\_CF\_STRUC\_FAILED**

(2373, X' 945 ') Coupling-tesis yapısı başarısız oldu.

#### **MQRC\_CF\_STRUC\_IN\_USE**

(2346, X'92A') Coupling-tesis yapısı kullanımda.

#### **MQRC\_CF\_STRUC\_LIST\_HDR\_IN\_USE**

(2347, X'92B') Coupling-tesis yapısı listesi-üstbilgi kullanımda.

#### **MQRC\_CHAR\_ATTR\_LENGTH\_ERROR**

(2006, X'7D6') Karakter öznitelikleri uzunluğu geçerli değil.

#### **MQRC\_CHAR\_ATTRS\_ERROR**

(2007, X'7D7') Karakter öznitelikleri dizgisi geçerli değil.

**MQRC\_CICS\_WAIT\_FAILED**  
(2140, X'85C') Bekleme isteđi CICStarafından reddedildi.

**MQRC\_CONNECTION\_BROKEN**  
(2009, X'7D9') Kuyruk yöneticisine yönelik bağlantı kaybedildi.

**MQRC\_CONNECTION\_NOT\_YETKILI**  
(2217, X'8A9') Bağlantı için yetkili deđil.

**MQRC\_CONNECTION\_DURATION**  
(2203, X'89B') Bağlantı sona erdiriliyor.

**MQRC\_DB2\_NOT\_AVAILABLE**  
(2342, X' 926 ') Db2 altsistemi kullanılamıyor.

**MQRC\_HCONN\_ERROR**  
(2018, X'7E2') Bağlantı tanıtıcısı geçerli deđil.

**MQRC\_HOBJ\_ERROR**  
(2019, X'7E3') Nesne tanıtıcısı geçerli deđil.

**MQRC\_INHIBIT\_VALUE\_ERROR**  
(2020, X'7E4') Inhibit-get ya da inhibit-put queue özniteliđinin deđeri geçerli deđil.

**MQRC\_INT\_ATTR\_COUNT\_ERROR**  
(2021, X'7E5') Tamsayı öznitelikleri sayısı geçerli deđil.

**MQRC\_INT\_ATTRS\_ARRAY\_ERROR**  
(2023, X'7E7') Tamsayı öznitelikleri dizisi geçerli deđil.

**MQRC\_NOT\_OPEN\_FOR\_SET**  
(2040, X'7F8') Kuyruk, küme için açık deđil.

**MQRC\_OBJECT\_CHANGED**  
(2041, X'7F9') Nesne tanımlaması açıldıđından beri deđiştirildi.

**MQRC\_OBJECT\_ZARAR**  
(2101, X'835 ') Nesne zarar gördü.

**MQRC\_PAGEES\_HATASI**  
(2193, X'891 ') Sayfa kümesi veri kümesine erişilirken hata oluştu.

**MQRC\_Q\_DELETED**  
(2052, X'804 ') Kuyruk silinmiştir.

**MQRC\_Q\_MGR\_NAME\_ERROR**  
(2058, X'80A') Kuyruk yöneticisi adı geçerli deđil ya da bilinmiyor.

**MQRC\_Q\_MGR\_NOT\_VAR**  
(2059, X'80B') Kuyruk yöneticisi bağlantı için kullanılabilir deđil.

**MQRC\_Q\_MGR\_STOPPING**  
(2162, X'872 ') Kuyruk yöneticisi sona erdiriliyor.

**MQRC\_RESOURCE\_SORUNU**  
(2102, X'836 ') Yetersiz sistem kaynakları var.

**MQRC\_SELECTOR\_COUNT\_ERROR**  
(2065, X'811 ') Seçici sayısı geçerli deđil.

**MQRC\_SELECTOR\_ERROR**  
(2067, X'813 ') Öznitelik seçicisi geçerli deđil.

**MQRC\_SELECTOR\_LIMIT\_EXACID**  
(2066, X'812 ') Seçicilerin sayısı çok büyük.

**MQRC\_STORAGE\_NOT\_AVAM**  
(2071, X'817 ') Kullanılabilir bellek yetersiz.

**MQRC\_SUPPRESSED\_BY\_EXIT**  
(2109, X'83D') Çıkış programı tarafından çağrı engellendi.

**MQRC\_TRIGGER\_CONTROL\_ERROR**  
(2075, X'81B') Tetikleme denetimi özniteliđi için deđer geçerli deđil.



### **MQRC\_TRIGGER\_DEPTH\_ERROR**

(2076, X'81C') Tetikleme derinliği özniteliği için değer geçerli değil.

### **MQRC\_TRIGGER\_MSG\_PRIORITY\_ERR**

(2077, X'81D') Tetikleme-Message-priority özniteliği için değer geçerli değil.

### **MQRC\_TRIGGER\_TYPE\_ERROR**

(2078, X'81E') Tetikleme tipi özniteliği için değer geçerli değil.

### **MQRC\_UNEXPECTED\_ERROR**

(2195, X'893 ') Beklenmeyen bir hata oluştu.

Bu kodlarla ilgili ayrıntılı bilgi için [İletiler ve neden kodları](#) başlıklı konuya bakın.

## **Kullanım notları**

1. Bu çağrıyı kullanarak, uygulama bir tamsayı öznitelikleri dizisi ya da bir karakter özniteliği dizileri derlemi ya da her ikisini birden belirleyebilir. Herhangi bir hata ortaya çıkmazsa, belirtilen öznitelikler aynı anda ayarlanır. Bir hata oluşursa (örneğin, bir seçici geçerli değilse ya da bir özniteliği geçerli olmayan bir değere ayarlama girişiminde bulunulduysa), çağrı başarısız olur ve öznitelikler ayarlanmaz.
2. Özniteliklerin değerleri MQINQ çağrısı kullanılarak saptlanabilir; ayrıntılar için [“MQINQ-Nesne özniteliklerinin sorgulması” sayfa 693](#) ' e bakın.

**Not:** MQINQ çağrısının kullanılarak sorgulanabilen değerlere sahip tüm öznitelikler, MQSET çağrısını kullanarak değerlerini değiştirebilirler. Örneğin, bu çağrıyla hiçbir süreç nesnesi ya da kuyruk yöneticisi özniteliği ayarlanamaz.

3. Öznitelik değişiklikleri, kuyruk yöneticisinin yeniden başlatma işlemlerinde (kuyruk yöneticisinin yeniden başlatılmalarından hayatta kalmayan geçici dinamik kuyruklara ilişkin değişiklikler dışında) korunur.
4. MQSET çağrısını kullanarak, bir model kuyruğunun özniteliklerini değiştiremezsiniz. Ancak, MQOO\_SET seçeneğiyle MQOP çağrısını kullanarak bir model kuyruğu açsanız, MQSET çağrısıyla yaratılan dinamik yerel kuyruğun özniteliklerini ayarlamak için MQSET çağrısını kullanabilirsiniz.
5. Ayarlanmakta olan nesne bir küme kuyruğunsa, açılacak açık için küme kuyruğunun yerel bir yönetim ortamı olmalıdır.

Nesne özniteliklerine ilişkin ek bilgi için aşağıdaki başlıklara bakın:

- [“Kuyruklara ilişkin öznitelikler” sayfa 821](#)
- [“Ad listelerine ilişkin öznitelikler” sayfa 854](#)
- [“Süreç tanımlamalarına ilişkin öznitelikler” sayfa 856](#)
- [“Kuyruk yöneticisine ilişkin öznitelikler” sayfa 785](#)

## **C çağırma**

```
MQSET (Hconn, Hobj, SelectorCount, Selectors, IntAttrCount, IntAttrs,  
CharAttrLength, CharAttrs, &CompCode, &Reason);
```

Değiştiregeleri aşağıdaki gibi bildirin:

```
MQHCONN  Hconn;           /* Connection handle */  
MQHOBJ   Hobj;           /* Object handle */  
MQLONG   SelectorCount; /* Count of selectors */  
MQLONG   Selectors[n];  /* Array of attribute selectors */  
MQLONG   IntAttrCount;  /* Count of integer attributes */  
MQLONG   IntAttrs[n];   /* Array of integer attributes */  
MQLONG   CharAttrLength; /* Length of character attributes buffer */  
MQCHAR   CharAttrs[n];  /* Character attributes */  
MQLONG   CompCode;      /* Completion code */  
MQLONG   Reason;        /* Reason code qualifying CompCode */
```

## Cobol çağırma

```
CALL 'MQSET' USING HCONN, HOBJ, SELECTORCOUNT, SELECTORS-TABLE,  
                  INTATTRCOUNT, INTATTRS-TABLE, CHARATTRLENGTH,  
                  CHARATTRS, COMPCODE, REASON.
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
** Connection handle  
01 HCONN          PIC S9(9) BINARY.  
** Object handle  
01 HOBJ          PIC S9(9) BINARY.  
** Count of selectors  
01 SELECTORCOUNT PIC S9(9) BINARY.  
** Array of attribute selectors  
01 SELECTORS-TABLE.  
  02 SELECTORS    PIC S9(9) BINARY OCCURS n TIMES.  
** Count of integer attributes  
01 INTATTRCOUNT PIC S9(9) BINARY.  
** Array of integer attributes  
01 INTATTRS-TABLE.  
  02 INTATTRS    PIC S9(9) BINARY OCCURS n TIMES.  
** Length of character attributes buffer  
01 CHARATTRLENGTH PIC S9(9) BINARY.  
** Character attributes  
01 CHARATTRS      PIC X(n).  
** Completion code  
01 COMPCODE       PIC S9(9) BINARY.  
** Reason code qualifying COMPCODE  
01 REASON         PIC S9(9) BINARY.
```

## PL/I çağırımı

```
call MQSET (Hconn, Hobj, SelectorCount, Selectors, IntAttrCount,  
            IntAttrs, CharAttrLength, CharAttrs, CompCode, Reason);
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
dcl Hconn          fixed bin(31); /* Connection handle */  
dcl Hobj          fixed bin(31); /* Object handle */  
dcl SelectorCount fixed bin(31); /* Count of selectors */  
dcl Selectors(n)  fixed bin(31); /* Array of attribute selectors */  
dcl IntAttrCount  fixed bin(31); /* Count of integer attributes */  
dcl IntAttrs(n)   fixed bin(31); /* Array of integer attributes */  
dcl CharAttrLength fixed bin(31); /* Length of character attributes  
buffer */  
dcl CharAttrs     char(n); /* Character attributes */  
dcl CompCode      fixed bin(31); /* Completion code */  
dcl Reason        fixed bin(31); /* Reason code qualifying  
CompCode */
```

## High Level Assembler çağırısı

```
CALL MQSET, (HCONN, HOBJ, SELECTORCOUNT, SELECTORS, INTATTRCOUNT, X  
            INTATTRS, CHARATTRLENGTH, CHARATTRS, COMPCODE, REASON)
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
HCONN          DS F      Connection handle  
HOBJ           DS F      Object handle  
SELECTORCOUNT DS F      Count of selectors  
SELECTORS      DS (n)F   Array of attribute selectors  
INTATTRCOUNT DS F      Count of integer attributes  
INTATTRS      DS (n)F   Array of integer attributes  
CHARATTRLENGTH DS F      Length of character attributes buffer  
CHARATTRS     DS CL(n)  Character attributes
```

COMPCODE	DS	F	Completion code
REASON	DS	F	Reason code qualifying COMPCODE

## Visual Basic çağırısı

```
MQSET Hconn, Hobj, SelectorCount, Selectors, IntAttrCount, IntAttrs,
CharAttrLength, CharAttrs, CompCode, Reason
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
Dim Hconn          As Long   'Connection handle'
Dim Hobj           As Long   'Object handle'
Dim SelectorCount  As Long   'Count of selectors'
Dim Selectors      As Long   'Array of attribute selectors'
Dim IntAttrCount   As Long   'Count of integer attributes'
Dim IntAttrs       As Long   'Array of integer attributes'
Dim CharAttrLength As Long   'Length of character attributes buffer'
Dim CharAttrs      As String 'Character attributes'
Dim CompCode       As Long   'Completion code'
Dim Reason         As Long   'Reason code qualifying CompCode'
```

## MQSETMP-İleti özelliğini ayarla

İleti tutamacının bir özelliğini ayarlamak ya da değiştirmek için MQSETMP çağırısını kullanın.

### Sözdizimi

MQSETMP (*Hconn, Hmsg, SetPropOpts, Name, PropDesc, Type, ValueLength, Value, Compcode, Reason*)

### Parametreler

#### Hconn

Tip: MQHCONN-input

Bu tanıtıcı, kuyruk yöneticisiyle olan bağlantıyı gösterir.

Değerin, **Hmsg** parametresinde belirtilen ileti tanıtıcısını yaratmak için kullanılan bağlantı tanıtıcısı ile eşleşmesi gerekir. İleti tanıtıcısı MQHC\_UNASSOCIATED\_HCONN kullanılarak yaratıldıysa, ileti tutamacının bir özelliği ayarlanırken iş parçasısında geçerli bir bağlantı oluşturulmalıdır; tersi durumda çağrı, MQRC\_CONNECTION\_BROKEN neden koduyla başarısız olur.

#### İtt

Tip: MQHMSG-giriş

Bu, değiştirilecek ileti işidir. Değer önceki bir MQCRTMH çağırısı tarafından döndürüldü.

#### SetPropOpts

Tip: MQSMPO-giriş

İleti özelliklerinin nasıl ayarlanacağını denetler.

Bu yapı, uygulamaların ileti özelliklerinin nasıl ayarlanmasını denetleyen seçenekleri belirlemesine olanak sağlar. Yapı, MQSETMP çağırısında bir giriş değiştirişidir. Ek bilgi için [MQSMPO](#) başlıklı konuya bakın.

#### Ad

Tip: MQCHARV-input

Ayarlanacak özelliğin adı.

Özellik adlarının kullanımına ilişkin ek bilgi için [Özellik adları](#) ve [Özellik adı kısıtlamaları](#) başlıklı konuya bakın.

#### PropDesc

Tip: MQPD-giriş/çıkış

Bu yapı, aşağıdakiler de içinde olmak üzere bir özelliğe ilişkin öznitelikleri tanımlamak için kullanılır:

- özellik desteklenmiyorsa ne olur
- özelliğin ait olduğu ileti bağılamı
- malın akındığı gibi kopyalandığı mesajlar

Bu yapıyla ilgili ek bilgi için [MQPD](#) başlıklı konuya bakın.

#### Tip

Tip: MQHOME-girişi

Ayarlanmakta olan özelliğin veri tipi. Aşağıdakilerden biri olabilir:

#### **MQTYPE\_BOOLEAN**

Bir Boole. *ValueLength* değeri 4 olmalıdır.

#### **MQTYPE\_BYTE\_STRING**

Bayt dizilimi. *ValueLength* değeri sıfır ya da daha büyük olmalıdır.

#### **MQTYPE\_INT8**

8 bitlik işaretli tamsayı. *ValueLength* değeri 1 olmalıdır.

#### **MQTYPE\_INT16**

16 bitlik bir işaretli tamsayı. *ValueLength* değeri 2 olmalıdır.

#### **MQTYPE\_INT32**

32 bitlik işaretli tamsayı. *ValueLength* değeri 4 olmalıdır.

#### **MQTYPE\_INT64**

Bir 64 bit imzalı tamsayı. *ValueLength* değeri 8 olmalıdır.

#### **MQTYPE\_FLOAT32**

32 bitlik kayan noktalı sayı. *ValueLength* değeri 4 olmalıdır.

Not: Bu tip, IBM COBOL for z/OSkullanan uygulamalarla desteklenmez.

#### **MQTYPE\_FLOAT64**

64 bitlik kayan noktalı sayı. *ValueLength* değeri 8 olmalıdır.

Not: Bu tip, IBM COBOL for z/OSkullanan uygulamalarla desteklenmez.

#### **MQTYPE\_STRING**

Bir karakter dizgisi. *ValueLength* değeri sıfır ya da daha büyük olmalı ya da MQVL\_NULL\_TERMINATED özel değeri olmalıdır.

#### **MQTYPE\_NULL**

Özellik var, ancak boş değere sahip. *ValueLength* değeri sıfır olmalıdır.

#### ValueLength

Tip: MQHOME-girişi

*Value* (Değer) parametresindeki özellik değerinin bayt cinsinden uzunluğu. Sıfır değeri yalnızca boş değerler için ya da dizgiler ya da byte dizgileri için geçerlidir. Sıfır değeri, özelliğin var olduğunu, ancak değer karakter ya da byte içermediği anlamına gelir.

*Type* değiştirgesi MQTYPE\_STRING kümesine sahipse, değer sıfırdan büyük ya da bu değere eşit ya da şu özel değere eşit olmalıdır:

#### **MQVL\_NULL\_TERMINATED**

Değer, dizgide karşılaşılan ilk boş değerle sınırlanır. Boş değer, dizginin bir parçası olarak içerilmedi. MQTYPE\_STRING de belirlenmezse, bu değer geçersiz olur.

Not: MQVL\_NULL\_TERMINATED ayarlıysa, bir dizgiyi sonlandırmak için kullanılan boş karakter, Değer karakter kümesinden boş değerde olur.

#### Değer

Tip: MQBYTExValueUzunluk-giriş

Ayarlanacak özelliğin değeri. Arabellek, değerdeki verilerin niteine uygun bir sınırdan hizalanmalıdır.

C programlama dilinde, parametre bir işaretçi olarak bildirilir; parametrenin herhangi bir tipinin adresi, parametre olarak belirtilebilir.

*ValueLength* sıfırsa, *Value* (Değer) ifadesine başvurulmaz. Bu durumda, C ya da System/390 çevirici programlarında yazılan programlarla geçirilen parametre adresi boş değerli olabilir.

### **CompCode**

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Tamamlanma kodu; bu kod aşağıdakilerden biridir:

#### **MQCC\_OK**

İşlem başarıyla tamamlandı.

#### **MQCC\_FAILED**

Arama başarısız oldu.

### **Neden**

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Neden kodu niteleyici *CompCode*.

*CompCode* , MQCC\_OK ise:

#### **MQRC\_NONE**

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* , MQCC\_UYARI ise:

#### **MQRC\_RFH\_FORMAT\_ERROR**

(2421, X'0975 ') Özellikler içeren bir MQRFH2 klasörü ayrıştırılamadı.

*CompCode* MQCC\_FAILED ise:

#### **MQRC\_ADAPTER\_NOT\_AVAM**

(2204, X'089C') Bağdaştırıcı kullanılamıyor.

#### **MQRC\_ADAPTER\_SERV\_LOAD\_ERROR**

(2130, X'852 ') Bağdaştırıcı hizmeti modülü yüklenemiyor.

#### **MQRC\_ASID\_MISSMACH**

(2157, X'86D') Birincil ve ana sunucu ASID değerleri farklı.

#### **MQRC\_BUFFER\_ERROR**

(2004, X'07D4') Değer parametresi geçerli değil.

#### **MQRC\_BUFFER\_LENGTH\_ERROR**

(2005, X'07D5') Değer uzunluğu parametresi geçerli değil.

#### **MQRC\_CALL\_IN\_PROGRESS**

(2219, X'08AB') Önceki arama tamamlanmadan önce girilen MQI çağırısı.

#### **MQRC\_HMSG\_ERROR**

(2460, X'099C') İleti tanıtıcısı göstergesi geçerli değil.

#### **MQRC\_MSG\_HANDLE\_IN\_USE**

(2499, X'09C3') İleti tanıtıcısı zaten kullanılıyor.

#### **MQRC\_OPTIONS\_ERROR**

(2046, X'07FE') Seçenekler geçerli değil ya da tutarlı değil.

#### **MQRC\_PD\_ERROR**

(2482, X'09B2') Özellik tanımlayıcı yapısı geçerli değil.

#### **MQRC\_ÖZELLİĞİ \_**

(2442, X'098A') Özellik adı geçersiz.

#### **MQRC\_ÖZELLİĞİ TYPE\_**

(2473, X'09A9') Özellik veri tipi geçersiz.

#### **MQRC\_PROP\_NUMBER\_FORMAT\_ERROR**

(2472, X'09A8') Değer verilerinde sayı biçimi hatası saptandı.

### **MQRC\_SMPO\_ERROR**

(2463, X'099F') İleti özelliği seçenekleri yapısını ayarlama geçerli değil.

### **MQRC\_SOURCE\_CCSID\_ERROR**

(2111, X'083F') Özellik adı kodlanmış karakter takımı tanıtıcısı geçerli değil.

### **MQRC\_STORAGE\_NOT\_AVAM**

(2071, X'817 ') Kullanılabilir bellek yetersiz.

### **MQRC\_UNEXPECTED\_ERROR**

(2195, X'893 ') Beklenmeyen bir hata oluştu.

Bu kodlarla ilgili ayrıntılı bilgi için [İletiler ve neden kodları](#) başlıklı konuya bakın.

## **C çağırma**

```
MQSETMP (Hconn, Hmsg, &SetPropOpts, &Name, &PropDesc, Type,  
ValueLength, &Value, &CompCode, &Reason);
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
MQHCONN  Hconn;          /* Connection handle */  
MQHMSG   Hmsg;          /* Message handle */  
MQSMPO   SetPropOpts; /* Options that control the action of MQSETMP */  
MQCHARV  Name;         /* Property name */  
MQPD     PropDesc;     /* Property descriptor */  
MQLONG   Type;         /* Property data type */  
MQLONG   ValueLength; /* Length of property value in Value */  
MQBYTE   Value[n];     /* Property value */  
MQLONG   CompCode;    /* Completion code */  
MQLONG   Reason;      /* Reason code qualifying CompCode */
```

## **Cobol çağırma**

```
CALL 'MQSETMP' USING HCONN, HMSG, SETMSGOPTS, NAME, PROPDSC, TYPE,  
VALUELENGTH, VALUE, COMPCODE, REASON.
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
** Connection handle  
01 HCONN      PIC S9(9) BINARY.  
** Message handle  
01 HMSG      PIC S9(18) BINARY.  
** Options that control the action of MQSETMP  
01 SETMSGOPTS.  
   COPY CMQSMPOV.  
** Property name  
01 NAME  
   COPY CMQCHRVV.  
** Property descriptor  
01 PROPDSC.  
   COPY CMQPDV.  
** Property data type  
01 TYPE      PIC S9(9) BINARY.  
** Length of property value in VALUE  
01 VALUELENGTH PIC S9(9) BINARY.  
** Property value  
01 VALUE     PIC X(n).  
** Completion code  
01 COMPCODE  PIC S9(9) BINARY.  
** Reason code qualifying COMPCODE  
01 REASON    PIC S9(9) BINARY.
```

## PL/I çağırımı

```
call MQSETMP (Hconn, Hmsg, SetPropOpts, Name, PropDesc, Type, ValueLength,  
Value, CompCode, Reason);
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
dcl Hconn          fixed bin(31); /* Connection handle */  
dcl Hmsg          fixed bin(63); /* Message handle */  
dcl SetPropOpts   like MQSMP0; /* Options that control the action of MQSETMP */  
dcl Name          like MQCHARV; /* Property name */  
dcl PropDesc      like MQPDA; /* Property descriptor */  
dcl Type          fixed bin(31); /* Property data type */  
dcl ValueLength   fixed bin(31); /* Length of property value in Value */  
dcl Value         char(n); /* Property value */  
dcl CompCode      fixed bin(31); /* Completion code */  
dcl Reason        fixed bin(31); /* Reason code qualifying CompCode */
```

## High Level Assembler çağırısı

```
CALL MQSETMP, (HCONN,HMSG,SETMSGHOPTS,NAME,PROPDSC,TYPE,VALUELENGTH,  
VALUE,COMPCODE,REASON)
```

Değiştirgeleri aşağıdaki gibi bildirin:

HCONN	DS	F	Connection handle
HMSG	DS	D	Message handle
SETMSGOPTS	CMQSMPOA	,	Options that control the action of MQSETMP
NAME	CMQCHRVA	,	Property name
PROPDSC	CMQPDA	,	Property descriptor
TYPE	DS	F	Property data type
VALUELENGTH	DS	F	Length of property value in VALUE
VALUE	DS	CL(n)	Property value
COMPCODE	DS	F	Completion code
REASON	DS	F	Reason code qualifying COMPCODE

## MQSTAT-Durum bilgilerini al

Durum bilgilerini almak için MQSTAT çağırısını kullanın. Döndürülen durum bilgisi tipi, çağırıda belirtilen Tip değeri tarafından belirlenir.

### Sözdizimi

MQSTAT (*Hconn*, *Tip*, *Stat*, *Compcode*, *Reason*)

### Parametreler

#### Hconn

Tip: MQHCONN-input

Bu tanırtıcı, kuyruk yöneticisiyle olan bağlantıyı gösterir. The value of *Hconn* was returned by a previous MQCONN or MQCONNX call.

On z/OS for CICS applications the MQCONN call can be omitted, and the following value specified for *Hconn* :

#### MQHC\_DEF\_HCONN

Varsayılan bağlantı tanırtıcısı.

#### Tip

Tip: MQHOME-girişi

İstenilen durum bilgilerinin tipi. > geçerli değerler şunlardır:

**MQSTAT\_TYPE\_ASYNC\_ERROR**

Önceki zamanuyumsuz put işlemleriyle ilgili bilgi döndürür.

**MQSTAT\_TYPE\_RECONNECTION**

Yeniden bağlantıyla ilgili bilgi döndürür. Bağlantı yeniden bağlanmaya devam ederse ya da yeniden bağlantı kurulamazsa, bu bilgiler bağlantının yeniden bağlanmasına neden olan hatayı açıklar.

Bu değer yalnızca istemci bağlantıları için geçerlidir. Diğer bağlantı tipleri için çağrı, **MQRC\_ENVIRONMENT\_ERROR** neden koduyla başarısız olur.

**MQSTAT\_TYPE\_RECONNECTION\_ERROR**

Yeniden bağlanmayla ilgili önceki bir hatayla ilgili bilgileri döndürür. Bağlantı yeniden bağlanmayı başaramadıysa, bilgi yeniden bağlanmanın başarısız olmasına neden olan hatayı açıklar.

Bu değer yalnızca istemci bağlantıları için geçerlidir. Diğer bağlantı tiplerinde, çağrı neden kodu **MQRC\_ENVIRONMENT\_ERROR** ile başarısız olur.

**Stat**

Tip: MQSTS-giriş/çıkış

Durum bilgisi yapısı. Ayrıntılar için bkz. [“MQSTS-Durum raporlama yapısı” sayfa 581.](#)

**CompCode**

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Tamamlanma kodu; bu kod aşağıdakilerden biridir:

**MQCC\_OK**

İşlem başarıyla tamamlandı.

**MQCC\_FAILED**

Arama başarısız oldu.

**Neden**

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Neden kodu niteleyici *CompCode*.

*CompCode* MQCC\_OK ise:

**MQRC\_NONE**

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_FAILED ise:

**MQRC\_API\_EXIT\_ERROR**

(2374, X' 946 ') API çıkışı başarısız oldu

**MQRC\_API\_EXIT\_LOAD\_ERROR**

(2183, X'887 ') API çıkışı yüklenemiyor.

**MQRC\_CALL\_IN\_PROGRESS**

(2219, X'8AB') Önceki arama tamamlanmadan önce girilen MQI çağrısı.

**MQRC\_CONNECTION\_BROKEN**

(2009, X'7D9') Kuyruk yöneticisine yönelik bağlantı kaybedildi.

**MQRC\_CONNECTION\_DURATION**

(2203, X'89B') Bağlantı sona erdiriliyor.

**MQRC\_FUNCTION\_NOT\_SUPPORTED**

(2298, X'8FA') İstenen işlev geçerli ortamda yok.

**MQRC\_HCONN\_ERROR**

(2018, X'7E2') Bağlantı tanıtıcısı geçerli değil.

**MQRC\_Q\_MGR\_STOPPING**

(2162, X'872')-Kuyruk yöneticisi durduruluyor



**MQRC\_RESOURCE\_SORUNU**

(2102, X'836 ') Yetersiz sistem kaynakları var.

**MQRC\_STAT\_TYPE\_ERROR**

(2430, X'97E' MQSTAT tipi ile ilgili hata

**MQRC\_STORAGE\_NOT\_AVAM**

(2071, X'817 ') Kullanılabilir bellek yetersiz.

**MQRC\_STS\_ERROR**

(2426, X'97A') MQSTS yapısıyla ilgili hata

**MQRC\_UNEXPECTED\_ERROR**

(2195, X'893 ') Beklenmeyen bir hata oluştu.

Bu kodlarla ilgili ayrıntılı bilgi için [İletiler ve neden kodları](#) başlıklı konuya bakın.

**Kullanım notları**

1. MQSTAT\_TYPE\_ASYNC\_ERROR tipini belirten bir MQSTAT çağrısı, önceki zamanuyumsuz MQPUT ve MQPUT1 işlemleriyle ilgili bilgileri döndürür. The MQSTS structure passed back on return from the MQSTAT call contains the first recorded asynchronous warning or error information for that connection. Daha fazla hata ya da uyarı ilk olarak izlenirse, bunlar olağan durumda bu değerleri değiştirmez. However, if an error occurs with a completion code of MQCC\_WARNING, a subsequent failure with a completion code of MQCC\_FAILED is returned instead.
2. If no errors have occurred since the connection was established or since the last call to MQSTAT then a CompCode of MQCC\_OK and Reason of MQRC\_NONE are returned in the MQSTS structure.
3. Bağlantı tanıtıcısı altında işlenen zamanuyumsuz çağrılarının sayısı, üç sayaç alanı yoluyla döndürülür; PutSuccessCount, PutWarningCount ve PutFailureCount. Bu sayaçlar, kuyruk yöneticisi tarafından her zamanuyumsuz işlemin başarıyla işlendiği, bir uyarıya sahip ya da başarısız olduğunda (dağıtım listesi başına bir dağıtım listesi için bir kez dağıtım listesi başına bir kez konduğunda) bir uyarı ya da başarısız olduğunda, bu sayaçlar artırılır (not, dağıtım listesi başına bir kez dağıtım listesi sayım işlemi için bir kez konduğunda). A counter is not incremented beyond the maximum positive value AMQ\_LONG\_MAX.
4. Önceki hata bilgilerinde ya da sıfırlanmakta olan önceki herhangi bir hata bilgisi ya da sayıyla MQSTAT sonuçlarına başarılı bir çağrı.
5. MQSTAT davranışı, sağladığınız **MQSTAT Type** parametresinin değerine bağlıdır.
6. **MQSTAT\_TYPE\_ASYNC\_ERROR**
  - a. MQSTAT\_TYPE\_ASYNC\_ERROR tipini belirten bir MQSTAT çağrısı, önceki zamanuyumsuz MQPUT ve MQPUT1 işlemleriyle ilgili bilgileri döndürür. The MQSTS structure passed back on return from the MQSTAT call contains the first recorded asynchronous warning or error information for that connection. Daha fazla hata ya da uyarı ilk olarak izlenirse, bunlar olağan durumda bu değerleri değiştirmez. However, if an error occurs with a completion code of MQCC\_WARNING, a subsequent failure with a completion code of MQCC\_FAILED is returned instead.
  - b. If no errors have occurred since the connection was established or since the last call to MQSTAT then a CompCode of MQCC\_OK and Reason of MQRC\_NONE are returned in the MQSTS structure.
  - c. Bağlantı tanıtıcısı altında işlenen zamanuyumsuz çağrılarının sayısı, üç sayaç alanı yoluyla döndürülür; PutSuccessCount, PutWarningCount ve PutFailureCount. Bu sayaçlar, kuyruk yöneticisi tarafından her zamanuyumsuz işlemin başarıyla işlendiği, bir uyarıya sahip ya da başarısız olduğunda (dağıtım listesi başına bir dağıtım listesi için bir kez dağıtım listesi başına bir kez konduğunda) bir uyarı ya da başarısız olduğunda, bu sayaçlar artırılır (not, dağıtım listesi başına bir kez dağıtım listesi sayım işlemi için bir kez konduğunda). A counter is not incremented beyond the maximum positive value AMQ\_LONG\_MAX.
  - d. Önceki hata bilgilerinde ya da sıfırlanmakta olan önceki herhangi bir hata bilgisi ya da sayıyla MQSTAT sonuçlarına başarılı bir çağrı.

## MQSTAT\_TYPE\_RECONNECTION

Suppose you call MQSTAT with Type set to MQSTAT\_TYPE\_RECONNECTION inside an event handler during reconnection. Bu örnekleri göz önünde bulundurun.

### İstemci yeniden bağlantı kurmayı deniyor ya da yeniden bağlantı kurulamamış.

CompCode in the MQSTS structure is MQCC\_FAILED and Reason might be either MQRC\_CONNECTION\_BROKEN or MQRC\_Q\_MGR QUIESCING. ObjectType , MQOT\_Q\_MGR, ObjectName kuyruk yöneticisi adıdır ve ObjectQMgrName boş.

### İstemci yeniden bağlantıyı başarıyla tamamladı ya da hiçbir zaman bağlantısı kesilmedi.

MQSTS yapısında CompCode , MQCC\_OK ve Reason ' dir: MQRC\_NONE

Subsequent calls to MQSTAT return the same results.

## MQSTAT\_TYPE\_RECONNECTION\_ERROR

Suppose you call MQSTAT with Type set to MQSTAT\_TYPE\_RECONNECTION\_ERROR in response to receiving MQRC\_RECONNECT\_FAILED to an MQI call. Bu örnekleri göz önünde bulundurun.

### Farklı bir kuyruk yöneticisinde yeniden bağlantı sırasında bir kuyruk yeniden açılırken bir yetki hatası oluştu.

CompCode in the MQSTS structure is MQCC\_FAILED and Reason is the reason that the reconnection failed, such as MQRC\_NOT\_AUTHORIZED. ObjectType is the type of object that caused the problem, such as MQOT\_QUEUE, ObjectName is the name of the queue and ObjectQMgrName the name of the queue manager owning the queue.

### Yeniden bağlanma sırasında yuva bağlantısı hatası oluştu.

CompCode in the MQSTS structure is MQCC\_FAILED and Reason is the reason that the reconnection failed, such as MQRC\_HOST\_NOT\_AVAILABLE. ObjectType , MQOT\_Q\_MGR, ObjectName kuyruk yöneticisi adıdır ve ObjectQMgrName boş.

Subsequent calls to MQSTAT return the same results.

## C çağırma

```
MQSTAT (Hconn, StatType, &Stat, &CompCode, &Reason);
```

Değişirgeleri aşağıdaki gibi bildirin:

```
MQHCONN Hconn;          /* Connection Handle */
MQLONG StatType;        /* Status type */
MQSTS Stat;             /* Status information structure */
MQLONG CompCode;        /* Completion code */
MQLONG Reason;          /* Reason code qualifying CompCode */
```

## Cobol çağırma

```
CALL 'MQSTAT' USING HCONN, STATTYPE, STAT, COMPCODE, REASON.
```

Değişirgeleri aşağıdaki gibi bildirin:

```
**      Connection handle
01     HCONN      PIC S9(9)      BINARY.
**      Status type
01     STATTYPE  PIC S9(9)      BINARY.
**      Status information
01     STAT.
      COPY CMQSTSV.
**      Completion code
01     COMPCODE  PIC S9(9)      BINARY.
**      Reason code qualifying COMPCODE
01     REASON    PIC S9(9)      BINARY.
```

## PL/I çağırımı

```
call MQSTAT (Hconn, StatType, Stat, Compcode, Reason);
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
dcl Hconn      fixed bin(31); /* Connection handle */
dcl StatType  fixed bin(31); /* Status type */
dcl Stat      like MQSTS;    /* Status information structure */
dcl CompCode  fixed bin(31); /* Completion code */
dcl Reason    fixed bin(31); /* Reason code qualifying CompCode */
```

## System/390 Çevirici çağırma

```
CALL MQSTAT, (HCONN, STATTYPE, STAT, COMPCODE, REASON)
```

Değiştirgeleri aşağıdaki gibi bildirin:

HCONN	DS	F	Connection handle
STATTYPE	DS	F	Status type
STAT	CMQSTSA,		Status information structure
COMPCODE	DS	F	Completion code
REASON	DS	F	Reason code qualifying COMPCODE

## MQSUB-Abonelikleri kaydet

Uygulama aboneliğini belirli bir konuya kaydettirmek için MQSUB çağırısını kullanın.

### Sözdizimi

MQSUB (*Hconn*, *SubDesc*, *Hobj*, *Hsub*, *Compcode*, *Reason*)

### Parametreler

#### Hconn

Tip: MQHCONN-input

Bu tanıtıcı, kuyruk yöneticisiyle olan bağlantıyı gösterir. The value of *Hconn* was returned by a previous MQCONN or MQCONNX call.

On z/OS for CICS applications the MQCONN call can be omitted, and the following value specified for *Hconn* :

#### MQHC\_DEF\_HCONN

Varsayılan bağlantı tanıtıcısı.

#### SubDesc

Tip: MQSD-giriş/çıkış

Bu yapı, uygulama tarafından kaydedilmekte olan kullanıdaki nesneyi tanımlayan bir yapıdır. Ek bilgi için [“MQSD-Abonelik tanımlayıcısı” sayfa 557](#) başlıklı konuya bakın.

#### Hobj

Tip: MQHOBJ-giriş/çıkış

Bu tanıtıcı, bu aboneliğe gönderilen iletileri almak için kurulmuş olan erişimi gösterir. Bu iletiler belirli bir kuyrukta saklanabilir ya da kuyruk yöneticisi, belirli bir kuyruk kullanılmadan depolamalarını yönetebilir.

Belirli bir kuyruğu kullanmak için, aboneliği abonelik oluşturulduğunda aboneliğe ilişkilendirmeniz gerekir. Bunu iki şekilde yapabilirsiniz:

- DEFINE SUB MQSC komutunu kullanarak ve bu komutu bir kuyruk nesnesinin adıyla birlikte sağlamış olabilir.
- MQSO\_CREATE ile MQSUB çağrılırken bu tanıtıcı değeri sağlanarak

Bu tanıtıcı çağrıya bir giriş parametresi olarak sağlandıysa, aşağıdaki seçeneklerden en az birini kullanarak, kuyruğun önceki bir MQOPEN çağrısından döndürülen geçerli bir nesne tanıtıcısı olmalıdır:

- MQOO\_INPUT\_\*
- MQOO\_BROWSE
- MQOO\_OUTPUT (kuyruk uzak bir kuyruksa)

Durum böyle değilse, çağrı MQRC\_HOBJ\_ERROR ile başarısız olur. Bir konu nesnesine çözülen bir diğer ad kuyruğunda nesne tanıtıcısı olamaz. Böyle bir durumda, çağrı MQRC\_HOBJ\_ERROR ile başarısız olur.

Kuyruk yöneticisi, bu aboneliğe gönderilen iletilerin depolarını yönetiyorsa, bu, aboneliği yarattığınızda, MQSO\_MANAGED seçeneğini kullanarak bu değer ayarlanmalıdır. Daha sonra kuyruk yöneticisi bu tanıtıcıyı çağrıya çıkış parametresi olarak döndürür. Döndürülen tanıtıcı, yönetilen tanıtıcı olarak bilinir. MQHO\_NONE belirlendiysse, ancak MQSO\_YÖNETİCİ belirlenmezse, çağrı MQRC\_HOBJ\_ERROR ile başarısız olur.

Kuyruk yöneticisi tarafından yönetilen bir tanıtıcı döndürdüğünde, bu değeri bir MQGET ya da MQCB çağrısında kullanarak ya da göz atma seçeneklerinde, bir MQINQ çağrısında ya da MQCLOSE üzerinde kullanabilirsiniz. Bunu MQPUT, MQSUB, MQSET üzerinde kullanamazsınız; bunu gerçekleştirme girişiminde MQRC\_NOT\_OPEN\_FOR\_OUTPUT, MQRC\_HOBJ\_ERROR ya da MQRC\_NOT\_OPEN\_FOR\_SET ile başarısız başarısız olun.

Bu abonelik MQSD yapısındaki MQSO\_RESUME seçeneği kullanılarak sürdürülüyorsa, MQSO\_YANATED olarak MQHO\_NONE değerine ayarlanarak, bu parametredeki uygulamaya geri döndürülebilirsiniz. Bu işlemi, aboneliğin yönetilen bir tanıtıcı kullanıp kullanmadığını ve bu komutta tanımlanan abonelik kuyruğunun tanıtıcısı ile DEFINE SUB ile yaratılan abonelikleri sağlamak için yararlı olup olmayacağını da yapabilirsiniz. Yönetimle ilgili olarak bir aboneliğin sürdürüldüğü durumlarda, kuyruk MQOO\_INPUT\_AS\_Q\_DEF ve MQOO\_BROWSE ile açılır. Diğer seçenekleri belirtmeniz gerekirse, uygulamanın abonelik kuyruğunu belirttik olarak açması ve çağrıdaki nesne tanıtıcısını belirtmesi gerekir. Kuyruğun açılması sırasında bir sorun varsa, çağrı MQRC\_INVALID\_DESTIND ile başarısız olur. *Hobj* sağlandıysa, özgün MQSUB çağrısındaki *Hobj* ile eşdeğer olmalıdır. Bu, bir MQOPER çağrısından döndürülen bir nesne tanıtıcı değeri sağlanırsa, tanıtıcı, önceden kullanılanlarla aynı kuyruğa olmalıdır. Aynı kuyruk değilse, çağrı MQRC\_HOBJ\_ERROR ile başarısız olur.

Bu abonelik MQSD yapısındaki MQSO ALTER seçeneği kullanılarak değiştiriliyorsa, farklı bir *Hobj* sağlanabilir. Kuyruğa teslim edilen ve daha önce bu parametre aracılığıyla tanımlanan yayınlar o kuyruğun üzerinde kalır ve **Hobj** parametresi artık farklı bir kuyruğu gösteriyorsa, bu iletileri almak için uygulamanın sorumluluğundadır.

Çizelge 556. Çeşitli abonelik seçenekleri ile hobj kullanılması		
Seçenekler	Hobj	Tanım
MQSO_CREATE + MQSO_MANAGED	Girişte yoksayıldı	Kuyruk yöneticisi tarafından yönetilen iletilerin depolanarak bir abonelik yaratır
MQSO_CREATE	Geçerli bir nesne tanıtıcısı	İletiler için hedef olarak belirli bir kuyruk sağlayan bir abonelik oluşturur.
MQSO_RESUME	MQHO_NONE	Yönetilen ya da olmayan önceden yaratılmış bir aboneliği sürdürür ve kuyruk yöneticisi, uygulama tarafından kullanılmak üzere nesne tanıtıcısını döndürmesini sağlar.

Çizelge 556. Çeşitli abonelik seçenekleri ile hobj kullanılması (devamı var)		
Seçenekler	Hobj	Tanım
MQSO_RESUME	Geçerli, eşleşen, nesne tanıtıcısı	İletilere ilişkin hedef olarak belirli bir kuyruğu kullanan, önceden oluşturulmuş bir aboneliği sürdürür ve belirli açık seçeneklerle bir nesne tanıtıcısı kullanır.
MQSO_ALTER + MQSO_MANAGED	MQHO_NONE	Önceden belirli bir kuyruk kullanan var olan bir aboneliği değiştirir, bu nedenle artık yönetilen bir aboneliktir. Hedef sınıfı (yönetilen ya da değil) değiştirilemez.
MQSO_ALTER	Geçerli bir nesne tanıtıcısı	Var olan bir aboneliği, yönetilen olsun ya da olmasın, artık belirli bir kuyruğu kullansın diye değiştirir. MQSO_MANAGED seçeneği kullanılmadığında, sağlanan kuyruk değiştirilebilir, ancak hedef sınıfı (yönetilen ya da değil) değiştirilemez.

Sağlanan ya da iade edilen *Hobj*, bu aboneliğe gönderilen yayın iletilerini almak isteyen sonraki MQGET ya da MQCB çağrılarında belirtilmelidir.

The *Hobj* handle is no longer valid when the MQCLOSE call is issued on it, or when the unit of processing that defines the scope of the handle terminates (until the application disconnects). Döndürülen nesne tanıtıcısı kapsamı, çağrıda belirtilen bağlantı tanıtıcısı ile aynı. Tutamaç kapsamına ilişkin bilgi için Hconn (MQHCONN)-output başlıklı konuya bakın. *Hobj* tutamacından bir MQCLOSE, *Hsub* tanıtıcısını etkilemez.

### Hsub

Tip: MQHOBJ-çıkış

Bu tanıtıcı, yapılan aboneliği temsil eder. Diğer iki işlem için de kullanılabilir:

- Sonraki bir MQSUBRQ çağrısında, abonelik yapılırken MQSO\_PUBLICATIONS\_ON\_REQUEST seçeneği kullanıldığında yayınların gönderilmesini istemek için kullanılabilir.
- Bu, yapılan bir MQCLOSE aboneliğinde, yapılan aboneliği kaldırmak için kullanılabilir. The *Hsub* handle ceases to be valid when the MQCLOSE call is issued, or when the unit of processing that defines the scope of the handle terminates. Döndürülen nesne tanıtıcısı kapsamı, çağrıda belirtilen bağlantı tanıtıcısı ile aynı. *Hsub* tutamacından bir MQCLOSE, *Hobj* tanıtıcısını etkilemez.

Bu tanıtıcı bir MQGET ya da MQCB çağrısına geçirilemez. **Hobj** parametresini kullanmanız gerekir. Bu tanıtıcıyı, MQCLOSE ya da MQSUBRQ dışında herhangi bir IBM MQ çağrısı üzerinde kullanamazsınız. Bu tanıtıcıyı diğer herhangi bir IBM MQ çağrısı sonuçlarıyla geçirirken MQRC\_HOBJ\_ERROR içinde yer alan sonuçlar.

### CompCode

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Tamamlanma kodu; bu kod aşağıdakilerden biridir:

#### MQCC\_OK

Başarıyla tamamlandı

#### MQCC\_UYARI

Uyarı (kısmi tamamlama)

#### MQCC\_FAILED

Arama başarısız oldu

## Neden

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Neden kodu niteleyici *CompCode*.

*CompCode* MQCC\_OK ise, neden kodu şu şekildedir:

### **MQRC\_NONE**

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_FAILED ise, neden kodu aşağıdakilerden biri olabilir:

### **MQRC\_CLUSTER\_RESOLUTION\_ERROR**

(2189, X'88D') Küme adının çözülmesi başarısız oldu.

### **MQRC\_DURABILITY\_NOT\_ALLOWALIZE**

2436 (X'0984 ') MQSO\_DAYİD seçeneğini kullanan bir MQSUB çağrısı başarısız oldu.

### **MQRC\_FUNCTION\_NOT\_SUPPORUL**

2298 (X'08FA') İstenen işlev, yürürlükteki ortamda yok.

### **MQRC\_HOBJ\_ERROR**

2019 (X'07E3') Nesne tanıtıcısı Hobj geçerli değil.

### **MQRC\_IDENTITY\_MISMATCH**

2434 (X'0982 ') Abonelik adı, var olan abonelikle eşleşiyor.

### **MQRC\_NOT\_YETKILI**

2035 (X'07F3') Kullanıcının bu işlemi gerçekleştirme yetkisi yok.

### **MQRC\_NO\_ALTYAPISI**

2428 (X'097C') Tanımlanmamış abonelik adı yok.

### **MQRC\_OBJECT\_STRING\_ERROR**

2441 (X'0989 ') Objectstring alanı geçerli değil.

### **MQRC\_OPTIONS\_ERROR**

2046 (X'07FE') Seçenekleri parametresi ya da alanında geçerli olmayan seçenekler ya da geçerli olmayan bir seçenek birleşimi var.

### **MQRC\_Q\_MGR QUIESCING**

2161 (X'0871 ') Kuyruk yöneticisi susturuyor.

### **MQRC\_RECONNECT\_Q\_MGR\_REQD**

2555 (X'09FB' X) MQCNO\_RECONNECT\_Q\_MGR seçeneği gerekli.

### **MQRC\_RETAINED\_MSG\_Q\_ERROR**

2525 (X'09DD') Abone olunan konu dizgisi için var olan yayınların alınması alınamıyor.

### **MQRC\_RETAINED\_NOT\_TESLIM**

2526 (X'09DE') Abone olunan konu dizgisi için var olan tutulan yayınlar, abonelik hedef kuyruğuna teslim edilemez ve ölü-mektup kuyruğuna teslim edilemez.

### **MQRC\_SD\_ERROR**

2424 (X'0978 ') Abonelik tanımlayıcısı (MQSD) geçerli değil.

### **MQRC\_SELECTION\_NOT\_AVAM**

2551 (X'09F7') Seçim dizgisi, IBM MQ seçici sözdizimini izlemiyor ve genişletilmiş ileti seçimi sağlayıcısı yok.

### **MQRC\_SELECTION\_STRING\_ERROR**

2519 (X'09D7') Bu seçim dizgisi, MQCHARV yapısı belgelerinde açıklandığı şekilde belirtilmelidir.

### **MQRC\_SELECTOR\_SYNTAX\_ERROR**

2459 (X'099B') Bir MQOP, MQPUT1ya da MQSUB çağrısı yayınlandı, ancak sözdizimi hatası içeren bir seçim dizgisi belirtildi.

### **MQRC\_SUB\_USER\_DATA\_ERROR**

2431 (X'097F') SubUserVeri alanı geçerli değil.

### **MQRC\_SUB\_NAME\_ERROR**

2440 (X'0988 ') SubName alanı geçerli değil.

**MQC\_SUB\_ALREADY\_EXISTS**

2432 (X'0980 ') Abonelik zaten var.

**MQRC\_SUB\_USER\_DATA\_ERROR**

2431 (X'097F') SubUserVeri alanı geçerli değil.

**MQRC\_TOPIC\_STRING\_ERROR**

2425 (X'0979 ') Konu dizgisi geçerli değil.

**MQRC\_UNKNOWN\_OBJECT\_NAME**

2085 (X'0825 ') MQSD ObjectName alanında tanıtılan nesne bulunamıyor.

**MQRC\_SUB\_JOIN\_NOT\_ALTERABLE**

29440 (X'7300 ') Abonelik paylaşım kipi, var olan abonelikle uyumsuz. JMS olmayan bir uygulamada JMS 2.0 paylaşılan aboneliği sürdürülmeye çalışılırken bu hata döndürülebilirdi.

Bu kodlarla ilgili ayrıntılı bilgi için [İletiler ve neden kodları](#) başlıklı konuya bakın.

**Kullanım notları**

- Abonelik, önceden tanımlanmış bir konu nesnesinin kısa adı, konu dizgisinin tam adı kullanılarak ya da iki kısmın bitişirmesiyle oluşturulmuş bir konuya eklenir. “MQSD-Abonelik tanımlayıcısı” sayfa 557’indeki *ObjectName* ve *ObjectString* açıklamalarına bakın.
  - Kuyruk yöneticisi, bir MQSUB çağrısı yayınlandığında güvenlik denetimlerini gerçekleştirir ve uygulamanın çalışmakta olduğu kullanıcı kimliğinin erişim izni verilmeden önce uygun yetki düzeyine sahip olduğunu doğrulamak için. Uygun konu nesnesi konu sıradüzeninde bulunur ve abone olma yetkisinin ayarlandığından emin olmak için bu konu nesnesinde bir yetki denetimi yapılır. MQSO\_PROMESSED seçeneği kullanılmıyorsa, çıkış için yetki denetimi yapılıp yapılmadığını denetlemek için hedef kuyruğun üzerinde bir yetki denetimi yapılır. MQSO\_MANAGED seçeneği kullanılırsa, çıkış ya da sorgu erişimi için yönetilen kuyruğun üzerinde herhangi bir yetki denetimi yapılmamaktadır.
  - Giriş olarak bir Hobj sağlamadıysanız, MQSUB çağrısı iki çekme noktası, bir nesne tanıtıcısı (Hobj) ve bir abonelik tanıtıcısı (Hsub) ayırır.
  - MQSO\_MANYED seçeneği kullanıldığında, MQSUB çağrısına geri döndürülen Hobj, Backout eşiği ve Exestionbackout requireue adı gibi öznitelikleri bulmak için sorgu sonlandırılabilir. Yönetilen kuyruğun adını da sorgulayabilirsiniz, ancak bu kuyruğu doğrudan açmayı denememeniz gerekir.
  - Abonelikler, grubun birden fazla yayınının yayınla eşleştiği durumlarda bile yalnızca tek bir yayının abonelikleri grubuna teslim edilmesine olanak tanıyabilir. Abonelikler MQSO\_GROUP\_SUB seçeneği kullanılarak gruplanır ve abonelikleri gruplamak için bu abonelikleri gruplamak gerekir:
    - MQSUB çağrısındaki Hobj parametresiyle gösterilen aynı kuyruk yöneticisinde aynı adı taşıyan (MQSO\_NICATED seçeneğini kullanmayan) aynı adı kullanarak (MQSO\_NICATED seçeneğini kullanmayan)
    - Aynı SubCorreltanıtıcısını paylaş
    - Aynı SubLevel' in
- Bu öznitelikler, grupta yer alınacak olan aboneliklerin kümesini tanımlar ve bir abonelik gruplandırılırsa, değiştirilemeyen öznitelikler de vardır. SubLevel sonuçları MQRC\_SUBLEVEL\_NOT\_ALTERABLE içinde değiştirilir ve diğer öğelerden herhangi birinin değiştirilmesi (abonelik gruplanmadıysa değiştirilebilir) MQRC\_GROUPING\_NOT\_ALTERNATEABLE içinde sonuçlar.
- MQSUB çağrısının başarıyla tamamlanması, işlemin tamamlandığı anlamına gelmez. Bu arama işleminin tamamlandığını denetlemek için [Dağıtılmış ağlara ilişkin zamanuyumsuz komutların sona erdiği denetleniyor](#) içindeki ALT adımına bakın.
  - MQSD ' deki alanlar, MQSO\_RESUME seçeneğini kullanan bir MQSUB çağrısından dönüş sırasında doldurulur. Döndürülen MQSD çağrısına, MQSO\_ALTER seçeneğini kullanan bir MQSUB çağrısına, MQSD ' ye uygulanan abonelik ile ilgili olarak yapmanız gereken değişikliklerle doğrudan aktarılabilir. Bazı alanlarda, çizelgede belirtildiği şekilde dikkat edilmesi gereken noktalar vardır.

Çizelge 557. MQSD ' deki alanlar için özel dikkat edilmesi gereken noktalar	
MQSD ' de alan adı	Özel noktalar
Erişim ya da yaratma seçenekleri	Bazı seçenekler MQSUB çağrısından döndürülebilecek şekilde ilk duruma getirilebilir. Daha sonra MQSUB çağrısında MQSD ' yi yeniden kullanırsanız, gerekli olan seçeneğin belirtik olarak ayarlanması gerekir.
Dayanıklılık seçenekleri, Hedef seçenekleri, Kayıt Seçenekleri ve Genel Arama Karakteri seçenekleri	Bu seçenekler uygun şekilde ayarlanır
Yayın seçenekleri	Bu seçenekler, yalnızca MQSO_CREATE işlemi için geçerli olan MQSO_NEW_PATICATIONS_ONLY dışında, uygun olarak ayarlanır.
Diğer seçenekler	Bu seçenekler, bir MQSUB çağrısından geri dönmeden değiştirilmez. Bunlar, API çağrısının yayınlanmalarını ve abonelikte birlikte saklanmadıklarını denetler. Bunlar, MQSD ' yi yeniden kullanan sonraki bir MQSUB çağrılarında gerektiği gibi ayarlanmalıdır.
ObjectName	Yalnızca bu giriş alanı, bir MQSUB çağrısından dönüşte değişmeden kalır.
ObjectString	Yalnızca bu giriş alanı, bir MQSUB çağrısından dönüşte değişmeden kalır. Bir arabellek sağlanırsa, <i>ResObjectString</i> alanında kullanılan tam konu adı döndürülür.
AlternateUserTanıtıcısı ve AlternateSecurityTanıtıcısı	Yalnızca bu giriş alanları, bir MQSUB çağrısından dönüşte değişmeden kalır. Bunlar, API çağrısının yayınlanmalarını ve abonelikte birlikte saklanmadıklarını denetler. Bu, MQSD ' yi yeniden kullanan sonraki bir MQSUB çağrılarında gerektiği şekilde ayarlanmalıdır.
SubExpiry	MQSO_RESUME seçeneğini kullanarak bir MQSUB çağrısından dönüşte, bu alan aboneliğin özgün süre bitimi olarak ayarlanır ve kalan süre bitimi değil olarak ayarlanır. Daha sonra, MQSO_ALTER seçeneğini kullanarak bir MQSUB çağrısında MQSD ' yi yeniden kullanırsanız, aboneliğin süre bitimini yeniden saymaya başlamak üzere sıfırlayın.
SubName	Bu alan, bir MQSUB çağrısına ilişkin bir giriş alanıdır ve çıkışta değiştirilmez.



Çizelge 557. MQSD ' deki alanlar için özel dikkat edilmesi gereken noktalar (devamı var)	
MQSD ' de alan adı	Özel noktalar
SubUserVerileri ve SelectionString	<p>Bu değişken uzunluk alanları, bir arabellek sağlanırsa, MQSU_RESUME seçeneği kullanılarak bir MQSUB çağrısından çıkışa döndürülür ve <i>VSBufSize</i> içinde artı bir arabellek uzunluğu da olabilir. Arabellek sağlanmazsa, MQCHARV ' in <i>VSLength</i> alanına yalnızca uzunluk değeri döndürülür. Sağlanan arabellek, alanı döndürmek için gereken alandan küçükse, sağlanan arabellekte yalnızca <i>VSBufSize</i> byte 'ları döndürülür.</p> <p>Daha sonra, MQSO_ALTER seçeneğini kullanarak bir MQSUB çağrısında MQSD ' yi yeniden kullanırsanız ve arabellek sağlanmaz, ancak sıfır dışında bir <i>VSLength</i> sağlanırsa, bu uzunluk alanının var olan uzunluğuna uygun olursa, bu alanda bir değişiklik yapılmamaktadır.</p>
SubCorrelTanıtıcısı ve PubAccountingSimgesi	<p>MQSO_SET_COREL_ID seçeneğini kullanmayacaksa, kuyruk yöneticisi <i>SubCorrelId</i> tarafından oluşturulur. MQSO_SET_IDENTITY_CONTEXT seçeneğini kullanmayacaksa, kuyruk yöneticisi <i>PubAccountingToken</i> tarafından oluşturulur.</p> <p>Bu alanlar, MQSO_RESUME seçeneği kullanılarak bir MQSUB çağrısından MQSD ' ye döndürülür. Kuyruk yöneticisi tarafından oluşturulduysa, oluşturulan değer MQSO_CREATE ya da MQSO_ALTER seçeneğini kullanarak bir MQSUB çağrısına döndürülür.</p>
PubPriority, SubLevel & PubApplIdentityData	Bu alanlar MQSD ' de döndürülür.
ResObjectDizgisi	Bir arabellek sağlanırsa, bu çıkış yalnızca MQSD ' ye döndürülür.

## C çağırma

```
MQSUB (Hconn, &SubDesc, &Hobj, &Hsub, &CompCode, &Reason)
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
MQHCONN Hconn; /* Connection handle */
MQSD SubDesc; /* Subscription descriptor */
MQHOBJ Hobj; /* Object handle */
MQHOBJ Hsub; /* Subscription handle */
MQLONG CompCode; /* Completion code */
MQLONG Reason; /* Reason code qualifying CompCode */
```

## Cobol çağırma

```
CALL 'MQSUB' USING HCONN, SUBDESC, HOBJ, HSUB, COMPCODE, REASON.
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
** Connection handle
01 HCONN PIC S9(9) BINARY.
```

```

** Subscription descriptor
  01 SUBDESC.
    COPY CMQSDV.
** Object handle
  01 HOBJ    PIC S9(9) BINARY.
** Subscription handle
  01 HSUB    PIC S9(9) BINARY.
** Completion code
  01 COMPCODE PIC S9(9) BINARY.
** Reason code qualifying COMPCODE
  01 REASON  PIC S9(9) BINARY.

```

## PL/I çağırımı

```
call MQSUB (Hconn, SubDesc, Hobj, Hsub, CompCode, Reason)
```

Değiştirgeleri aşağıdaki gibi bildirin:

```

dcl Hconn    fixed bin(31); /* Connection handle */
dcl SubDesc  like MQSD;    /* Subscription descriptor */
dcl Hobj     fixed bin(31); /* Object handle */
dcl Hsub     fixed bin(31); /* Subscription handle */
dcl CompCode fixed bin(31); /* Completion code */
dcl Reason   fixed bin(31); /* Reason code qualifying CompCode */

```

## High Level Assembler çağırısı

```
CALL MQSUB, (HCONN, SUBDESC, HOBJ, HSUB, COMPCODE, REASON)
```

Değiştirgeleri aşağıdaki gibi bildirin:

```

HCONN    DS      F  Connection handle
SUBDESC  CMQSDA  ,  Subscription descriptor
HOBJ     DS      F  Object handle
HSUB     DS      F  Subscription handle
COMPCODE DS      F  Completion code
REASON   DS      F  Reason code qualifying COMPCODE

```

## MQSUBRQ-Abonelik isteği

Abonenin MQSO\_PUBLICATIONS\_ON\_REQUEST ile kaydedildiği zaman alıkonan yayınlara ilgili bir istek yapmak için MQSUBRQ çağırısını kullanın.

### Sözdizimi

MQSUBRQ (*Hconn, Hsub, Action, SubRqOpts, Compcode, Reason*)

### Parametreler

#### Hconn

Tip: MQHCONN-input

Bu tanırtıcı, kuyruk yöneticisiyle olan bağlantıyı gösterir. The value of *Hconn* was returned by a previous MQCONN or MQCONNX call.

On z/OS for CICS applications the MQCONN call can be omitted, and the following value specified for *Hconn* :

#### MQHC\_DEF\_HCONN

Varsayılan bağlantı tanırtıcısı.

## Hsub

Tip: MQHOBJ-input

Bu tanıtıcı, bir güncellenmenin isteneceği aboneliği temsil eder. *Hsub* değeri önceki bir MQSUB çağrısından döndürüldü.

## İşlem

Tip: MQHOME-girişi

Bu değıştirge, abonelikte istenmekte olan işlemi denetler. Aşağıdaki değer belirtilmelidir:

### **MQSR\_ACTION\_YAYINI**

Bu işlem, belirtilen konu için bir güncelleme yayınının gönderildiğini ister. Yalnızca abone, aboneliği gerçekleştirdiğinde MQSUB çağrısında MQSO\_PUBLICATIONS\_ON\_REQUEST seçeneğini belirttiyse kullanılabilir. Kuyruk yöneticisinin konu için saklanacak bir yayını varsa, bu, aboneye gönderilir. Yoksa, arama başarısız olur. Bir uygulama tutulan bir yayını gönderdiyse, bu, yayının MQIsRetained ileti özelliği ile gösterilir.

Hsub parametresiyle gösterilen var olan abonelikte konu joker karakterler içerebileceğinden, abone birden çok tutulan yayınları alabilir.

## SubRqOpts

Tip: MQSRO-giriş/çıkış

Bu seçenekler MQSUBRQ 'nun işlemi denetler, ayrıntılar için [“MQSRO-Abonelik isteği seçenekleri” sayfa 578](#) ' e bakın.

Herhangi bir seçenek gerekmiyorsa, C ya da S/390 çevirici programında yazılan programlar, bir MQSRO yapısının adresini belirtmek yerine boş değerli bir parametre adresi belirtebilir.

## CompCode

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Tamamlanma kodu; bu kod aşağıdakilerden biridir:

### **MQCC\_OK**

Başarıyla tamamlandı

### **MQCC\_UYARI**

Uyarı (kısmi tamamlama)

### **MQCC\_FAILED**

Arama başarısız oldu

## Neden

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Neden kodu niteleyici *CompCode*.

*CompCode* MQCC\_OK ise:

### **MQRC\_NONE**

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_FAILED ise:

### **MQRC\_FUNCTION\_NOT\_SUPPORTUL**

2298 (X'08FA') İstenen işlev, yürürlükteki ortamda yok.

### **MQRC\_NO\_RETAINED\_MSG**

2437 (X'0985 ') Bu konu için saklanmış yayınların saklanmasına neden yok.

### **MQRC\_OPTIONS\_ERROR**

2046 (X'07FE') Seçenekleri parametresi ya da alanında geçerli olmayan seçenekler ya da geçerli olmayan bir seçenek birleşimi var.

### **MQRC\_Q\_MGR QUIESCING**

2161 (X'0871 ') Kuyruk yöneticisi susturuyor.

### **MQRC\_SRO\_ERROR**

2438 (X'0986 ') MQSUBRQ çağrısında, Subscription Request Options MQSRO geçerli değil.

### **MQRC\_RETAINED\_MSG\_Q\_ERROR**

2525 (X'09DD') Abone olunan konu dizgisi için var olan yayınların alınması alınamıyor.

### **MQRC\_RETAINED\_NOT\_TESLIM**

2526 (X'09DE') Abone olunan konu dizgisi için var olan tutulan yayınlar, abonelik hedef kuyruğuna teslim edilemez ve ölü-mektup kuyruğuna teslim edilemez.

Bu kodlarla ilgili ayrıntılı bilgi için [İletiler ve neden kodları](#) başlıklı konuya bakın.

## **Kullanım notları**

Aşağıdaki kullanım notları, MQSR\_ACTION\_YAYIN işlem kodu kullanımı için geçerli olur:

1. Bu komut başarıyla tamamlanırsa, belirtilen abonelikte eşleşen alıkonan yayınlar aboneliğe gönderildi ve bu aboneliği yaratan özgün MQSUB yüklemi üzerinde döndürülen Hobj kullanılarak MQGET ya da MQCB kullanılarak alınabilir.
2. Aboneliği yaratan özgün MQSUB yüklemi tarafından abone olunan konu genel arama karakteri içeriyorsa, birden çok alıkonabilen yayın gönderilebilir. Bu çağrı sonucu olarak gönderilen yayınların sayısı, SubRqOpts yapısındaki NumPubs alanına kaydedilir.
3. Bu fiil, MQRC\_NO\_RETAINED\_MSG neden koduyla tamamlanırsa, belirtilen konu için yürürlükte tutulan yayınlar yoktur. #
4. Bu komut, bir MQRC\_RETAINED\_MSG\_Q\_ERROR ya da MQRC\_RETAINED\_NOT\_TESLIM edilen bir neden koduyla tamamlanırsa, belirtilen konuya ilişkin güncel yayınlar vardır; ancak, teslim edilemedikleri anlamına gelen bir hata ortaya çıktı.
5. Uygulamanın, bu çağrıyı gerçekleştirebilmesi için önce konuya ilişkin geçerli bir aboneliği olmalıdır. Abonelik, uygulamanın önceki bir örneğinde yapıldıysa ve abonelik için geçerli bir tanıtıcı kullanılamıyorsa, uygulamanın bu çağrıda kullanılmak üzere bir tanıtıcı elde etmek için önce MQSO\_RESUME seçeneğiyle birlikte MQSUB seçeneğini çağırmalıdır.
6. Bu yayınlar, bu uygulamanın yürürlükteki abonelikleriyle kullanılmak üzere kayıtlı olan hedefe gönderilir. Yayınların başka bir yerde gönderilmesi gerekiyorsa, önce MQSO\_ALTER seçeneğiyle MQSUB çağrısıyla aboneliğin değiştirilmelidir.

## **C çağırma**

```
MQSUB (Hconn, Hsub, Action, &SubRqOpts, &CompCode, &Reason)
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
MQHCONN Hconn; /* Connection handle */
MQHOBJ Hsub; /* Subscription handle */
MQLONG Action; /* Action requested by MQSUBRQ */
MQSRO SubRqOpts; /* Options that control the action of MQSUBRQ */
MQLONG CompCode; /* Completion code */
MQLONG Reason; /* Reason code qualifying CompCode */
```

## **Cobol çağırma**

```
CALL 'MQSUBRQ' USING HCONN, HSUB, ACTION, SUBRQOPTS, COMPCODE, REASON.
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
** Connection handle
01 HCONN PIC S9(9) BINARY.
** Subscription handle
01 HSUB PIC S9(9) BINARY.
```

```
** Action requested by MQSUBRQ
01 ACTION PIC S9(9) BINARY.
** Options that control the action of MQSUBRQ
01 SUBRQOPTS.
COPY CMQSROV.
** Completion code
01 COMPCODE PIC S9(9) BINARY.
** Reason code qualifying COMPCODE
01 REASON PIC S9(9) BINARY.
```

## PL/I çağırımı

```
call MQSUBRQ (Hconn, Hsub, Action, SubRqOpts, CompCode, Reason)
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
dcl Hconn fixed bin(31); /* Connection handle */
dcl Hsub fixed bin(31); /* Subscription handle */
dcl Action fixed bin(31); /* Action requested by MQSUBRQ */
dcl SubRqOpts like MQSRO; /* Options that control the action of MQSUBRQ */
dcl CompCode fixed bin(31); /* Completion code */
dcl Reason fixed bin(31); /* Reason code qualifying CompCode */
```

## High Level Assembler çağırısı

```
CALL MQSUBRQ,(HCONN, HSUB, ACTION, SUBRQOPTS,COMPCODE,REASON)
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
HCONN DS F Connection handle
HSUB DS F Subscription handle
ACTION DS F Action requested by MQSUBRQ
SUBRQOPTS CMQSROA , Options that control the action of MQSUBRQ
COMPCODE DS F Completion code
REASON DS F Reason code qualifying COMPCODE
```

## Nesnelerin öznitelikleri

Bu konu derlemi, yalnızca bir MQINQ işlev çağırısının konusu olabilecek IBM MQ nesnelerini listeler ve sorgulanabilen özniteliklerin ve kullanılacak seçicilerin ayrıntılarını verir.

## Kuyruk yöneticisine ilişkin öznitelikler

Some queue manager attributes are fixed for particular implementations; others can be changed by using the MQSC command ALTER QMGR.

Öznitelikler, DISPLAY QMGRkomutu kullanılarak da görüntülenebilir. Çoğu kuyruk yöneticisi öznitelikleri, özel bir MQOT\_Q\_MGR nesnesi açarak ve döndürülen tanıtıcı ile MQINQ çağırısını kullanarak sorgulanabilir.

Aşağıdaki çizelge, kuyruk yöneticisine özgü öznitelikleri özetlemektedir. Öznitelikler alfabetik sırayla açıklanır.

**Not:** Bu kısımda gösterilen özniteliklerin adları, MQINQ çağırısıyla kullanılan açıklayıcı adlardır; bu adlar PCF komutlarıyla aynı olur. MQSC komutları, öznitelikleri tanımlamak, değiştirmek ya da görüntülemek için kullanıldığında, diğer kısa adlar kullanılır; ek bilgi için [MQSC komutları](#) konusuna bakın.

Çizelge 558. Kuyruk yöneticisine ilişkin öznitelikler	
Öznitelik	Tanım
AccountingConnGeçersiz Kılma	Hesap ayarlarını geçersiz kıl.
AccountingInterval	Ara muhasebe kayıtlarının ne sıklıkta yazılabileceği.

Çizelge 558. Kuyruk yöneticisine ilişkin öznitelikler (devamı var)

Öznitelik	Tanım
<a href="#">ActivityConnGeçersiz Kılma</a>	Etkinlik ayarlarını geçersiz kıl.
<a href="#">ActivityTrace</a>	IBM MQ MQI uygulaması etkinlik izleme derlemine denetler.
<a href="#">AdoptNewMCACheck</a>	Yeni MCA ' yı kabul edip etmeyeceğini belirlemek için kullanılan öğeler imlendi.
<a href="#">AdoptNewMCATYPE</a>	Belirli bir kanal tipindeki bir MCA ' nın (MCA) otomatik olarak başlatılıp başlatılmayacağı.
<a href="#">AlterationDate</a>	Tanımlamanın son olarak değiştirildiği tarih
<a href="#">AlterationTime</a>	Tanımlamanın son olarak değiştirildiği saat
<a href="#">AuthorityEvent</a>	Yetkilendirme (Yetkili Değil) olaylarının oluşturulup oluşturulmayacağını denetler
<a href="#">BridgeEvent</a>	Köprü olaylarına ilişkin denetim özniteliği.
<a href="#">ChannelAutoDef</a>	Otomatik kanal tanımlamasının kullanılmasına izin verilip verilmediğini denetler
<a href="#">ChannelAutoDefEvent</a>	Kanal otomatik tanımlama olaylarının oluşturulup oluşturulmayacağını denetler
<a href="#">ChannelAutoDefExit</a>	Otomatik kanal tanımlaması için kullanıcı çıkışının adı
<a href="#">ChannelEvent</a>	Kanal olaylarına ilişkin denetim özniteliği.
<a href="#">ChannelInitiatorDenetimi</a>	Kanal başlatıcısı için denetim özniteliği
<a href="#">ChannelMonitoring</a>	Kanallar için çevrimiçi izleme verileri
<a href="#">ChannelStatistics</a>	Kanallar için istatistik verilerinin toplanmasını denetler.
<a href="#">ChinitAdapters</a>	IBM MQ çağrılarının işlenmesine ilişkin bağdaştırıcı alt görevi sayısı.
<a href="#">ChinitDispatchers</a>	Kanal başlatıcısı için kullanılacak dağıtıcıların sayısı.
	IBM kullanımı için ayrılmıştır.
<a href="#">ChinitTraceAutoStart</a>	Kanal başlatıcı izlemesinin otomatik olarak başlatılıp başlatılmayacağını.
<a href="#">ChinitTraceTableSize</a>	Kanal başlatıcısının izleme verileri alanı büyüklüğü.
<a href="#">ClusterSenderMonitoringDefault</a>	Küme gönderen kanalları için çevrimiçi izleme verileri varsayılanı
<a href="#">ClusterSenderİstatistikleri</a>	Küme gönderen kanallarına ilişkin istatistik izleme bilgileri derlemine denetler.
<a href="#">ClusterWorkloadVerileri</a>	Küme iş yükü çıkışa ilişkin kullanıcı verileri
<a href="#">ClusterWorkloadÇıkış</a>	Küme iş yükü yönetimi için kullanıcı çıkışının adı
<a href="#">ClusterWorkloadUzunluğu</a>	Küme iş yükü çıkışa aktarılan ileti verisi uzunluğu üst sınırı
<a href="#">CLWLMRUChannels</a>	Küme iş yükü dengelemesi için en son kullanılan kanalların sayısı
<a href="#">CLWLUseQ</a>	Küme iş yükü, uzak kuyruğu kullanır.
<a href="#">CodedCharSetId</a>	Kodlanmış karakter takımı tanıtıcısı
<a href="#">CommandEvent</a>	Komut olaylarına ilişkin denetim özniteliği.
<a href="#">CommandInputQName özniteliği</a>	Komut giriş kuyruğu adı
<a href="#">CommandLevel</a>	Komut düzeyi
<a href="#">CommandServerControl özniteliği</a>	Komut sunucusuna ilişkin denetim özniteliği.
<a href="#">Yapılandırma Olayı özniteliği</a>	Yapılandırma olaylarına ilişkin denetim özniteliği.
<a href="#">DeadLetterQName</a>	Ölü-mektup kuyruğunun adı
<a href="#">DEFCLXQ</a>	Varsayılan küme iletim kuyruğu tipi
<a href="#">DefXmitQName</a>	Varsayılan iletim kuyruğu adı
<a href="#">DistLists</a>	Dağıtım listesi desteği
<a href="#">DNSGroup</a>	İş Yükü Yöneticisi Dinamik Etki Alanı Adı Hizmetleri desteği kullanılırken TCP dinleyicisine ilişkin grubun adı.
<a href="#">DNSWLM</a>	TCP dinleyicisinin, Dinamik Etki Alanı Adı Hizmetleri için İş Yükü Yöneticisi ile birlikte kaydolup olmadığı.
<a href="#">ExpiryInterval</a>	Süresi dolan iletiler için taramalar arasındaki aralık
<a href="#">IGQPutAuthority</a>	Grup içi kuyruğa alma yetkisi
<a href="#">IGQUserId</a>	Grup içi kuyruğa alma kullanıcı kimliği
<a href="#">InhibitEvent</a>	Engelleyici (Inhibit Get ve Inhibit Sput) olaylarının oluşturulup oluşturulmayacağını denetler

Çizelge 558. Kuyruk yöneticisine ilişkin öznitelikler (devamı var)

Öznitelik	Tanım
<a href="#">IPAddressVersion</a>	Internet Protocol adresinin sürümü
<a href="#">IntraGroupQueueing</a>	Grup içi kuyruğa alma desteği
<a href="#">ListenerTimer</a>	APPC ya da TCP/IP hatasından sonra dinleyiciyi yeniden başlatma girişimleri arasındaki zaman aralığı.
<a href="#">LocalEvent</a>	Yerel hata olaylarının oluşturulup oluşturulmadığını denetler
<a href="#">LoggerEvent</a>	Günlüğe kaydedici olaylarının oluşturulup oluşturulmayacağını denetler
<a href="#">LUGroupName</a>	Kuyruk paylaşım grubu için gelen iletimleri işleyen LU 6.2 dinleyicisi için genel LU adı.
<a href="#">LUAdı</a>	Giden LU 6.2 iletimleri için kullanılacak LU adı.
<a href="#">LU62ARMSuffix</a>	SYS1.PARMLIB üyesi APPCPMxx, bu kanal başlatıcısı için LUADD ' yi aday göstermektedir.
<a href="#">LU62Channels</a>	Maximum number of current channels or connected clients that use LU 6.2.
<a href="#">MaxActiveKanalları</a>	Herhangi bir zamanda etkin olabilecek kanal sayısı üst sınırı.
<a href="#">MaxChannels</a>	Geçerli kanal sayısı üst sınırı.
<a href="#">MaxHandles</a>	Tanıtıcı sayısı üst sınırı
<a href="#">MaxMsgUzunluğu</a>	İleti uzunluğu üst sınırı (byte)
<a href="#">MaxPriority özniteliği</a>	En yüksek öncelik
<a href="#">MaxPropertiesUzunluğu</a>	Özellik verileri uzunluğu üst sınırı (byte)
<a href="#">MaxUncommittedMsgs</a>	Bir iş birimi içinde kesinleştirilmemiş ileti sayısı üst sınırı
<a href="#">MQIAccounting</a>	MQI verilerine ilişkin muhasebe bilgilerinin toplanmasını denetler.
<a href="#">MQIStatistics</a>	Kuyruk yöneticisine ilişkin istatistik izleme bilgilerinin derlemesini denetler.
<a href="#">MsgMarkBrowseInterval</a>	Kuyruk yöneticisinin işaretlenen iletilerden işareti kaldırılabileceği aralık.
<a href="#">OutboundPortAlt Sınır</a>	<i>OutboundPortMinile</i> , giden kanalları bağlarken kullanılacak kapı numaralarının aralığını tanımlar.
<a href="#">OutboundPortAlt Sınır</a>	<i>OutboundPortMaxile</i> , giden kanalları bağlarken kullanılacak kapı numaralarının aralığını tanımlar.
<a href="#">PerformanceEvent</a>	Başarımla ilgili olayların oluşturulup oluşturulmayacağını denetler
<a href="#">Altyapı</a>	Kuyruk yöneticisinin çalışmakta olduğu altyapı
<a href="#">PubSubNPInputMsg</a>	Teslim edilmemiş bir giriş iletiminin atılıp atılmayacağını (ya da alıkoymak)
<a href="#">PubSubNPreponse</a>	Teslim edilmemesinin davranışını denetler
<a href="#">PubSubMaxMsgRetryCount</a>	Başarısız bir komut iletimi işlenirken (syncpoint altında) işlem sırasında yeniden deneme sayısı
<a href="#">PubSubSyncPoint</a>	Yalnızca kalıcı (ya da tüm) iletilerin uyumluluk noktası altında işlenip işlenmeyeceğini
<a href="#">PubSubKipi</a>	Kuyruğa alınan yayınlama/abone olma arabiriminin çalıştırılıp çalıştırılmayacağı
<a href="#">QMGrDesc</a>	Kuyruk yöneticisi tanımı
<a href="#">QMGrIdentifier</a>	Kuyruk yöneticisinin iç sistem tarafından üretilen benzersiz tanıtıcısı
<a href="#">QMGrName</a>	Kuyruk yöneticisi adı
<a href="#">QSGName</a>	Kuyruk paylaşım grubunun adı
<a href="#">QueueAccounting</a>	Kuyruklara ilişkin muhasebe bilgilerinin toplanmasını denetler.
<a href="#">QueueMonitoring</a>	Kuyruklar için çevrimiçi izleme verileri
<a href="#">QueueStatistics</a>	Kuyruklar için istatistik verilerinin toplanmasını denetler.
<a href="#">ReceiveTimeout</a>	TCP/IP kanalı, etkin olmayan duruma dönmeden önce verilerin ne kadar süreyle bekleyeceğini belirler.
<a href="#">ReceiveTimeoutAlt Sınır</a>	<i>ReceiveTimeout</i> niteleyicisi.
<a href="#">ReceiveTimeoutTipi</a>	TCP/IP kanalı etkin olmayan duruma dönmeden önce veri için bekleme süresi alt sınırı.
<a href="#">RemoteEvent</a>	Uzak hata olaylarının oluşturulup oluşturulmadığını denetler
<a href="#">RepositoryName</a>	Bu kuyruk yöneticisinin havuz hizmetleri sağladığı kümenin adı
<a href="#">RepositoryNamelist</a>	Bu kuyruk yöneticisinin havuz hizmetleri sağladığı kümelerin adlarını içeren ad listesi nesnesinin adı
<a href="#">ScyCase</a>	Güvenlik profilleri vakası
<a href="#">SharedQMGrAd</a>	Paylaşılan kuyruk kuyruk yöneticisi adı

Çizelge 558. Kuyruk yöneticisine ilişkin öznelikler (devamı var)

Öznelik	Tanım
"KYR" sayfa 818	IBM MQ Advanced Message Security protection for a queue Manager açık ya da kapalı.
SSLCRLNameList <u>1</u>	Kimlik doğrulama bilgileri nesnelere adlarını içeren ad listesi nesnelere ad.
SSLCryptoHardware <u>1</u>	Şifreleme donanımı yapılandırma dizesi.
SSLECC	TLS olaylarına ilişkin denetim özneliği.
SSLFIPSTRequired	Kriptografi için yalnızca FIPS onaylı algoritmalar kullanın.
SSLKeyRepository <u>1</u>	TLS anahtarı havuzunun yeri.
SSLKeyResetSayı	TLS anahtarı sıfırlama sayısı.
SSLTasks <u>1</u>	TLS çağrılarının işlenmesine ilişkin sunucu alt görevi sayısı.
StatisticsInterval	İstatistik izleme verileri yazma sıklığı.
StartStopOlayı	Başlatma ve durdurma olaylarının oluşturulup oluşturulmayacağını denetler
SyncPoint	Eşitleme noktası kullanılabilirliği
TCPKanallar	TCP/IP kullanan yürürlükteki kanal ya da bağlı istemci sayısı üst sınırı.
TCPKeepAlive	Bağlantının diğer uçlarını denetlemek için TCP KEPAVE ile birlikte kullanılmalı.
TCPName	Kullanmakta olduğunuz TCP/IP sisteminin adı.
TCPStackType	Kanal başlatıcının TCP/IP adreslerini nasıl kullanabileceğini gösterir.
TraceRouteKayıt özneliği	İzleme rotası bilgilerinin kaydedilmesini denetler.
TriggerInterval	Tetikleme-ileti aralığı
S\u00f0fcr\u00f0fcm	S\u00f0fcr\u00f0fcm
XrCapability	Telemetri komutlarının desteklenip desteklenmediğini belirtir.

**Notlar:**

1. Bu öznelik, MQINQ çağrısını kullanarak sorgulanamaz ve bu bölümde anlatılmaz. Bu özneliğe ilişkin ayrıntılar için [Kuyruk Yöneticisinin Değiştirilmesi](#) başlıklı konuya bakın.

### İlgili görevler

MQI istemcisinde çalıştırma sırasında yalnızca FIPS onaylı CipherSpecs ' in kullanıldığını belirtme

### İlgili başvurular

[Federal Information Processing Standards \(FIPS\) for AIX, Linux, and Windows](#)

### AccountingConnOverride (MQUZE)

Bu, uygulamaların, Qmgr öznelideki ACCTMQI ve ACCTQDATA değerlerinin ayarını geçersiz kılmasına olanak sağlar.

Değer aşağıdakilerden biridir:

#### MQMON\_DEVRE DIŞI

Uygulamalar, MQCONNX çağrısındaki MQCNO yapısındaki Seçenekler alanını kullanarak, ACCTMQI ve ACCTQ Qmgr özneliklerinin ayarını geçersiz kılabilir. Bu varsayılan değerdir.

#### MQMON\_ENABLED

Uygulamalar, MQCNO yapısındaki Options alanını kullanarak, ACCTQ ve ACCTMQI Qmgr özneliklerini geçersiz kılabilir.

Bu değerdeki değişiklikler yalnızca, öznelik üzerindeki değişiklikten sonra kuyruk yöneticisine yönelik bağlantılar için geçerlidir.

Bu öznelik yalnızca aşağıdaki altyapılarda desteklenir:

- ▶ **IBM i** IBM i
- ▶ **Linux** ▶ **AIX** AIX and Linux
- ▶ **Windows** Windows



Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_ACCOUNTING\_CONN\_OVERRIDE seçiciyi kullanın.

### **AccountingInterval (MQUZE)**

Bu, ara muhasebe kayıtlarının saniye cinsinden ne kadar önce yazılacağını belirtir (saniye cinsinden).

Değer, 0 ile 604800 arasındaki bir tamsayıdır ve varsayılan değer olan 1800 (30 dakika) değeridir. Ara düzey kayıtları kapatmak için 0 değerini belirtin.

Bu öznitelik yalnızca aşağıdaki altyapılarda desteklenir:

-  IBM i
-  Linux  AIX and Linux
-  Linux
-  Windows

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_ACCOUNTING\_INTERVAL seçiciyi kullanın.

### **ActivityConnGeçersiz Kılma (MQUZE)**

Bu, uygulamaların kuyruk yöneticisi özniteindeki ACTVTRC değerinin ayarını geçersiz kılmasına olanak sağlar.

Değer aşağıdakilerden biridir:

#### **MQMON\_DEVRE DIŞI**

Uygulamalar, MQCONNX çağrısındaki MQCNO yapısındaki Seçenekler alanını kullanarak ACTVTRC kuyruk yöneticisi özniteliğinin ayarını geçersiz kılamaz. Bu varsayılan değerdir.

#### **MQMON\_ENABLED**

Uygulamalar, MQCNO yapısındaki Seçenekler alanını kullanarak ACTVTRC kuyruk yöneticisi özniteliğini geçersiz kılabilir.

Bu değerdeki değişiklikler yalnızca, öznitelik üzerindeki değişiklikten sonra kuyruk yöneticisine yönelik bağlantılar için geçerlidir.

Bu öznitelik yalnızca Multiplatforms üzerinde desteklenir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_ACTIVITY\_CONN\_OVERRIDE seçiciyi kullanın.

### **ActivityTrace (MQUZE)**

Bu, IBM MQ MQI uygulaması etkinlik izleme derlemine denetler.

Değer aşağıdakilerden biridir:

#### **MQMON\_ON**

Collect IBM MQ MQI application activity trace.

#### **MQMON\_OFF**

IBM MQ MQI uygulama etkinliği izlemesini toplamaz. Bu varsayılan değerdir.

If you set the queue manager attribute ACTVCON0 to ENABLED, this value might be overridden for individual connections using the Options field in the MQCNO structure.

Bu değerdeki değişiklikler yalnızca, öznitelik üzerindeki değişiklikten sonra kuyruk yöneticisine yönelik bağlantılar için geçerlidir.

Bu öznitelik yalnızca Multiplatforms üzerinde desteklenir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_ACTIVITY\_TRACE seçicisini kullanın.

### **AdoptNewMCACheck (MQUBE)**

Bu, önceden etkin bir MCA ile aynı adı taşıyan yeni bir gelen kanal saptandığında MCA ' yı kabul edip etmeyeceğini belirlemek için denetilecek öğeleri tanımlar.

Değer aşağıdakilerden biridir:

**MQADOPT\_CHECK\_Q\_MGR\_NAME**

Kuyruk yöneticisi adını denetleyin.

**MQADOPT\_CHECK\_NET\_ADDR**

Ağ adresini denetleyin.


**MQADOPT\_CHECK\_ALL**

Kuyruk yöneticisi adını ve ağ adresini denetleyin. Mümkünse, kanallarınızı kapalı, yanlışlıkla ya da kötü bir şekilde korumak için bu denetimi gerçekleştirin. Bu varsayılan değerdir.

**MQADOPT\_CHECK\_NONE**

Hiçbir öğeyi işaretlemeyin.

Bu özniteliğe ilişkin değişiklikler, kanal bir kanalı kullanmaya ilişkin bir sonraki seferde yürürlüğe girmektedir.

 Bu öznitelik yalnızca z/OS üzerinde desteklenir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_ADOPTNEWMCA\_CHECK seçicisini kullanın.

***AdoptNewMCAType (MQUZE)***

Bu, AdoptNewMCACheck özniteyle eşleşen yeni bir gelen kanal isteği algılandığında, belirli bir kanal tipinin bir MCA ' nın otomatik olarak başlatılıp başlatılmayacağını belirtir.

Bu değer aşağıdaki değerlerden biridir:

**MQADOPT\_TYPE\_NO**

Artık kanal eşgörünümlerinin benimsenmesi gerekli değil. Bu varsayılan değerdir.

**MQADOPT\_TYPE\_ALL**

Tüm kanal tiplerini kabul edin.

Bu öznitelik yalnızca z/OS üzerinde desteklenir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_ADOPTNEWMCA\_TYPE seçiciyi kullanın.

***AlterationDate (MQCHAR12)***

Bu, tanımın en son değiştirildiği tarihtir. Tarih biçimi şöyledir: YYYY-MM-DD; uzunluğun 12 byte 'ı olması için sondaki iki boşlukla doldurulması.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQCA\_ALTERATION\_DATE seçiciyi kullanın. Bu özniteliğin uzunluğu MQ\_DATE\_LENGTH tarafından verilir.

***AlterationTime (MQCHAR8)***

Bu, tanımın en son değiştirildiği dönemdir. Saatin biçimi şöyledir: HH.MM.SS.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQCA\_ALTERATION\_TIME seçiciyi kullanın. Bu özniteliğin uzunluğu MQ\_TIME\_LENGTH tarafından verilir.

***AuthorityEvent (MQUZE)***

Bu, yetkilendirme (Yetkili Değil) olaylarının oluşturulup oluşturulmayacağını denetler. Bu değer aşağıdaki değerlerden biridir:

**MQEVR\_DISABLE**

Olay raporlama devre dışı bırakıldı.

**MQEVRENABLED**

Olay raporlaması etkinleştirildi.

Olaylarla ilgili daha fazla bilgi için [Olay izleme](#) başlıklı konuya bakın.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_AUTHORITY\_ETO seçiciyi kullanın.

### **BridgeEvent (MQUZE)**

Bu, IMS köprüsü olaylarının oluşturulup oluşturulmayacağını belirtir.

Değer aşağıdakilerden biridir:

#### **MQEVRENABLED**

IMS köprüsü olayları aşağıdaki gibi oluşturulur:

MQRC\_BRIDGE\_STARTED  
MQRC\_BRIDGE\_DURDU

#### **MQEVR\_DISABLE**

IMS köprüsü olayları oluşturmayın; bu varsayılan değerdir.

Bu öznelik yalnızca z/OS üzerinde desteklenir.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_BRIDGE\_EVENT seçiciyi kullanın.

### **ChannelAutoDef (MQUZE)**

Bu öznelik, MQCHT\_RECEIVER ve MQCHT\_SVRCONN tipli kanalların otomatik tanımlamasını denetler. MQCHT\_CLUSSDR kanallarının otomatik tanımlaması her zaman etkindir. Değer aşağıdakilerden biridir:

#### **MQCHAD\_ENGELLI**

Kanal otomatik tanımlama devre dışı bırakıldı.

#### **MQCHAD\_ENABLED**

Kanal otomatik tanımlama etkinleştirildi.

 Bu öznelik yalnızca [Multiplatforms](#) üzerinde desteklenir.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_CHANNEL\_AUTO\_DEF seçiciyi kullanın.

### **ChannelAutoDefEvent (MQlong)**

Bu, kanal otomatik tanımlama olaylarının oluşturulup oluşturulmayacağını denetler. Bu, MQCHT\_RECEIVER, MQCHT\_SVRCONN ve MQCHT\_CLUSSDR tipi kanallara uygulanır. Değer aşağıdakilerden biridir:

#### **MQEVR\_DISABLE**

Olay raporlama devre dışı bırakıldı.

#### **MQEVRENABLED**

Olay raporlaması etkinleştirildi.

Olaylarla ilgili daha fazla bilgi için [Olay izleme](#) başlıklı konuya bakın.

 Bu öznelik yalnızca [Multiplatforms](#) üzerinde desteklenir.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_CHANNEL\_AUTO\_DEF\_EVENT seçiciyi kullanın.


### **ChannelAutoDefExit (MQCHARn)**

Bu, otomatik kanal tanımlamasına ilişkin kullanıcı çıkışıdır. Bu ad boş değilse ve *ChannelAutoDef* MQCHAD\_ENABLED değeri varsa, kuyruk yöneticisinin kanal tanımlaması yaratmak üzere olduğu her saat çıkışı çağrılır. Bu, MQCHT\_RECEIVER, MQCHT\_SVRCONN ve MQCHT\_CLUSSDR tipi kanallara uygulanır. Bundan sonra, çıkış aşağıdakilerden birini yapabilir:

- Kanal tanımlamasını değiştirmeden yaratın.
- Oluşturulan kanal tanımlamasının özneliklerini değiştirin.

- Kanalin oluşturulmasını tamamen engelle.

**Not:** Bu özniteliğin hem uzunluk hem de değeri, ortama özgü bir değerdir. Çeşitli ortamlardaki bu özniteliğin değerinin ayrıntıları için, “MQCD-Kanal tanımlaması” sayfa 1456 içindeki MQCD yapısına giriş yapın.

 z/OS üzerinde bu öznitelik yalnızca küme gönderici ve küme alıcı kanalları için geçerlidir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQCA\_CHANNEL\_AUTO\_DEF\_EXIT seçiciyi kullanın. Bu özniteliğin uzunluğu, MQ\_EXIT\_NAME\_LENGTH tarafından verilir.

### **ChannelEvent (MQUZE)**

Bu, kanal olaylarının oluşturulup oluşturulmayacağını belirtir.

Bu değer aşağıdaki değerlerden biridir:

#### **MQEVR\_EXCEPTION**

Yalnızca aşağıdaki kanal olaylarını oluşturun:

- MQRC\_CHANNEL\_ACTIVATED
- MQRC\_CHANNEL\_CONV\_ERROR
- MQRC\_CHANNEL\_NOT\_ETKINED
- MQRC\_CHANNEL\_DURDURDU, şu ReasonQualifiers:

MQRQ\_CHANNEL\_STOPPED\_ERROR  
MQRQ\_CHANNEL\_STOPPED\_RETRY  
MQRQ\_CHANNEL\_STOPPED\_DISABET

MQRC\_CHANNEL\_STOPPED\_BY\_USER

#### **MQEVRENABLED**

Tüm kanal olaylarını oluşturun. Kural Dışı Durum (EXCEPTION) tarafından oluşturulanların yanı sıra, aşağıdaki kanal olaylarını oluşturun:

- MQRC\_CHANNEL\_STARTED
- MQRC\_CHANNEL\_DURG, şu ReasonQualifier ile durduruldu:

MQRQ\_CHANNEL\_STOPPED\_OK

#### **MQEVR\_DISABLE**

Kanal olayları oluşturmayın; varsayılan değer budur.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_CHANNEL\_EVENT seçiciyi kullanın.

### **ChannelInitiatorControl (MQlong)**

Bu, kuyruk yöneticisi başlatıldığında kanal başlatıcının başlatılıp başlatılmayacağını belirtir.

Bu değer aşağıdaki değerlerden biridir:

#### **MQSVC\_CONTROL\_MANUAL**

Kanal başlatıcı otomatik olarak başlatılamaz.

#### **MQSVC\_CONTROL\_Q\_MGR**

Kuyruk yöneticisi başlatıldığında, kanal başlatıcı otomatik olarak başlatılabilecektir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_CHINIT\_CONTROL seçiciyi kullanın.

### **ChannelMonitoring (MQUZE)**

Bu öznitelik, kanallara ilişkin çevrimiçi izleme verilerini belirtir.

Değer aşağıdakilerden biridir:

### **MQMON\_NONE**

MONCHL kanal özniteliğinin ayarından bağımsız olarak, tüm kanallara ilişkin kanal izleme için veri toplama işlemini geçersiz kılın. Bu varsayılan değerdir.

### **MQMON\_OFF**

MONCHL kanal öznitelideki QMGR değerini belirten kanallara ilişkin izleme verileri toplama işlemi kapatılmasına neden olur.

### **MQMON\_LOW**

MONCHL kanal öznitelide QMGR belirten kanallar için veri toplama oranı düşük oranlı bir izleme verileri toplama işlemi sağlar.

### **MQMON\_ORTAMı**

MONCHL kanal öznitelide QMGR belirten kanallar için, veri toplama işlemi için ılımlı bir veri toplama işlemi ile ilgili izleme verileri toplama seçeneğini açın.

### **MQMON\_YÜKSEK**

MONCHL kanal öznitelide QMGR belirten kanallar için veri toplama oranı yüksek olan veri toplama verilerini izleme.



z/OS sistemlerinde, bu parametrenin etkinleştirilmesi, seçtiğiniz değerden bağımsız olarak istatistik verilerinin toplanmasını açar. LOW, MEDIUMya da HIGH değerinin belirlenmesi sonuçlarınız için fark yaratmaz.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_MONITORING\_CHANNEL seçiciyi kullanın.

## ***ChannelStatistics (MQUZE)***

Bu, kanallara ilişkin istatistik verilerinin toplanmasını denetler.

Değer aşağıdakilerden biridir:

### **MQMON\_NONE**

STATCHL kanal özniteliğinin ayarından bağımsız olarak, tüm kanallara ilişkin kanal istatistikleri için veri toplama işlemini geçersiz kılın. Bu varsayılan değerdir.

### **MQMON\_OFF**

STATCHL kanal öznitelideki QMGR değerini belirten kanallar için istatistik verileri derlemeni kapatın.

### **MQMON\_LOW**

İstatistik verileri derlemeni, STATCHL kanal öznitelideki QMGR belirten kanallar için düşük bir veri toplama oranına sahip olarak döndürün.

### **MQMON\_ORTAMı**

İstatistik verileri derlemeni, STATCHL kanal öznitelideki QMGR belirten kanallar için ılımlı bir veri toplama oranına sahip olarak döndürün.

### **MQMON\_YÜKSEK**

İstatistik verileri derlemeni, STATCHL kanal öznitelideki QMGR belirten kanallar için yüksek bir veri toplama oranına sahip olarak döndürün.

Çoğu sistem için, MEDIUM kullanmanız önerilir. Ancak, her saniye yüksek bir ileti hacmi işleyen bir kanal için, LOW (Düşü) seçeneğini belirleyerek örnekleme düzeyini azaltmak isteyebilirsiniz. Ayrıca, yalnızca birkaç ileti işleyen bir kanal için ve en güncel bilgilerin önemli olduğu bir kanal için HIGH (Yüksek) seçeneğini belirlemek isteyebilirsiniz.



z/OS sistemlerinde, bu parametrenin etkinleştirilmesi, seçtiğiniz değerden bağımsız olarak istatistik verilerinin toplanmasını açar. LOW, MEDIUMya da HIGH değerinin belirlenmesi sonuçlarınız için fark yaratmaz. Kanal muhasebe kayıtlarını toplamak için bu parametre etkinleştirilmelidir.

Bu özniteliğin değerini saptamak için MQINQ çağrısına sahip MQIA\_STATISTICS\_CHANNEL seçiciyi kullanın.

## ***ChinitAdapters (MQUZE)***

Bu, IBM MQ çağrılarını işlemek için kullanılacak bağdaştırıcı alt görevlerinin sayısıdır. Değer 0-9999 arasında olmalıdır; varsayılan değer 8 olmalıdır.

Bağdaştırıcıların dağıtıcılara oranı ( ChinitDispatchers özniteliği) 8-5 arasında olmalıdır. Ancak, yalnızca birkaç kanalınız varsa, bu parametrenin değerini varsayılan değerden düşürmenize gerek yoktur. Bir test sistemi için 8 (varsayılan değer); bir üretim sistemi için 20 değerini kullanabilirsiniz. İdeal olarak, IBM MQ çağrılarında daha fazla koşutluk sağlayan 20 bağdaştırıcıya sahip olmalısınız. Bu, kalıcı iletiler için önemlidir. Kalıcı olmayan iletiler için daha az bağdaştırıcı daha iyi olabilir.

Bu öznitelik yalnızca z/OS üzerinde desteklenir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_CHINIT\_ADAPTERS seçiciyi kullanın.

### ***ChinitDispatchers (MQUZA)***

Bu, kanal başlatıcısı için kullanılacak dağıtıcıların sayısıdır. Varsayılan değer olan 5 olan 0-9999 arasında bir değer olmalıdır.

Bir kılavuz olarak, 50 akım kanalı için bir dağıtıcıya izin verin. Ancak, yalnızca birkaç kanalınız varsa, bu özniteliğin değerini varsayılan değerden düşürmek zorunda kalmayınız. TCP/IP kullanıyorsanız, burada daha büyük bir değer belirtseniz bile, TCP/IP kanalları için kullanılan dağıtıcıların en çok sayısı 100 'dür. Aşağıdaki ayarları kullanabilirsiniz: test sistemleri, 5 (varsayılan); üretim sistemleri, 20 (1000 etkin kanala kadar işlem yapmak için 20 dağıtıcıya ihtiyacınız vardır).

Bu öznitelik yalnızca z/OS üzerinde desteklenir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_CHINIT\_DISTRIBUTOR seçiciyi kullanın.

### ***ChinitTraceAutoStart (MQUZE)***

Bu, kanal başlatıcı izinin otomatik olarak başlatılıp başlatılmayacağını belirtir.

Değer aşağıdakilerden biridir:

#### **MQTRAXSTR\_YES**

Kanal başlatıcı izlemesini otomatik olarak başlat. Bu varsayılan değerdir.

#### **MQTRAXSTR\_NO**

Kanal başlatıcı izlemesini otomatik olarak başlatmayın.

Bu öznitelik yalnızca z/OS üzerinde desteklenir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_CHINIT\_TRACE\_AUTO\_START seçiciyi kullanın.

### ***ChinitTraceTableSize (MQUZE)***

Bu, kanal başlatıcısının izleme verileri alanının (MB) büyüklüğünün büyüklüğüdür.

Değer, 0 ile 2048 arasında bir değer olmalı ve varsayılan değer 2 olmalıdır.

**Not:** Büyük z/OS veri alanlarını her kullandığınızda, ilgili z/OS sayfalama etkinliğini desteklemek için sisteminizde yeterli yardımcı depolama alanına sahip olmadığınızdan emin olun. Ayrıca, SYS1.DUMP veri kümeleri.

Bu öznitelik yalnızca z/OS üzerinde desteklenir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_CHINIT\_TRACE\_TABLE\_SIZE seçiciyi kullanın.

### ***ClusterSenderMonitoringDefault (MQLONG)***

Bu, otomatik olarak tanımlanmış küme gönderen kanallarının ChannelMonitoring özniteliği için yerine konabilecek değeri belirtir.

Değer aşağıdakilerden biridir:

#### **MQMON\_Q\_MGR**

Çevrimiçi izleme verilerinin toplanması, kuyruk yöneticisi **ChannelMonitoring** özniteliğinin ayarından devralınır. Bu varsayılan değerdir.

**MQMON\_OFF**

Kanala ilişkin izleme geçersiz kılındı

**MQMON\_LOW**

*ChannelMonitoring* MQMON\_NONE ise, izleme, sistem başarımı üzerinde en az etkiye sahip bir düşük veri toplama oranı ile etkinleştirilir. Toplanan veriler büyük olasılıkla en güncel veri kaynağı değil.

**MQMON\_ORTAMı**

*ChannelMonitoring* MQMON\_NONE ise, izleme, sistem başarımı üzerinde sınırlı etkili olan bir veri toplama işlemi hızının etkinleştirilmesini sağlar.

**MQMON\_YÜKSEK**

*ChannelMonitoring* MQMON\_NONE ise, izleme, yüksek oranda veri toplama işlemi uygulanmıyorsa, sistem başarımı üzerinde etkili bir etki sağlar. Toplanan veriler, en güncel kullanılabilir verilerdir.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_MONITORING\_AUTO\_CLUSSDR seçiciyi kullanın.

**ClusterSenderStatistics (MQUZE)**

Küme gönderen kanalları, havuzda CLUSRCVR tanımından otomatik olarak tanımlanabileceğinden, ALTER kanalını kullanarak bu otomatik olarak tanımlanmış küme gönderen kanallarına ilişkin STATCHL özneliğinin ayarını değiştiremezsiniz. Bu kanallar için, çevrimiçi izleme verilerinin toplanıp toplanmayacağı konusunda karar, bu kuyruk yöneticisi özneliğinin ayarına dayalıdır.

Değer aşağıdakilerden biridir:

**MQMON\_Q\_MGR**

Otomatik olarak tanımlanmış küme gönderen kanallarına ilişkin istatistik verileri derlemi, kuyruk yöneticisi özneliği STATCHL ' nin değerine dayalıdır. Bu varsayılan değerdir.

**MQMON\_OFF**

Otomatik tanımlı küme gönderen kanalları için istatistik verileri derlemine kapatın.

**MQMON\_LOW**

Veri toplama oranı düşük olan otomatik olarak tanımlanmış küme gönderen kanalları için istatistik verileri toplamasını etkinleştirin.

**MQMON\_ORTAMı**

Veri toplama orta oranı olan, otomatik olarak tanımlanmış küme gönderen kanalları için istatistik verileri toplamasını etkinleştirin.

**MQMON\_YÜKSEK**

Veri toplama oranı yüksek olan otomatik tanımlı küme gönderen kanalları için istatistik verileri toplamasını etkinleştirin.

Çoğu sistem için MEDIUM öneriyoruz. Ancak, her saniye yüksek bir ileti hacmi işleyen otomatik tanımlı bir küme gönderen kanalı için, LOW (Düşü) seçeneğini belirleyerek örnekleme düzeyini azaltmak isteyebilirsiniz. Ayrıca, yalnızca birkaç ileti işleyen bir kanal için ve en güncel bilgilerin önemli olduğu bir kanal için HIGH (Yüksek) seçeneğini belirlemek isteyebilirsiniz.



z/OS sistemlerinde, bu parametrenin etkinleştirilmesi, seçtiğiniz değerden bağımsız olarak istatistik verilerinin toplanmasını açar. LOW, MEDIUM ya da HIGH değerinin belirlenmesi sonuçlarınız için fark yaratmaz. Kanal muhasebe kayıtlarını toplamak için bu parametre etkinleştirilmelidir.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_STATISTICS\_AUTO\_CLUSTSDR seçiciyi kullanın.

**ClusterWorkloadVerileri (MQCHAR32)**

Bu, çağrıldığında küme iş yükü çıkışa geçilen, kullanıcı tanımlı 32 baytlık karakter dizilimidir. Çıkışa geçilecek veri yoksa, dizgi boş olur.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQCA\_CLUSTER\_WORKLOAD\_DATA seçicisini kullanın.

### **ClusterWorkloadExit (MQCHARn)**

Bu, küme iş yükü yönetimine ilişkin kullanıcı çıkışıdır. Bu ad boş değilse, çıkış, bir küme kuyruğuna her ileti konursa ya da bir kümeden diğerine taşındığında çağrılır. Çıkış, ileti için hedef olarak kuyruk yöneticisi tarafından seçilen kuyruk örneğini kabul edebilir ya da başka bir kuyruk eşgörünümü seçebilirler.

**Not:** Bu özniteliğin hem uzunluk hem de değeri, ortama özgü bir değerdir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQCA\_CLUSTER\_WORKLOAD\_EXIT seçiciyi kullanın. Bu özniteliğin uzunluğu, MQ\_EXIT\_NAME\_LENGTH tarafından verilir.

### **ClusterWorkloadUzunluğu (MQUZE)**

Bu, küme iş yükü çıkışa geçirilen ileti verilerinin uzunluk üst sınısıdır. Çıkışa aktarılan verilerin gerçek uzunluğu, aşağıdakilerin alt sınırlarından biri olur:

- İletinin uzunluğu.
- Kuyruk yöneticisinin **MaxMsgLength** özniteliği.
- **ClusterWorkloadLength** özniteliği.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_CLUSTER\_WORKLOAD\_LENGTH seçiciyi kullanın.

### **CLWLMRUChannels (MQUZE)**

Bu, küme iş yükü seçimi algoritması tarafından kullanılmak üzere değerlendirilmek üzere en son kullanılan en çok kullanılan küme kanallarının sayısını belirtir.

Bu değer, 1-999999999 aralığında bir değerdir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_CLWL\_MRU\_CHANGES seçiciyi kullanın.

### **CLWLUseQ (MQUSEL)**

Bu, küme iş yükü için uzak kuyrukların kullanılmayacağını belirtir.

Değer aşağıdakilerden biridir:

#### **MQCLWL\_USEQ\_ANY**

Hem yerel, hem de uzak kuyrukları kullanın.

#### **MQCLWL\_USEQ\_LOCAL**

Uzak kuyrukları kullanmayın. Bu varsayılan değerdir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_CLWL\_USEQ seçiciyi kullanın.

### **CodedCharSetId (MQHOMER)**

Bu, kuyruk yöneticisi tarafından, nesnelerin adları ve kuyruk yaratma tarihi ve saati gibi, MQI ' de tanımlı olan tüm karakter dizgisi alanları için kuyruk yöneticisi tarafından kullanılan karakter kümesini tanımlar. Karakter takımı, nesne adlarında geçerli olan karakterler için tek byte 'lık karakterler içeren bir karakter takımı olmalıdır. İletide taşınan uygulama verileri için geçerli değildir. Değer, ortama bağlıdır:

- z/OS üzerinde, kuyruk yöneticisi başlatıldığında değer sistem değiştiregelerinden ayarlanır; varsayılan değer 500 'dür.
- Windows üzerinde değer, kuyruk yöneticisini yaratan kullanıcının birincil CODEPAGE değeridir.
- IBM üzerinde, kuyruk yöneticisi ilk yaratıldığında, bu değer, ortamdaki ayarlandır.
- AIX and Linux' ta değer, kuyruk yöneticisini yaratan kullanıcının yerel ayarı için varsayılan CODESET değeridir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_CODED\_CHAR\_SET\_ID seçicisini kullanın.



## **CommandEvent (MQUZE)**

Bu, komut olaylarının oluşturulup oluşturulmayacağını belirtir:

### **MQEVR\_DISABLE**

Komut olayları oluşturmayın. Bu varsayılandır.

### **MQEVRENABLED**

Komut olayları oluşturur.

### **MQEVR\_NO\_DISPLAY**

MQINQ dışında başarılı olan tüm komutlar için komut olayları oluşturulur.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_COMMAND\_OLAY seçiciyi kullanın.

## **CommandInputQName (MQCHAR48)**

Bu ad, yerel kuyruk yöneticisinde tanımlı olan komut giriş kuyruğunun adıdır. Bu, kullanıcıların komut gönderebileceği bir kuyruğdur (eğer yetkiniz varsa). Kuyruğun adı ortama bağlıdır:

- z/OS' ta, kuyruğun adı SYSTEM.COMMAND.INPUT; MQSC ve PCF komutları bu komutlara gönderilebilir. MQSC komutlarının ayrıntıları için bkz. [MQSC komutları](#) ve PCF komutlarının ayrıntıları için [Programlanabilir Komut Biçimlerinin Tanımlamaları](#) .
- Diğer tüm ortamlarda, kuyruğun adı SYSTEM.ADMIN.COMMAND.QUEUE(Kuyruk) ve yalnızca PCF komutları gönderilebilir. Ancak, MQSC komutu MQCMD\_ESCAPE tipi bir PCF komutu içindeyse bu kuyruğa bir MQSC komutu gönderilebilir. Escape komutuyla ilgili bilgi için [Escape \(Esc\)](#) tuşuna bakın.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQCA\_COMMAND\_INPUT\_Q\_NAME seçiciyi kullanın. Bu özniteliğin uzunluğu MQ\_Q\_NAME\_LENGTH tarafından verilir.

## **CommandLevel (MQUZE)**

**Not:** Sunucu ve istemciler de içinde olmak üzere tüm IBM MQ bileşenleri için HP-UX işletim sistemine ilişkin destek IBM MQ 9.1' den kaldırılır.

Bu, kuyruk yöneticisi tarafından desteklenen sistem denetimi komutlarının düzeyini gösterir. Bu, aşağıdaki değerlerden biri olabilir:

### **MQCMDL\_LEVEL\_800**

Sistem denetimi komutlarının 800. düzeyi.

Bu değer aşağıdaki sürümler tarafından döndürülür:

- IBM MQ for AIX 8.0
- IBM MQ for IBM i 8.0
- IBM MQ for Linux 8.0
- IBM MQ for Windows 8.0
- IBM MQ for z/OS 8.0

### **MQCMDL\_LEVEL\_801**

Sistem denetimi komutlarının 801 düzeyi.

Bu değer aşağıdaki sürümler tarafından döndürülür:

- IBM MQ for AIX 8.0.0 Fix Pack 2
- IBM MQ for HP-UX 8.0.0 Fix Pack 2
- IBM MQ for IBM i 8.0.0 Fix Pack 2
- IBM MQ for Linux 8.0.0 Fix Pack 2

### **MQCMDL\_LEVEL\_802**

Sistem denetimi komutlarının 802 düzeyi.

Bu değer aşağıdaki sürümler tarafından döndürülür:

- IBM MQ for AIX 8.0.0 Fix Pack 3
- IBM MQ for IBM i 8.0.0 Fix Pack 3
- IBM MQ for Linux 8.0.0 Fix Pack 3
- IBM MQ for Windows 8.0.0 Fix Pack 3

#### **MQCMDL\_LEVEL\_900**

Sistem denetimi komutlarının 900. düzeyi.

Bu değer aşağıdaki sürümler tarafından döndürülür:

- IBM MQ for AIX 9.0
- IBM MQ for IBM i 9.0
- IBM MQ for Linux 9.0
- IBM MQ for Windows 9.0
- IBM MQ for z/OS 9.0

#### **MQCMDL\_LEVEL\_901**

Sistem denetimi komutlarının 901 düzeyi.

Bu değer aşağıdaki sürümler tarafından döndürülür:

- IBM MQ for Linux 9.0.1
- IBM MQ for Windows 9.0.1
- IBM MQ for z/OS 9.0.1

#### **MQCMDL\_LEVEL\_902**

Sistem denetimi komutlarının 902 düzeyi.

Bu değer aşağıdaki sürümler tarafından döndürülür:

- IBM MQ for Linux 9.0.2
- IBM MQ for Windows 9.0.2
- IBM MQ for z/OS 9.0.2

#### **MQCMDL\_LEVEL\_903**

Sistem denetimi komutlarının 903 düzeyi.

Bu değer aşağıdaki sürümler tarafından döndürülür:

- IBM MQ for Linux 9.0.3
- IBM MQ for Windows 9.0.3
- IBM MQ for z/OS 9.0.3

#### **MQCMDL\_LEVEL\_904**

Sistem denetimi komutlarının 904 düzeyi.

Bu değer aşağıdaki sürümler tarafından döndürülür:

- IBM MQ for AIX 9.0.4
- IBM MQ for Linux 9.0.4
- IBM MQ for Windows 9.0.4
- IBM MQ for z/OS 9.0.4

#### **MQCMDL\_LEVEL\_905**

Sistem denetimi komutlarının 905 düzeyi.

Bu değer aşağıdaki sürümler tarafından döndürülür:

- IBM MQ for AIX 9.0.5
- IBM MQ for Linux 9.0.5
- IBM MQ for Windows 9.0.5

- IBM MQ for z/OS 9.0.5

#### **MQCMDL\_LEVEL\_910**

Sistem denetimi komutlarının düzey 910 'unu.

Bu değer aşağıdaki sürümler tarafından döndürülür:

- IBM MQ for AIX 9.1.0
- IBM MQ for IBM i 9.1.0
- IBM MQ for Linux 9.1.0
- IBM MQ for Windows 9.1.0
- IBM MQ for z/OS 9.1.0

#### **MQCMDL\_LEVEL\_911**

Sistem denetimi komutlarının 911 'i.

Bu değer aşağıdaki sürümler tarafından döndürülür:

- IBM MQ for AIX 9.1.1
- IBM MQ for Linux 9.1.1
- IBM MQ for Windows9.1.1
- IBM MQ for z/OS 9.1.1

#### **MQCMDL\_LEVEL\_912**

Sistem denetimi komutlarının 912 düzeyi.

Bu değer aşağıdaki sürümler tarafından döndürülür:

- IBM MQ for AIX 9.1.2
- IBM MQ for Linux 9.1.2
- IBM MQ for Windows9.1.2
- IBM MQ for z/OS 9.1.2

#### **MQCMDL\_LEVEL\_913**

Sistem denetimi komutlarının 913. düzeyi.

Bu değer aşağıdaki sürümler tarafından döndürülür:

- IBM MQ for AIX 9.1.3
- IBM MQ for Linux 9.1.3
- IBM MQ for Windows9.1.3
- IBM MQ for z/OS 9.1.3

#### **MQCMDL\_LEVEL\_914**

Sistem denetimi komutlarının 914. düzeyi.

Bu değer aşağıdaki sürümler tarafından döndürülür:

- IBM MQ for AIX 9.1.4
- IBM MQ for Linux 9.1.4
- IBM MQ for Windows9.1.4
- IBM MQ for z/OS 9.1.4

#### **MQCMDL\_LEVEL\_915**

Sistem denetimi komutlarının 915 düzeyi.

Bu değer aşağıdaki sürümler tarafından döndürülür:

- IBM MQ for AIX 9.1.5
- IBM MQ for Linux 9.1.5
- IBM MQ for Windows9.1.5

- IBM MQ for z/OS 9.1.5

#### **MQCMDL\_LEVEL\_920**

Sistem denetimi komutlarının 920. düzeyi.

Bu değer aşağıdaki sürümler tarafından döndürülür:

- IBM MQ for AIX 9.2.0
- IBM MQ for IBM i 9.2.0
- IBM MQ for Linux 9.2.0
- IBM MQ for Windows 9.2.0
- IBM MQ for z/OS 9.2.0

#### **MQCMDL\_LEVEL\_921**

Sistem denetimi komutlarının 921 'i.

Bu değer aşağıdaki sürümler tarafından döndürülür:

- IBM MQ for AIX 9.2.1
- IBM MQ for Linux 9.2.1
- IBM MQ for Windows9.2.1
- IBM MQ for z/OS 9.2.1

#### **MQCMDL\_LEVEL\_922**

Sistem denetimi komutlarının 922 düzeyi.

Bu değer aşağıdaki sürümler tarafından döndürülür:

- IBM MQ for AIX 9.2.2
- IBM MQ for Linux 9.2.2
- IBM MQ for Windows9.2.2
- IBM MQ for z/OS 9.2.2

#### **MQCMDL\_LEVEL\_923**

Sistem denetimi komutlarının 923 düzeyi.

Bu değer aşağıdaki sürümler tarafından döndürülür:

- IBM MQ for AIX 9.2.3
- IBM MQ for Linux 9.2.3
- IBM MQ for Windows9.2.3
- IBM MQ for z/OS 9.2.3

#### **MQCMDL\_LEVEL\_924**

Sistem denetimi komutlarının 924. düzeyi.

Bu değer aşağıdaki sürümler tarafından döndürülür:

- IBM MQ for AIX 9.2.4
- IBM MQ for Linux 9.2.4
- IBM MQ for Windows9.2.4
- IBM MQ for z/OS 9.2.4

#### **MQCMDL\_LEVEL\_925**

Düzy 925of sistem denetimi komutları.

Bu değer aşağıdaki sürümler tarafından döndürülür:

- IBM MQ for AIX 9.2.5
- IBM MQ for Linux 9.2.5
- IBM MQ for Windows9.2.5

- IBM MQ for z/OS 9.2.5

**CommandLevel** özniteliğinin belirli bir değerine karşılık gelen sistem denetimi komutları kümesi, **Platform** özniteliğinin değerine göre değişiklik gösterir; her ikisi de hangi sistem denetimi komutlarının destekleneceğine karar vermek için kullanılmalıdır.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_COMMAND\_LEVEL seçicisini kullanın.

### **CommandServerDenetimi (MQUZE)**

Kuyruk yöneticisi başlatıldığında, komut sunucusunun başlatılıp başlatılmayacağını belirler.

Değer, aşağıdaki değerlerden herhangi biri olabilir:

#### **MQSVC\_CONTROL\_MANUAL**

Komut sunucusu otomatik olarak başlatılamaz.

#### **MQSVC\_CONTROL\_Q\_MGR**

Kuyruk yöneticisi başlatıldığında, komut sunucusu otomatik olarak başlatılacaktır.

Bu öznitelik z/OS üzerinde desteklenmez.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_CMD\_SERVER\_CONTROL seçiciyi kullanın.

### **ConfigurationEvent (MQUWN)**

Yapılandırma olaylarının oluşturulup oluşturulmadığını denetler.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_CONFIGURATION\_OLAY seçiciyi kullanın.

Değer, aşağıdaki değerlerden herhangi biri olabilir:

#### **MQEVR\_DISABLE**

Olay raporlama devre dışı bırakıldı.

#### **MQEVRENABLED**

Olay raporlaması etkinleştirildi.

### **Multi V 9.2.0 CurrentQFileBoyutu (MQUZE)**

Kuyruk dosyasının megabayt cinsinden yürürlükteki büyüklüğü, en yakın megabayt yuvarlatılmış olarak yuvarlanır.

<i>Çizelge 559. Bu özniteliğin geçerli olduğu kuyruk tipleri</i>				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Bu kuyruk durumu özniteliğine ilişkin değer, kuyruğun şu anda ne boyutta olduğu, en yakın megabayta yuvarlanmış olmalıdır. Varsayılan özniteliklere sahip yeni bir kuyruk için **CurrentQFileSize** değeri 1 olur.

Bu özniteliğin en yüksek değeri 99,999.9999 MB ' dir ve bu öznitelik için varsayılan değer yoktur.

### **Multi V 9.2.0 CurrentMaxQFileSize (MQUBE)**

Kuyrukta kullanılan yürürlükteki blok büyüklüğü göz önüne alındığında, kuyruk dosyasının büyüyebileceği büyüklük üst sınırı en yakın megabyte değerine yuvarlanır.

<i>Çizelge 560. Bu özniteliğin geçerli olduğu kuyruk tipleri</i>				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Bu alanın kullanımı iki katlıdır:

- **MaxQFileSize** değerini geçerli blok boyutu için varsayılan değere ayarlıyorsanız, **CurrentMaxQFileSize** , varsayılan değerini eşitlediği gerçek değeri gösterir.

- **CurrentMaxQFileSize** , **MaxQFileSize** ile eşleşmezse, daha büyük bir ayrıntı düzeyi benimsemek için kuyruğun boşaltılması gerektiğini biliyorsunuz.

**Not:** Kuyruk dosyalarının ve öbek büyüklüğünün ve ayrıntı düzeylerinin büyüklüğünün değiştirilmesine ilişkin ek bilgi için [IBM MQ kuyruk dosyalarının değiştirilmesi](#) başlıklı konuya bakın.

Bu özniteliğin en yüksek değeri 99,999.9999 MB ' dir ve varsayılan değer yoktur. Değer şu anda ayarlanan en yüksek değer her neyse, varsayılan özniteliklere sahip yeni bir kuyruk için **CurrentMaxQFileSize** değeri 2,088,960 MB 'dir.

### **DeadLetterQName (MQCHAR48)**

Bu ad, yerel kuyruk yöneticisinde, ölü-mektup (teslim edilemeyen ileti) kuyruğu olarak tanımlanan bir kuyruğun adıdır. İletiler doğru hedefe yönlendirilemiyorsa, bu kuyruğa ileti gönderilir.

Örneğin, aşağıdaki durumlarda iletiler şu kuyruğa konabiliyor:

- Kuyruk yöneticisinde henüz tanımlı olmayan bir kuyruğa yollayan bir ileti kuyruk yöneticisinde gönderilir.
- Kuyruk yöneticisinde bir ileti gönderilir, ancak gönderileceği kuyruk bunu alamıyor; nedeni şunlar olabilir:
  - Kuyruk dolu
  - Put istekleri engellenir
  - Gönderme düğümünün kuyruğa ileti koyma yetkisi yok

Uygulamalar, ileti kuyruğunda da ileti yerleştirebilir.

Rapor iletileri sıradan iletilerde aynı şekilde işlem görür; rapor letisi hedef kuyruğuna teslim edilemiyorsa (genellikle özgün iletinin ileti tanımlayıcısındaki *ReplyToQ* alanı tarafından belirlenen kuyruk), rapor letisi ölü-mektup (teslim edilemeyen ileti) kuyruğuna yerleştirilir.

**Not:** Süre bitim sürelerini geçen iletiler (bkz. MQMD-Süre bitimi alanı ) **değil** , atıldığında bu kuyruğa aktarılır. Ancak, gönderme uygulaması tarafından istenirse, bir süre bitimi rapor letisi (MQRO\_EXPIRATION) hala oluşturulur ve *ReplyToQ* kuyruğuna gönderilir.

Bir put isteğini yayınlayan uygulama, MQPUT ya da MQPUT1 çağrısı tarafından döndürülen neden kodu yoluyla sorunun zamanuyumlu olarak bildirildiğini bildiren iletiler (teslim edilemeyen ileti) kuyruğunda bulunmaz (örneğin, istekleri koyan yerel bir kuyruğa ileti konması).

Ölü-mektup (teslim edilemeyen ileti) kuyruğunda iletiler bazen bir MQDLH yapısıyla önekli olarak uygulama letisi verilerine sahip olur. Bu yapı, iletinin ölü harf (teslim edilemeyen ileti) kuyruğuna neden yerleştirildiğini gösteren ek bilgiler içerir. Bu yapıyla ilgili daha fazla bilgi için bkz. "[MQDLH-Harf üstbilgisi](#)" sayfa 347 .

This queue must be a local queue, with a **Usage** attribute of MQUS\_NORMAL.

Bir kuyruk yöneticisi, bir ölü-mektubu (teslim edilemeyen ileti) kuyruğunu desteklemiyorsa ya da tanımlanmamışsa, ad boyluk YekliYtir. Tüm IBM MQ kuyruk yöneticileri bir dead-letter (teslim edilemeyen ileti) kuyruğunu destekler, ancak varsayılan olarak tanımlı değildir.

Ölmeyen harf (teslim edilemeyen ileti) kuyruğu başka bir nedenle tanımlı değil, tam ya da kullanılamaz durumda değilse, ileti kanalı aracısı tarafından iletim kuyruğunda yerine bir ileti kanalı aracısı tarafından aktarılmış olacak bir ileti saklanır.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQCA\_DEAD\_LETTER\_Q\_NAME seçiciyi kullanın. Bu özniteliğin uzunluğu MQ\_Q\_NAME\_LENGTH tarafından verilir.

### **DefClusterXmitQueueType (MQUBE)**

DefClusterXmitQueueType özniteliği, iletileri küme alıcı kanallarına göndermek için küme gönderen kanalları tarafından varsayılan olarak hangi iletim kuyruğunun seçileceğini denetler.

**DefClusterXmitQueueType** değerleri MQCLXQ\_SCTQ ya da MQCLXQ\_CHANNEL' dir.

## **MQCLXQ\_SCTQ**

Tüm küme gönderen kanallar SYSTEM . CLUSTER . TRANSMIT . QUEUE' den ileti gönderir. İletim kuyruğuna yerleştirilen iletilerin correlID , iletinin gönderileceği küme gönderen kanalı belirler.

SCTQ , bir kuyruk yöneticisi tanımlandığında ayarlanır. Bu davranış, IBM WebSphere MQ 7.5' dan önceki IBM WebSphere MQ sürümlerinde örtük olarak bulunur. Önceki sürümlerde, kuyruk yöneticisi özniteliği DefClusterXmitQueueType yoktu.

## **MQCLXQ\_CHANNEL**

Her küme gönderen kanal farklı bir iletim kuyruğundan ileti gönderir. Her iletim kuyruğu, SYSTEM . CLUSTER . TRANSMIT . MODEL . QUEUEmodel kuyruğundan kalıcı dinamik bir kuyruk olarak yaratılır.

If the queue manager attribute, DefClusterXmitQueueTürü, is set to KANAL, varsayılan yapılandırma, tek tek küme iletim kuyruklarıyla ilişkilendirilmiş küme gönderen kanalları olarak değiştirilir. İletim kuyrukları, SYSTEM . CLUSTER . TRANSMIT . MODEL . QUEUEmodel kuyruğundan yaratılan kalıcı dinamik kuyruklardır. Her iletim kuyruğu bir küme-gönderen kanalla ilişkilendirilir. Bir küme gönderen kanal bir küme iletim kuyruğuna hizmet sunarken, iletim kuyruğu tek bir kümede yalnızca bir kuyruk yöneticisine ilişkin iletiler içerir. Kümeleri, bir kümedeki her kuyruk yöneticisinin tek bir küme kuyruğu içereceği şekilde yapılandırabilirsiniz. Bu durumda, bir kuyruk yöneticisinden her küme kuyruğuna ileti trafiği iletilerden diğer kuyruklara ayrı olarak aktarılır.

Değeri sorgulamak için MQINQ' u arayın ya da MQIA\_DEF\_CLUSTER\_XMIT\_Q\_TYPE seçicisini ayarlamak için bir Query queue Manager ( MQCMD\_INQUIRE\_Q\_MGR) PCF komutunu gönderin. Değeri değiştirmek için, MQIA\_DEF\_CLUSTER\_XMIT\_Q\_TYPE seçicisini ayarlamak için bir Change Queue Manager ( MQCMD\_CHANGE\_Q\_MGR) PCF komutu gönderin.

### **İlgili başvurular**

[Kuyruk Yöneticisini Değiştir](#)

[Sorgu Kuyruğu Yöneticisi](#)

[“MQINQ-Nesne özniteliklerinin sorgulması” sayfa 693](#)

MQINQ çağrısı, bir tamsayılar dizisi ve bir nesnenin özniteliklerini içeren bir karakter dizgileri kümesini döndürür.

## **DefXmitQName (MQCHAR48)**

Bu ad, hangi iletim kuyruğunun kullanılacağına ilişkin başka bir gösterge yoksa, iletilerin uzak kuyruk yöneticilerine iletilmesi için kullanılan iletim kuyruğunun adıdır.

Varsayılan iletim kuyruğu yoksa, ad tümüyle boştur. Bu özniteliğin ilk değeri boştur.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQCA\_DEF\_XMIT\_QNAME seçicisini kullanın. Bu özniteliğin uzunluğu MQ\_Q\_NAME\_LENGTH tarafından verilir.

## **DistLists (MQUZE)**

Bu, yerel kuyruk yöneticisinin MQPUT ve MQPUT1 çağrılarında dağıtım listelerini destekleyip desteklemediğini gösterir. Bu değer aşağıdaki değerlerden biridir:

### **MQDL\_DESTEKLENEN**

Dağıtım listeleri desteklenir.

### **MQDL\_NOT\_SUPPORTANT**

Dağıtım listeleri desteklenmiyor.

Bu özniteliğin değerini saptamak için, MQINQ çağrısına sahip MQIA\_DIST\_LISTS seçiciyi kullanın.

## **DNSGroup (MQCHAR18)**

Bu parametre artık kullanılmıyor. Bkz. [IBM MQ 8.0içinde değişen özellikler](#).

Bu öznitelik yalnızca z/OS üzerinde desteklenir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQCA\_DNS\_GROUP seçiciyi kullanın. Bu özniteliğin uzunluğu MQ\_DNS\_GROUP\_NAME\_LENGTH tarafından verilir.

### **DNSWLM (MQUZA)**

Bu parametre artık kullanılmıyor. Bkz. [IBM MQ 8.0 içinde değişen özellikler](#).

Değer aşağıdakilerden biridir:

#### **MQDNSWLM\_YES**

Bu değer, daha önceki bir yayından geçirilen bir kuyruk yöneticisinden görülebilir. Değer yoksayılr.

#### **MQDNSWLM\_NO**

Bu, kuyruk yöneticisi tarafından desteklenen tek değerdir.

Bu öznitelik yalnızca z/OS üzerinde desteklenir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_DNS\_WLM seçiciyi kullanın.


### **ExpiryInterval (MQUZE)**

Kuyruk yöneticisinin süresi dolmuş iletileri aramak için kuyrukları tarayan sıklık değeri belirtilir. 1 ile 99 999 999 arasındaki bir aralıktaki bir zaman aralığı ya da aşağıdaki özel değer olabilir:

#### **MQEXPI\_OFF**

Kuyruk yöneticisi, süresi dolan iletileri arayan kuyrukları taramaz.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_EXPIRY\_INTERVAL seçiciyi kullanın.

 Bu öznitelik yalnızca z/OS üzerinde desteklenir.

### **IGQPutAuthority (MQHOT)**

Bu öznitelik yalnızca, yerel kuyruk yöneticisi bir kuyruk paylaşım grubunun bir üyesiye geçerlidir. Yerel grup içi kuyruğa alma aracı (IGQ aracı), paylaşılan iletim kuyruğundan bir iletiyi kaldırdığında ve iletiyi yerel bir kuyruğa yerlediğinde gerçekleştirilen yetki denetimi tipini gösterir. Değer aşağıdakilerden biridir:

#### **MQIGQPA\_VARSAYILAN**

Yetki için denetlenen kullanıcı kimliği, ileti paylaşılan iletim kuyruğunda olduğunda, iletiyle ilişkili ayrı MQMD 'deki *UserIdentifier* alanının değeridir. Bu, iletiyi paylaşılan iletim kuyruğuna yerleştiren programın kullanıcı kimliğidir ve genellikle, uzak kuyruk yöneticisinin altında çalıştığı kullanıcı kimliğiyle aynıdır.

RESFIELL tanıtımı, birden çok kullanıcı kimliğinin denetleneceğini belirtiyorsa, yerel IGQ aracısının (*IGQUserId*) kullanıcı kimliği de denetlenir.

#### **MQIGQPA\_CONTEXT**

Yetki için denetlenen kullanıcı kimliği, ileti paylaşılan iletim kuyruğunda olduğunda, iletiyle ilişkili ayrı MQMD 'deki *UserIdentifier* alanının değeridir. Bu, iletiyi paylaşılan iletim kuyruğuna yerleştiren programın kullanıcı kimliğidir ve genellikle, uzak kuyruk yöneticisinin altında çalıştığı kullanıcı kimliğiyle aynıdır.

RESFIELL tanıtımında birden çok kullanıcı kimliği denetlenecekse, yerel IGQ aracısının (*IGQUserId*) kullanıcı kimliği ve *embedded* MQMD 'deki *UserIdentifier* alanının değeri de denetlenir. İkinci kullanıcı kimliği genellikle, iletiyi oluşturan uygulamanın kullanıcı kimliğidir.

#### **MQIGQPA\_ONLY\_IGQ**

Yetki için denetlenen kullanıcı kimliği, yerel IGQ aracısının (*IGQUserId*) kullanıcı kimliğidir.

RESFIELL tanıtımı, birden çok kullanıcı kimliğinin denetleneceğini belirtiyorsa, bu kullanıcı kimliği tüm denetimler için kullanılır.

#### **MQIGQPA\_ALTERNATE\_OR\_IGQ**

Yetki için denetlenen kullanıcı kimliği, yerel IGQ aracısının (*IGQUserId*) kullanıcı kimliğidir.



RESFIELL tanıtımı birden çok kullanıcı kimliğinin denetleneceğini gösteriyorsa, *yerleşik* MQMD ' deki *UserIdentifier* alanının değeri de denetlenir. Bu kullanıcı kimliği genellikle, iletiyi oluşturan uygulamanın kullanıcı kimliğidir.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_IGQ\_PUT\_AUTHORITY seçicisini kullanın.

**z/OS** Bu öznelik yalnızca z/OS üzerinde desteklenir.

### **IGQUserId (MQUZN)**

Bu öznelik, yerel kuyruk yöneticisi bir kuyruk paylaşım grubunun üyesiye geçerlidir. Yerel grup içi kuyruğa alma aracısıyla (IGQ aracı) ilişkilendirilmiş olan kullanıcı kimliğini belirtir. Bu tanıtıcı, IGQ aracı yerel kuyruklara ileti yerleştirdiğinde yetki için denetlenebilecek kullanıcı tanıtıcılarından biridir. Denetlenen gerçek kullanıcı tanıtıcıları, **IGQPutAuthority** özneliğinin ayarına ve dış güvenlik seçeneklerine bağlıdır.

*IGQUserId* boşsa, IGQ aracısıyla herhangi bir kullanıcı kimliği ilişkilendirilmemiş ve ilgili yetki denetimi gerçekleştirilmez (ancak, diğer kullanıcı kimlikleri yetki için hala denetlenebilir).

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQCA\_IGQ\_USER\_ID seçiciyi kullanın. Bu özneliğin uzunluğu MQ\_USER\_ID\_LENGTH tarafından verilir.

**z/OS** Bu öznelik yalnızca z/OS üzerinde desteklenir.

### **InhibitEvent (MQUBE)**

Bu, Inhibit (Inhibit Get ve Inhibit Put) olaylarının engelleyip üretilmediğini denetler. Değer aşağıdakilerden biridir:

#### **MQEVR\_DISABLE**

Olay raporlama devre dışı bırakıldı.

#### **MQEVRENABLED**

Olay raporlaması etkinleştirildi.

Olaylarla ilgili daha fazla bilgi için [Olay izleme](#) başlıklı konuya bakın.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_INHIBIT\_EVENT seçiciyi kullanın.

z/OS' ta, bu özneliğin değerini saptamak için MQINQ çağrısını kullanamazsınız.

### **IntraGroupqueuing (MQUZE)**

Bu öznelik yalnızca, yerel kuyruk yöneticisi bir kuyruk paylaşım grubunun bir üyesiye geçerlidir. Kuyruk paylaşım grubu için grup içi kuyruğa alma özelliğinin etkinleştirilip etkinleştirilmediğini belirtir. Değer aşağıdakilerden biridir:

#### **MQIGQ\_DISABLE**

Kuyruk paylaşım grubundaki diğer kuyruk yöneticilerine gönderilen tüm iletiler geleneksel kanallar kullanılarak iletilir.

#### **MQIGQ\_ENABLED**

Kuyruk paylaşım grubundaki diğer kuyruk yöneticilerine gönderilen iletiler, aşağıdaki koşul yerine getirildiyse, paylaşılan iletim kuyruğu kullanılarak iletilir:

- İleti verileri artı iletim üstbilgisinin uzunluğu 63 KB ' yi (64 512 bayt) aşmaz.

İletim üstbilgisi için MQXQH büyüklüğünden biraz daha fazla alan ayrılması önerilir; bu amaçla MQ\_MSG\_HEADER\_LENGTH değışmezi sağlanmıştır.

Bu koşul karşılanmazsa, ileti geleneksel kanallar kullanılarak iletilir.

**Not:** Grup içi kuyruğa alma etkinleştirildiğinde, paylaşılan iletim kuyruğu kullanılarak iletilen iletilerin sırası, konvansiyonel kanallar kullanılarak iletilen iletilerle görel olarak korunmaz.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_INTRA\_GROUP\_queuing seçiciyi kullanın.

Bu öznitelik yalnızca z/OSüzerinde desteklenir.

### ***IPAddressVersion (MQUZE)***

Hangi IP adresi sürümünü ( IPv4 ya da IPv6) kullanılacağını belirtir.

Bu öznitelik yalnızca hem IPv4 , hem de IPv6 çalıştıran sistemler için geçerlidir ve aşağıdaki koşullardan biri geçerli olduğunda yalnızca *TransportType* MQXPY\_TCP olarak tanımlanmış kanalları etkiler:

- Kanal *ConnectionName* , hem IPv4 , hem de IPv6 adresine ve **LocalAddress** parametresine çözülen bir anasistem adıdır.
- Kanal *ConnectionName* ve *LocalAddress* , hem IPv4 , hem de IPv6 adreslerine çözümleyen anasistem adlarıdır.

Değer, aşağıdaki değerlerden herhangi biri olabilir:

#### **MQIPADDR\_IPV4**

IPv4 kullanılır.

#### **MQIPADDR\_IPV6**

IPv6 kullanılır.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_IP\_ADDRESS\_VERSION seçiciyi kullanın.

### ***ListenerTimer (MQUZE)***

Bu, bir APPC ya da TCP/IP hatası olması durumunda, IBM MQ dinleyicisini yeniden başlatma girişimleri arasındaki zaman aralığından (saniye cinsinden). Değer, 5 ile 9999 arasında olmalıdır; varsayılan değer 60 'dır.

Bu öznitelik yalnızca z/OS üzerinde desteklenir.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_LISTENER\_TIMER seçiciyi kullanın.

### ***LocalEvent (MQUZE)***

Bu, yerel hata olaylarının oluşturulup oluşturulmadığını denetler. Değer aşağıdakilerden biridir:

#### **MQEVR\_DISABLE**

Olay raporlama devre dışı bırakıldı.

#### **MQEVRENABLED**

Olay raporlaması etkinleştirildi.

Olaylarla ilgili daha fazla bilgi için [Olay izleme](#) başlıklı konuya bakın.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_LOCAL\_EVENT seçiciyi kullanın.

z/OS' ta, bu özneliğin değerini saptamak için MQINQ çağrısını kullanamazsınız.

### ***LoggerEvent (MQUZE)***

Bu, kurtarma günlüğü olaylarının oluşturulup oluşturulmayacağını denetler. Değer aşağıdakilerden biridir:

#### **MQEVR\_DISABLE**

Olay raporlama devre dışı bırakıldı.

#### **MQEVRENABLED**

Olay raporlaması etkinleştirildi.

Olaylarla ilgili daha fazla bilgi için [Olay izleme](#) başlıklı konuya bakın.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_LOGGER\_EVENT seçiciyi kullanın.

Bu öznitelik yalnızca [Multiplatforms](#) üzerinde desteklenir.

### ***LUGroupName (MQCHAR8)***

Bu ad, kuyruk paylaşım grubu için gelen iletimleri işleyen LU 6.2 dinleyicisinin genel LU adıdır. Bu adı boş bırakırsanız, bu dinleyiciyi kullanamazsınız.

Bu öznitelik yalnızca z/OS üzerinde desteklenir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQCA\_LU\_GROUP\_NAME seçiciyi kullanın. Bu özniteliğin uzunluğu MQ\_LU\_NAME\_LENGTH tarafından verilir.

### **LUAdı (MQCHAR8)**

Bu, giden LU 6.2 iletimleri için kullanılacak LU ' nun adıdır. Bunu dinleyici tarafından gelen iletimler için kullandığı LU ' ya ayarlayın. Bu adı boş bırakırsanız, APPC/MVS varsayılan LU kullanılır; bu bir deyişkendir; bu nedenle, LU6.2 kullanıyorsanız, LUName değerini her zaman ayarlayın.

Bu öznitelik yalnızca z/OS üzerinde desteklenir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQCA\_LU\_NAME seçiciyi kullanın. Bu özniteliğin uzunluğu MQ\_LU\_NAME\_LENGTH tarafından verilir.

### **LU62ARMSuffix (MQCHAR2)**

Bu, SYS1.PARMLIB üyesi APPCPMxx, bu kanal başlatıcısı için LUADD ' yi (LUADD) atamaz. ARM kanal başlatıcısını yeniden başlattığında, SET APPC=xx z/OS komutu verilir. Bu adı bırakırsanız, SET APPC=xx komutu verilmez.

Bu öznitelik yalnızca z/OS üzerinde desteklenir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQCA\_LU62\_ARM\_SUFFIX seçicisini kullanın. Bu özniteliğin uzunluğu MQ\_ARM\_SUFFIX\_LENGTH tarafından verilir.

### **LU62Channels (MQUZE)**

Bu, LU 6.2 iletim protokolünü kullanan, geçerli olabilecek kanal sayısı üst sınırı ya da bağlanabilen istemci sayısı üst sınışıdır.

Değer, 0-9999 aralığında olmalıdır; varsayılan değer 200 olmalıdır. Bu değeri sıfır olarak ayarladıysanız, LU 6.2 iletim protokolü kullanılmaz.

Bu öznitelik yalnızca z/OS üzerinde desteklenir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_LU62\_CHANNELS seçicisini kullanın.

### **MaxActiveKanalları (MQUZE)**

Bu öznitelik, herhangi bir zamanda *etkin* olabilecek kanal sayısı üst sınışıdır.

Varsayılan değer, MaxChannels özniteliği için belirtilen değerdir.

z/OS için, değer 1-9 999 aralığında olmalıdır.

Diğer tüm altyapılar için varsayılan değer 999 999 999 'tır. Bu değer, etkin kanal sayısının sınırsız olduğu ya da bir sınır koymak için gerçek bir sayıya ayarlanabileceği anlamına gelir.

**MQ Appliance** IBM MQ Appliance üzerindeki **MaxActiveChannels** değerini değiştirmemelisiniz. İstemci kanalı sayısı üst sınırını sınırlamak istiyorsanız, her bir SVRCONN kanalı için sınır tanımlamak üzere SVRCONN kanalı tanımlarındaki kanal başına MAXINST ve MAXINSTC özniteliklerini kullanın, IBM MQ Appliance belgelerinde IBM MQ Appliance' ta kuyruk yöneticisi yapılandırması başlıklı konuya bakın.

**MaxActiveChannels** parametresi, yalnızca z/OS üzerinde bir kuyruk yöneticisi öznitesidir. Diğer platformlarda **MaxActiveChannels**, qm. ini dosyasındaki bir öznitedir. Diğer platformlarda **MaxActiveChannels** özniteliğini ayarlamaya ilişkin bilgi için bkz. Distributed queuing için yapılandırma dosyası stanzas for distributed queuing.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_ACTIVE\_CHANGES seçiciyi kullanın.

### **İlgili kavramlar**

Kanal durumları

## **MaxChannels (MQUZE)**

Bu öznitelik, *yürürlükteki* olabilecek kanal sayısı üst sınısıdır (bağlı istemcilere sahip sunucu bağlantı kanalları da içinde olmak üzere).

z/OS için, varsayılan değer 200 olan 1 ile 9 999 arasında bir değer olmalıdır.

**MQ Appliance** IBM MQ Appliance için varsayılan değer 999 999 999 'tır ve değiştirilmemelidir. İstemci kanalı sayısı üst sınırını sınırlamak istiyorsanız, her bir SVRCONN kanalı için sınır tanımlamak üzere SVRCONN kanalı tanımlarındaki kanal başına MAXINST ve MAXINSTC özniteliklerini kullanın, IBM MQ Appliance belgelerinde [IBM MQ Appliance' ta kuyruk yöneticisi yapılandırması](#) başlıklı konuya bakın.

Ağdan bağlantı hizmeti sunmakla meşgul olan bir sistemin varsayılan ayardan daha yüksek bir sayıya ihtiyacı olabilir. Test sırasında sisteminizin davranışını gözlemleyerek, ortamınız için doğru olan değeri belirleyin.

Diğer tüm altyapılar için varsayılan değer 100 'dür. **MaxChannels** değerini, gerekiyorsa, yürürlükteki kanal sayısı üst sınırını sınırlamak için farklı bir değer olarak ayarlayabilirsiniz.

**MaxChannels** parametresi, yalnızca z/OS üzerinde bir kuyruk yöneticisi öznitesidir. Diğer platformlarda **MaxChannels** , qm . ini dosyasındaki bir öznitedir. Diğer platformlarda **MaxChannels** öznitelikliğini ayarlamaya ilişkin bilgi için bkz. [Distributed queuing için yapılandırma dosyası stanzas for distributed queuing](#) .

Bu öznitelikliğini değeri saptamak için, **MQINQ** çağrısıyla MQIA\_MAX\_CHANGES seçiciyi kullanın.

### **İlgili kavramlar**

[Kanal durumları](#)

## **MaxHandles (MQUZE)**

Bu, herhangi bir görevin eşzamanlı olarak kullanabileceği açık tutamaç sayısı üst sınısıdır. Tek bir kuyruk (ya da kuyruk olmayan bir nesne için) için her başarılı MQOPEN çağrısı tek bir tanıtıcı kullanır. Bu tanıtıcı, nesne kapatıldığında yeniden kullanım için kullanılabilir olur. Ancak, bir dağıtım listesi açıldığında, dağıtım listesindeki her bir kuyruk ayrı bir tanıtıcı olarak ayrılır; böylece, MQOPER çağrısı dağıtım listesindeki kuyruklar olduğu için, bu tanıtıcıyı bir çok tanıtıcı olarak kullanır. Bu, *MaxHandles* için uygun bir değer üzerinde karar verilirken dikkate alınmalıdır.

MQPUT1 çağrısı, işlenmesinin bir parçası olarak bir MQOPEN çağrısı gerçekleştirir; sonuç olarak, MQPUT1 , MQXX\_ENCODE\_CASE\_ONE open işlevi kadar tanıtıcı kullanır; ancak, tutamaçlar yalnızca MQPUT1 çağrısının süresi boyunca kullanılır.

z/OS üzerinde, *görev* , bir CICS görevi, MVS görevi ya da IMS bağımlı bölgesi anlamına gelir.

Değer, 1-999 999 999 aralığında olmalıdır. Varsayılan değer, ortam tarafından belirlenir:

- z/OS üzerinde varsayılan değer 100 'dür.
- Diğer tüm ortamlarda, varsayılan değer 256 'tır.

Bu öznitelikliğini değeri saptamak için, MQINQ çağrısıyla MQIA\_MAX\_HANDHANS seçiciyi kullanın.

## **MaxMsgUzunluğu (MQUZE)**

Bu, kuyruk yöneticisinin işleyebileceği en uzun *fiziksel* iletinin uzunluğudur. Ancak, **MaxMsgLength** kuyruk yöneticisi öznitelikliğini **MaxMsgLength** kuyruk öznitelikden bağımsız olarak ayarlanabildiği için, bir kuyruğa yerleştirilebilecek en uzun fiziksel ileti, bu iki değerden küçük olan en küçük fiziksel iletidir.

Kuyruk yöneticisi kesimlere ayırma özelliğini destekliyorsa, uygulama, iki **MaxMsgLength** öznitelikinden daha küçük olan bir mantıksal iletiyi, ancak uygulama MQMD ' de MQMF\_SEGMENTATION\_ALLOWLY işaretini belirtirse, bu iletiyi daha uzun bir mantıksal ileti koyabilir. Bu işaret belirlenirse, mantıksal ileti uzunluğuna ilişkin üst sınır 999 999 999 bayttır, ancak genellikle işletim sistemi tarafından uygulanan kaynak kısıtlamaları ya da uygulamanın çalıştığı ortam tarafından uygulanan kaynak kısıtları alt sınırdadır.

**MaxMsgLength** öznitelikliğini için alt sınır 32 KB ' dir (32 768 bayt). Üst sınır 100 MB ' dir (104 857 600 bayt).

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_MAX\_MSG\_LENGTH seçiciyi kullanın.

### **MaxPriority (MQUZE)**

Bu, kuyruk yöneticisi tarafından desteklenen ileti önceliği üst sınığıdır. Öncelikler, sıfır (en düşük) ile *MaxPriority* (en yüksek) arasında değişiyor.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_MAX\_PRIORITY seçiciyi kullanın.

### **MaxPropertiesUzunluk (MQUZE)**

Bu, bir iletiyle akabilecek özelliklerin boyutunu denetlemek için kullanılır. Bu, hem özellik adını bayt cinsinden, hem de özellik değerinin bayt cinsinden boyutunu içerir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_MAX\_PROPERTIES\_LENGTH seçiciyi kullanın.

### **Multi V 9.2.0 MaxQFileBoyutu (MQUZE)**

Bir kuyruk dosyasının büyüyebileceği büyüklük üst sınırı (megabyte).

Çizelge 561. Bu özniteliğin geçerli olduğu kuyruk tipleri

Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Bir kuyruk dosyası, yürürlükteki kuyruk dosyası büyüklüğünden daha düşük bir değer olarak yapılandırıldıysa, büyüklük üst sınırını aşmalıdır. Bu gerçekleşirse, kuyruk dosyası artık yeni iletileri kabul etmez, ancak var olan iletilerin tüketilmesine izin verir. Kuyruk dosyası büyüklüğü yapılandırılan değer altına düştüğünde, yeni iletilerin kuyruğa konmasına izin verilir.

**Not:** Kuyruk yöneticisinin dahili olarak seçilen boyuta ulaşmak için daha büyük bir blok büyüklüğü kullanması gerekebileceğinden, bu şekil, kuyrukta yapılandırılan öznitelik değerinden farklı olabilir. Kuyruk dosyalarının ve öbek büyüklüğünün ve ayrıntı düzeylerinin büyüklüğünün değiştirilmesine ilişkin ek bilgi için [IBM MQ kuyruk dosyalarının değiştirilmesi](#) başlıklı konuya bakın.

When the granularity needs changing because this attribute has been increased, warning message AMQ7493W Öge boyu deđiřti is written to the AMQERR logs. Bu durum, IBM MQ ' un yeni ayrıntı düzeyini benimsemesi için kuyruğun boşaltılması için planlamaya gereksinim dumanıza olanak tanır.

Bu özniteliğin en yüksek değeri 267.386.880 MB, varsayılan değer ve geçirilen değer, bir kuyruk için 512 değeri olan bir kuyruk için geçerli üst sınır olan 2,088,960 MB ' dir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_MAX\_Q\_FILE\_SIZE seçicisini kullanın.

### **MaxUncommittedMsgs (MQUZE)**

Bu, bir iş birimi içinde bulunabilecek kesinleştirilmemiş ileti sayısı üst sınığıdır. Kesinleştirilmemiş ileti sayısı, yürürlükteki iş biriminin başlangıcından bu yana aşağıdaki toplamın toplamını içerir:

- Uygulama tarafından MQPMO\_SYNCPOINT seçeneđi ile gönderilen iletiler
- Uygulama tarafından MQGMO\_SYNCPOINT seçeneđi ile alınan iletiler
- MQPMO\_SYNCPOINT seçeneđi ile gönderilen iletiler için kuyruk yöneticisi tarafından oluşturulan iletileri ve COA rapor iletilerini tetikler
- MQGMO\_SYNCPOINT seçeneđiyle alınan iletiler için kuyruk yöneticisi tarafından oluşturulan COD raporu iletileri

Şu iletiler kesinleştirilmemiş olarak sayılmaz:

- Uygulama tarafından bir iş biriminin dışında konarak ya da alınan iletiler
- Kuyruk yöneticisi tarafından oluşturulan ya da bir iş birimi dışında alınan iletilerin sonucu olarak, kuyruk yöneticisi tarafından oluşturulan iletileri ya da COA/COD rapor iletilerini tetikler.

- Kuyruk yöneticisi tarafından oluşturulan süre bitim raporu iletileri (son kullanma rapor iletilerinin MQGMO\_SYNCPOINT belirtimine neden olmasına yol açan çağrı olsa bile)
- Kuyruk yöneticisi tarafından oluşturulan olay iletileri (olay iletilerinin MQPMO\_SYNCPOINT ya da MQGMO\_SYNCPOINT belirtmesine neden olmasına yol açan çağrı olsa bile)

**Not:**

1. Kural dışı durum rapor iletileri Message Channel Agent (MCA) tarafından ya da uygulama tarafından oluşturulur ve uygulama tarafından konulan ya da alınan sıradan iletiler ile aynı şekilde işlem görür.
2. Bir ileti ya da bölüm MQPMO\_SYNCPOINT seçeneği ile konduğunda, kesinleştirilmemiş iletilerin sayısı, ne kadar fiziksel ileti konmasına neden olursa olsun, birden çok ileti artırılır. (Kuyruk yöneticisinin iletiyi ya da bölümü ayırmaması gerekiyorsa, birden çok fiziksel ileti sonuçlanabilir.)
3. MQPMO\_SYNCPOINT seçeneği ile bir dağıtım listesi konduğunda, kesinleştirilmemiş iletilerin sayısı, oluşturulan her fiziksel ileti için bir *tarafından artırılır*. Bu, en az biri kadar küçük ya da dağıtım listesindeki hedef sayısı kadar büyük olabilir.

Bu öznitelik için alt sınır 1; üst sınır 999 999 999 'tır. Varsayılan değer 10000 'dir.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_MAX\_UNCOMMITTED\_MSGS seçicisini kullanın.

### **MQIAccounting (MQUZE)**

Bu, MQI verilerine ilişkin muhasebe bilgilerinin toplanmasını denetler.

Değer aşağıdakilerden biridir:

**MQMON\_ON**

API muhasebe verilerini toplar.

**MQMON\_OFF**

API muhasebe verilerini toplamaz. Bu varsayılan değerdir.

ACCTCONO kuyruk yöneticisi özneliğini ENABLE değerine ayarladıysanız, bu değer MQCNO yapısındaki Seçenekler alanını kullanan tek tek bağlantılar için geçersiz kılınabilir. Bu değerdeki değişiklikler yalnızca, özneliğe yapılan değişiklikten sonra oluşan kuyruk yöneticisine yönelik bağlantılar için geçerlidir.

Bu öznitelik yalnızca Multiplatforms üzerinde desteklenir.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_ACCOUNTING\_MQI seçiciyi kullanın.

### **MQIStatistics (MQUZN)**

Kuyruk yöneticisine ilişkin istatistik izleme bilgilerinin toplanmasını denetler.

Değer aşağıdakilerden biridir:

**MQMON\_ON**

MQI istatistiklerini toplar.

**MQMON\_OFF**

MQI istatistikleri toplanmasın. Bu varsayılan değerdir.

Bu öznitelik yalnızca Multiplatforms üzerinde desteklenir.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_STATISTICS\_MQI seçiciyi kullanın.

### **MsgMarkBrowseInterval (MQlong)**

Kuyruk yöneticisinin iletileri göz atarak otomatik olarak kaldırabileceği süre (milisaniye) cinsinden zaman aralığı.

Kuyruk yöneticisinin göz atma iletilerinden işareti otomatik olarak kaldırabileceği bir zaman aralığı (milisaniye olarak).

Bu öznitelik, MQGMO\_MARK\_BROWSE\_CO\_OP ileti alma seçeneği kullanılarak, MQGET çağrısıyla göz atıldığı gibi işaretlenen iletilerin göz atıldığı gibi işaretlenmesinin beklendiğini gösterir.

Kuyruk yöneticisi, bu yaklaşık aralıktan daha fazla aralık için işaretlendiğinde, iş birliği yapan tutamaçlar için göz atılmış olarak işaretlenen browsed iletilerini otomatik olarak işaretleyebilir.

Bu, MQGMO\_MARK\_BROWSE\_HANDLE ileti alma seçeneğini kullanarak, göz atma olarak imlenmiş herhangi bir iletinin durumunu etkilemez. Bu, MQGET çağrısıyla elde edilen bir çağrıdır.

Değer üst sınırı 999 999 999 ve varsayılan değer 5000 'dir. *MsgMarkBrowseInterval* için -1 özel değeri, sınırsız zaman aralığını temsil eder.



**Uyarı:** Bu değer, varsayılan 5000 değerinin altında olmamalıdır.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_MSG\_MARK\_BROWSE\_INTERVAL seçiciyi kullanın.

### **OutboundPortMax (MQUZE)**

Giden kanalları bağlamak için kullanılacak kapı numaralarının OutboundPortMin ve OutboundPortMax tarafından tanımlanan, aralıktaki en yüksek kapı numarasıdır.

Değer, 0 ile 65535 aralığında bir tamsayıdır ve OutboundPortMin değerine eşit ya da ondan büyük olmalıdır. Varsayılan değer 0'dır.

Bu öznelik yalnızca z/OS üzerinde desteklenir.

Bu özneliğin değerini saptamak için, MQINQ çağrısına sahip MQIA\_OUTBOUND\_PORT\_MAX seçicisini kullanın.

### **OutboundPortMin (MQUZE)**

Giden kanalları bağlamak için kullanılacak kapı numaralarının OutboundPortMin ve OutboundPortMax tarafından tanımlanan, aralıktaki en düşük kapı numarasıdır.

Değer, 0 ile 65535 aralığındaki bir tamsayıdır ve OutboundPortMax değerine eşit ya da bu değerden küçük olmalıdır. Varsayılan değer 0'dır.

Bu öznelik yalnızca z/OS üzerinde desteklenir.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_OUTBOUND\_PORT\_MIN seçiciyi kullanın.

### **PerformanceEvent (MQUZE)**

Bu, performansa ilişkin olayların oluşturulup oluşturulmayacağını denetler. Bu değer aşağıdaki değerlerden biridir:

#### **MQEVR\_DISABLE**

Olay raporlama devre dışı bırakıldı.

#### **MQEVRENABLED**

Olay raporlaması etkinleştirildi.

Olaylarla ilgili daha fazla bilgi için [Olay izleme](#) başlıklı konuya bakın.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_PERFORMANCE\_EVENT seçiciyi kullanın.

### **Altyapı (MQLONG)**

Kuyruk yöneticisinin çalışmakta olduğu işletim sistemini gösterir:

#### **MQPL\_AIX**

AIX (MQPL\_UNIX ile aynı değer).

#### **MQP\_APPLIANCE**

IBM MQ Appliance

#### **MQPL\_MVS**

z/OS (MQPL\_ZOS ile aynı değer).

#### **MQPL\_OS390**

z/OS (MQPL\_ZOS ile aynı değer).

**MQPL\_OS400**

IBM i.

**MQPL\_UNIX**

UNIX.

**MQPL\_WINDOWS\_NT**

Windows sistemleri.

**MQPL\_ZOS**

z/OS.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_PLATFORM seçiciyi kullanın.

**PubSubNPInputMsg (MQHOD)**

Teslim edilmemiş bir giriş iletisinin atılıp atılmayacağını ya da saklanmayacağını.

Değer aşağıdakilerden biridir:

**MQUNDELIVERED\_ATIN**

İşlenemezlerse, kalıcı olmayan giriş iletileri atılabilir.

Bu varsayılan değerdir.

**MQUNDELIVERED\_KEEP**

İşlenemezlerse, kalıcı olmayan giriş iletileri atılmaz. Bu durumda, kuyruklanan yayınlama/abone olma arabirimi, işlemi uygun aralıklarla yeniden denemeye devam eder ve sonraki iletileri işlemeye devam etmez.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_PUBSU\_NP\_NP\_MSG seçiciyi kullanın.

**PubSubNPResponse (MQLONG)**

Teslim edilemeyen yanıt iletilerinin davranışını denetler.

Değer aşağıdakilerden biridir:

**MQUNDELIVERED\_NORMAL**

Yanıt kuyruğuna yerleştirilemeyen, kalıcı olmayan yanıtlar, DLQ ' ya yerleştirilemezlerse, bunlar atılır.

**MQUNDELIVERED\_SAFE**

Yanıt kuyruğuna yerleştirilemeyen, kalıcı olmayan yanıtlar, çıkmaz mektup kuyruğuna yerleştirilir. Yanıt ayarlanamaz ve DLQ ' ya yerleştirilemezse, kuyruğa alınan yayınlama/abone olma arabirimi yürürlükteki işlemi geri alacak ve uygun aralıklarla yeniden denenecek ve sonraki iletileri işlemeye devam etmez.

**MQUNDELIVERED\_ATIN**

Kalıcı olmayan yanıtlar, yanıt kuyruğuna yerleştirilmez.

Bu, yeni kuyruk yöneticileri için varsayılan değerdir.

**MQUNDELIVERED\_KEEP**

Kalıcı olmayan yanıtlar, ölü mektup kuyruğuna yerleştirilmez ya da atılmaz. Bunun yerine, kuyruğa alınan yayınlama/abone olma arabirimi yürürlükteki işlemi geri alacak ve uygun aralıklarla yeniden deneyecek.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_PUBSU\_NP\_RESP seçiciyi kullanın.

**Geçirilen kuyruk yöneticilerine ilişkin varsayılan değer.**

Kuyruk yöneticisi IBM MQ V6.0' dan geçirildiyse, bu özniteliğin başlangıç değeri, aşağıdaki tabloda gösterildiği gibi, geçişten önce *DiscardNonPersistentResponse* ve *DLQNonPersistentResponse* değerlerine bağlıdır.



		DLQNonPersistentyanıtı		
		Evet	Hayır	Ayarlanmadı
DiscardNonPersistentResponse	Evet	MQUNDELIVERED_NORMAL	MQUNDELIVERED_ATIN	MQUNDELIVERED_NORMAL
	Hayır	MQUNDELIVERED_SAFE	MQUNDELIVERED_KEEP	MQUNDELIVERED_SAFE
	Ayarlanmadı	SyncPointKalıcı = Hayır ise, MQUNDELIVERED_SAFE else MQUNDELIVERED_NORMAL	SyncPointKalıcı = Hayır ise, MQUNDELIVERED_KEEP else MQUNDELIVERED_ATMA	SyncPointKalıcı = Hayır ise, MQUNDELIVERED_SAFE else MQUNDELIVERED_NORMAL

### **PubSubMaxMsgRetryCount (MQlong)**

Eşitleme noktası altında başarısız olan bir komut iletisini işlerken yeniden deneme sayısı.

Değer aşağıdakilerden biridir:

**0 - 999 999 999**

Varsayılan değer 5'tir.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_PUBSU\_MAXMSG\_RETRY\_COUNT seçiciyi kullanın.

### **PubSubSyncPoint (MQUBE)**

Yalnızca kalıcı iletilerin ya da tüm iletilerin uyumluluk noktası altında işlenip işlenmeyeceğini belirleyin.

Değer aşağıdakilerden biridir:

**MQSYNCPOINT\_IFPER**

Bu, kuyruğa alınan yayınlama/abone olma arabiriminin, eşitleme noktası dışında kalıcı olmayan iletileri almasına neden olur. Yardımcı program dışında bir yayın alırsa, yardımcı program yayını dış eşitleme noktası dışında bilinen abonelere iletir.

Bu varsayılan değerdir.

**MQSYNCPOINT\_YES**

Bu, kuyruğa alınan yayınlama/abone olma arabiriminin tüm iletileri eşitleme noktası altında almasına neden olur.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_PUBSU\_SYNC\_PT seçiciyi kullanın.

### **PubSubKipi (MQUZE)**

Yayınlama/abone olma motorunun ve kuyruğa alınan yayınlama/abone olma arabiriminin çalışıp çalışmadığı, bu nedenle uygulama programlama arabirimini ve kuyruğa alınan yayınlama/abone olma arabirimi tarafından izlenmekte olan kuyrukları kullanarak uygulamaların yayınlanmasına/abone olmasına izin verilmesine olanak sağlar.

Değer aşağıdakilerden biridir:

**MQPSM\_COMPAT**

publish/subscreen motoru çalışıyor. Bu nedenle, uygulama programlama arabirimi kullanılarak yayınlama/abone olma olanaklıdır. Kuyruğa alınan yayınlama/abone olma arabirimi çalışmıyor; bu nedenle, kuyruğa alınan yayınlama/abone olma arabirimi tarafından izlenen kuyruklara yapılan tüm iletiler üzerinde işlem yapmamaktadır. Bu ayar, bu kuyruk yöneticisini kullanan WebSphere Message Broker V6 ya da önceki sürümleriyle uyumluluk için kullanılır; bunun nedeni, kuyruğa yollanmış yayınlama/abone olma arabiriminin normalde okuduğu kuyrukları okuması gerekir.

**MQPSM\_DISABLE**

publish/subscreen motoru ve kuyruğa alınan yayınlama/abone olma arabirimi çalışmıyor. Bu nedenle, uygulama programlama arabirimi kullanılarak yayınlama/abone olma olanaklı değildir. Kuyruğa alınan yayınlama/abone olma arabirimiyle izlenen kuyruklara yapılan yayınlama/abone olma iletileri üzerinde işlem yapmamış olur.

**MQPSMENABLED**

publish/subscreen motoru ve kuyruğa alınan yayınlama/abone olma arabirimi çalışıyor. Bu nedenle, uygulama programlama arabirimi ve kuyruklanan yayınlama/abone olma arabirimi tarafından izlenmekte olan kuyruklar kullanılarak yayınlama/abone olma olanaklıdır. Kuyruk yöneticisinin ilk varsayılan değeri budur.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_PUBSU\_MODE seçiciyi kullanın.

### **QMGrDesc (MQCHAR64)**

Kuyruk yöneticisini açıklayan bir açıklama için bu alanı kullanın. Alanın içeriği kuyruk yöneticisi için önemli değildir; ancak kuyruk yöneticisi, alanın yalnızca görüntülenebilecek karakterleri içermesini gerektirebilir. Boş değerli karakterler içeremez; gerekirse, boşlukla doldurulabilir. Bir DBCS kuruluşunda, bu alan DBCS karakterleri içerebilir (en çok 64 bayt alan uzunluğuna bağlı).

**Not:** Bu alan, kuyruk yöneticisinin karakter kümesinde olmayan karakterler içeriyorsa ( **CodedCharSetId** kuyruk yöneticisi öznitelemesiyle tanımlandığı gibi), bu alan başka bir kuyruk yöneticisine gönderilirse, bu karakterler yanlış çevrilebilir.

- z/OS' ta varsayılan değer, ürün adı ve sürüm numarasıdır.
- Diğer tüm ortamlarda, varsayılan değer boşluktur.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQCA\_Q\_MGR\_DESC seçiciyi kullanın. Bu özniteliğin uzunluğu MQ\_Q\_MGR\_DESC\_LENGTH tarafından verilir.

### **QMGrIdentifier (MQCHAR48)**

Bu, kuyruk yöneticisi için dahili olarak oluşturulmuş benzersiz bir addır.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQCA\_Q\_MGR\_IDENTIFIER seçiciyi kullanın. Bu özniteliğin uzunluğu MQ\_Q\_MGR\_IDENTIFIER\_LENGTH tarafından verilir.

Bu öznitelik aşağıdaki ortamlarda desteklenir:

-  AIX
-  IBM i
-  Linux
-  Windows
-  z/OS

ve IBM MQ istemcileri bu sistemlere bağlıdır.

### **QMGrName (MQCHAR48)**

Bu ad, uygulamanın bağlı olduğu kuyruk yöneticisinin adı olan yerel kuyruk yöneticisinin adıdır.

Adın ilk 12 karakteri, benzersiz bir ileti tanıtıcısı oluşturmak için kullanılır (bkz. [MQMD- MsgId alanı](#) ). İletişim kurabilen kuyruk yöneticilerinin, ileti tanıtıcılarının kuyruk yöneticisi ağında benzersiz olması için ilk 12 karakterde farklı olan adları olması gerekir.


z/OS' ta ad, 4 boşluk olmayan karakterle sınırlı olan altsistem adıyla aynıdır.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQCA\_Q\_MGR\_NAME seçicisini kullanın. Bu özniteliğin uzunluğu, MQ\_Q\_MGR\_NAME\_LENGTH tarafından verilir.

### **QSGName (MQCHAR4)**

Bu ad, yerel kuyruk yöneticisinin ait olduğu kuyruk paylaşım grubunun adıdır. Yerel kuyruk yöneticisi bir kuyruk paylaşım grubuna ait değilse, ad boş olur.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQCA\_QSG\_NAME seçicisini kullanın. Bu özniteliğin uzunluğu MQ\_QSG\_NAME\_LENGTH tarafından verilir.

 Bu öznitelik yalnızca z/OS üzerinde desteklenir.

### **QueueAccounting (MQHOMER)**

Bu, kuyruklar için muhasebe bilgilerinin toplanmasını denetler.

Değer aşağıdakilerden biridir:

#### **MQMON\_NONE**

Kuyruk hesap özneliği ACCTQ ayarından bağımsız olarak, kuyruklar için muhasebe verilerini toplamaz. Bu varsayılan değerdir.

#### **MQMON\_OFF**

ACCTQ kuyruk öznelide QMGR belirten kuyruklar için muhasebe verileri toplanmayın.

#### **MQMON\_ON**

ACCTQ kuyruk öznelide QMGR belirten kuyruklar için muhasebe verilerini toplayın.

Bu değerdeki değişiklikler yalnızca, özneliğe yapılan değişiklikten sonra oluşan kuyruk yöneticisine yönelik bağlantılar için geçerlidir.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_ACCOUNTING\_Q seçiciyi kullanın.

### **QueueMonitoring (MQUZE)**

Bu, kuyrukların çevrimiçi izlenmesine ilişkin varsayılan ayarı belirtir.

**QueueMonitoring** kuyruk özneliği MQMON\_Q\_MGR olarak ayarlandıysa, bu öznelik, kanal tarafından kabul edilen değeri belirtir. Değer şu şekilde olabilir:

#### **MQMON\_OFF**

Çevrimiçi izleme verileri toplama kapatıldı. Kuyruk yöneticisinin ilk varsayılan değeri budur.

#### **MQMON\_NONE**

Çevrimiçi izleme verileri toplama, **QueueMonitoring** özneliğinin ayarından bağımsız olarak kuyruklar için kapatılır.

#### **MQMON\_LOW**

Veri toplama oranı düşük olan çevrimiçi izleme verileri toplama işlemi açık bir şekilde açık olmalıdır.

#### **MQMON\_ORTAMı**

Çevrimiçi izleme verileri toplama, veri toplama işlemi için ılımlı bir oran ile açık bir şekilde açıktır.

#### **MQMON\_YÜKSEK**

Veri toplama oranı yüksek olan çevrimiçi izleme verileri toplama işlemi açık bir şekilde açıktır.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_MONITORING\_Q seçiciyi kullanın.

### **QueueStatistics (MQLONG)**

Bu, kuyruklar için istatistik verilerinin toplanmasını denetler.

Bu değer aşağıdaki değerlerden biridir:

#### **MQMON\_NONE**

**QueueStatistics** kuyruk özneliğinin ayarından bağımsız olarak kuyruklar için kuyruk istatistiği toplanmayın. Bu varsayılan değerdir.

#### **MQMON\_OFF**

**QueueStatistics** kuyruk öznelide Kuyruk Yöneticisi belirten kuyruklar için istatistik verileri toplanmayın.

#### **MQMON\_ON**

**QueueStatistics** kuyruk öznelide Kuyruk Yöneticisi belirten kuyruklar için istatistik verileri toplayın.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_STATISTICS\_Q seçiciyi kullanın.

### **ReceiveTimeout (MQUZE)**

Bu, bir TCP/IP kanalının etkin olmayan duruma dönmeden önce iş ortasından kalp atışları da içinde olmak üzere verileri ne kadar süreyle bekleyeceğini belirler. Bu, yalnızca ileti kanalları için geçerlidir ve MQI kanallarına uygulanmaz.

ReceiveTimeout değerinin tam anlamı, ReceiveTimeouttipinde belirtilen değere göre değiştirilir. ReceiveTimeoutTipi aşağıdakilerden birine ayarlanabilir:

- MQRCTIME\_EQUAL-Bu değer, kanala ilişkin bekleyeceği saniye cinsinden sayıdır. 0-999999 aralığında bir değer belirtin.
- MQRCTIME\_ADD-bu değer, kararlaştırılan HBINT ' ye eklenecek saniye cinsinden sayıdır ve bir kanal ne kadar süreyle bekleyeceğini belirler. 1-999999 aralığında bir değer belirtin.
- MQRCTIME\_MULTIPLE-Bu değer, anlaşmalı HBINT için uygulanacak bir çarpıcı. 2-99 aralığında 0 ya da bir değer belirtin.

Varsayılan değer 0'dır.

Set ReceiveTimeoutType to MQRCTIME\_MULTIPLY or MQRCTIME\_EQUAL, and ReceiveTimeout to 0, to stop a channel from timing out its wait to receive data from its partner.

Bu öznelik yalnızca z/OS üzerinde desteklenir.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_RECEIVE\_TIMEOUT seçiciyi kullanın.

### **ReceiveTimeoutMin (MQUZE)**

Bu, etkin olmayan duruma dönmeden önce, bir TCP/IP kanalının iş ortasından kalp atışları da içinde olmak üzere verileri almayı bekleyeceği en düşük süre (saniye olarak).

Bu, MQI kanallarına değil, yalnızca ileti kanallarına uygulanır. Değer, 0 ile 999999 arasında bir değer olmalıdır; varsayılan değer 0 olmalıdır.

TCP/IP kanalı bekleme süresinin, HBINT ' nin kararlaştırılan değeriyle görel olarak hesaplanacağını belirtmek için ReceiveTimeouttipini kullanırsanız ve sonuçtaki değer bu parametrenin değerinden küçükse, bunun yerine bu değer kullanılır.

Bu öznelik yalnızca z/OS üzerinde desteklenir.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_RECEIVE\_TIMEOUT\_MIN seçiciyi kullanın.

### **ReceiveTimeoutTip (MQUZE)**

Bu niteleyici, bir TCP/IP kanalının etkin olmayan duruma dönmeden önce iş ortasından kalp atışları da içinde olmak üzere veri almayı ne kadar süreyle bekleyeceğini tanımlamak için ReceiveTimeout değerine uygulanmış bir niteleyicidir. Bu, MQI kanallarına değil, yalnızca ileti kanallarına uygulanır.

Değer aşağıdakilerden biridir:

#### **MQRCTIME\_ÇARP**

ReceiveTimeout , bir kanalın ne kadar bekleyeceğini belirlemek üzere anlaşmalı HBINT değerine uygulanacak bir çarpıcıdır. Bu varsayılan değerdir.

#### **MQRCTIME\_ADD**

ReceiveTimeout is a value, in seconds, to add to the negotiated HBINT value to determine how long a channel waits.

#### **MQRCTIME\_EQUAL**

ReceiveTimeout , kanal bekleyeceği saniye cinsinden bir değer.

Bir kanal zamanlamasını, ortasından veri almak için beklemesini durdurmak için, ReceiveTimeouttipini MQRCTIME\_ÇOĞAL ya da MQRCTIME\_EQUAL olarak ayarlayın ve ReceiveTimeout değerini 0 olarak ayarlayın.

Bu öznelik yalnızca z/OS üzerinde desteklenir.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_RECEIVE\_TIMEOUT\_TYPE seçiciyi kullanın.

### **RemoteEvent (MQUZE)**

Bu, uzak hata olaylarının oluşturulup oluşturulmayacağını denetler. Bu değer aşağıdaki değerlerden biridir:

## **MQEVN\_DISABLE**

Olay raporlama devre dışı bırakıldı.

## **MQEVNENABLED**

Olay raporlaması etkinleştirildi.

Olaylarla ilgili daha fazla bilgi için [Olay izleme](#) başlıklı konuya bakın.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_REMOTE\_EVENT seçiciyi kullanın.

## **RepositoryName (MQCHAR48)**

Bu, kuyruk yöneticisinin bir havuz yöneticisi hizmeti sağladığı bir kümenin adıdır. Kuyruk yöneticisi bu hizmeti birden çok küme için sağlıyorsa, *RepositoryNameList* kümeleri tanımlayan bir ad listesi nesnesinin adını belirtir ve *RepositoryName* boş olur. *RepositoryName* ve *RepositoryNameList* ' dan en az biri boş bırakılmalı.

Bu özneliğin değerini saptamak için, MQINQ çağrısına sahip MQCA\_REPOSITORY\_NAME seçiciyi kullanın. Bu özneliğin uzunluğu, MQ\_Q\_MGR\_NAME\_LENGTH tarafından verilir.

## **RepositoryNameList (MQCHAR48)**

Bu ad, bu kuyruk yöneticisinin bir havuz yöneticisi hizmeti sağladığı kümelerin adlarını içeren bir ad listesi nesnesinin adıdır. Kuyruk yöneticisi bu hizmeti yalnızca bir küme için sağlıyorsa, ad listesi nesnesi yalnızca bir ad içerir. Alternatively, *RepositoryName* can be used to specify the name of the cluster, in which case *RepositoryNameList* is blank. *RepositoryName* ve *RepositoryNameList* ' dan en az biri boş bırakılmalı.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQCA\_REPOSITORY\_NAMELIST seçiciyi kullanın. Bu özneliğin uzunluğu MQ\_NAMELIST\_NAME\_LENGTH tarafından verilir.

## **ScyCase(MQCHAR8)**

Kuyruk yöneticisinin, büyük ve küçük harf karışık olarak ya da yalnızca büyük harfli olarak güvenlik profili adlarını destekleyip desteklemediğini belirtir.

Değer aşağıdakilerden biridir:


### **MQSCYC\_UPPER**

Güvenlik profili adları büyük harfli olmalıdır.

### **MQSCYC\_MIXED**

Güvenlik profili adları büyük harfli ya da büyük/küçük harf karışık olabilir.

Changes to this attribute take effect when a Refresh Security command is run with *SecurityType* (MQSECTYPE\_CLASSES) specified.

 Bu öznelik yalnızca z/OS üzerinde desteklenir.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_SECURITY\_CASE seçiciyi kullanın.

## **SharedQMGrAd (MQUZE)**

This specifies whether the *ObjectQmgrName* should be used or treated as the local queue manager on an MQOPEN call, for a shared queue, when the *ObjectQmgrName* is that of another queue manager in the queue sharing group.

Değer, aşağıdaki değerlerden herhangi biri olabilir:

### **MQSQM\_USE**

*ObjectQmgrName* kullanılır ve uygun iletim kuyruğu açılır.

### **MQSQM\_IGNORE**

Hedef kuyruk paylaşılıyorsa ve *ObjectQmgrName* , aynı kuyruk paylaşım grubundaki bir kuyruk yöneticisine, açık yerel olarak gerçekleştirilir.

Bu öznelik yalnızca z/OS üzerinde geçerlidir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_SHARED\_Q\_Q\_MGR\_NAME seçicisini kullanın.

### **KYR**

Bir kuyruk yöneticisi için Advanced Message Security güvenlik yeteneklerinin kullanılabilir olup olmadığını belirtir.

### **MQCAP\_DESTEKLELEN**

Kuyruk yöneticisinin çalışmakta olduğu kuruluş için AMS bileşeni kuruluysa, bu varsayılan değerdir.

### **MQCAP\_NOT\_SUPPORTANT**

### **SSLEVENT (MQUZA)**

Bu, TLS olaylarının oluşturulup oluşturulmayacağını belirtir.

Bu değer aşağıdaki değerlerden biridir:

#### **MQEVRENABLED**

TLS olaylarını şu şekilde oluşturun:

MQRC\_CHANNEL\_SSL\_ERROR

#### **MQEVDR\_DISABLE**

TLS olayları oluşturmayın; bu varsayılan değerdir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_SSL\_OLAY seçiciyi kullanın.

### **SSLFIPSRequired (MQLONG)**

**Not:** AIX, Linux, and Windows işletim sistemlerinde IBM MQ , "IBM Crypto for C" şifreleme modülü aracılığıyla FIPS 140-2 uyumluluğu sağlar. Bu modüle ilişkin sertifika Geçmiş durumuna taşındı. Müşteriler, [IBM Crypto for C certificate](#) belgesini görüntüleyip NIST tarafından sağlanan tüm tavsiyeleri dikkate almalıdır. Yeni bir FIPS 140-3 modülü şu anda devam ediyor ve durumu [İşlem listesindeki NIST CMVP modüllerinde](#) aranarak görüntülenebilir.

Bu, şifreleme, şifreleme donanımı yerine IBM MQ içinde yürütüldüğünde yalnızca FIPS onaylı algoritmaların kullanılacağını belirtmenizi sağlar. Şifreleme donanımı yapılandırıldıysa, kullanılan şifreleme modülleri donanım ürünü tarafından sağlanan modüllerdir; bu modüller, kullanılmakta olan donanım ürününe bağlı olarak belirli bir düzeyde FIPS onaylı olabilir ya da olmayabilir.

Değer, aşağıdaki değerlerden biridir:

#### **MQSSL\_FIPS\_NO**

Kullanılmakta olan platformda desteklenen herhangi bir CipherSpec kullanın. Bu değer varsayılan değerdir.

#### **MQSSL\_FIPS\_YES**

Bu kuyruk yöneticisinden ve bu kuyruk yöneticisine tüm TLS bağlantılarında izin verilen CipherSpecs içinde yalnızca FIPS onaylı şifreleme algoritmalarını kullanın.

Bu parametre yalnızca z/OS, AIX, Linux, and Windows platformlarında geçerlidir.

Bu özniteliğin değerini saptamak için MQINQ çağrısıyla MQIA\_SSL\_FIPS\_REQUIRED seçicisini kullanın.

### **İlgili görevler**

[MQI istemcisinde çalıştırma zamanında yalnızca FIPS onaylı CipherSpecs kullanılmasının belirtilmesi](#)

### **İlgili başvurular**

[AIX, Linux, and Windows için Federal Bilgi İşleme Standartları \(FIPS\)](#)

### **SSLKeyResetSayı (MQUZE)**

Bu, iletişim başlatan TLS kanalı ileti kanalı araçlarının (MCA ' lar) kanalda şifreleme için kullanılan gizli anahtarı ilk durumuna getirdiğinde belirtir.

Bu değer, gizli anahtar yeniden anlaşılmadan önce kanalda gönderilen ve alınan toplam şifrelenmemiş bayt sayısını temsil eder. Bayt sayısı, MCA tarafından gönderilen denetim bilgilerini içerir.

Değer, 0 ile 999 999 999 aralığındaki bir sayıdır ve varsayılan değer olarak 0 değerini içerir. 1 bayt-32 KB aralığında bir TLS gizli anahtar sıfırlama sayısı belirtirseniz, TLS kanalları 32 KB ' lik gizli anahtar sıfırlama sayısını kullanır. Bu, küçük TLS gizli anahtar ilk duruma getirme değerleri için oluşacağı aşırı anahtar sıfırlamalarının işlem maliyetinden kaçınmak içindir.

Başlatan kanal MCA tarafından gönderilen ve alınan toplam şifrelenmemiş bayt sayısı belirtilen değeri aştığında, gizli anahtar yeniden görüşülmektedir. Kanal kalp atışları etkinleştirilirse, bir kanal sağlıklı işletim bildirim gönderilmeden ya da bir kanal sağlıklı işletim bildirim alındıktan sonra gizli anahtar yeniden görüşülür ya da şifrelenmemiş byte sayısı belirtilen değeri aştığında, hangisi önce olursa olsun, yeniden görüşülür.

Yeniden görüşme için gönderilen ve alınan bayt sayısı, kanal MCA tarafından gönderilen ve alınan denetim bilgilerini içerir ve yeniden görüşme gerçekleştiğinde sıfırlanır.

Gizli anahtarların hiçbir zaman yeniden anlaşılmadığını belirtmek için 0 değerini kullanın.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_SSL\_RESET\_COUNT seçiciyi kullanın.

### ***StartStopOlayı (MQUZE)***

Bu, başlatma ve durdurma olaylarının oluşturulup oluşturulmayacağını denetler. Değer aşağıdakilerden biridir:

#### **MQEVR\_DISABLE**

Olay raporlama devre dışı bırakıldı.

#### **MQEVRENABLED**

Olay raporlaması etkinleştirildi.

Olaylarla ilgili daha fazla bilgi için [Olay izleme](#) başlıklı konuya bakın.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_START\_STOP\_EVENT seçiciyi kullanın.

### ***StatisticsInterval (MQUZE)***

Bu, istatistik izleme verilerinin izleme kuyruğuna ne kadar sıklıkta (saniye cinsinden) yazılacağını belirtir.

Değer, 0 ile 604800 arasındaki bir tamsayıdır ve varsayılan değer olan 1800 (30 dakika) değeridir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_STATISTICS\_INTERVAL seçiciyi kullanın.

### ***SyncPoint (MQUZE)***

Bu durum, yerel kuyruk yöneticisinin, MQGET, MQPUT ve MQPUT1 çağrılarında sahip iş birimlerini destekleyip desteklemediğini gösterir.

#### **MQSP\_AVALABILIR**

İş birimleri ve syncpoint'leme kullanılabilir.

#### **MQSP\_NOT\_VAR**

İş birimi ve uyumluluk noktası kullanılamıyor.

- z/OS üzerinde bu değer hiçbir zaman döndürülmez.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_SYNCPOINT seçiciyi kullanın.

### ***TCPKanalları (MQUZE)***

Bu, TCP/IP iletim protokolünü kullanan, geçerli olabilecek kanal sayısı üst sınırı ya da bağlanabilen istemci sayısı üst sınırı.

Değer, 0-9999 aralığında olmalıdır; varsayılan değer 200 olmalıdır. 0 değerini belirlerseniz, TCP/IP kullanılmaz.

Bu öznitelik yalnızca z/OS üzerinde desteklenir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_TCP\_KANALS seçiciyi kullanın.

### ***TCPKeepAlive (MQLONG)***

Bu, bağlantının diğer ucunun hala kullanılabilir olup olmadığını denetlemek için TCP KEPENVE kullanıp kullanmayacağınızı belirtir. Uygun değilse, kanal kapatılır.

Değer aşağıdakilerden biridir:

#### **MQTCPKEEP\_YES**

TCP tanıtım konfigürasyonu veri kümesinde belirtildiği gibi, TCP KEPAVE 'ı kullanın. Kanal özniteliği KeepAliveInterval (KAINTE) değerini belirlerseniz, bu özniteliğin ayarının kullanılacağı değer kullanılır.

#### **MQTCPKEEP\_NO**

TCP KEPAVE kullanılmayı kullanmanız. Bu varsayılan değerdir.

Bu öznitelik yalnızca z/OS üzerinde desteklenir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_TCP\_KEEP\_ALIVE seçiciyi kullanın.

#### **TCPName (TCPName) (MQCHAR8)**

Bu ad, TCPStackType değerine bağlı olarak, kullanılacak tek ya da tercih edilen TCP/IP yığınının adıdır. Bu parametre yalnızca CINET çoklu yığın ortamları için geçerlidir. Varsayılan değer TCPIP 'dir.

Bu öznitelik yalnızca z/OS üzerinde desteklenir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQCA\_TCP\_NAME seçicisini kullanın. Bu özniteliğin uzunluğu MQ\_TCP\_NAME\_LENGTH tarafından verilir.

#### **TCPStackType (MQLONG)**

Bu, kanal başlatıcının yalnızca TCPName 'de belirtilen TCP/IP yığını kullanıp kullanamayacağını ya da isteğe bağlı olarak seçilen herhangi bir TCP/IP yığına bağ tanımlayıp bağlamayacağını belirtir. Bu parametre yalnızca CINET çoklu yığın ortamları için geçerlidir.

Değer aşağıdakilerden biridir:

#### **MQTCPSTACK\_SATIR**

Kanal başlatıcı yalnızca TCPName (TCPName) adlı TCP/IP adres alanlarını kullanabilir. Bu varsayılan değerdir.

#### **MQTCPSTACK\_MULTIPLE**

Kanal başlatıcı, kullanılabilir herhangi bir TCP/IP adres alanını kullanabilir. Bir kanal ya da dinleyici için başka bir değer belirtilmediyse, bu değer, TCPName olarak belirlenmiş olan varsayılan değer olarak ayarlanır.

Bu öznitelik yalnızca z/OS üzerinde desteklenir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_TCP\_STACK\_TYPE seçiciyi kullanın.

#### **TraceRouteKaydı (MQUZE)**

Bu, izleme rotası bilgilerinin kaydedilmesini denetler.

Değer aşağıdakilerden biridir:

#### **MQRECORDING\_DEVRE Dışı**

İzlenecek rota izleme iletilerine ekleme yapılamaz.

#### **MQRECORDING\_Q**

İzleme rotası iletilerini sabit adlandırılmış kuyruğa konmasını sağlar.

#### **MQRECORDING\_MSG**

İzleme rotası iletilerini, iletinin kendisini kullanarak belirlenen bir kuyruğa koyun. Bu varsayılan değerdir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_TRACE\_ROUTE\_RECORD seçiciyi kullanın.

#### **TriggerInterval (MQUZE)**

Tetikleme iletilerinin sayısını sınırlamak için kullanılan bir zaman aralığı (milisaniye cinsinden). Bu, yalnızca *TriggerType* MQTT\_FIRST olduğunda ilgili olur. Bu durumda, tetikleme iletileri genellikle



kuyrukta uygun bir ileti geldiğinde üretilir ve kuyruk daha önce boştur. Ancak, belirli koşullar altında, kuyruk boş olmasa da, MQTT\_FIRST tetikleyicisiyle ek bir tetikleyici iletisi oluşturulabilir. Bu ek tetikleyici iletileri her *TriggerInterval* milisaniyeden daha sık oluşturulmaz.

Tetikleme ile ilgili ek bilgi için [Tetikleme kanalları](#) başlıklı konuya bakın.

Değer 0 'dan küçük değil ve 999 999 'dan büyük değil. Varsayılan değer 999 999 999 'tır.

Bu özneliğin değerini saptamak için MQINQ çağrısıyla MQIA\_TRIGGER\_INTERVAL seçiciyi kullanın.

### **TriggerInterval (MQUZE)**

Tetikleme iletilerinin sayısını sınırlamak için kullanılan bir zaman aralığı (milisaniye cinsinden). Bu, yalnızca *TriggerType* MQTT\_FIRST olduğunda ilgili olur. Bu durumda, tetikleme iletileri genellikle kuyrukta uygun bir ileti geldiğinde üretilir ve kuyruk daha önce boştur. Ancak, belirli koşullar altında, kuyruk boş olmasa da, MQTT\_FIRST tetikleyicisiyle ek bir tetikleyici iletisi oluşturulabilir. Bu ek tetikleyici iletileri her *TriggerInterval* milisaniyeden daha sık oluşturulmaz.

Tetikleme ile ilgili ek bilgi için [Tetikleme kanalları](#) başlıklı konuya bakın.

Değer 0 'dan küçük değil ve 999 999 'dan büyük değil. Varsayılan değer 999 999 999 'tır.

Bu özneliğin değerini saptamak için MQINQ çağrısıyla MQIA\_TRIGGER\_INTERVAL seçiciyi kullanın.

### **Sürüm (MQCFST)**

Bu, VVRRMFF biçiminde IBM MQ kodunun sürüşüdür. Burada:

VV-Sürüm

RR-Serbest Bırak

MM-Bakım düzeyi

FF-Düzeltilme düzeyi

### **XrCapability(MQUZE)**

This controls whether MQ Telemetry commands are supported by the queue manager.

Değer aşağıdakilerden biridir:

#### **MQCAP\_DESTEKLENEN**

MQ Telemetry bileşeni kurulu ve Telemetry komutları desteklenmektedir.

#### **MQCAP\_NOT\_SUPPORTANT**

MQ Telemetry bileşeni kurulu değil.

Bu öznelik yalnızca [Multiplatforms](#) üzerinde desteklenir.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_XR\_CAPABPROCICE seçiciyi kullanın.

## **Kuyruklara ilişkin öznelikler**

Beş tip kuyruk tanımlaması vardır. Bazı kuyruk öznelikleri tüm kuyruk tipleri için geçerlidir; diğer kuyruk öznelikleri yalnızca belirli kuyruk tipleri için geçerlidir.

### **Kuyruk tipleri**

Kuyruk yöneticisi, aşağıdaki kuyruk tanımlaması tiplerini destekler:

#### **Yerel kuyruk**

İletileri yerel bir kuyruğun üzerinde saklayabilirsiniz.

 z/OS üzerinde, bunu paylaşılan ya da özel bir kuyruk yapabilirsiniz.

Programın bağı olduğu kuyruk yöneticisine aitse, bir program kuyruğa *yerel* olarak bilinir. Yerel kuyruklardan ileti alabilir ve yerel kuyruklara ileti yerleştirebilirsiniz.

Kuyruk tanımlaması nesnesi, kuyruğa konan fiziksel iletilerin yanı sıra, kuyruğun tanımlama bilgilerini de içerir.

### Yerel kuyruk yöneticisi kuyruğu

Kuyruk yerel kuyruk yöneticisinde var.

**z/OS** Kuyruk, z/OSüzerinde özel bir kuyruk olarak bilinir.

### **z/OS** Paylaşılan kuyruk (yalnızcaz/OS)

Kuyruk, paylaşılan havuzun sahibi olan kuyruk paylaşım grubuna ait olan tüm kuyruk yöneticilerinin erişebildiği paylaşılan bir havuzda var.

Kuyruk paylaşım grubundaki herhangi bir kuyruk yöneticisine bağlı uygulamalar, bu tipteki kuyruklara ileti yerleştirebilir ya da bu tipteki kuyruklara ileti kaldırılabilir. Bu tür kuyruklar, yerel kuyruklar ile aynı şekilde etkili olur. **QType** kuyruk özniteliğinin değeri MQQT\_LOCAL' dir.

Yerel kuyruk yöneticisine bağlı uygulamalar, bu tipteki kuyruklara ileti yerleştirebilir ya da bu tipteki kuyruklara ileti kaldırılabilir. **QType** kuyruk özniteliğinin değeri MQQT\_LOCAL' dir.

### Küme kuyruğu

Bir küme kuyruğunda, tanımlandığı yerde, bir küme kuyruğunda ileti saklayabilirsiniz. Küme kuyruğu, bir küme kuyruk yöneticisi tarafından barındırılan ve kümedeki diğer kuyruk yöneticilerinin kullanımına sunulan bir kuyruktur. **QType** kuyruğu özniteliğinin değeri MQQT\_CLUSTER' dir.

Kümedeki diğer kuyruk yöneticilerine bir küme kuyruğu tanımlaması tanıtılır. Kümedeki diğer kuyruk yöneticileri, iletileri karşılık gelen uzak kuyruk tanımlamasına gerek kalmadan bir küme kuyruğuna yerleştirebilirler. Küme kuyruğu, küme adı kullanılarak birden çok kümede duyurulabilir.

Bir kuyruk duyurulduğunda, kümedeki herhangi bir kuyruk yöneticisi bu kuyruğa ileti yerleştirebilir. Bir ileti koymak için kuyruk yöneticisinin, kuyruğun bulunduğu tam havuzlardan bunu öğrenmesi gerekir. Daha sonra iletiye bazı yöneltme bilgileri ekler ve iletiyi bir küme iletim kuyruğuna yerleştirir.

Kuyruk yöneticisi, birden çok iletim kuyruğundaki bir kümedeki diğer kuyruk yöneticilerine ilişkin iletileri saklayabilir. Bir kuyruk yöneticisini, iletileri birden çok küme iletim kuyruğunda iki farklı şekilde saklaması için yapılandırabilirsiniz. **DEFCLXQ** kuyruk yöneticisi özniteliğini CHANNELolarak ayarlarsanız, her küme gönderen kanalı için SYSTEM.CLUSTER.TRANSMIT.MODEL.QUEUE ' den otomatik olarak farklı bir küme iletim kuyruğu yaratılır. CLCHNAME iletim kuyruğu seçeneğini bir ya da daha çok küme gönderen kanalla eşleşecek şekilde ayarlarsanız, kuyruk yöneticisi, o iletim kuyruğundaki eşleşen kanallara ilişkin iletileri saklayabilir.



**Uyarı:** IBM WebSphere MQ 7.5'un önceki bir sürümünden yükseltilmiş bir kuyruk yöneticisiyle özel olarak ayrılmış SYSTEM.CLUSTER.TRANSMIT.QUEUES kullanıyorsanız, SYSTEM.CLUSTER.TRANSMIT.MODEL.QUEUE ' de SHARE/NOSHARE seçeneğinin **SHARE**olarak ayarlandığından emin olun.

**z/OS** Küme kuyruğu, IBM MQ for z/OS' daki bir kuyruk paylaşım grubunun üyeleri tarafından paylaşılan bir kuyruk olabilir.

### Uzak kuyruk

Uzak kuyruk fiziksel bir kuyruk değildir; uzak kuyruk yöneticisinde bulunan bir kuyruğun yerel tanımlamasıdır. Uzak kuyruğun yerel tanımlaması, yerel kuyruk yöneticisine iletilerin uzak kuyruk yöneticisine nasıl yönelteceğini bildiren bilgiler içerir.

Yerel kuyruk yöneticisine bağlı uygulamalar, bu tipteki kuyruklara ileti yerleştirebilir; iletiler, iletileri uzak kuyruk yöneticisine yöneltmek için kullanılan yerel iletim kuyruğuna yerleştirilir. Uygulamalar uzak kuyruklardan iletileri kaldıramaz. **QType** kuyruk özniteliğinin değeri MQQT\_REMOTE' dir.

Ayrıca, aşağıdakiler için uzak bir kuyruk tanımlaması da kullanabilirsiniz:

- Yanıt kuyruğu yöneltmesi

Bu durumda, tanımlamanın adı bir yanıtlama kuyruğu adıdır. Daha fazla bilgi için bakınız: [Reply-to queue aliases and clusters](#).

- Kuyruk yöneticisi yumuşatması

Bu durumda, tanımın adı bir kuyruk yöneticisinin diğer adı ve kuyruğun adı değildir. Daha fazla bilgi için bkz. [Kuyruk yöneticisi diğer adları ve kümeleri](#).

### Diğer ad kuyruğu

Bu bir fiziksel kuyruk değildir; yerel bir kuyruk, paylaşılan bir kuyruk, bir küme kuyruğu ya da uzak bir kuyruk için alternatif bir addir. Diğer ad çözücünün bulunduğu kuyruğun adı, diğer ad kuyruğunun bir parçası.

Yerel kuyruk yöneticisine bağlı uygulamalar, bu tipteki kuyruklara ileti yerleştirebilir; iletiler, diğer adın çözümleneceği kuyruğa yerleştirilir. Diğer ad, yerel bir kuyruğa, paylaşılan bir kuyruğa ya da yerel bir yönetim ortamına sahip olan bir küme kuyruğuna çözülürse, uygulamalar bu tipteki kuyruklardan iletileri kaldırabilir. **QType** kuyruk özniteliğinin değeri MQQT\_ALIAS' dır.

### Model kuyruğu

Bu bir fiziksel kuyruk değildir; yerel bir kuyruğun yaratılabileceği bir kuyruk öznitelikleri kümesidir.

İletiler bu tipteki kuyruklarda saklanamaz.

## kuyruk sınırları

V 9.2.0

IBM MQ 9.2.0' tan, daha önceki IBM MQ yayınlarında kullanılan iki terabayt varsayılan limitinden daha fazlasını destekleyecek kuyrukları yapılandırma ve izleme seçeneğiniz vardır. Ayrıca, kuyruk dosyasının büyüebileceği boyutu azaltma seçeneğiniz de vardır.

Kuyrukları yapılandırmanızı sağlamak için, yerel ve model kuyruklarında **MAXFSIZE** özniteliğini kullanabilir ve kuyrukları izlemek için, **CURFSIZE** ve **CURMAXFS** kuyruk durumu özniteliklerini kullanabilirsiniz.

Ek bilgi için [IBM MQ kuyruk dosyalarının değiştirilmesibaşlıklı konuya](#) bakın.

## Kuyruk öznitelikleri

Bazı kuyruk öznitelikleri tüm kuyruk tipleri için geçerlidir; diğer kuyruk öznitelikleri yalnızca belirli kuyruk tipleri için geçerlidir. Bir özniteliğin geçerli olduğu kuyruk tipleri [Çizelge 562 sayfa 824](#) içinde ve sonraki tablolarda gösterilir.

[Çizelge 562 sayfa 824](#) , kuyruklara özgü öznitelikleri özetler. Öznitelikler alfabetik sırayla açıklanır.

**Not:** Bu bölümde gösterilen özniteliklerin adları, MQINQ ve MQSET çağrıları ile kullanılan açıklayıcı adlardır; bu adlar PCF komutlarıyla aynıdır. MQSC komutları, öznitelikleri tanımlamak, değiştirmek ya da görüntülemek için kullanıldığında, diğer kısa adlar kullanılır; ayrıntılar için [MQSC komutları](#) konusuna bakın.

Aşağıdaki tabloda, sütunlar aşağıdaki gibi geçerlidir:

- Yerel kuyruklar için kolon da paylaşılan kuyruklara uygulanır.
- Model kuyruklarına ilişkin kolon, hangi özniteliklerin, model kuyruğundan yaratılan yerel kuyruk tarafından edinilmiş olduğunu gösterir.
- Küme kuyruklarına ilişkin kolon, küme kuyruğu tek başına sorgu için açıldığında ya da sorgu ve çıkış için açıldığında sorgulanabilen öznitelikleri gösterir. Herhangi bir öznitelik sorgulanırsa, çağrı tamamlanma kodu MQCC\_UYARCı değerini ve neden kodu MQRC\_SELECTOR\_NOT\_FOR\_TYPE (2068) sonucunu döndürür.

Bir ya da daha fazla giriş, göz atma ya da küme sorgulamak üzere küme kuyruğu açılırsa, bunun yerine yerel kuyruklar için kolon uygulanır.

Küme kuyruğu tek başına sorgulanmak üzere açılırsa ya da sorgu ve çıkış için artı temel kuyruk yöneticisi adını belirtirse, bunun yerine yerel kuyruklar için sütun uygulanır.

Çizelge 562. Kuyruklara ilişkin öznitelikler

Öznitelik	Tanım	Yerel	Model	Diğer Ad	Uzak	Küme
<a href="#">AlterationDate</a>	Tanımlamanın son olarak değiştirildiği tarih	X		X	X	
<a href="#">AlterationTime</a>	Tanımlamanın son olarak değiştirildiği saat	X		X	X	
<a href="#">BackoutRequeueQName</a>	Aşırı geriletme yeniden kuyruğa alma kuyruğu adı	X	X			
<a href="#">BackoutThreshold</a>	Geriletme eşiği	X	X			
<a href="#">BaseQName</a>	Diğer adın çözümleyileceği kuyruk adı			X		
<a href="#">CFStrucName</a>	Bağlaşım-tesis yapısı adı	X	X			
<a href="#">CLCHNAME</a>	Küme gönderici kanal adları	✓	✓			
<a href="#">ClusterName</a>	Kuyruğun ait olduğu kümenin adı	X		X	X	X
<a href="#">ClusterNamelist</a>	Kuyruğun ait olduğu kümelerin adlarını içeren ad listesi nesnesinin adı	X		X	X	
<a href="#">CLWLQueuePriority</a>	Küme iş yükü kuyruğu önceliği	X		X	X	X
<a href="#">CLWLQueueRank</a>	Küme iş yükü kuyruğu sırası	X		X	X	X
<a href="#">CLWLUseQ</a>	Uzak kuyruğu kullan	X				
<a href="#">CreationDate</a>	Kuyruğun yaratıldığı tarih	X				
<a href="#">CreationTime</a>	Kuyruğun yaratıldığı saat	X				
<a href="#">CurrentQDepth</a>	Yürürlükteki kuyruk derinliği	X				
<a href="#">DefaultPutResponse</a>	Varsayılan yanıt koyma yanıtı	✓	✓	✓	✓	
<a href="#">DefBind</a>	Varsayılan bağ tanımı	X		X	X	X
<a href="#">DefinitionType attribute</a>	Kuyruk tanımlaması tipi	X	X			
<a href="#">DefInputOpenOption</a>	Varsayılan giriş açma seçeneği	X	X			
<a href="#">DefPersistence</a>	Varsayılan ileti kalıcılığı	X	X	X	X	X
<a href="#">DefPriority</a>	Varsayılan ileti önceliği	✓	✓	✓	✓	✓
<a href="#">DefReadAhead</a>	Varsayılan okuma önden okuma	X	X	X		
<a href="#">DistLists</a>	Dağıtım listesi desteği	X	X			
<a href="#">HardenGetBackout</a>	Doğru bir geriletme sayısının sağlanıp korunmayacağı	X	X			
<a href="#">IndexType</a>	Dizin tipi	X	X			
<a href="#">InhibitGet</a>	Kuyruğa ilişkin alma işlemlerine izin verilip verilmeyeceği	X	X	X		
<a href="#">InhibitPut</a>	Kuyruğa ilişkin koyma işlemlerine izin verilip verilmeyeceği	X	X	X	X	X
<a href="#">InitiationQName</a>	Başlatma kuyruğunun adı	X	X			
<a href="#">MaxMsgLength</a>	İleti uzunluğu üst sınırı (byte)	X	X			
<a href="#">MaxQDepth</a>	Kuyruk derinliği üst sınırı	X	X			
<a href="#">MsgDeliverySequence attribute</a>	İleti teslim sırası	X	X			
<a href="#">NonPersistentMessage Class</a>	Kalıcı olmayan iletiler için güvenilirlik hedefi	X	X			
<a href="#">OpenInputCount</a>	Giriş için açma sayısı	X				

Çizelge 562. Kuyruklara ilişkin öznitelikler (devamı var)

Öznitelik	Tanım	Yerel	Model	Diğer Ad	Uzak	Küme
<a href="#">OpenOutputCount</a>	Çıkış için açma sayısı	X				
<a href="#">PropertyControl</a>	Özellik denetimi	✓	✓	✓		
<a href="#">ProcessName</a>	İşlem adı	X	X			
<a href="#">QDepthHighEvent attribute</a>	Kuyruk Derinliği Yüksek olaylarının oluşturulup oluşturulmayacağı	X	X			
<a href="#">QDepthHighLimit</a>	Kuyruk derinliği üst sınırı	X	X			
<a href="#">QDepthLowEvent attribute</a>	Kuyruk Derinliği Düşük olaylarının oluşturulup oluşturulmayacağı	X	X			
<a href="#">QDepthLowLimit attribute</a>	Kuyruk derinliği için alt sınır	X	X			
<a href="#">QDepthMaxEvent</a>	Kuyruk Dolu olaylarının oluşturulup oluşturulmayacağı	X	X			
<a href="#">QDesc</a>	Kuyruk tanımlaması	X	X	X	X	X
<a href="#">QName</a>	Kuyruk adı	X		X	X	X
<a href="#">QServiceInterval</a>	Kuyruk hizmet aralığı hedefi	X	X			
<a href="#">QServiceIntervalEvent attribute</a>	Hizmet Aralığı Yüksek ya da Hizmet Aralığı Tamam olaylarının oluşturulup oluşturulmayacağı	X	X			
<a href="#">QSGDisp attribute</a>	Kuyruk paylaşımı grubu yok etme	X		X	X	
<a href="#">QueueAccounting</a>	Kuyruk muhasebesi veri toplama	X	X	X	X	X
<a href="#">QueueMonitoring</a>	Kuyruklar için çevrimiçi izleme verileri	X	✓			
<a href="#">QueueStatistics</a>	Kuyruk istatistikleri veri toplama	X	X	X	X	X
<a href="#">QType</a>	Kuyruk tipi	X		X	X	X
<a href="#">RemoteQMgrName</a>	Uzak kuyruk yöneticisinin adı				X	
<a href="#">RemoteQName</a>	Uzak kuyruğun adı				X	
<a href="#">RetentionInterval</a>	Alıkoyma aralığı	X	X			
<a href="#">Scope</a>	Kuyruğa ilişkin bir girişin, bir hücre dizininde de bulunup bulunmayacağını	X		X	X	
<a href="#">Shareability</a>	Kuyruk paylaşılabilirliği	X	X			
<a href="#">StorageClass</a>	Kuyruğa ilişkin depolama sınıfı	X	X			
<a href="#">TriggerControl</a>	Tetikleyici denetimi	X	X			
<a href="#">TriggerData</a>	Verileri tetikle	X	X			
<a href="#">TriggerDepth</a>	Tetikleyici derinliği	X	X			
<a href="#">TriggerMsgPriority</a>	Tetikleyiciler için eşik iletisi önceliği	X	X			
<a href="#">TriggerType</a>	Tetikleme Tipi	X	X			
<a href="#">Usage attribute</a>	Kuyruk kullanımı	X	X			
<a href="#">XmitQName</a>	İletim kuyruğu adı				X	

**İlgili kavramlar**  
 Küme kuyrukları  
 Yerel kuyruklar

Kullanılacak küme iletim kuyruğunun tipini nasıl seçecek?

### **AlterationDate (MQCHAR12)**

Tanımlamanın en son değiştirildiği tarih.

Çizelge 563. Bu özneliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X		X	X	

Bu, tanımın en son değiştirildiği tarihtir. The format of the date is YYYY-MM-DD, padded with two trailing blanks to make the length 12 bytes (for example, 1992-09-23-- , where - represents a single blank character).

Kuyruk yöneticisi olarak, belirli özneliklerin değerleri (örneğin, *CurrentQDepth*) değişir. Bu özneliklerde yapılan değişiklikler *AlterationDate* işlemini etkilemez.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQCA\_ALTERATION\_DATE seçiciyi kullanın. Bu özneliğin uzunluğu MQ\_DATE\_LENGTH tarafından verilir.

### **AlterationTime (MQCHAR8)**

Tanımlamanın en son değiştirildiği saat.

Çizelge 564. Bu özneliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X		X	X	

Bu, tanımın en son değiştirildiği dönemdir. Saatin biçimi, saatin 10 'dan küçük olması (örneğin, 09.10.20gibi), 24 saatlik zaman biçimini kullanarak HH.MM.SS ' dir.

- z/OS'ta, sistem saatinin GMT' ye doğru olarak ayarlanmakta olduğu saat Greenwich Saat (GMT) olur.
- Diğer ortamlarda, saat yerel saattir.

Kuyruk yöneticisi olarak, belirli özneliklerin değerleri (örneğin, *CurrentQDepth*) değişir. Bu özneliklerde yapılan değişiklikler *AlterationTime* işlemini etkilemez.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQCA\_ALTERATION\_TIME seçiciyi kullanın. Bu özneliğin uzunluğu MQ\_TIME\_LENGTH tarafından verilir.

### **BackoutRequeueQName (MQCHAR48)**

Bu, aşırı geriletme yeniden kuyruğa alma kuyruğu adı. Değerin sorgulanmasına izin verilmesinin dışında, kuyruk yöneticisi bu özneliğin değerini temel alarak bir işlem gerçekleştirmez.

Çizelge 565. Bu özneliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

WebSphere Application Server içinde çalışan ve IBM MQ Application Server Faciliti'lerini kullanan uygulamalar, yedeklenen iletilerin nereye gidileceğini belirlemek için bu özneliği kullanır. Diğer tüm uygulamalar için, kuyruk yöneticisi özneliğin değerini temel alarak herhangi bir işlem gerektirir.

IBM MQ classes for JMS uses this attribute to determine where to transfer a message that has already been backed out the maximum number of times as specified by the *BackoutThreshold* attribute.

Bu özniteliğin deęerini saptamak için, MQINQ çağrısıyla MQCA\_BACKUUT\_REQ\_Q\_NAME seçicisini kullanın. Bu özniteliğin uzunluęu MQ\_Q\_NAME\_LENGTH tarafından verilir.

### **BackoutThreshold (MQLONG)**

Bu geriletme eřięi. Deęerin sorgulanmasına izin verilmesinin dıřında, kuyruk yöneticisi bu özniteliğin deęerini temel olarak bir iřlem geręekleřtirmez.

<i>Çizelge 566. Bu özniteliğin geęerli olduęu kuyruk tipleri</i>				
Yerel	Model	Dięer Ad	Uzak	Küme
X	X			

WebSphere Application Server içinde çalıřan uygulamalar ve IBM MQ Application Server Faciliti'lerini kullanan uygulamalar, bir iletinin geriletilmesi gerekip gerekmedięini belirlemek için bu öznitelięi kullanır. Dięer tüm uygulamalar için, kuyruk yöneticisi özniteliğin deęerini temel olarak herhangi bir iřlem gerektirir.

IBM MQ classes for JMS , iletiyi *BackoutQueueName* öznitelięi tarafından belirtilen kuyruęa aktarılmadan önce bir iletinin kaç kez geriletileceęini belirlemek için bu öznitelięi kullanır.

Bu özniteliğin deęerini saptamak için MQINQ çağrısıyla MQIA\_BACKOUT\_THRESHOLD seçiciyi kullanın.

### **BaseQName (MQCHAR48)**

Bu ad, yerel kuyruk yöneticisinde tanımlı olan bir kuyruęun adıdır.

<i>Çizelge 567. Bu özniteliğin geęerli olduęu kuyruk tipleri</i>				
Yerel	Model	Dięer Ad	Uzak	Küme
		X		

(Kuyruk adlarıyla ilgili daha fazla bilgi için bkz. [MQOD- ObjectName alanı.](#)) Kuyruk, ařaęıdaki tiplerden biridir:

#### **MQQT\_LOCAL**

Yerel kuyruk.

#### **MQQT\_REMOTE**

Uzak kuyruęun yerel tanımlaması.

#### **MQQT\_CLUSTER**

Küme kuyruęu.

Bu özniteliğin deęerini saptamak için, MQINQ çağrısıyla MQCA\_BASE\_QNAME seçiciyi kullanın. Bu özniteliğin uzunluęu MQ\_Q\_NAME\_LENGTH tarafından verilir.

### **BaseType (MQCFIN)**

Dięer adın çözdüęü nesne tipi.

<i>Çizelge 568. Bu özniteliğin geęerli olduęu kuyruk tipleri</i>				
Yerel	Model	Dięer Ad	Uzak	Küme
		X		

Bu deęer ařaęıdaki deęerlerden biridir:

#### **MQOT\_Q**

Temel nesne tipi bir kuyruktur

#### **MQOT\_KONUSU**

Temel nesne tipi bir konudur

### **CFStrucName (MQCHAR12)**


Bu ad, kuyruklardaki iletilerin saklandığı bağlantı olanağı yapısının adıdır. Adın ilk karakteri A-Z aralığında, kalan karakterler A-Z, 0-9 aralığında ya da boşluk karakterleridir.

Çizelge 569. Bu özneliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Bağlaşım tesisinde yapının tam adını almak için, **QSGName** kuyruk yöneticisi özneliğinin değerini **CFStrucName** kuyruk özneliği değeriyle sonek olarak yazın.

Bu öznelik yalnızca paylaşılan kuyruklar için geçerlidir; *QSGDi.sp* ' un MQQSGD\_SHARED değeri yoksa, yoksayılr.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQCA\_CF\_STRUC\_NAME seçicisini kullanın. Bu özneliğin uzunluğu, MQ\_CF\_STRUC\_NAME\_LENGTH tarafından verilir.

 Bu öznelik yalnızca z/OSüzerinde desteklenir.

### **ClusterChannelAd (MQCHAR20)**

ClusterChannelName , bu kuyruğu bir iletim kuyruğu olarak kullanan küme gönderen kanallarının sosyal adıdır. Öznelik, hangi küme-gönderen kanallarının bu küme iletim kuyruğundan bir küme-alıcı kanalına ileti gönderdiğini belirtir.

Çizelge 570. Bu özneliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Varsayılan kuyruk yöneticisi yapılandırması, tek bir iletim kuyruğundan ( SYSTEM . CLUSTER . TRANSMIT . QUEUE ) ileti göndermek için tüm küme gönderen kanalları içindir. Varsayılan yapılanış, **DefClusterXmitQueueType** kuyruk yöneticisi özneliği değiştirilerek değiştirilebilir. Özneliğin varsayılan değeri SCTQ' dur. Değeri CHANNELolarak değiştirebilirsiniz. **DefClusterXmitQueueType** özneliğini CHANNELolarak ayarlarsanız, her küme gönderen kanalı varsayılan olarak belirli bir küme iletim kuyruğunu ( SYSTEM . CLUSTER . TRANSMIT . ChannelName ) kullanır.

İletim kuyruğu özneliğini ClusterChannelName özneliğini kümeyi gönderen bir kanala el ile ayarlayabilirsiniz. Kümeyi gönderen kanal tarafından bağlanan kuyruk yöneticisine yönlendirilen iletiler, küme gönderen kanalı tanımlayan iletim kuyruğunda saklanır. Bunlar varsayılan küme iletim kuyruğunda saklanmaz. ClusterChannelName özneliğini boş olarak ayarlarsanız, kanal yeniden başlatıldığında kanal varsayılan küme iletim kuyruğuna geçer. Varsayılan kuyruk, kuyruk yöneticisinin değerine DefClusterXmitQueueType (Tip) özneliğine bağlı olarak SYSTEM . CLUSTER . TRANSMIT . ChannelName ya da SYSTEM . CLUSTER . TRANSMIT . QUEUEşeklinde olur.

**ClusterChannelName** içinde yıldız işaretlerini ( "\*" ) belirterek, bir iletim kuyruğunu bir küme gönderen kanalları kümesiyle ilişkilendirebilirsiniz. Yıldız işaretleri, kanal adı dizgisinin başında, sonunda ya da ortasında herhangi bir sayıda yer olabilir. **ClusterChannelName** 20 karakterlik bir uzunlukla sınırlıdır: MQ\_CHANNEL\_NAME\_LENGTH.

### **ClusterName (MQCHAR48)**

Bu, kuyruğun ait olduğu kümenin adıdır.

Çizelge 571. Bu özneliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X		X	X	X



Kuyruk birden çok kümeye aitse, *ClusterNameList* kümeleri tanımlayan bir ad listesi nesnesinin adını belirtir ve *ClusterName* boş olur. *ClusterName* ve *ClusterNameList* ' dan en az biri boş bırakılmalı.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQCA\_CLUSTER\_NAME seçiciyi kullanın. Bu özniteliğin uzunluğu MQ\_CLUSTER\_NAME\_LENGTH tarafından verilir.

### **ClusterNameList (MQCHAR48)**

Bu, bu kuyruğun ait olduğu kümelerin adlarını içeren bir ad listesi nesnesinin adıdır.

Çizelge 572. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X		X	X	

Kuyruk yalnızca bir kümeye aitse, ad listesi nesnesi yalnızca bir ad içerir. Alternatif olarak, *ClusterName* can be used to specify the name of the cluster, in which case *ClusterNameList* is blank. *ClusterName* ve *ClusterNameList* ' dan en az biri boş bırakılmalı.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQCA\_CLUSTER\_NAMELIST seçiciyi kullanın. Bu özniteliğin uzunluğu MQ\_NAMELIST\_NAME\_LENGTH tarafından verilir.

### **CLWLQueuePriority (MQUWN)**

Bu, küme iş yükü kuyruğu önceliğidir, kuyruğun önceliğini gösteren 0-9 aralığındaki bir değerdir.

Çizelge 573. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X		X	X	X

Ek bilgi için [Küme kuyruklarına](#) başlıklı konuya bakın.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_CLWL\_Q\_PRIORITY seçiciyi kullanın.

### **CLWLQueueRank (MQUZE)**

Bu, küme iş yükü kuyruğu sırasıdır; 0 ile 9 aralığındaki bir değer, kuyruğun sırasını simgeleyen bir değerdir.

Çizelge 574. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X		X	X	X

Ek bilgi için [Küme kuyruklarına](#) başlıklı konuya bakın.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_CLWL\_Q\_RANK seçiciyi kullanın.

### **CLWLUseQ (MQUSEL)**

Bu, hedef kuyruқта hem yerel bir yönetim ortamı, hem de en az bir uzak küme eşgörünümü varsa, MQPUT ' un davranışını tanımlar. Put originates from a cluster channel, this attribute does not apply.

Çizelge 575. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X				

Değer aşağıdakilerden biridir:

#### **MQCLWL\_USEQ\_ANY**

Uzak ve yerel kuyrukları kullanın.

### **MQCLWL\_USEQ\_LOCAL**

Uzak kuyrukları kullanmayın.

### **MQCLWL\_USEQ\_AS\_Q\_MGR**

Tanımlamayı kuyruk yöneticisinin MQIA\_CLWL\_USEQ ' dan devralır.

Ek bilgi için [Küme kuyruklar](#) başlıklı konuya bakın.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_CLWL\_USEQ seçiciyi kullanın. Bu özniteliğin uzunluğu MQ\_CLWL\_USEQ\_LENGTH tarafından verilir.

### **CreationDate (MQCHAR12)**

Bu, kuyruğun oluşturulduğu tarihtir.

<i>Çizelge 576. Bu özniteliğin geçerli olduğu kuyruk tipleri</i>				
Yerel	Model	Diğer Ad	Uzak	Küme
X				

Tarihin biçimi, uzunluğu 12 baytı (örneğin, 2013-09-23-- , -- , tek bir boş karakteri temsil eder) izleyen iki sondaki boşlukla doldurulmuş olarak YYYY-MM-DD biçimidir.

- IBM i' ta, bir kuyruğun oluşturulma tarihi, kuyruğu temsil eden temel işletim sistemi varlığından (dosya ya da kullanıcı alanı) farklı olabilir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQCA\_CREATION\_DATE seçiciyi kullanın. Bu özniteliğin uzunluğu MQ\_CREATION\_DATE\_LENGTH tarafından verilir.

### **CreationTime (MQCHAR8)**

Bu, kuyruğun yaratıldığı zamandır.

<i>Çizelge 577. Bu özniteliğin geçerli olduğu kuyruk tipleri</i>				
Yerel	Model	Diğer Ad	Uzak	Küme
X				

Saatin biçimi, saatin 10 'dan küçük olması (örneğin, 09 . 10 . 20 gibi), 24 saatlik zaman biçimini kullanarak HH.MM.SS ' dir.

- z/OS'ta, sistem saatinin GMT' ye doğru olarak ayarlanmakta olduğu saat Greenwich Saat (GMT) olur.
- Diğer ortamlarda, saat yerel saattir.
- IBM i' ta, bir kuyruğun oluşturulma zamanı, kuyruğu temsil eden temel işletim sistemi varlığından (dosya ya da kullanıcı alanı) farklı olabilir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQCA\_CREATION\_TIME seçiciyi kullanın. Bu özniteliğin uzunluğu MQ\_CREATION\_TIME\_LENGTH tarafından verilir.

### **CurrentQDepth (MQUZE)**

Bu, şu anda kuyruktaki iletilerin sayısıdır.

<i>Çizelge 578. Bu özniteliğin geçerli olduğu kuyruk tipleri</i>				
Yerel	Model	Diğer Ad	Uzak	Küme
X				

Bir MQPUT çağrısı sırasında ve bir MQGET çağrısının geri tepesi sırasında artırılır. Göz atma olmayan bir MQGET çağrısı sırasında ve bir MQPUT çağrısının geri aşımı sırasında bu değer azaltılır. Bunun etkisi, sayımın bir iş birimi içinde kuyruğa yerleştirilmiş olan, ancak MQGET çağrısıyla alınıp alınmalarına rağmen henüz kesinleştirilmemiş iletilerin içerilmesinin de kapsadığı bir etkidir. Benzer şekilde, MQGET

çağrısını kullanarak bir iş birimi içinde alınan iletileri dışlar, ancak henüz kesinleştirilmemiş iletiler de içerilecektir.

Bu iletiler, son kullanma saatlerini geçmiş, ancak henüz atılmamış olan iletileri de içerir, ancak bu iletiler alınmak için uygun değildir. Ek bilgi için [MQMD-Expiry field](#) başlıklı konuya bakın.

Unit-of-work processing and the segmentation of messages can both cause *CurrentQDepth* to exceed *MaxQDepth*. Ancak, bu ileti iletilerin alınabilirliğini etkilemez; kuyrukta bulunan tüm iletiler, olağan biçimde MQGET çağrısını kullanarak alınabilir.

Bu özniteliğin değeri kuyruk yöneticisi olarak işlev görmektedir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_CURRENT\_Q\_DEPTH seçiciyi kullanın.

### **DefaultPutYanıtı (MQUZE)**

Bir uygulama MQPMO\_RESPONSE\_AS\_Q\_DEF 'i belirlerken, kuyruğa veri yerleştirmek için kullanılacak yanıtın tipini belirtir.

Çizelge 579. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X	X	X	

Bu değer aşağıdaki değerlerden biridir:

#### **MQPRT\_SYNC\_RESPONSE**

Koyma işlemi zamanuyumlu olarak yayınlandı, yanıt döndürülüyor.

#### **MQPRT\_ASYNC\_RESPONSE**

Koyma işlemi zamanuyumsuz olarak yayınlanır ve MQMD alanlarının bir alt kümesini döndürür.

### **DefBind (MQLONG)**

Bu, MQOPED çağrısında MQOO\_BIND\_AS\_Q\_DEF belirtildiğinde ve kuyruk bir küme kuyruğu olduğunda, bu varsayılan bağ tanımıdır.

Çizelge 580. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X		X	X	X

Değer aşağıdakilerden biridir:

#### **MQBND\_BIND\_ON\_AçIK**

Bağ tanımı, MQOPER çağrısı tarafından düzeltilen

#### **MQBND\_BIND\_NOT\_FIXED**

Bağ tanımı düzeltilmedi.

#### **MQBND\_BIND\_ON\_GROUP**

Bir uygulamanın, bir ileti grubunun tümünün aynı hedef eşgörünümüne ayrılmasını istemesine olanak tanır. Bu değer IBM WebSphere MQ 7.1' de yeni olduğu için, bu kuyruğu açan uygulamaların herhangi biri IBM WebSphere MQ 7.0.1 ya da daha önceki kuyruk yöneticilerine bağlanıyorsa, bu değer kullanılmamalıdır.

Bu özniteliğin değerini saptamak için, MQINQ çağrısına sahip MQIA\_DEF\_BIND seçiciyi kullanın.

### **DefinitionType (MQUZE)**

Bu, kuyruğun nasıl tanımlandığını gösterir.

Çizelge 581. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Değer aşağıdakilerden biridir:

### **MQODT\_ÖNCEDEN tanımlı**

Kuyruk, sistem yöneticisi tarafından oluşturulan kalıcı bir kuyruktır; yalnızca sistem yöneticisi bunu silebilir.

Önceden tanımlanmış kuyruklar, DEFINE MQSC komutu kullanılarak yaratılır ve yalnızca DELETE MQSC komutu kullanılarak silinebilir. Önceden tanımlanmış kuyruklar, model kuyruklarından yaratılamaz.

Komutlar bir işletmen tarafından ya da komut giriş kuyruğuna bir komut iletisi gönderen yetkili bir kullanıcı tarafından yayınlanabilir (ek bilgi için [CommandInputQName özniteliği](#) başlıklı konuya bakın).

### **MQODT\_PERMANENT\_DYNAMIC**

Kuyruk, bir uygulama tarafından yaratılan kalıcı bir kuyruksa, MQOD nesne tanımlayıcısında belirtilen bir model kuyruğu adıyla MQOPER çağrısı yayınlayan bir çağrıdır. Model kuyruğu tanımı, **DefinitionType** özniteliği için MQODT\_PERMANENT\_DYNAMIC değerini aldı.

Bu kuyruk tipi, MQCLOSE çağrısını kullanarak silinebilir. Daha ayrıntılı bilgi için bkz. "[MQCLOSE-Nesneyi kapat](#)" sayfa 638 .

Kalıcı dinamik kuyruğa ilişkin **QSGDisp** özniteliğinin değeri MQQSGD\_Q\_MGR ' dir.

### **MQODT\_TEMPORARY\_DYNAMIC**

Kuyruk, bir uygulama tarafından yaratılan ve MQOD nesne tanımlayıcısı MQOD ' da belirtilen bir model kuyruğu adıyla MQOPER çağrısı yayınlayan geçici bir kuyruktır. Model kuyruğu tanımlaması, **DefinitionType** özniteliği için MQODT\_TEMPORARY\_DYNAMIC değerini aldı.

Bu kuyruk tipi, yaratılan uygulama tarafından kapatıldığında, MQCLOSE çağrısıyla otomatik olarak silinir.

Geçici dinamik kuyruğa ilişkin **QSGDisp** özniteliğinin değeri MQQSGD\_Q\_MGR ' dir.

### **MQODT\_SHARED\_DYNAMIC**

Kuyruk, bir uygulama tarafından yaratılan paylaşılan kalıcı bir kuyruksa, MQOD nesne tanımlayıcısında belirtilen bir model kuyruğu adıyla bir MQOPER çağrısı yayınlayan bir uygulama tarafından yaratılmış. Model kuyruğu tanımı, **DefinitionType** özniteliği için MQODT\_SHARED\_DYNAMIC değerini aldı.

Bu kuyruk tipi, MQCLOSE çağrısını kullanarak silinebilir. Daha ayrıntılı bilgi için bkz. "[MQCLOSE-Nesneyi kapat](#)" sayfa 638 .

Paylaşılan bir dinamik kuyruk için **QSGDisp** özniteliğinin değeri MQQSGD\_SHARED ' tir.

Bir model kuyruğu tanımlamasındaki bu öznitelik, model kuyruklarının her zaman önceden tanımlanmış olduğu için, model kuyruğunun nasıl tanımlandığını göstermez. Bunun yerine, model kuyruğundaki bu özniteliğin değeri, MQOPEN çağrısını kullanarak model kuyruğu tanımlamasından yaratılan devingen kuyrukların her birinin *DefinitionType* değerini saptamak için kullanılır.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_DEFINITION\_TYPE seçiciyi kullanın.

### **DefInputOpenOption (MQUZE)**

Bu, giriş için kuyruğun açılmasına ilişkin varsayılan yöntemdir.

Çizelge 582. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Kuyruk açıldığında, MQOPEN çağrısında MQOO\_INPUT\_AS\_Q\_DEF seçeneği belirtilirse geçerlidir. Değer aşağıdakilerden biridir:

### **MQOO\_INPUT\_EXCLUSIVE**

Dışlayıcı erişimi olan iletileri almak için kuyruğu açın.

Kuyruk sonraki MQGET çağrılarıyla kullanılmak üzere açılmıştır. Kuyruk şu anda bu ya da herhangi bir tip (MQOO\_INPUT\_SHARED ya da MQOO\_INPUT\_EXCLUSIVE) girişi için başka bir uygulama tarafından açıldıysa, çağrı neden kodu MQRC\_OBJECT\_IN\_USE ile başarısız olur.

### **MQOO\_INPUT\_SHARED**

Paylaşılan erişimi olan iletileri almak için kuyruğu açın.

Kuyruk sonraki MQGET çağrılarıyla kullanılmak üzere açılmıştır. Kuyruk şu anda bu ya da MQOO\_INPUT\_SHARED ile başka bir uygulama tarafından açılrsa başarılı olabilir, ancak kuyruk şu anda MQOO\_INPUT\_EXCLUSIVE ile açıldıysa, MQRC\_OBJECT\_IN\_USE neden koduyla başarısız olursa, çağrı başarılı olabilir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_DEF\_INPUT\_OPEN\_OPTION seçiciyi kullanın.

### **DefPersistence (MQUZE)**

Bu, kuyruktaki iletilerin varsayılan kalıcısıdır. İleti konulduğunda ileti tanımlayıcısında MQPER\_PERSISTENCE\_AS\_Q\_DEF değeri belirtilirse geçerlidir.

Çizelge 583. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X	X	X	X

Kuyruk adı çözümleme yolunda birden çok tanımlama varsa, MQPUT ya da MQPUT1 çağrısı sırasında yol içindeki ilk tanımlamasında varsayılan kalıcılık varsayılan kalıcılık değeri alınır. Bu durum şöyle olabilir:

- Diğer ad kuyruğu
- Yerel kuyruk
- Uzak kuyruğun yerel tanımlaması
- Kuyruk yöneticisi diğer adı
- Bir iletim kuyruğu (örneğin, *DefXmitQName* kuyruğu)

Değer aşağıdakilerden biridir:

### **MQPER\_PERISPER**

İleti, sistem hatalarını ve kuyruk yöneticisini yeniden başlatma işlemini sürdürür. Kalıcı iletiler üzerine yerleştirilemiyor:

- Geçici dinamik kuyruklar
- CFLEVEL (2) ya da aşağısında bir CFSTRUCT nesnesiyle eşlenen ya da CFSTRUCT nesnesinin RECOVER (NO) olarak tanımlı olduğu kuyruklar paylaşılır.

Kalıcı iletiler kalıcı dinamik kuyruklara ve önceden tanımlanmış kuyruklara yerleştirilebilir.

### **MQPER\_NOT\_PERSISTENT**

Bu ileti olağan durumda sistem arızalarına ya da kuyruk yöneticisi yeniden başlatılmalarına neden olmaz. Bu durum, bir kuyruk yöneticisi yeniden başlatıldığında yardımcı bellekte iletinin el değmemiş bir kopyası bulunsa bile geçerlidir.

Paylaşılan kuyruklar durumunda, kalıcı olmayan iletiler *do* kuyruk paylaşım grubundaki kuyruk yöneticilerinin yeniden başlatılmasını sağlar, ancak paylaşılan kuyruklarda iletileri depolamak için kullanılan bağlaşım tesisinin başarısızlıklarını atlatmaz.

Hem kalıcı hem de kalıcı olmayan iletiler aynı kuyruğun üzerinde var olabilir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_DEF\_PERSISTENCE seçiciyi kullanın.

### **DefPriority (MQUZE)**

Kuyruktaki iletiler için varsayılan öncelik budur. Bu, ileti kuyruğa konduğunda ileti tanımlayıcısında MQPRI\_PRIORITY\_AS\_Q\_DEF değeri belirtilirse geçerlidir.

Çizelge 584. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X	X	X	X

Kuyruk adı çözümleme yolunda birden çok tanımlama varsa, ileti için varsayılan öncelik, bu özniteliğin değerinden sonra, put işlemi sırasında yol içindeki *ilk* tanımlamasından alınır. Bu durum şöyle olabilir:

- Diğer ad kuyruğu
- Yerel kuyruk
- Uzak kuyruğun yerel tanımlaması
- Kuyruk yöneticisi diğer adı
- Bir iletim kuyruğu (örneğin, *DefXmitQName* kuyruğu)

Bir iletinin kuyruğa konacağı yol, kuyruğun **MsgDeliverySequence** özniteliğinin değerine bağlıdır:

- **MsgDeliverySequence** özniteliği MQMDS\_PRIORITY ise, bir iletinin kuyruğa konacağı mantıksal konum, ileti tanımlayıcısındaki *Priority* alanının değerine bağlıdır.
- **MsgDeliverySequence** özniteliği MQMDS\_FIFO ise, iletiler kuyrukta, ileti tanımlayıcısındaki *Priority* alanının değerinden bağımsız olarak, çözülmüş kuyruğun *DefPriority* değerine eşit bir önceliğe sahip olduğu gibi, kuyruğa yerleştirilir. Ancak, *Priority* alanı, iletiyi koyan uygulama tarafından belirtilen değeri korur. Daha fazla bilgi için bkz. [MsgDeliverySequence özniteliği](#) .

Priorities are in the range zero (lowest) through *MaxPriority* (highest); see [MaxPriority özniteliği](#).

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_DEF\_PRIORITY seçiciyi kullanın.

### DefReadİleri (MQUZE)

İstemciye teslim edilen kalıcı olmayan iletiler için varsayılan okuma tamamlama davranışını belirtir.

Çizelge 585. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X	X		

DefReadİleri düzey, aşağıdaki değerlerden birine ayarlanabilir:

#### MQREADA\_NO

Kalıcı olmayan iletiler, bir uygulama tarafından istekte bulunmadan önce istemciye gönderilmez. İstemci olağandışı şekilde sona ererse, en çok bir kalıcı olmayan ileti kaybedilebilir.

#### MQREADA\_YES

Kalıcı olmayan iletiler, bir uygulama tarafından istekte bulunmadan önce istemciye gönderilmektedir. İstemcinin olağandışı bir şekilde sona ermesi ya da istemcinin gönderdiği tüm iletileri tüketmemesi durumunda, kalıcı olmayan iletiler kaybedilebilir.

#### MQREADA\_DEVRE Dışı

Bu kuyruk için kalıcı olmayan iletilerin etkinleştirilmesini oku. İleriye ilişkin okuma, istemci uygulaması tarafından istenip istenmemesinden bağımsız olarak istemciye gönderilmez.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_DEF\_READ\_AHEAD seçiciyi kullanın.

### DefPResp (MQUZA)

MQPMO içindeki PutResponsetipi MQPMO\_RESPONSE\_AS\_Q\_DEF olarak ayarlanınca, varsayılan yanıt tipi (DEFPRESP) özniteliği, uygulamalar tarafından kullanılan değeri tanımlar. Bu öznitelik, tüm kuyruk tipleri için geçerlidir.

Çizelge 586. Bu özneteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X	X	X	X

Değer aşağıdakilerden biridir:

#### Sync

Koyma işlemi, zamanuyumlu olarak bir yanıt döndürerek yayınlanır.

#### Async

Koyma işlemi zamanuyumsuz olarak yayınlanır ve MQMD alanlarının bir alt kümesini döndürür.

Bu özneteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_DEF\_PUT\_RESPONSE\_TYPE seçiciyi kullanın.

#### DistLists (MQUZE)

Bu, dağıtım listesi iletilerinin kuyruğa yerleştirilip yerleştirilemeyeceğini gösterir.

Çizelge 587. Bu özneteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Bir ileti kanalı aracısı (MCA), kanalın diğer ucundaki kuyruk yöneticisinin dağıtım listelerini destekleyip desteklemediğini yerel kuyruk yöneticisine bildirmek için bu özneteliği ayarlar. Bu ikinci kuyruk yöneticisi (iş ortaklığı kuyruk yöneticisi olarak adlandırılır), bir gönderen MCA tarafından yerel iletim kuyruğundan kaldırıldıktan sonra, bir sonraki iletiyi alan kuyruk yöneticidir.

MCA'nın gönderilmesi, iş ortaklığı kuyruk yöneticisinde alınan MCA'ya yönelik bir bağlantı kurduğunda özneteliği ayarlar. Bu şekilde, MCA'nın gönderilmesi, yerel kuyruk yöneticisinin yalnızca iş ortağı kuyruk yöneticisinin doğru işleyebileceği iletileri ileti kuyruğuna yerleştirmesine neden olabilir.

Bu öznetelik öncelikle iletim kuyruklarıyla kullanım içindir, ancak tanımlanan işleme, kuyruk için tanımlanan kullanımdan bağımsız olarak gerçekleştirilir (bkz. [Kullanım özneteliği](#)).

Değer aşağıdakilerden biridir:

#### MQDL\_DESTEKLENEN

Dağıtım listesi iletileri kuyrukta saklanabilir ve bu formdaki iş ortaklığı kuyruk yöneticisine iletilebilir. Bu işlem, iletiyi birden çok hedefe göndermek için gereken işlem miktarını azaltır.

#### MQDL\_NOT\_SUPPORTANT

Ortak kuyruk yöneticisi dağıtım listelerini desteklemediğinden, dağıtım listesi iletileri kuyruğun üzerinde saklanamaz. Bir uygulama dağıtım listesi iletisi koyarsa ve bu ileti bu kuyruğa konacaksa, kuyruk yöneticisi dağıtım listesi iletisini böler ve her bir iletiyi kuyruğa yerleştirir. Bu, iletiyi birden çok hedefe göndermek için gereken işleme miktarını artırır, ancak iletilerin iş ortaklığı kuyruk yöneticisi tarafından doğru bir şekilde işlenmesini sağlar.

Bu özneteliğin değerini saptamak için, MQINQ çağrısına sahip MQIA\_DIST\_LISTS seçiciyi kullanın. Bu özneteliğin değerini değiştirmek için MQSET çağrısını kullanın.

Bu öznetelik z/OS üzerinde desteklenmez.

#### HardenGetGeri Al (MQUZE)

Her ileti için, bir iş birimi içindeki bir MQGET çağrısıyla ve daha sonra bu iş biriminin geriletmediği ileti sayısı, her ileti için bir MQGET çağrısının tutulmasını sağlar.

Çizelge 588. Bu özneteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Bu sayı, MQGET çağrısının tamamlanmasından sonra ileti tanımlayıcısında bulunan *BackoutCount* alanında bulunur.

İleti geriletme sayısı, kuyruk yöneticisinin yeniden başlatılabilmesine neden olur. Ancak, sayımın doğru olduğundan emin olmak için, bir MQGET çağrısının bu kuyruk için bir iş birimi içinde bir ileti aldığı her defasında bilgilerin *sertleştirilmiş* (disk ya da başka bir kalıcı depolama aygıtı üzerine kaydedilmelidir) olması gerekir. Bu işlem yapılmazsa, kuyruk yöneticisi başarısız olur ve MQGET çağrıları geri gelir, sayı artırılabilir ya da olmayabilir.

Hardening information for each MQGET call within a unit of work, however, imposes additional processing cost, so set the **HardenGetBackout** attribute to MQQA\_BACKOUT\_HARDENED only if it is essential that the count is accurate.

Multiplatforms' ta, bu özneliğin ayarından bağımsız olarak, ileti geri alma sayısı her zaman sertleştirilir.

Aşağıdaki değerler olanaklı:

#### **MQQA\_BACKOUT\_HARDENED**

Bu kuyruktaki iletilere ilişkin geriletme sayısının doğru olduğundan emin olmak için sertleştirme kullanılır.

#### **MQQA\_BACKOUT\_NOT\_HARDENED**

Bu kuyruktaki iletilere ilişkin geriletme sayısının doğru olduğundan emin olmak için sertleştirme kullanılmaz. Bu nedenle, sayı olması gerekenden daha düşük olabilir.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_HARDEN\_GET\_BACKUP seçiciyi kullanın.

### **IndexType (MQUZE)**

Kuyruk yöneticisinin kuyruklardaki iletiler için sakladığı dizin tipini belirtir.

Çizelge 589. Bu özneliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Gereken dizin tipi, uygulamanın iletileri nasıl aldığına ve kuyruğun paylaşılan bir kuyruk ya da paylaşılmayan bir kuyruk olup olmadığına bağlıdır (bkz. [QSGDisp özneliği](#)). *IndexType* için aşağıdaki değerler kullanılabilir:

#### **MQIT\_YOK**

Bu kuyruk için kuyruk yöneticisi tarafından hiçbir dizin korunmadı. Bu değeri, tipik olarak, MQGET çağrısında seçim ölçütü kullanmadan, sıralı olarak işlenen kuyruklar için kullanın.

#### **MQIT\_MSG\_ID**

Kuyruk yöneticisi, kuyruklardaki iletilerin ileti tanıtıcılarını kullanan bir dizini tutar. Uygulamanın genellikle ileti tanıtıcısını MQGET çağrısında seçim ölçütü olarak kullanarak iletileri aldığı bu değer kuyruklarını kullanın.

#### **MQIT\_COREL\_ID**

Kuyruk yöneticisi, kuyruklardaki iletilerin ilinti tanıtıcılarını kullanan bir dizini tutar. Uygulamanın genellikle, MQGET çağrısındaki seçim ölçütü olarak ilinti tanıtıcısını kullanarak iletileri aldığı kuyruklar için bu değeri kullanın.

#### **MQIT\_MSG\_TOKEN**

**Önemli:** Bu dizin tipi, yalnızca z/OS Workflow için IBM MQ Workflow ile kullanılan kuyruklar için kullanılmalıdır.

The queue manager maintains an index that uses the message tokens of the messages on the queue for use with the workload manager (WLM) functions of z/OS.

Siz *gerekir*, WLM tarafından yönetilen kuyruklar için bu seçeneği belirleyin; başka bir kuyruk tipi için bu seçeneği belirlemeyin. Ayrıca, bu değeri, uygulamanın z/OS iş yükü yöneticisi işlevlerini kullanmadığı, ancak MQGET çağrısında bir seçim ölçütü olarak ileti simgesini kullanarak iletileri almaya devam ettiği bir kuyruk için de kullanmayın.



## MQIT\_GROUP\_ID

Kuyruk yöneticisi, kuyruklardaki iletilerin grup tanıtıcılarını kullanan bir dizini tutar. Bu değer, uygulamanın MQGET çağrısındaki MQGMO\_LOGICAL\_ORDER seçeneğini kullanarak iletileri aldığı kuyruklar için kullanılmalıdır.

Bu dizin tipine sahip bir kuyruk iletim kuyruğu olamaz. Bu dizin tipini içeren bir paylaşılan kuyruk, CFLEVEL (3) ya da daha sonraki bir CFRUCT nesnesindeki bir CFSTRUCT nesnesiyle eşlenmek üzere tanımlanmalıdır.

### Not:

- MQGET çağrısındaki MQGMO\_LOGICAL\_ORDER seçeneğini kullanarak iletilerin verimli bir şekilde alınması için eniyilendiğinden, MQIT\_GROUP\_ID dizin tipli bir kuyrukla ilgili iletilerin fiziksel sırası tanımlanmadı. Başka bir deyişle, iletilerin fiziksel sırası, genellikle iletilerin bulunduğu sıraya göre değil, kuyrukta yer alan iletiler değildir.
- Bir MQIT\_GROUP\_ID kuyruğu *MsgDeliverySequence* MQMDS\_PRIORITY değerine sahipse, kuyruk yöneticisi, iletilerin mantıksal sırada alınmasını eniyilemek için 0 ve 1 numaralı ileti önceliklerini kullanır. Sonuç olarak, bir gruptaki ilk iletinin sıfır ya da bir önceliğine sahip olmamaları gerekir; eğer varsa, ileti iki önceliğe sahip olduğu gibi işlenir. MQMD yapısındaki *Priority* alanı değiştirilmez.

İleti gruplarıyla ilgili daha fazla bilgi için, [MQGMO-Seçenekler](#) alanındaki grup ve kesim seçeneklerine ilişkin açıklamalara bakın.

Çeşitli durumlarda kullanılması gereken dizin tipi [Çizelge 590 sayfa 837](#) ve [Çizelge 591 sayfa 838](#) içinde gösterilir.

Çizelge 590. MQGMO_LOGICAL_ORDER belirtilmediğinde, kuyruk dizini tipi için önerilen ya da gerekli değerler		
MQGET çağrısındaki seçim ölçütleri	Paylaşılmayan kuyruk için dizin tipi	Paylaşılan kuyruğa ilişkin dizin tipi
Yok	Herhangi	Herhangi
<b>Bir tanıtıcı kullanarak seçim:</b>		
İleti Tanıtıcısı	MQIT_MSG_ID önerilen	MQIT_NONE ya da MQIT_MSG_ID gerekli; MQIT_MSG_ID önerilendi
İlinti tanıtıcısı	MQIT_COREL_ID önerisi	MQIT_COREL_ID gerekli
Grup tanıtıcısı	MQIT_GROUP_ID önerilendi	MQIT_GROUP_ID gerekli
<b>İki tanıtıcı kullanarak seçim:</b>		
İleti tanıtıcısı ve ilinti tanıtıcısı	MQIT_MSG_ID ya da MQIT_COREL_ID önerilmişti	MQIT_NONE ya da MQIT_MSG_ID ya da MQIT_COREL_ID gerekli (Verimlilik için, dizin tipinin, en belirgin anahtarlarla sahip olacak MQMD alanıyla eşleştirmek için seçildiği önerilir)
İleti tanıtıcısı ve grup tanıtıcısı	MQIT_MSG_ID ya da MQIT_GROUP_ID önerilendi	Desteklenmiyor
İlinti tanıtıcısı ve grup tanıtıcısı	MQIT_COREL_ID ya da MQIT_GROUP_ID önerilendi	Desteklenmiyor
<b>Üç tanıtıcı kullanarak seçim:</b>		


Çizelge 590. MQGMO\_LOGICAL\_ORDER belirtilmediğinde, kuyruk dizini tipi için önerilen ya da gerekli değerler (devamı var)

<b>MQGET çağrısındaki seçim ölçütleri</b>	<b>Paylaşılmayan kuyruk için izin tipi</b>	<b>Paylaşılan kuyruğa ilişkin izin tipi</b>
İleti tanıtıcısı artı ilinti tanıtıcısı artı grup tanıtıcısı	MQIT_MSG_ID ya da MQIT_COREL_ID ya da MQIT_GROUP_ID önerilendi	Desteklenmiyor
<b>Grup ile ilgili ölçütler kullanarak seçim:</b>		
Grup tanıtıcısı ve ileti sıra numarası	MQIT_GROUP_ID gerekli	MQIT_GROUP_ID gerekli
İleti sıra numarası (1 olmalıdır)	MQIT_GROUP_ID gerekli	MQIT_GROUP_ID gerekli
<b>İleti simgesi kullanılarak seçim:</b>		
Uygulama kullanımı için ileti simgesi	MQIT_MSG_TOKEN kullanımını kullanmayın	
WLM kullanımı için ileti simgesi	MQIT_MSG_TOKEN gerekli	Desteklenmiyor

Çizelge 591. MQGMO\_LOGICAL\_ORDER belirlendiğinde, kuyruk dizini tipi için önerilen ya da gereken değerler

<b>MQGET çağrısındaki seçim ölçütleri</b>	<b>Paylaşılmayan kuyruk için izin tipi</b>	<b>Paylaşılan kuyruğa ilişkin izin tipi</b>
Yok	MQIT_GROUP_ID gerekli	MQIT_GROUP_ID gerekli
<b>Bir tanıtıcı kullanarak seçim:</b>		
İleti Tanıtıcısı	MQIT_GROUP_ID gerekli	Desteklenmiyor
İlinti tanıtıcısı	MQIT_GROUP_ID gerekli	Desteklenmiyor
Grup tanıtıcısı	MQIT_GROUP_ID gerekli	MQIT_GROUP_ID gerekli
<b>İki tanıtıcı kullanarak seçim:</b>		
İleti tanıtıcısı artı ilinti tanıtıcısı	MQIT_GROUP_ID gerekli	Desteklenmiyor
İleti tanıtıcısı ve grup tanıtıcısı	MQIT_GROUP_ID gerekli	Desteklenmiyor
İlinti tanıtıcısı ve grup tanıtıcısı	MQIT_GROUP_ID gerekli	Desteklenmiyor
<b>Üç tanıtıcı kullanarak seçim:</b>		
İleti tanıtıcısı artı ilinti tanıtıcısı artı grup tanıtıcısı	MQIT_GROUP_ID gerekli	Desteklenmiyor

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_INDEX\_TYPE seçiciyi kullanarak MQINQ adını kullanın.

 Bu öznitelik yalnızca z/OS üzerinde desteklenir.

### **InhibitGet (MQUZE)**

Bu, kuyruğa ilişkin alma işlemlerinin izin verilip verilmediğini denetler.

Çizelge 592. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X	X		

Kuyruk bir diğer ad kuyruksa, MQGET çağrısının başarılı olması için, alma işlemi sırasında hem diğer ad, hem de temel kuyruk için alma işlemlerine izin verilmelidir. Değer aşağıdakilerden biridir:

#### **MQQA\_GET\_INHIBITED**

Operasyonların engellenmesini engelliyorsunuz.

MQGET çağrıları, MQRC\_GET\_INHIBITED neden koduyla başarısız olur. Bu, MQGMO\_BROWSE\_FIRST ya da MQGMO\_BROWSE\_NEXT değerini belirten MQGET çağrıları içerir.

**Not:** Bir iş birimi içinde çalışan bir MQGET çağrısı başarıyla tamamlanırsa, **InhibitGet** özniteliğinin daha sonra MQQA\_GET\_INHIBLOCA değerine değiştirilmesi, iş biriminin kesinleştirilmesini engellememektedir.

#### **MQQA\_GET\_ALLOWED**

Alma işlemlerine izin verilir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_INHIBIT\_GET seçicisini kullanın. Bu özniteliğin değerini değiştirmek için MQSET çağrısını kullanın.

#### **InhibitPut (MQUZE)**

Bu, kuyruğa ilişkin olarak gerçekleştirilen işlemlerin yapılmasına izin verilip verilmediğini denetler.

Çizelge 593. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X	X	X	X

Kuyruk adı çözme yolunda birden çok tanımlama varsa, put işlemi sırasında, MQPUT ya da MQPUT1 çağrısının başarılı olması için yoldaki *her* tanımlaması (kuyruk yöneticisi diğer adı tanımlamaları da içinde olmak üzere) için işlemlere izin verilmelidir. Değer aşağıdakilerden biridir:

#### **MQQA\_PUT\_INHIBITED**

Put operasyonları engellenir.

MQPUT ve MQPUT1 çağrıları, MQRC\_PUT\_INHIBITED neden kodlarıyla başarısız olur.

**Not:** Bir iş birimi içinde çalışan bir MQPUT çağrısı başarıyla tamamlanırsa, daha sonra **InhibitPut** özniteliğinin değeri MQQA\_PUT\_ENGELLEYICI olarak değiştirilirse, iş biriminin kesinleştirilmesini engellememektedir.

#### **MQQA\_PUT\_ALLOWED**

Put işlemlerine izin verilir.

Bu özniteliğin değerini saptamak için MQINQ çağrısına sahip MQIA\_INHIBIT\_PUT seçiciyi kullanın. Bu özniteliğin değerini değiştirmek için MQSET çağrısını kullanın.

#### **InitiationQName (MQCHAR48)**

Bu, yerel kuyruk yöneticisinde tanımlı bir kuyruğun adıdır; kuyruğun MQQT\_LOCAL tipinde olması gerekir.

Çizelge 594. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X				

Kuyruk yöneticisi, bu özniteliğin ait olduğu kuyruğa gelen iletinin bir sonucu olarak, uygulama başlatma işlemi gerektiğinde, başlatma kuyruğuna bir tetikleyici iletisi gönderir. Başlatma kuyruğu, tetikleme

iletisinin alınmasından sonra uygun uygulamayı başlatan bir tetikleyici izleme uygulaması tarafından izlenmelidir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQCA\_INITIATION\_Q\_NAME seçicisini kullanın. Bu özniteliğin uzunluğu MQ\_Q\_NAME\_LENGTH tarafından verilir.

### **MaxMsgUzunluğu (MQUZE)**

Bu, kuyruğa yerleştirilebilecek en uzun *fiziksel* iletinin uzunluğuna ilişkin bir üst sınırdır.

Çizelge 595. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Ancak, **MaxMsgLength** kuyruk özniteliği **MaxMsgLength** kuyruk yöneticisi özniteinden bağımsız olarak ayarlanabildiği için, kuyruğa yerleştirilebilecek en uzun fiziksel iletinin uzunluğuna ilişkin gerçek üst sınır, bu iki değerden küçük olan uzunluğun en küçük olan sınırlarından biri olabilir.

Kuyruk yöneticisi kesimlere ayırma özelliğini destekliorsa, uygulama, iki **MaxMsgLength** özniteliğinin daha küçük olan *mantıksal* bir iletiyi, ancak uygulama MQMD ' de MQMF\_SEGMENTATION\_ALLOWLY işaretini belirtirse, bu iletiyi daha uzun bir yere koymanın mümkün olduğunu gösterir. Bu işaret belirlenirse, mantıksal ileti uzunluğuna ilişkin üst sınır 999 999 999 bayttır, ancak genellikle işletim sistemi tarafından uygulanan kaynak kısıtlamaları ya da uygulamanın çalıştığı ortam tarafından uygulanan kaynak kısıtları alt sınırdır olur.

Kuyruk üzerinde çok uzun bir ileti kuyruğa alma girişimi, aşağıdaki neden kodlarından biriyle başarısız olur:

- MQRC\_MSG\_TOO\_BIG\_FOR\_Q ileti kuyruk için çok büyükse
- MQRC\_MSG\_TOO\_BIG\_FOR\_Q\_MGR, ileti kuyruk yöneticisi için çok büyükse, ancak kuyruk için çok büyük değilse

**MaxMsgLength** özniteliği için alt sınır sıfır; üst sınır 100 MB ' dir (104 857 600 bayt).

Daha fazla bilgi için bakınız: [MQPUT- BufferLength](#) parameter.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_MAX\_MSG\_LENGTH seçiciyi kullanın.

### **MaxQDepth (MQUZE)**

Bu, herhangi bir zamanda kuyruksa bulunabilecek fiziksel ileti sayısı için tanımlanan üst sınırdır.

Çizelge 596. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

**MaxQDepth** iletilerini içeren bir kuyruğa ileti koyma girişimi başarısız oldu; neden kodu MQRC\_Q\_FULL neden kodlarıyla başarısız oldu.

İş biriminin işlenmesi ve iletilerin bölünmesinin her ikisi de kuyruksa gerçek fiziksel ileti sayısının **MaxQDepth** ' u aşmasına neden olabilir. Ancak, kuyruksa tüm iletiler MQGET çağrısını kullanarak alınabildiği için, bu ileti iletinin alınabilirliğini etkilemez.

Bu özniteliğin değeri sıfır ya da daha büyük. Üst sınır, ortam tarafından belirlenir:

- Aşağıdaki altyapılarda, değer 999 999 999 ' u aşamaz:

-  AIX
-  Linux
-  Windows

– z/OS z/OS

- IBM i IBM üzerinde değer 640 000 'i aşamaz.

**Not:** Kuyruktaki **MaxQDepth** iletisinden daha az ileti olsa da, kuyruğun kullanabileceği saklama alanı çok yorulmuş olabilir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_MAX\_Q\_DEPTH seçiciyi kullanın.

### **MsgDeliverySırası (MQUBE)**

Çizelge 597. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Bu, MQGET çağrısının uygulamaya ileti döndürdüğü sırayı belirler:

#### **MQMDS\_FIFO**

İletiler FIFO sırasına geri döndürülür (ilk olarak ilk giren ilk çıkar).

MQGET çağrısı, iletinin önceliği ne olursa olsun, çağrıda belirtilen seçim ölçütlerini karşılayan ilk iletisini döndürür.

#### **MQMDS\_PRIORITY**

İletiler öncelik sırasına geri döndürülür.

MQGET çağrısı, çağrıda belirtilen seçim ölçütlerine uyan *en yüksek öncelikli* iletiyi döndürür. Her bir öncelik düzeyi içinde, iletiler FIFO sırasına geri döndürülür (ilk olarak ilk giren ilk çıkar).

- On z/OS, if the queue has an *IndexType* of MQIT\_GROUP\_ID, the **MsgDeliverySequence** attribute specifies the order in which message groups are returned to the application. Grupların döndürüldüğü belirli sıra, her gruptaki ilk iletinin konumuna ya da önceliğine göre belirlenir. MQGET çağrısındaki MQGMO\_LOGICAL\_ORDER seçeneğini kullanarak iletilerin verimli bir şekilde alınması için eniyilendiği için, kuyruklardaki iletilerin fiziksel sırası tanımlanmadı.
- z/OS üzerinde, *IndexType* MQIT\_GROUP\_ID ve *MsgDeliverySequence* ise MQMDS\_PRIORITY ise, kuyruk yöneticisi ileti önceliklerini sıfır ve mantıksal düzende iletilerin alınmasını eniyilemek için kullanır. Sonuç olarak, bir gruptaki ilk iletinin sıfır ya da bir önceliğine sahip olmamaları gerekir; eğer varsa, ileti iki önceliğe sahip olduğu gibi işlenir. MQMD yapısındaki *Priority* alanı değiştirilmez.

Kuyruğun üzerinde iletiler varken ilgili öznitelikler değiştirilirse, teslim sırası aşağıdaki gibidir:

- The order in which messages are returned by the MQGET call is determined by the values of the **MsgDeliverySequence** and **DefPriority** attributes in force for the queue at the time that the message arrives on the queue:
  - If *MsgDeliverySequence* is MQMDS\_FIFO when the message arrives, the message is placed on the queue as though its priority were *DefPriority*. Bu, iletinin ileti tanımlayıcısındaki *Priority* alanının değerini etkilemez; alan, ileti ilk kez konulduğunda sahip olduğu değeri korur.
  - İleti geldiğinde *MsgDeliverySequence* MQMDS\_PRIORITY ise, ileti, ileti tanımlayıcısındaki *Priority* alanı tarafından verilen önceliğe uygun yerde kuyruğa yerleştirilir.

Kuyruktaki iletiler varken **MsgDeliverySequence** özniteliğinin değeri değiştirilirse, kuyruklardaki iletilerin sırası değiştirilmez.

Kuyrukta iletiler varken, **DefPriority** özniteliğinin değeri değiştirilirse, **MsgDeliverySequence** özniteliği MQMDS\_FIFO; olarak ayarlanmış olsa da iletiler, FIFO sırasına göre teslim edilmeyebilir; daha yüksek öncelikte kuyruğa yerleştirilen bu iletiler önce teslim edilir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_MSG\_DELIVERY\_SEQUENCE seçiciyi kullanın.

## NonPersistentMessageClass (MQUBE)

Kalıcı olmayan iletiler için güvenilirlik hedefi.

Çizelge 598. Bu özneliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Bu, bu kuyruğa ilişkin kalıcı olmayan iletilerin hangi koşullarda atıldığı belirtileceğini belirtir:

### MQNPM\_CLASS\_NORMAL

Kalıcı olmayan iletiler, kuyruk yöneticisi oturumunun geçerlilik süresi ile sınırlıdır; iletiler, kuyruk yöneticisi yeniden başlatılabilir. Bu yalnızca paylaşılmayan kuyruklar için geçerlidir ve varsayılan değerdir.

### MQNPM\_CLASS\_YÜKSEK

Kuyruk yöneticisi, kuyruğun ömrü boyunca kalıcı olmayan iletileri alıkoymayı dener. Bir hata durumunda kalıcı olmayan iletiler kaybolabilir. Bu değer, paylaşılan kuyruklar için zorlanır.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_NPM\_CLASS seçiciyi kullanın.

## OpenInputSayı (MQUZE)

Bu, şu anda MQGET çağrısıyla kuyruktan ileti kaldırmak için geçerli olan tanıtıcı sayısıdır.

Çizelge 599. Bu özneliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X				

Bu, yerel kuyruk yöneticisi tarafından bilinen tanıtıcı sayıların toplam sayısıdır. Kuyruk paylaşılan bir kuyruksa, yerel kuyruk yöneticisinin ait olduğu kuyruk paylaşım grubundaki diğer kuyruk yöneticilerindeki kuyruk için gerçekleştirilen giriş için sayım açılmaz.

Sayı, bu kuyruğa çözülen bir diğer ad kuyruğunun giriş için açıldığı tanıtıcıyı içerir. Sayı, giriş içermeyen işlemler için kuyruğun açıldığı noktaları içermez (örneğin, kuyruk yalnızca göz atma için açılmış).

Bu özneliğin değeri kuyruk yöneticisi olarak işlev görmektedir.

Bu özneliğin değerini saptamak için, MQINQ çağrısına sahip MQIA\_OPEN\_INPUT\_COUNT seçiciyi kullanın.

## OpenOutputSayı (MQUZE)

Bu, MQPUT çağrısıyla kuyruğa ileti eklenmesi için geçerli olan tanıtıcıların sayısıdır.

Çizelge 600. Bu özneliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X				

Bu, yerel kuyruk yöneticisi tarafından bilinen bu tür tanıtıcıların toplam sayısıdır; uzak kuyruk yöneticilerindeki bu kuyruk için gerçekleştirilen çıkışa ilişkin açma işlemlerini içermez. Kuyruk paylaşılan bir kuyruksa, yerel kuyruk yöneticisinin ait olduğu kuyruk paylaşım grubundaki diğer kuyruk yöneticilerindeki kuyruk için gerçekleştirilen çıkışa ilişkin sayı açılmaz.

Sayı, bu kuyruğa çözülen bir diğer ad kuyruğunun çıkış için açıldığı noktaları içerir. Sayı, çıkış içermeyen işlemler için kuyruğun açıldığı noktaları içermez (örneğin, kuyruk yalnızca sorgu için açılır).

Bu özneliğin değeri kuyruk yöneticisi olarak işlev görmektedir.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_OPEN\_OUTPUT\_COUNT seçiciyi kullanın.

### **ProcessName (MQCHAR48)**

Yerel kuyruk yöneticisinde tanımlı olan bir süreç nesnesinin adıdır. Süreç nesnesi, kuyruğa hizmet verebilen bir programı tanımlar.

<i>Çizelge 601. Bu özneliğin geçerli olduđu kuyruk tipleri</i>				
Yerel	Model	Diđer Ad	Uzak	Küme
X	X			

Bu özneliğin deđerini saptamak için, MQINQ çağrısıyla MQCA\_PROCESS\_NAME seçiciyi kullanın. Bu özneliğin uzunluđu MQ\_PROCESS\_NAME\_LENGTH tarafından verilir.

### **PropertyControl (MQUZE)**

MQGMO\_PROPERTIES\_AS\_Q\_DEF seçeneđiyle MQGET çağrısını kullanan kuyruklardan alınan iletiler için ileti özelliklerinin nasıl işleneceđini belirtir.

<i>Çizelge 602. Bu özneliğin geçerli olduđu kuyruk tipleri</i>				
Yerel	Model	Diđer Ad	Uzak	Küme
X	X	X		

Deđer aşağıdakilerden biridir:

#### **MQPROP\_ALL**

İletin tüm özellikleri, uygulamaya teslim edildiğinde ileti ile birlikte gönderilir. İleti tanımlayıcısında (ya da uzantısında) olanlar dışında, özellikler ileti verilerinde bir ya da daha çok MQRFH2 üstbilgilerine yerleştirilir. Bir ileti tanıtıcısı sağlanırsa, davranış, ileti tutamacındaki özellikleri döndürmeye neden olur.

#### **MQPROP\_UYUMLULUK**

İleti, mcd. öneğine sahip bir özellik içeriyorsa, jms., Usr. ya da mqext., Tüm ileti özellikleri, uygulamaya MQRFH2 üstbilgisinde teslim edilir. Ters durumda, ileti tanımlayıcısında (ya da uzantısında) yer alan durumlar dışında, iletinin tüm özellikleri atılır ve artık bu uygulamanın erişilemeyecek şekilde bulunur. Bu varsayılan deđerdir; JMS ile ilgili özelliklerin, deđiştirilmemiş çalışmaya devam etmek için ileti verilerindeki bir MQRFH2 üstbilgisinde olmasını bekleyen uygulamalara izin verir. Bir ileti tanıtıcısı sağlanırsa, davranış, ileti tanıtıcısındaki özellikleri döndürmeye neden olur.

#### **MQPROP\_FORCE\_MQRFH2**

Uygulamanın ileti tanıtıcısını belirtmesine bakılmaksızın, özellikler her zaman bir MQRFH2 üstbilgisindeki ileti verilerinde döndürülür. MQGET çağrısına ilişkin MQGMO yapısının MsgHandle alanında belirtilen geçerli bir ileti tanıtıcısı yok sayılır. İletin özelliklerine, ileti tanıtıcısı üzerinden erişilemez.

#### **MQPROP\_NONE**

İleti tanımlayıcısında (ya da uzantısında) olanlar dışında, iletinin tüm özellikleri, iletiden uygulamaya alınmadan önce kaldırılır. Bir ileti tanıtıcısı sağlanırsa, davranış, ileti tutamacındaki özellikleri döndürmeye neden olur.

Bu parametre Yerel, Diđer Ad ve Model kuyrukları için geçerlidir. Deđerini saptamak için, MQINQ çağrısıyla MQIA\_PROPERTY\_CONTROL seçiciyi kullanın.

### **QDepthHighOlayı (MQUZE)**

Kuyruk Derinliđi Yüksek olaylarının oluşturulup oluşturulmayacağını denetler.

<i>Çizelge 603. Bu özneliğin geçerli olduđu kuyruk tipleri</i>				
Yerel	Model	Diđer Ad	Uzak	Küme
X	X			

Kuyruk Derinliği Yüksek olayı, bir uygulamanın bir kuyruğa ileti yerleştirdiğini ve bu, kuyruktaki ileti sayısının kuyruk derinliği üst eşliğinden büyük ya da ona eşit olmasına neden olduğunu gösterir (**QDepthHighLimit** özneliğine bakın).

**Not:** Bu özneliğin değeri dinamik olarak değışebilir.

Değer ařağıdakilerden biridir:

**MQEVN\_DISABLE**

Olay raporlama devre dıřı bırakıldı.

**MQEVN\_ENABLED**

Olay raporlaması etkinleřtirildi.

Olaylarla ilgili daha fazla bilgi için [Olay izlemebařlıklı konuya](#) bakın.

Bu özneliğin değeri saptamak için, MQINQ çağrısıyla MQIA\_Q\_DEPTH\_HIGH\_EVENT seğıiciyi kullanın.

Bu öznelik z/OSüzerinde desteklenir, ancak bu özneliğin değeri belirlemek için MQINQ çağrısı kullanılamaz.

**QDepthHighSınırı (MQUZE)**

Kuyruk Derinliği Yüksek bir olay oluřturmak için kuyruk derinliğinin karřılařtırıldıđı eşik değeri budur.

<i>Çizelge 604. Bu özneliğin geđerli olduđu kuyruk tipleri</i>				
Yerel	Model	Diđer Ad	Uzak	Küme
X	X			

Bu olay, bir uygulamanın bir kuyruğa ileti koyduđunu ve kuyruğun kuyruk derinliği üst eşliğinden büyük ya da bu değere eşit olacak ileti sayısına neden olduđunu gösterir. Bkz. [QDepthHighOlay özneliđi](#).

Değer, kuyruk derinliği üst sınırı (**MaxQDepth** özneliđi) yüzdesi olarak ifade edilir ve 0 'dan büyük ya da ona eşit ve 100 'den küçük ya da 100 'e eşit ya da daha küçük bir değeri. Varsayılan değeri 80 'dir.

Bu özneliğin değeri saptamak için, MQINQ çağrısıyla MQIA\_Q\_DEPTH\_HIGH\_LIMIT seğıicisini kullanın.

Bu öznelik z/OSüzerinde desteklenir, ancak bu özneliğin değeri belirlemek için MQINQ çağrısı kullanılamaz.

**QDepthLowOlayı (MQUZE)**

Bu, Kuyruk Derinliği Düşük olaylarının oluřturulup oluřturulmayacađını denetler.

<i>Çizelge 605. Bu özneliğin geđerli olduđu kuyruk tipleri</i>				
Yerel	Model	Diđer Ad	Uzak	Küme
X	X			

Kuyruk Derinliği Düşük olayı, bir uygulamanın bir kuyruktan ileti alındıđını ve kuyruktaki ileti sayısının, kuyruk derinliği düşük eşliğinden küçük ya da bu değere eşit olmasına neden olduđunu gösterir (bkz. [QDepthLowSınır özneliđi](#)).

**Not:** Bu özneliğin değeri dinamik olarak değışebilir.

Değer ařağıdakilerden biridir:

**MQEVN\_DISABLE**

Olay raporlama devre dıřı bırakıldı.

**MQEVN\_ENABLED**

Olay raporlaması etkinleřtirildi.

Olaylarla ilgili daha fazla bilgi için [Olay izlemebařlıklı konuya](#) bakın.

Bu özneliğin değeri saptamak için, MQINQ çağrısıyla MQIA\_Q\_DEPTH\_LOW\_EVENT seğıiciyi kullanın.



Bu öznitelik z/OSüzerinde desteklenir, ancak bu özneliğin deęerini belirlemek için MQINQ çağrısı kullanılmaz.

### **QDepthLowSınırı (MQUZE)**

Kuyruk Derinlięi Düşük bir olay oluşturmak için kuyruk derinlięinin karşılaştırıldığı eşik budur.

<i>Çizelge 606. Bu özneliğin geçerli olduęu kuyruk tipleri</i>				
Yerel	Model	Dięer Ad	Uzak	Küme
X	X			

Bu olay, bir uygulamanın kuyruktan ileti aldığına ve kuyruğun kuyruk derinlięi düşük eşiğinden küçük ya da bu deęere eşit olmasına neden olduğunu gösterir. Bkz. QDepthLowOlay öznelięi.

Deęer, kuyruk derinlięi üst sınırı (**MaxQDepth** öznelięi) yüzdesi olarak ifade edilir ve 0 'dan büyük ya da ona eşit ve 100 'den küçük ya da 100 'e eşit ya da daha küçük bir deęer. Varsayılan deęer 20'dir.

Bu özneliğin deęerini saptamak için, MQINQ çağrısıyla MQIA\_Q\_DEPTH\_LOW\_LIMIT seçicisini kullanın.

Bu öznitelik z/OSüzerinde desteklenir, ancak bu özneliğin deęerini belirlemek için MQINQ çağrısı kullanılmaz.

### **QDepthMaxOlayı (MQUZE)**

Bu, Kuyruk Tam olaylarının oluşturulup oluşturulmayacağını denetler. Kuyruk Dolu olayı, kuyruğun dolu olduęu için, kuyruğun bir kuyruęa koyma deęerinin reddedildiğini, yani kuyruk derinlięinin zaten üst sınır deęerine ulaştığını gösterir.

<i>Çizelge 607. Bu özneliğin geçerli olduęu kuyruk tipleri</i>				
Yerel	Model	Dięer Ad	Uzak	Küme
X	X			

**Not:** Bu özneliğin deęeri dinamik olarak deęişebilir.

Deęer aşağıdakilerden biridir:

#### **MQEVR\_DISABLE**

Olay raporlama devre dışı bırakıldı.

#### **MQEVRENABLED**

Olay raporlaması etkinleştirildi.

Olaylarla ilgili daha fazla bilgi için [Olay izleme](#) başlıklı konuya bakın.

Bu özneliğin deęerini saptamak için, MQINQ çağrısıyla MQIA\_Q\_DEPTH\_MAX\_EVENT seçiciyi kullanın.

Bu öznitelik z/OSüzerinde desteklenir, ancak bu özneliğin deęerini belirlemek için MQINQ çağrısı kullanılmaz.

### **QDesc (MQCHAR64)**

Açıklayıcı açıklamalar için bu alanı kullanın.

<i>Çizelge 608. Bu özneliğin geçerli olduęu kuyruk tipleri</i>				
Yerel	Model	Dięer Ad	Uzak	Küme
X	X	X	X	X

Alanın içerięi kuyruk yöneticisi için önemli deęildir; ancak kuyruk yöneticisi, alanın yalnızca görüntülenebilecek karakterleri içermesini gerektirebilir. Boş deęerli karakterler içeremez; gerekirse, boşlukla doldurulabilir. Bir DBCS kuruluşunda, alan DBCS karakterleri içerebilir (en çok 64 baytlık alan uzunluęuna tabi).

**Not:** Bu alan, kuyruk yöneticisinin karakter kümesinde olmayan karakterler içeriyorsa ( **CodedCharSetId** kuyruk yöneticisi öznitelemesiyle tanımlandığı gibi), bu alan başka bir kuyruk yöneticisine gönderilirse, bu karakterler yanlış çevrilebilir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQCA\_Q\_DESC seçiciyi kullanın. Bu özniteliğin uzunluğu MQ\_Q\_DESC\_LENGTH tarafından verilir.

### **QName (MQCHAR48)**

Bu ad, yerel kuyruk yöneticisinde tanımlı olan bir kuyruğun adıdır.

<i>Çizelge 609. Bu özniteliğin geçerli olduğu kuyruk tipleri</i>				
Yerel	Model	Diğer Ad	Uzak	Küme
X		X	X	X

Bir kuyruk yöneticinde tanımlı olan tüm kuyruklar, aynı kuyruk ad alanını paylaşır. Bu nedenle, bir MQQT\_LOCAL kuyruğu ve MQQT\_ALIAS kuyruğu aynı ada sahip olamaz.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQCA\_Q\_NAME seçicisini kullanın. Bu özniteliğin uzunluğu MQ\_Q\_NAME\_LENGTH tarafından verilir.

### **QServiceInterval (MQUZE)**

Bu, Hizmet Aralığı Yüksek ve Hizmet Aralığı Tamam olayları oluşturmak için karşılaştırma için kullanılan hizmet aralığından bir değer.

<i>Çizelge 610. Bu özniteliğin geçerli olduğu kuyruk tipleri</i>				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Bkz. [QServiceIntervalOlay](#) özniteliği.

Değer, milisaniye cinsinden, ve sıfırdan büyük ya da sıfıra eşit ve 999 999 999 'dan küçük ya da 999 'a eşit.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_Q\_SERVICE\_INTERVAL seçiciyi kullanın.

Bu öznitelik z/OSüzerinde desteklenir, ancak bu özniteliğin değerini belirlemek için MQINQ çağrısı kullanılamaz.

### **QServiceIntervalOlayı (MQUZE)**

Bu, Hizmet Aralığı Yüksek ya da Hizmet Aralığı Tamam olaylarının oluşturulup oluşturulmayacağını denetler.

<i>Çizelge 611. Bu özniteliğin geçerli olduğu kuyruk tipleri</i>				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

- Bir Hizmet Aralığı Yüksek olayı, **QServiceInterval** özniteliğinden en az belirtilen zaman için kuyruktan herhangi bir ileti alınmadığına işaret edildiğinde oluşturulur.
- Bir Hizmet Aralığı Tamam olayı, **QServiceInterval** özniteliğinden belirtilen süre içinde kuyruktan iletilerin alındığını gösterdiğinde oluşturulur.

**Not:** Bu özniteliğin değeri dinamik olarak değişebilir.

Değer aşağıdakilerden biridir:

#### **MQQSI\_YÜKSEK**

Kuyruk Hizmet Aralığı Yüksek olayları etkinleştirildi.

- Kuyruk Hizmet Aralığı Yüksek olayları **etkin** ve

- Kuyruk Hizmet Aralığı Tamam olayları **devre dışı**.

### **MQCSI\_OK**

Kuyruk Hizmet Aralığı Tamam olayları etkinleştirildi.

- Kuyruk Hizmet Aralığı Yüksek olayları **geçersiz kılındı** ve
- Kuyruk Hizmet Aralığı Tamam olayları **etkindir**.

### **MQCSI\_NONE**

Kuyruk hizmeti aralık olayı etkinleştirilmedi.

- Kuyruk Hizmet Aralığı Yüksek olayları **geçersiz kılındı** ve
- Kuyruk Hizmet Aralığı Tamam olayları da **devre dışı bırakıldı**.

Paylaşılan kuyruklar için, bu özniteliğin değeri yoksayılır; MQCSIE\_NONE değeri kabul edilir.

Olaylarla ilgili daha fazla bilgi için [Olay izleme](#) başlıklı konuya bakın.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_Q\_SERVICE\_INTERVAL\_EVENT seçiciyi kullanın.

z/OS' ta, bu özniteliğin değerini saptamak için MQINQ çağrısını kullanamazsınız.

### **QSGDisp (MQHOT)**

Bu, kuyruğun yok edilmesini belirtir.

<i>Çizelge 612. Bu özniteliğin geçerli olduğu kuyruk tipleri</i>				
<b>Yerel</b>	<b>Model</b>	<b>Diğer Ad</b>	<b>Uzak</b>	<b>Küme</b>
X		X	X	

Değer aşağıdakilerden biridir:

### **MQQSGD\_Q\_MGR**

Nesnenin kuyruk yöneticisi yok etme işlemi var. Bu, nesne tanımlamasının yalnızca yerel kuyruk yöneticisinde bilindiği anlamına gelir; tanımlama, kuyruk paylaşım grubundaki diğer kuyruk yöneticilerine bilinmez.

Kuyruk paylaşım grubundaki her kuyruk yöneticisinin adı aynı olan ve yürürlükteki nesneyle aynı olan bir nesne olabilir, ancak bunlar ayrı nesnedir ve bunlar arasında ilinti yoktur. Onların özniteliklerinin birbiriyle aynı olması engellenir.


### **MQQSGD\_COPY**

Nesne, paylaşılan havuzda var olan bir ana nesne tanımlamasının yerel bir kopyasıdır. Kuyruk paylaşım grubundaki her kuyruk yöneticisi nesnenin kendi kopyasına sahip olabilir. Başlangıçta, tüm kopyalar aynı özniteliklere sahiptir, ancak MQSC komutlarını kullanarak, her bir kopyayı değiştirerek, özniteliklerinin diğer kopyalardaki özniteliklerine göre değişiklik göstermesini de yapabilirsiniz. Paylaşılan havuzdaki ana tanım değiştirildiğinde, kopyaların öznitelikleri yeniden uyumlulaştırılır.

### **MQQSGD\_SHARED**

Nesne, paylaşılmış yok etme içeriyor. Bu, paylaşılan havuzda, kuyruğun paylaşım grubundaki tüm kuyruk yöneticileri tarafından bilinen tek bir nesnenin tek bir eşgörünümü var demektir. Gruptaki bir kuyruk yöneticisi nesneye eriştiğinde, nesneye ilişkin tek paylaşılan eşgörünüme erişir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_QSG\_DISP seçicisini kullanın.

 Bu öznitelik yalnızca z/OS üzerinde desteklenir.

### **QueueAccounting (MQHOMER)**

Çizelge 613. Bu özneliğin geçerli olduđu kuyruk tipleri				
Yerel	Model	Diđer Ad	Uzak	Küme
X	X	X	X	

Bu, kuyruđa iliřkin muhasebe verilerinin toplanmasını denetler. Bu kuyruk için toplanacak hesap verileri için, MQCONNX çağrısındaki MQCNO yapısındaki QMGR özneliđi ACCTQ ya da Options alanı kullanılarak, bu bađlantıya iliřkin hesap verileri de etkinleřtirilmelidir.

Bu öznelik ařađıdaki deđerlerden birine sahiptir:

#### **MQMON\_Q\_MGR**

Bu kuyruđa iliřkin muhasebe verileri, ACCTQ QMGR özneliđi ayarına dayalı olarak toplanır. Bu varsayılan ayardır.

#### **MQMON\_OFF**

Bu kuyruk için hesap verileri toplanmasın.

#### **MQMON\_ON**

Bu kuyruk için muhasebe verilerini topla.

Bu özneliđin deđerini saptamak için, MQINQ çağrısıyla MQIA\_ACCOUNTING\_Q seçiciyi kullanın.

### **QueueMonitoring (MQUZE)**

Kuyruklar için çevrimiçi izleme verilerinin toplanmasını denetler.

Çizelge 614. Bu özneliğin geçerli olduđu kuyruk tipleri				
Yerel	Model	Diđer Ad	Uzak	Küme
X	X			

Deđer ařađıdakilerden biridir:

#### **MQMON\_Q\_MGR**

İzleme verilerini **QueueMonitoring** kuyruk yöneticisi özneliđinin ayarına göre toplayın. Bu varsayılan deđerdir.

#### **MQMON\_OFF**

Bu kuyruk için çevrimiçi izleme verileri toplama kapatılır.

#### **MQMON\_LOW**

**QueueMonitoring** kuyruk yöneticisi özneliđinin deđeri MQMON\_NONE deđilse, çevrimiçi izleme verileri derlemi açık olur ve bu kuyruk için düşük veri toplama işlemleri olur.

#### **MQMON\_ORTAMı**

**QueueMonitoring** kuyruk yöneticisi özneliđinin deđeri MQMON\_NONE deđilse, çevrimiçi izleme verileri derlemi açık olur ve bu kuyruk için orta düzeyde veri toplama işlemleri vardır.

#### **MQMON\_YÜKSEK**

**QueueMonitoring** kuyruk yöneticisi özneliđinin deđeri MQMON\_NONE deđilse, çevrimiçi izleme verileri derlemi açık olur ve bu kuyruk için yüksek oranda veri toplama işlemleri vardır.

Bu özneliđin deđerini saptamak için, MQINQ çağrısıyla MQIA\_MONITORING\_Q seçiciyi kullanın.

### **QueueStatistics (MQCHAR12)**

Çizelge 615. Bu özneliğin geçerli olduđu kuyruk tipleri				
Yerel	Model	Diđer Ad	Uzak	Küme
X	X	X	X	

Bu, kuyruğa ilişkin istatistik verilerinin toplanmasını denetler.

Bu öznitelik aşağıdaki değerlerden birine sahiptir:

#### **MQMON\_Q\_MGR**

Bu kuyruğa ilişkin muhasebe verileri, QMGR özniteliği STATQ ' nun ayarına dayalı olarak toplanır. Bu varsayılan ayardır.

#### **MQMON\_OFF**

Bu kuyruk için istatistik verileri toplama işlemi kapatılıyor.

#### **MQMON\_ON**

Bu kuyruk için istatistik verileri toplamasını etkinleştirin.

### **QType (MQUZE)**

Çizelge 616. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X		X	X	X

Bu, kuyruğun tipidir; aşağıdaki değerlerden birine sahiptir:

#### **MQQT\_ALIAS**

Diğer ad kuyruğu tanımlaması.

#### **MQQT\_CLUSTER**

Küme kuyruğu.

#### **MQQT\_LOCAL**

Yerel kuyruk.

#### **MQQT\_REMOTE**

Uzak kuyruğun yerel tanımlaması.

Bu özniteliğin değerini saptamak için, MQINQ çağırısıyla MQIA\_Q\_TYPE seçiciyi kullanın.

### **RemoteQMgrAd (MQCHAR48)**

Çizelge 617. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
			X	

This is the name of the remote queue manager on which the queue **RemoteQName** is defined.

**RemoteQName** kuyruğu, MQQSGD\_COPY ya da MQQSGD\_SHARED 'in bir **QSGDisp** değerine sahipse, **RemoteQMgrName** , **RemoteQName** ' un sahibi olan kuyruk paylaşım grubunun adı olabilir.

Bir uygulama uzak kuyruğun yerel tanımlamasını açarsa, **RemoteQMgrName** boş bırakılmamalı ve yerel kuyruk yöneticisinin adı olmamalıdır. **XmitQName** boşsa, iletim kuyruğu olarak **RemoteQMgrName** ile aynı adı taşıyan yerel kuyruk kullanılır. **RemoteQMgrName** adlı bir kuyruk yoksa, **DefXmitQName** kuyruk yöneticisi özniteliği tarafından tanımlanan kuyruk kullanılır.

Bu tanımlama bir kuyruk yöneticisi diğer adı için kullanılırsa, **RemoteQMgrName** , alışan olan kuyruk yöneticisi adıdır. Yerel kuyruk yöneticisinin adı olabilir. Ters durumda, açık ortaya çıktığında **XmitQName** boşsa, **RemoteQMgrName** ile aynı olan bir yerel kuyruk olması gerekir; bu kuyruk iletim kuyruğu olarak kullanılır.

If this definition is used for a reply-to alias, this name is the name of the queue manager that is to be the **ReplyToQMgr**.

**Not:** Kuyruk tanımlaması yaratıldığında ya da değiştirildiğinde, bu öznitelik için belirtilen değerde geçerlilik denetimi gerçekleştirilmez.

Bu özniteliğin deęerini saptamak için, MQINQ çağrısıyla MQCA\_REMOTE\_Q\_MGR\_NAME seçicisini kullanın. Bu özniteliğin uzunluęu, MQ\_Q\_MGR\_NAME\_LENGTH tarafından verilir.

### **RemoteQName (MQCHAR48)**

Çizelge 618. Bu özniteliğin geçerli olduęu kuyruk tipleri				
Yerel	Model	Dięer Ad	Uzak	Küme
			X	

Bu, uzak kuyruk yöneticisinde bilindięi gibi, kuyruęun adıdır. *RemoteQMgrName*.

Bir uygulama uzak bir kuyruęun yerel tanımlamasını açarsa, açma işlemi *RemoteQName* boş bırakılmamalı.

Bu tanımlama bir kuyruk yöneticisi dięer adı tanımlaması için kullanılırsa, açma işlemi *RemoteQName* boş bırakılmalıdır.

Tanımlama bir yanıt-yanıt dięer adı için kullanıldıysa, bu ad, *ReplyToQ* olacak olan kuyruęun adıdır.

**Not:** Kuyruk tanımlaması yaratıldıęında ya da deęiştirildięinde, bu öznitelik için belirtilen deęerde geçerlilik denetimi gerçekleştirilmez.

Bu özniteliğin deęerini saptamak için, MQINQ çağrısıyla MQCA\_REMOTE\_Q\_NAME seçicisini kullanın. Bu özniteliğin uzunluęu MQ\_Q\_NAME\_LENGTH tarafından verilir.

### **RetentionInterval (MQUZE)**

Bu, kuyruęun alıkonacaęı süre (time). Bu süre geçtikten sonra kuyruk silinmeye uygun olur.

Çizelge 619. Bu özniteliğin geçerli olduęu kuyruk tipleri				
Yerel	Model	Dięer Ad	Uzak	Küme
X	X			

Saat, kuyruęun oluşturulduęu tarih ve saatten başlayarak saat olarak ölçülür. Kuyruęun oluşturulma tarihi ve saati **CreationDate** ve **CreationTime** özniteliklerinde kaydedilir.

Bu bilgiler, artık gerekli olmayan kuyrukları tanımlamak ve silmek için bir ev bakımı uygulamasının ya da işletmenin etkinleştirilmesini sağlamak amacıyla sağlanmışır.

**Not:** Kuyruk yöneticisi, bu öznitelige dayalı olarak kuyrukları silmek ya da süresi dolmayan bir alıkoyma aralıęıyla kuyrukların silinmesini önlemek için hiçbir zaman herhangi bir işlem kullanmaz; bu, gereken işlemi yapmak kullanıcının sorumluluęundadır.

Kalıcı dinamik kuyrukların birikmesini önlemek için gerçekçi bir alıkoyma aralıęı kullanın (bkz. [DefinitionType öznitelięi](#)). Ancak, bu öznitelik önceden tanımlanmış kuyruklarla da kullanılabilir.

Bu özniteliğin deęerini saptamak için, MQINQ çağrısıyla MQIA\_RETENTION\_INTERVAL seçiciyi kullanın.

### **Kapsam (MQLONG)**

Bu, bu kuyruęa ilişkin bir girişin bir hücre dizininde de olup olmadıęını denetler.

Çizelge 620. Bu özniteliğin geçerli olduęu kuyruk tipleri				
Yerel	Model	Dięer Ad	Uzak	Küme
X		X	X	

Kurulabilir bir Ad hizmeti tarafından bir hücre dizini sağlanır. Deęer aşıęıdakilerden biridir:

#### **MQSCO\_Q\_MGR**

Kuyruk tanımlamasının kuyruk yöneticisi kapsamı var: Kuyruęun tanımlaması, sahip olduęu kuyruk yöneticisinin ötesine geçmiyor. Kuyruęu başka bir kuyruk yöneticisinden çıkış için açmak için,

sahip olan kuyruk yöneticisinin adı belirtilmelidir ya da diğer kuyruk yöneticisinin kuyruğun yerel tanımlaması olmalıdır.

### **MQSCO\_CEL**

Kuyruk tanımlamasının hücre kapsamı vardır: Kuyruk tanımlaması, hücredeki tüm kuyruk yöneticilerinin kullanımına sunulan bir hücre dizinine de yerleştirilir. Kuyruk, kuyruğun adını belirterek, hücredeki kuyruk yöneticilerinden herhangi birinden çıkış için açılabilir; kuyruğun sahibi olan kuyruk yöneticisinin adı belirtilmemelidir. Ancak, kuyruk tanımlaması, yerel tanımlama önceliğe sahip olduğu için, bu adı taşıyan bir kuyruğun yerel tanımlamasını da içeren bir kuyrukta bulunan kuyruk yöneticisinde kullanılamaz.

Kurulabilir bir Ad hizmeti tarafından bir hücre dizini sağlanır.

Model ve dinamik kuyrukların hücre kapsamı olamaz.

Bu değer yalnızca, bir hücre dizinini destekleyen bir ad hizmeti yapılandırıldıysa geçerlidir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_SCOPE seçicisini kullanın.

Bu özniteliğe ilişkin destek aşağıdaki kısıtlamalara tabidir:

- IBM üzerinde öznitelik desteklenir, ancak yalnızca MQSCO\_Q\_MGR geçerli olur.
- z/OS üzerinde öznitelik desteklenmiyor.

### **Paylaşılabilirlik (MQUZE)**

Bu, kuyruğun koştuzamanlı olarak birden çok kez giriş için açılıp açılmayacağını gösterir.

Çizelge 621. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Değer aşağıdakilerden biridir:

#### **MQQA\_SHAREABLE**

Kuyruk paylaşılabilir.

MQOO\_INPUT\_SHARED seçeneğiyle birden çok açma işlemine izin verilir.

#### **MQQA\_NOT\_SHAREABLE**

Kuyruk paylaşılabilir değil.

MQOO\_INPUT\_SHARED seçeneğiyle MQOPEN çağrısı MQOO\_INPUT\_EXCLUSIVE olarak işlem görür.


Bu özniteliğin değerini saptamak için MQINQ çağrısıyla MQIA\_SHAREABILITY seçiciyi kullanın.

### **StorageClass (MQCHAR8)**

Bu ad, kuyruğu tutmak için kullanılan fiziksel depolamayı tanımlayan kullanıcı tanımlı bir addır. Uygulamada, yalnızca bellek arabelleğinden sayfalanır olması gerekiyorsa, bir ileti diske yazılır.

Çizelge 622. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Bu özniteliğin değerini saptamak için, MQINQ çağrısına sahip MQCA\_STORAGE\_CLASS seçiciyi kullanın. Bu özniteliğin uzunluğu MQ\_STORAGE\_CLASS\_LENGTH tarafından verilir.

 Bu öznitelik yalnızca z/OS üzerinde desteklenir.

### **TriggerControl (MQUZE)**

Bu, tetikleme iletilerinin, kuyruğa hizmet vermek üzere bir uygulama başlatmak için bir başlatma kuyruğuna yazılıp yazılmayacağını denetler.

Çizelge 623. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Bu, aşağıdakilerden biridir:

#### **MQTC\_OFF**

Bu kuyruk için herhangi bir tetikleme iletisi yazılmasına izin verilmeyecek. *TriggerType* değeri, bu durumda ilgisiz bir değerdir.

#### **MQTC\_ON**

Tetikleme iletileri, uygun tetikleyici olayları oluştuğunda bu kuyruk için yazılmalıdır.

Bu özniteliğin değerini saptamak için MQINQ çağrısıyla MQIA\_TRIGGER\_CONTROL seçiciyi kullanın. Bu özniteliğin değerini değiştirmek için MQSET çağrısını kullanın.

#### **TriggerData (MQCHAR64)**

Bu, kuyruk yöneticisinin, bu kuyruğa gelen bir ileti bir tetikleme iletisinin başlatma kuyruğuna yazılmasına neden olduğunda, tetikleme iletisine eklediği serbest biçimli verilerdir.

Çizelge 624. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Bu verilerin içeriği kuyruk yöneticisi için bir önem göstermiyor. Bu, başlatma kuyruğunu işleyen tetikleyici izleme uygulamasına ya da tetikleme izleyicinin başlattığı uygulamaya anlamlı bir şekilde neden olur.

Karakter dizisi boş boş değer içermemelidir. Gerekirse boşluklarla doldurulabilir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQCA\_TRIGGER\_DATA seçiciyi kullanın. Bu özniteliğin değerini değiştirmek için MQSET çağrısını kullanın. Bu özniteliğin uzunluğu MQ\_TRIGGER\_DATA\_LENGTH tarafından verilir.

#### **TriggerDepth (MQUZE)**

Çizelge 625. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Bu, bir tetikleme iletisi yazılmadan önce kuyruğun üzerinde olması gereken, öncelikli *TriggerMsgPriority* ya da daha büyük iletilerin sayısıdır. Bu, *TriggerType* değeri MQTT\_DEPTH değerine ayarlandığında geçerlidir. *TriggerDepth* değeri bir ya da daha fazlasıdır. Bu öznitelik, başka bir şekilde kullanılmamaktadır.

Bu özniteliğin değerini saptamak için MQINQ çağrısına sahip MQIA\_TRIGGER\_DEPTH seçiciyi kullanın. Bu özniteliğin değerini değiştirmek için MQSET çağrısını kullanın.

#### **TriggerMsgÖnceliği (MQUZE)**

Bu, iletilerin tetikleme iletilerinin oluşturulmasına katkıda bulunmadığı (yani, bir tetikleme iletisi oluşturulup oluşturulmamaya karar verirken kuyruk yöneticisi bu iletileri yoksayar) ileti önceliğidir.

Çizelge 626. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			



*TriggerMsgPriority* , sıfır (en düşük) ile *MaxPriority* (en düşük) aralığında olabilir (highest; bkz. *MaxPriority* özneliği ); sıfır değeri, tüm iletilerin, tetikleyici iletileri oluşturulmasına katkıda bulunmasına neden olur.

Bu özneliğin değeri saptamak için MQINQ çağrısıyla MQIA\_TRIGGER\_MSG\_PRIORITY seçiciyi kullanın. Bu özneliğin değeri değiştirmek için MQSET çağrısını kullanın.

### **TriggerType (MQUZE)**

Bu, kuyruğa gelen iletilerin bir sonucu olarak tetikleme iletilerinin yazıldığı koşulları denetler.

Çizelge 627. Bu özneliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Aşağıdaki değerlerden birine sahiptir:

#### **MQTT\_NONE**

Bu kuyruktaki iletilerin sonucu olarak hiçbir tetikleme iletisi yazılmadı. Bu, *TriggerControl* ' un MQTC\_OFF değerine ayarlandığı gibi aynı etkiye sahiptir.

#### **MQTT\_BIRINCI**

Kuyruk üzerindeki önceliğin *TriggerMsgPriority* ya da daha büyük ileti sayısı 0 'dan 1 'e kadar olduğunda, bir tetikleme iletisi yazılır.

#### **MQTT EVERY**

Kuyruğa bir öncelik iletisi *TriggerMsgPriority* ya da daha büyük bir ileti geldiğinde bir tetikleme iletisi yazılır.

#### **MQTT\_DERINLIK**

A trigger message is written whenever the number of messages of priority *TriggerMsgPriority* or greater on the queue equals or exceeds *TriggerDepth*. Tetikleme iletisi yazıldıktan sonra, *TriggerControl* yeniden açık bir şekilde yeniden açılıncaya kadar tetiklemeyi önlemek için MQTC\_OFF olarak ayarlanır.

Bu özneliğin değeri saptamak için, MQINQ çağrısına sahip MQIA\_TRIGGER\_TYPE seçiciyi kullanın. Bu özneliğin değeri değiştirmek için MQSET çağrısını kullanın.

### **Kullanım (MQUZE)**

Bu, kuyruğun ne için kullanıldığını gösterir.

Çizelge 628. Bu özneliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Değer aşağıdakilerden biridir:

#### **MQUS\_NORMAL**

Bu, uygulamaların iletileri yerleştirirken ve alırken kullandığı bir kuyruktur; kuyruk bir iletim kuyruğu değildir.

#### **MQUS\_ILETİMİ**

Bu, uzak kuyruk yöneticilerine gönderilen iletileri tutmak için kullanılan bir kuyruktur. Bir uygulama uzak bir kuyruğa ileti gönderdiğinde, yerel kuyruk yöneticisi iletiyi özel bir biçimde geçici olarak uygun iletim kuyruğunda saklar. Daha sonra bir ileti kanalı aracısında ileti, iletim kuyruğundan okur ve iletiyi uzak kuyruk yöneticisine aktarır. Uzak denetimi yapılandırmaya ilişkin ek bilgi için [Uzak denetim için kuyruk yöneticilerinin yapılandırılması](#) başlıklı konuya bakın.

Yalnızca ayrıcalıklı uygulamalar, iletileri doğrudan doğruya koymak için MQOO\_OUTPUT için bir iletim kuyruğu açabilir. Genellikle, yalnızca yardımcı program uygulamaları bunu yapar. İleti veri biçiminin doğru olduğundan emin olun (bkz. "[MQXQH-İletim kuyruğu üstbilgisi](#)" sayfa 608 ) ya da iletim işlemi

sırasında hata oluşabilir. Bağlam, MQPMO\_\*\_CONTEXT bağlam seçeneklerinden biri belirlenmedikçe geçirilmez ya da ayarlanmaz.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_USAGE seçiciyi kullanın.

### **XmitQName (MQCHAR48)**

Bu, iletim kuyruğu adıdır. Bir uzak kuyruk ya da kuyruk yöneticisi diğer adı tanımlaması için açık bir durum olduğunda bu öznelik boş değilse, iletiyi iletmek için kullanılacak yerel iletim kuyruğunun adını belirler.

Çizelge 629. Bu özneliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
			X	

**XmitQName** boş bırakılırsa, adı **RemoteQMgrName** olan yerel kuyruk iletim kuyruğu olarak kullanılır. **RemoteQMgrName** adlı bir kuyruk yoksa, **DefXmitQName** kuyruk yöneticisi özneliği tarafından tanımlanan kuyruk kullanılır.

Tanımlama, bir kuyruk yöneticisi diğer adı olarak kullanılıyorsa ve **RemoteQMgrName** yerel kuyruk yöneticisinin adı ise bu öznelik dikkate alınmaz. Tanımlama, bir yanıt kuyruğu diğer adı tanımı olarak kullanılırsa da yoksayılr.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQCA\_XMIT\_QNAME seçicisini kullanın. Bu özneliğin uzunluğu MQ\_Q\_NAME\_LENGTH tarafından verilir.

### **Ad listelerine ilişkin öznelikler**

Aşağıdaki çizelge, ad listelerine özgü öznelikleri özetlemektedir. Öznelikler alfabetik sırayla açıklanır.

Ad listeleri tüm IBM MQ sistemlerinde desteklenir, artı IBM MQ MQI clients bu sistemlere bağlı olarak desteklenir.

**Not:** Bu kısımda gösterilen özneliklerin adları, MQINQ ve MQSET çağrılarıyla kullanılan açıklayıcı adlardır; bu adlar, PCF komutlarıyla aynı olur. MQSC komutları, öznelikleri tanımlamak, değiştirmek ya da görüntülemek için kullanıldığında, diğer kısa adlar kullanılır; ek bilgi için [MQSC komutları](#) konusuna bakın.

Çizelge 630. Ad listelerine ilişkin öznelikler	
Öznelik	Tanım
<a href="#">AlterationDate</a>	Tanımlamanın son olarak değiştirildiği tarih
<a href="#">AlterationTime</a>	Tanımlamanın son olarak değiştirildiği saat
<a href="#">NameCount</a>	Ad listesindeki ad sayısı
<a href="#">NamelistDesc</a>	Ad listesi açıklaması
<a href="#">NamelistName</a>	Ad listesi adı
<a href="#">Adlar</a>	<i>NameCount</i> adlarının listesi
<a href="#">NamelistType</a>	Ad listesi tipi
<a href="#">QSGDisp</a>	Kuyruk paylaşımı grubu yok etme

### **AlterationDate (MQCHAR12)**

Bu, tanımın en son değiştirildiği tarihtir. Tarih biçimi şöyledir: YYYY-MM-DD; uzunluğun 12 byte 'ı olması için sondaki iki boşlukla doldurulması.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQCA\_ALTERATION\_DATE seçiciyi kullanın. Bu özneliğin uzunluğu MQ\_DATE\_LENGTH tarafından verilir.

### **AlterationTime (MQCHAR8)**

Bu, tanımın en son değiştirildiği dönemdir. Saatin biçimi şöyledir: HH.MM.SS.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQCA\_ALTERATION\_TIME seçiciyi kullanın. Bu özniteliğin uzunluğu MQ\_TIME\_LENGTH tarafından verilir.

### **NameCount (MQUZE)**

Bu ad, ad listesindeki adların sayısıdır. Sıfır değerinden büyük ya da sıfıra eşit. Aşağıdaki değer tanımlıdır:

#### **MQNC\_MAX\_NAMELIST\_NAME\_COUNT**

Bir ad listesindeki ad sayısı üst sınırı.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_NAME\_COUNT seçiciyi kullanın.

### **NamelistDesc (MQCHAR64)**

Açıklayıcı açıklamalar için bu alanı kullanın; değeri, tanımlama işlemi tarafından oluşturulur. Alanın içeriği kuyruk yöneticisi için önemli değildir; ancak kuyruk yöneticisi, alanın yalnızca görüntülenebilecek karakterleri içermesini gerektirebilir. Boş değerli karakterler içeremez; gerekirse, boşlukla doldurulabilir. Bir DBCS kuruluşunda, bu alan DBCS karakterleri içerebilir (en çok 64 bayt alan uzunluğuna bağlı).

**Not:** Bu alan, kuyruk yöneticisinin karakter kümesinde olmayan karakterler içeriyorsa ( **CodedCharSetId** kuyruk yöneticisi öznitelemesiyle tanımlandığı gibi), bu alan başka bir kuyruk yöneticisine gönderilirse, bu karakterler yanlış çevrilebilir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQCA\_NAMELIST\_DESC seçiciyi kullanın.

Bu özniteliğin uzunluğu MQ\_NAMELIST\_DESC\_LENGTH tarafından verilir.

### **NamelistName (MQCHAR48)**

Bu ad, yerel kuyruk yöneticisinde tanımlı bir ad listesinin adıdır. Ad listesi adlarıyla ilgili daha fazla bilgi için [Diğer nesne adları](#) bölümüne bakın.

Her bir ad listesi, kuyruk yöneticisine ait diğer ad listelerinin adlarından farklı bir ada sahiptir, ancak farklı tiplerdeki diğer kuyruk yöneticisi nesnelere ait adlarını çoğaltabilir (örneğin, kuyruklar).

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQCA\_NAMELIST\_NAME seçiciyi kullanın.

Bu özniteliğin uzunluğu MQ\_NAMELIST\_NAME\_LENGTH tarafından verilir.

### **NamelistType (MQUZE)**

Bu, ad listesindeki adların doğasını belirtir ve ad listesinin nasıl kullanıldığını gösterir. Bu değer aşağıdaki değerlerden biridir:

#### **MQNT\_NONE**

Atanmış tipi olmayan ad listesi.

#### **MQNT\_Q**

Kuyrukların adlarını içeren ad listesi.

#### **MQNT\_CLUSTER**

Kümelerin adlarını içeren ad listesi.

#### **MQNT\_AUTH\_INFO**

Kimlik doğrulama bilgileri nesnelere ait adlarını içeren ad listesi.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_NAMELIST\_TYPE seçiciyi kullanın.



Bu öznitelik yalnızca z/OS üzerinde desteklenir.

### **Adlar (MQCHAR48xNameCount)**

Bu, *NameCount* adlarının bir listesidir; burada her ad, yerel kuyruk yöneticisinde tanımlı olan bir nesnenin adıdır. Nesne adlarıyla ilgili daha fazla bilgi için bkz. [IBM MQ nesnelere ait adlandırılmasına ilişkin kurallar](#).

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQCA\_NAMES seçiciyi kullanın.

Listedeki her adın uzunluğu MQ\_OBJECT\_NAME\_LENGTH tarafından verilir.

## **QSGDisp (MQHOT)**

Bu, ad listesinin yok edilmesini belirtir. Değer aşağıdakilerden biridir:

### **MQQSGD\_Q\_MGR**


Nesnenin kuyruk yöneticisi yok etme değeri vardır: nesne tanımlaması yalnızca yerel kuyruk yöneticisinde bilinir; tanımlama, kuyruk paylaşım grubundaki diğer kuyruk yöneticilerine bilinmez.

Kuyruk paylaşım grubundaki her kuyruk yöneticisinin adı aynı olan ve yürürlükteki nesneyle aynı olan bir nesne olabilir, ancak bunlar ayrı nesnedir ve bunlar arasında ilinti yoktur. Onların özniteliklerinin birbiriyle aynı olması engellenir.

### **MQQSGD\_COPY**

Nesne, paylaşılan havuzda var olan bir ana nesne tanımlamasının yerel bir kopyasıdır. Kuyruk paylaşım grubundaki her kuyruk yöneticisi nesnenin kendi kopyasına sahip olabilir. Başlangıçta, tüm kopyalar aynı özniteliklere sahiptir, ancak MQSC komutlarını kullanarak her bir kopyayı değiştirebilirsiniz, böylece öznitelikleri diğer kopyalardaki özniteliklerden farklı olur. Paylaşılan havuzdaki ana tanım değiştirildiğinde, kopyaların öznitelikleri yeniden uyumlulaştırılır.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_QSG\_DISP seçicisini kullanın.

 Bu öznitelik yalnızca z/OS üzerinde desteklenir.

## **Süreç tanımlamalarına ilişkin öznitelikler**

Aşağıdaki çizelge, süreç tanımlamalarına özgü öznitelikleri özetlemektedir. Öznitelikler alfabetik sırayla açıklanır.

**Not:** Bu kısımdaki özniteliklerin adları, MQINQ ve MQSET çağrılılarıyla kullanılan açıklayıcı adlardır; bu adlar PCF komutlarıyla aynıdır. MQSC komutları, öznitelikleri tanımlamak, değiştirmek ya da görüntülemek için kullanıldığında, diğer kısa adlar kullanılır; ek bilgi için [MQSC komutları](#) konusuna bakın.

Çizelge 631. Süreç tanımlamalarına ilişkin öznitelikler	
Öznitelik	Tanım
<a href="#">AlterationDate</a>	Tanımlamanın son olarak değiştirildiği tarih
<a href="#">AlterationTime</a>	Tanımlamanın son olarak değiştirildiği saat
<a href="#">AppId</a>	Uygulama tanıtıcısı
<a href="#">AppType</a>	Uygulama tipi
<a href="#">EnvData</a>	Ortam verileri
<a href="#">ProcessDesc</a>	Süreç Tanımı
<a href="#">ProcessName</a>	İşlem adı
<a href="#">QSGDisp</a>	Kuyruk paylaşımı grubu yok etme
<a href="#">UserData</a>	Kullanıcı verileri

### **AlterationDate (MQCHAR12)**

Bu, tanımın en son değiştirildiği tarihtir. Tarih biçimi şöyledir: YYYY-MM-DD; uzunluğun 12 byte 'ı olması için sondaki iki boşlukla doldurulması.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQCA\_ALTERATION\_DATE seçiciyi kullanın. Bu özniteliğin uzunluğu MQ\_DATE\_LENGTH tarafından verilir.

### **AlterationTime (MQCHAR8)**

Bu, tanımın en son değiştirildiği dönemdir. Saatin biçimi şöyledir: HH.MM.SS.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQCA\_ALTERATION\_TIME seçiciyi kullanın. Bu özniteliğin uzunluğu MQ\_TIME\_LENGTH tarafından verilir.

## **AppId (MQCHAR256)**

Bu, başlatılacak uygulamayı tanımlayan bir karakter dizilimidir. Bu bilgiler, başlatma kuyruğunda iletileri işleyen bir tetikleme izleme programı uygulaması tarafından kullanılır; bilgiler, tetikleme iletilerinin bir parçası olarak başlatıcı kuyruğuna gönderilir.

The meaning of *AppId* is determined by the trigger-monitor application. The trigger monitor provided by IBM MQ requires *AppId* to be the name of an executable program. Aşağıdaki notlarda belirtilen ortamlar için geçerlidir:

- z/OS üzerinde, *AppId* şu olmalıdır:
  - CICS tetikleyicisi kullanılarak başlatılan uygulamalar için bir CICS işlem tanıtıcısı. CKTI
  - IMS tetikleyici izleyicisi CSQQTRMN kullanılarak başlatılan uygulamalar için bir IMS işlem tanıtıcısı
- Windows' ta, program adının başına bir sürücü ve izin yolu eklenerek önekli bir ad olabilir.
- AIX and Linux üzerinde, program adının başına bir izin yolu önek olarak eklenebilir.

Karakter dizisi boş boş (nulls) içeremez. Gerekirse boşluklarla doldurulabilir.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQCA\_APPL\_ID seçiciyi kullanın. Bu özneliğin uzunluğu, MQ\_PROCESS\_APPL\_ID\_LENGTH tarafından verilir.

## **AppType (MQLONG)**

Bu, bir tetikleyici iletilerinin alınmasına yanıt olarak başlatılacak programın niteliğini tanımlar. Bu bilgiler, başlatma kuyruğundaki iletileri işleyen bir tetikleyici uygulaması tarafından kullanılır; bilgiler, tetikleme iletilerinin bir parçası olarak başlatma kuyruğuna gönderilir.

*AppType* ' in herhangi bir değeri olabilir, ancak standart tipler için şu değerler önerilir; kullanıcı tanımlı uygulama tiplerini MQAT\_USER\_LAST ile MQAT\_USER\_FIRST aralığındaki değerlerle sınırlandırın:

### **MQAT\_AIX**

AIX uygulaması (MQAT\_UNIX ile aynı değer).

### **MQAT\_BATCH**

Toplu iş uygulaması

### **MQAT\_CICS**

CICS hareketi.

### **MQAT\_IMS**

IMS uygulaması.

### **MQAT\_IMS\_BRIDGE**

IMS köprü uygulaması.

### **MQAT\_JAVA**

Java uygulaması.

### **MQAT\_MVS**

MVS ya da TSO uygulaması (MQAT\_ZOS ile aynı değer).

### **MQAT\_OS390**

OS/390 uygulaması (MQAT\_ZOS ile aynı değer).

### **MQAT\_OS400**

IBM i uygulaması.

### **MQAT\_UNIX**

UNIX uygulaması.

### **MQAT\_BILINMIYOR**

Bilinmeyen tipte uygulama.

### **MQAT\_KULLANICISI**

Kullanıcı uygulaması.

**MQAT\_WINDOWS**

64 bit Windows uygulaması.

**MQAT\_WINDOWS\_NT**

32 bit Windows uygulaması.

**MQAT\_WLM**

z/OS iş yükü yöneticisi uygulaması.

**MQAT\_ZOS**

z/OS uygulaması.

**MQAT\_USER\_FIRST**

Kullanıcı tanımlı uygulama tipi için en düşük değer.

**MQAT\_USER\_LAST**

Kullanıcı tanımlı uygulama tipi için en yüksek değer.

Bu özniteliğin değerini saptamak için MQINQ çağrısıyla MQIA\_APPL\_TYPE seçicisini kullanın.

**EnvData (MQCHAR128)**

Bu dizgi, başlatılacak uygulamaya ilişkin ortamla ilgili bilgileri içeren bir karakter dizilimidir. Bu bilgiler, başlatma kuyruğunda iletileri işleyen bir tetikleme izleme programı uygulaması tarafından kullanılır; bilgiler, tetikleme iletilerinin bir parçası olarak başlatıcı kuyruğuna gönderilir.

The meaning of *EnvData* is determined by the trigger-monitor application. The trigger monitor provided by IBM MQ appends *EnvData* to the parameter list passed to the started application. Parametre listesi, MQTMC2 yapısından oluşur, ardından bir boşluk, ardından *EnvData* sonunda boşluklar kaldırılmış olur. Aşağıdaki notlarda belirtilen ortamlar için geçerlidir:

- z/OS'ta:
  - *EnvData* , IBM MQ tarafından sağlanan tetikleme izleme uygulamaları tarafından kullanılmaz.
  - ApplType , MQAT\_WLM ise, iş bilgileri üstbilgisindeki (MQWIH) ServiceName ve ServiceStep alanları için EnvData ' da varsayılan değerler sağlayabilirsiniz.
- AIX and Linux üzerinde, başlangıç uygulamasını arka planda çalıştırmak için *EnvData* , & karakterine ayarlanabilir.

Karakter dizgisi boş boş (nulls) içeremez. Gerekirse boşluklarla doldurulabilir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQCA\_ENV\_DATA seçiciyi kullanın. Bu özniteliğin uzunluğu MQ\_PROCESS\_ENV\_DATA\_LENGTH tarafından verilir.

**ProcessDesc (MQCHAR64)**

Açıklayıcı açıklamalar için bu alanı kullanın. Alanın içeriği kuyruk yöneticisi için önemli değildir; ancak kuyruk yöneticisi, alanın yalnızca görüntülenebilecek karakterleri içermesini gerektirebilir. Boş değerli karakterler içeremez; gerekirse, boşlukla doldurulabilir. Bir DBCS kuruluşunda, alan DBCS karakterleri içerebilir (en çok 64 baytlık alan uzunluğuna tabi).

**Not:** Bu alan, kuyruk yöneticisinin karakter kümesinde olmayan karakterler içeriyorsa ( **CodedCharSetId** kuyruk yöneticisi öznitelemesiyle tanımlandığı gibi), bu alan başka bir kuyruk yöneticisine gönderilirse, bu karakterler yanlış çevrilebilir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQCA\_PROCESS\_DESC seçiciyi kullanın.

Bu özniteliğin uzunluğu MQ\_PROCESS\_DESC\_LENGTH tarafından verilir.

**ProcessName (MQCHAR48)**

Yerel kuyruk yöneticisinde tanımlı olan bir süreç tanımlamasının adı.

Her süreç tanımlaması, kuyruk yöneticisine ait diğer süreç tanımlamalarının adlarından farklı bir ada sahiptir. Ancak süreç tanımlamasının adı, farklı tiplerde bulunan diğer kuyruk yöneticisi nesnelerinin adları (örneğin, kuyruklar) ile aynı olabilir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQCA\_PROCESS\_NAME seçiciyi kullanın.

Bu özniteliğin uzunluğu MQ\_PROCESS\_NAME\_LENGTH tarafından verilir.

### **QSGDisp (MQHOT)**

Bu, süreç tanımlamasının yok etmeyi belirtir. Değer aşağıdakilerden biridir:

#### **MQQSGD\_Q\_MGR**


Nesnenin kuyruk yöneticisi yok etme değeri vardır: nesne tanımlaması yalnızca yerel kuyruk yöneticisinde bilinir; tanımlama, kuyruk paylaşım grubundaki diğer kuyruk yöneticilerine bilinmez.

Kuyruk paylaşım grubundaki her kuyruk yöneticisinin adı aynı olan ve yürürlükteki nesneyle aynı olan bir nesne olabilir, ancak bunlar ayrı nesnedir ve bunlar arasında ilinti yoktur. Onların özniteliklerinin birbiriyle aynı olması engellenir.

#### **MQQSGD\_COPY**

Nesne, paylaşılan havuzda var olan bir ana nesne tanımlamasının yerel bir kopyasıdır. Kuyruk paylaşım grubundaki her kuyruk yöneticisi nesnenin kendi kopyasına sahip olabilir. Başlangıçta, tüm kopyalar aynı özniteliklere sahiptir, ancak MQSC komutlarını kullanarak her bir kopyayı değiştirebilirsiniz, böylece öznitelikleri diğer kopyalardaki özniteliklerden farklı olur. Paylaşılan havuzdaki ana tanım değiştirildiğinde, kopyaların öznitelikleri yeniden uyumlulaştırılır.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA\_QSG\_DISP seçicisini kullanın.

 Bu öznitelik yalnızca z/OS üzerinde desteklenir.

### **UserData (MQCHAR128)**

UserData , başlatılacak uygulamayla ilgili kullanıcı bilgilerini içeren bir karakter dizimidir. Bu bilgiler, başlatma kuyruğunda iletileri işleyen bir tetikleme izleme uygulaması ya da tetikleme izleme programı tarafından başlatılan uygulama tarafından kullanılmak içindir. Bilgiler, tetikleme iletilerinin bir parçası olarak başlatıcı kuyruğuna gönderilir.

The meaning of *UserData* is determined by the trigger-monitor application. The trigger monitor provided by IBM MQ passes *UserData* to the started application as part of the parameter list. Parametre listesi, MQTMC2 yapısından ( *UserData* içeren) oluşur, ardından bir boşluk, ardından *EnvData* sonunda boşluklar kaldırılmış olarak bulunur.

Karakter dizisi boş boş (nulls) içeremez. Gerekirse boşluklarla doldurulabilir. For Microsoft Windows, the character string must not contain double quotation marks if the process definition is going to be passed to **runmqtrm**.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQCA\_USER\_DATA seçiciyi kullanın. Bu özniteliğin uzunluğu MQ\_PROCESS\_USER\_DATA\_LENGTH tarafından verilir.

## **Dönüş kodları**

Her bir IBM MQ Message Queue Interface (MQI) ve IBM MQ Administration Interface (MQAI) çağrısı için, bir **completion** kodu ve bir **reason** kodu, arama işleminin başarılı olduğunu ya da başarısız olduğunu göstermek için kuyruk yöneticisi ya da bir çıkış yordamı tarafından döndürülür.

Uygulamaların, özel olarak belirtildiği durumlar dışında, belirli bir sırada denetlenmekte olan hatalara bağlı olmamaları gerekir. Bir çağrıdan birden fazla tamamlanma kodu ya da neden kodu ortaya çıkarsa, bildirilen belirli bir hata uygulamaya bağlıdır.

Bir IBM MQ API çağrısının ardından başarıyla tamamlanma olup olmadığını denetleyen uygulamalar, her zaman tamamlama kodunu denetlememelidir. Neden kodunun değerine dayalı olarak, tamamlanma kodu değerini varsaymayın.

## Tamamlanma kodları

Tamamlanma kodu parametresi (*CompCode*), çağırının, çağrılarını başarıyla tamamlanıp tamamlanmadığını, kısmen mi, yoksa başarısız mı olduğunu hızlı bir şekilde görmelerini sağlar. Aşağıda, arama tanımlarında yer alan daha fazla ayrıntı içeren bir tamamlama kodlarının listesi verilmiştir:

### MQCC\_OK

Çağrı tam olarak tamamlandı; tüm çıkış parametreleri ayarlandı. **Reason** parametresi her zaman bu durumda MQRC\_NONE değerine sahip olur.

### MQCC\_UYARI

Arama kısmen tamamlandı. *CompCode* ve *Reason* çıkış parametrelerinin yanı sıra bazı çıkış parametreleri de ayarlanmış olabilir. **Reason** parametresi, kısmi tamamlama hakkında ek bilgi verir.

### MQCC\_FAILED

Çağrıya ilişkin işlem tamamlanmadı. Kuyruk yöneticisinin durumu, özellikle belirtilenler dışında değişmeden kalır. *CompCode* ve *Reason* çıkış parametreleri ayarlandı; burada belirtilenler dışında, diğer parametreler değiştirilmez.

Neden, uygulama programında bir hata olabilir ya da programın dışındaki bazı durumların sonucu olabilir; örneğin, kullanıcının yetkisi iptal edilmiş olabilir. **Reason** parametresi hatayla ilgili ek bilgi verir.

## Neden kodları

Neden kodu parametresi (*Reason*), tamamlanma kodu parametresini niteliyor (*CompCode*).

Raporlamak için özel bir neden yoksa, MQRC\_NONE döndürülür. Başarılı bir çağrı, MQCC\_OK ve MQRC\_NONE değerini döndürür.

Tamamlanma kodu MQCC\_UYARY ya da MQCC\_FAILED ise, kuyruk yöneticisi her zaman uygun bir neden bildirir; ayrıntılar her çağrı açıklaması altında verilir.

Kullanıcı çıkış yordamlarının tamamlanma kodlarını ve nedenlerini belirledikleri durumlarda, bu kuralların bu kurallara uyması gerekir. Ayrıca, kullanıcı çıkışlarının tanımladığı özel neden değerlerinin, kuyruk yöneticisi tarafından tanımlanan değerlerle çakışmadığından emin olmak için sıfırdan küçük olması gerekir. Çıkışlar, uygun olduğu yerlerde kuyruk yöneticisi tarafından önceden tanımlanmış nedenleri ayarlayabilir.

Neden kodları da aşağıdaki yerde oluşur:

- MQDLH yapısının *Reason* alanı
- MQMD yapısındaki *Feedback* alanı

Neden kodlarına ilişkin eksiksiz açıklamalar için [İletiler ve neden kodları](#) başlıklı konuya bakın.

## MQI seçeneklerinin geçerliliğini denetleme kuralları

Bu kısımda, bir MQPUT, MQPUT, MQPUT1, MQGET, MQCLOSE ya da MQSUB çağrısından bir MQRC\_OPTIONS\_ERROR neden kodu üreten durumlar listelenir.

### MQOPEN çağrısı

MQOPER çağrısına ilişkin seçenekler için:

- Aşağıdakilerden en az *biri* belirtilmeli:
  - MQOO\_BROWSE
  - MQOO\_INPUT\_EXCLUSIVE <sup>1</sup>
  - MQOO\_INPUT\_SHARED <sup>1</sup>
  - MQOO\_INPUT\_AS\_Q\_DEF <sup>1</sup>
  - MQOO\_SORGULAMA
  - MQOO\_OUTPUT



- MQOO\_SET
- MQOO\_BIND\_ON\_AçIK 2
- MQOO\_BIND\_NOT\_FIXED 2
- MQOO\_BIND\_ON\_GROUP 2
- MQOO\_BIND\_AS\_Q\_DEF 2
- Aşağıdakilerin yalnızca *bir* ' e izin verilir:
  - MQOO\_READ\_AHEAD
  - MQOO\_NO\_READ\_AHEAD
  - MQOO\_READ\_AHEAD\_AS\_Q\_DEF

1. Aşağıdakilerin yalnızca *bir* ' e izin verilir:

- MQOO\_INPUT\_EXCLUSIVE
- MQOO\_INPUT\_SHARED
- MQOO\_INPUT\_AS\_Q\_DEF

2. Aşağıdakilerin yalnızca *bir* ' e izin verilir:

- MQOO\_BIND\_ON\_AçIK
- MQOO\_BIND\_NOT\_FIXED
- MQOO\_BIND\_ON\_GROUP
- MQOO\_BIND\_AS\_Q\_DEF

**Not:** Daha önce listelenmiş seçenekler karşılıklı olarak birbirini dışlar. Ancak, MQOO\_BIND\_AS\_Q\_DEF değeri sıfır olduğu için, diğer iki bağlama seçeneğinden biriyle birlikte belirtildiğinde, MQRC\_OPTIONS\_ERROR neden kodunda sonuç verilmez. Program belgelerine yardımcı olmak için MQOO\_BIND\_AS\_Q\_DEF sağlanıyor.

- MQOO\_SAVE\_ALL\_CONTEXT belirtilirse, MQOO\_INPUT\_ \* seçeneğinden biri de belirtilmelidir.
- MQOO\_SET\_ \* \_CONTEXT ya da MQOO\_PASS\_ \* \_CONTEXT seçeneklerinden biri belirtilirse, MQOO\_OUTPUT da belirtilmelidir.
- MQOO\_CO\_OP belirtilirse, MQOO\_BROWSE de belirtilmelidir.
- MQOO\_NO\_MULTICAST belirtilirse, MQOO\_OUTPUT da belirtilmelidir.

## MQPUT çağırısı

Put-message seçenekleri için:

- MQPMO\_SYNCPOINT ve MQPMO\_NO\_SYNCPOINT birleşimine izin verilmiyor.
- Aşağıdakilerin yalnızca *bir* ' e izin verilir:
  - MQPMO\_DEFAULT\_CONTEXT
  - MQPMO\_NO\_BAĞLAMı
  - MQPMO\_PASS\_ALL\_CONTEXT
  - MQPMO\_PASS\_IDENTITY\_CONTEXT
  - MQPMO\_SET\_ALL\_CONTEXT
  - MQPMO\_SET\_IDENTITY\_CONTEXT
- Aşağıdakilerin yalnızca *bir* ' e izin verilir:
  - MQPMO\_ASYNC\_RESPONSE
  - MQPMO\_SYNC\_RESPONSE
  - MQPMO\_RESPONSE\_AS\_TOPIC\_DEF
  - MQPMO\_RESPONSE\_AS\_Q\_DEF

- MQPMO\_ALTERNATE\_USER\_AUTHORITY ögesine izin verilmiyor (yalnızca MQPUT1 çağrısında geçerli).

## **MQPUT1 çağrısı**

Put-message seçenekleri için, aşağıdaki durumlar dışında, kural MQPUT çağrısına ilişkin kurallarla aynıdır:

- MQPMO\_ALTERNATE\_USER\_AUTHORITY ögesine izin verilir.
- MQPMO\_LOGICAL\_ORDER 'a izin verilmiyor.

## **MQGET çağrısı**

Get-message seçenekleri için:

- Aşağıdakilerin yalnızca *bir* ' e izin verilir:
  - MQGMO\_NO\_SYNCPOINT
  - MQGMO\_SYNCPOINT
  - MQGMO\_SYNCPOINT\_IF\_PERSISTENT
- Aşağıdakilerin yalnızca *bir* ' e izin verilir:
  - MQGMO\_BROWSE\_FIRST
  - MQGMO\_BROWSE\_MSG\_UNDER\_CURSOR
  - MQGMO\_BROWSE\_NEXT
  - MQGMO\_MSG\_UNDER\_CURSOR
- MQGMO\_SYNCPOINT, aşağıdakilerden herhangi biriyle izin verilmiyor:
  - MQGMO\_BROWSE\_FIRST
  - MQGMO\_BROWSE\_MSG\_UNDER\_CURSOR
  - MQGMO\_BROWSE\_NEXT
  - MQGMO\_LOCK
  - MQGMO\_UNLOCK
- MQGMO\_SYNCPOINT\_IF\_PERSISTENT, aşağıdakilerin hiçbiriyle kullanılamaz:
  - MQGMO\_BROWSE\_FIRST
  - MQGMO\_BROWSE\_MSG\_UNDER\_CURSOR
  - MQGMO\_BROWSE\_NEXT
  - MQGMO\_COMPLE\_MSG
  - MQGMO\_UNLOCK
- MQGMO\_MARK\_SKIP\_BACKUP, MQGMO\_SYNCPOINT belirtilmesini gerektirir.
- MQGMO\_WAIT ve MQGMO\_SET\_SIGNAL birleşimine izin verilmiyor.
- MQGMO\_LOCK belirtilirse, aşağıdakilerden birinin de belirtilmesi gerekir:
  - MQGMO\_BROWSE\_FIRST
  - MQGMO\_BROWSE\_MSG\_UNDER\_CURSOR
  - MQGMO\_BROWSE\_NEXT
- MQGMO\_UNLOCK belirtilirse, yalnızca aşağıdaki değerlere izin verilir:
  - MQGMO\_NO\_SYNCPOINT
  - MQGMO\_NO\_BEKLEME

## **MQCLOSE çağrısı**

MQCLOSE çağrısına ilişkin seçenekler için:

- MQCO\_DELETE ve MQCO\_DELETE\_PURGE birleşimine izin verilmiyor.
- Aşağıdakilerin yalnızca birine izin verilir:
  - MQCO\_KEEP\_SUB
  - MQCO\_REMOVE\_SUB

## MQSUB çağırısı

MQSUB çağırısına ilişkin seçenekler için:

- Aşağıdakilerden en az birinin belirtilmesi gerekir:
  - MQSO ALTER
  - MQSO RESUME
  - MQSO CREATE
- Aşağıdakilerin yalnızca birine izin verilir:
  - MQSO DAYANIKLI
  - MQSO\_NON\_DAYANIKLI

**Not:** Daha önce listelenmiş seçenekler karşılıklı olarak birbirini dışlar. Ancak, MQSO\_NON\_DAYANIKLI değeri sıfır olduğu için, bu değeri MQSO\_DAYANIKLI ile belirtme, MQRC\_OPTIONS\_ERROR neden kodunda sonuç vermiyor. MQSO\_NON\_DAYANIKLI, program belgelerine yardımcı olmak için sağlanmıştır.

- MQSO\_GROUP\_SUB ve MQSO\_MANAGED birleşimine izin verilmiyor.
- MQSO\_GROUP\_SUB için MQSO\_SET\_COREL\_ID belirtilmeli.
- Aşağıdakilerin yalnızca birine izin verilir:
  - MQSO\_ANY\_USERID
  - MQSO\_FIXED\_USERID
- MQSO\_NEW\_PUBLICATIONS\_ONLY ile birlikte şu şekilde izin verilir:
  - MQSO\_CREATE
  - MQSO ALTER, MQSO\_NEW\_PUBLICATIONS\_ONLY ilk abonelikte ayarlandıysa
- MQSO\_PUBLICATIONS\_ON\_REQUEST ve 1 'den büyük SubLevel birleşimine izin verilmez.
- Aşağıdakilerin yalnızca birine izin verilir:
  - MQSO\_WILDCARD\_CHAR
  - MQSO\_WILDCARD\_KONUSU
- MQSO\_NO\_MULTICAST için MQSO\_MANAGED belirtilmeli.

## Kuyruğa alınan yayınlama/abone olma komutu iletileri

Bir uygulama, kuyruğa alınan bir yayınlama/abone olma uygulamasını denetlemek için MQRFH2 komut iletilerini kullanabilir.

Yayınlama/abone olma için MQRFH2 komutunu kullanan bir uygulama, SYSTEM.BROKER.CONTROL.QUEUE:

- [“Yayın iletisini sil” sayfa 864](#)
- [“Abone İletisi Deregister İletisi” sayfa 865](#)
- [“İletiyi yayınla” sayfa 869](#)
- [“Abone iletisini kaydet” sayfa 871](#)
- [“Güncelleme İletisi İste” sayfa 875](#)

Kuyruğa alınmış yayınlama/abone olma uygulamaları yazıyorsanız, bu iletileri, kuyruk yöneticisi yanıt iletilerini ve ileti tanımlayıcısını (MQMD) anlamamız gerekir; aşağıdaki bilgilere bakın:

- [“Kuyruk Yöneticisi Yanıt iletileri” sayfa 877](#)
- [“Kuyruk yöneticisi tarafından iletilen yayınlara ilişkin MQMD ayarları” sayfa 883](#)
- [“Kuyruk yöneticisi yanıt iletilerinde MQMD ayarları” sayfa 883](#)
- [“Yayınlama/abone olma neden kodları” sayfa 879](#)

Komutlar, MQRFH2 üstbilgisinin **NameValueData** alanındaki bir psc klasöründe yer alır. Bir komut iletilerine yanıt olarak bir aracı tarafından gönderilebilecek ileti bir psc1 klasöründe yer alır.

Her komutun açıklamaları, bir klasörde bulunabilecek özellikleri listelemektedir. Ters belirtilmediği sürece, özellikler isteğe bağlıdır ve yalnızca bir kez oluşabilir.

Özelliklerin adları <Command>olarak gösterilir.

Değerler dizgi biçiminde olmalıdır; örneğin: Publish.

Bir özelliğin değerini temsil eden bir dizgi değişmezi parantez içinde gösterilir; örneğin: (MQPSC\_PUBLISH).

Dizgi değişmezleri, kuyruk yöneticisiyle birlikte sağlanan cmqpsc . h üstbilgi dosyasında tanımlanır.

## Yayın iletilerini sil

Kuyruk yöneticisine, belirtilen konulara ilişkin alıkonan yayınları silmesini sağlamak için, **Delete Publication** komut iletileri bir yayıncıdan kuyruk yöneticisine ya da başka bir kuyruk yöneticisinden bir kuyruk yöneticisine gönderilir.

Bu ileti, kuyruk yöneticisinin kuyruğa alınan yayınlama/abone olma arabirimi tarafından izlenen bir kuyruğa gönderilir.

Giriş kuyruğunun, özgün yayının gönderildiği kuyruk olması gerekir.

**Delete Publication** komut iletilerinde belirtilen konuların tümünü değil, ancak tümü için yetkiniz varsa, yalnızca bu konular silinir. Bir **Broker Response** iletileri hangi konuların silinmediğini belirtir.

Similarly, if a **Publish** command contains more than one topic, a **Delete Publication** command matching some, but not all, of those topics deletes only the publications for the topics that are specified in the **Delete Publication** command.

Kuyruk yöneticisine bir komut iletileri gönderirken gerekli olan ileti tanımlayıcı (MQMD) parametrelerinin ayrıntıları için [“Kuyruk yöneticisi tarafından iletilen yayınlara ilişkin MQMD ayarları” sayfa 883](#) başlıklı konuya bakın.

## Özellikler

### Komut (MQPSC\_COMMAND)

Değer: DeletePub (MQPSC\_DELETE\_YAYINI).

Bu özellik belirtilmeli.

### Konu > (MQPSC\_KONUSU)

Değer, alıkonan yayınların silineceği bir konu içeren bir dizgidir. Birden çok konuyla ilgili yayınları silmek için, dizgide genel arama karakterleri kullanılabilir.

Bu özellik belirtilmeli; gerektiği kadar çok konu için yinelenmelidir.

### DelOpt (MQPSC\_DELETE\_OPTION)

Seçenek silme özelliği aşağıdaki değerlerden birini alabilir:

### Yerel (MQPSC\_LOCAL)

Belirtilen konulara ilişkin alıkonan tüm yayınlar yerel kuyruk yöneticisinde (yani, bu iletilerin gönderildiği kuyruk yöneticisi) silinir; bunlar Yerel seçeneğiyle yayınlanıp yayınlanmayacaklarını belirtmez.

Diğer kuyruk yöneticilerindeki yayınlar etkilenmez.

### **Yok (MQPSC\_NONE)**

Tüm seçenekler varsayılan değerlerini alır. Bu, De10pt özelliğinin atlanması ile aynı etkiye sahiptir. Diğer seçenekler aynı zamanda belirtilirse, Yok yoksayılır.

Bu özellik atlanırsa, varsayılan değer olarak, belirtilen konulara ilişkin alıkonan tüm yayınların, Yerel seçeneğiyle yayınlanıp yayınlanmamasından bağımsız olarak ağdaki tüm kuyruk yöneticilerinde silineceği yer alıdır.

### **Örnek**

Burada, bir **Delete Publication** komut iletisi için bir NameValueData örneği yer alıyor. This is used by the sample application to delete, at the local queue manager, the retained publication that contains the latest score in the match between Team1 and Team2.

```
<psc>
  <Command>DeletePub</Command>
  <Topic>Sport/Soccer/State/LatestScore/Team1 Team2</Topic>
  <De10pt>Local</De10pt>
</psc>
```

## **Abone İletisi Deregister İletisi**

**Deregister Subscriber** komut iletisi, bir abone tarafından ya da bir abone adına başka bir uygulama tarafından, verili deęiřtirgelerle eşleşen iletileri almak istemediğini belirtmek için bir kuyruk yöneticisine gönderilir.

Bu ileti SYSTEM.BROKER.CONTROL.QUEUE(kuyruk), kuyruk yöneticisinin denetim kuyruęu. Kullanıcının bu kuyruęa bir ileti koymak için gereken yetkiye sahip olması gerekir.

Kuyruk yöneticisine bir komut iletisi gönderirken gerekli olan ileti tanımlayıcı (MQMD) parametrelerinin ayrıntıları için [Kuyruk yöneticisi tarafından iletilen yayınlara ilişkin MQMD ayarları](#) başlıklı konuya bakın.

Özgün aboneliğın ilgili konusu, abonelik noktası ve süzgeç deęerleri belirtilerek, bireysel abonelikte kayıt dışı olabilir. Deęerlerden herhangi biri belirtilmediyse (varsayılan deęerleri almışlardı), özgün abonelikte, abonelik kayıttan kaldırıldığında atlanmalırlar.

Bir aboneye ya da bir aboneye ilişkin tüm abonelikler, DeregAll seçeneği kullanılarak kayıttan kaldırılabilir. Örneğın, bir abonelik noktası (ancak konu ya da süzgeç yok) ile birlikte DeregAll belirtilirse, konu ve süzgecin ne olursa olsun, belirtilen abonelik noktasındaki aboneye ilişkin tüm abonelikler kayıttan kaldırılır. Herhangi bir konu, süzgeç ve abonelik noktasına izin verilir; üçü yalnızca bir abonelik eşleşiyorsa ve DeregAll seçeneği yoksayılır.

İleti, aboneliği kaydetmiş olan abone tarafından gönderilmelidir; bu, abonenin kullanıcı kimliği denetlenerek onaylanır.

Abonelikler, MQSC ya da PCF komutlarını kullanan bir sistem denetimcisi tarafından da kayıttan kaldırılabilir. Ancak, geçici bir dinamik kuyrukla kayıtlı abonelikler yalnızca kuyruk adı deęil, kuyrukla ilişkilendirilir. Kuyruk, belirtik olarak ya da kuyruk yöneticisinden gelen uygulama baęlantısını keserek silinirse, artık o kuyruęa ilişkin abonelikleri kayıttan kaldırmak için **Deregister Subscriber** komutunu kullanmanız artık mümkün deęildir. Abonelikler, geliřtirici çalıřma ortamı kullanılarak kayıttan kaldırılabilir ve kuyruk yöneticisi tarafından bir sonraki yayınlı ya da kuyruk yöneticisi yeniden bařlatıldığında otomatik olarak kuyruk yöneticisi tarafından kaldırılabilir. Normal kořullar altında, uygulamaların kuyruęu silmeden önce aboneliklerini silmesi ya da kuyruk yöneticisinden baęlantıyı kesmeleri gerekir.

Bir abone, bir aboneliği kayıttan kaldırmak için bir ileti gönderirse ve bu iletinin bařarıyla işlendiğini belirten bir yanıt iletisi alırsa, abonelik kayıttan kaldırıldığını için aynı zamanda kuyruk yöneticisi tarafından işlenmekte olan bazı yayınlar abone kuyruęuna erişmeye devam edebilir. İletiler kuyruktan kaldırılmamışsa, abone kuyruęunda işlenmemiş iletiler birikmesi olabilir. Uygulama, bir süre uyuduktan sonra uygun CorrelId ile bir MQGET çağrısını içeren bir döngü yürütürse, bu iletiler kuyruktan temizlenir.

Benzer bir şekilde, abone kalıcı bir dinamik kuyruk kullanıyorsa ve bir MQCLOSE çağrısında *MQCO\_DELETE\_PURGE* seçeneğiyle kuyruğu kapasa, kuyruk boş olmayabilir. Kuyruk yöneticisinden gelen herhangi bir yayın, kuyruk silindiğinde henüz kesinleştirilmediyse, MQCLOSE çağrısı tarafından bir MQRC\_Q\_NOT\_EMPTY dönüş kodu verilir. Uygulama, bu sorunu uyutarak ve zaman zaman MQCLOSE çağrısını yeniden yayınlayarak önleyebilir.

## Özellikler

### Komut (MQPSC\_COMMAND)

Değer: DeregSub (MQPSC\_DEREGISTER\_SUBSCRIBER).

Bu özellik belirtilmeli.

### Konu (MQPSC\_KONUSU)

Değer, kaydı kayıttan kaldırılacak konuyu içeren bir dizgidir.

Bu özellik, isteğe bağlı olarak birden çok konu kaydı kaldırılacaksa yinelenebilir. It can be omitted if DeregAll is specified in <RegOpt>.

Belirtilen konular, abonenin diğer konulara ilişkin abonelikleri korumak istemesi durumunda kaydolanların bir alt kümesi olabilir. Genel arama karakterlerine izin verilir, ancak genel arama karakterleri içeren bir konu dizgisinin, **Deregister Subscriber** komut iletilisinde belirtilen karşılık gelen dizgiyle tam olarak eşleşmesi gerekir.

### SubPoint (MQPSC\_SUBSCRIPTION\_POINT)

Değer, aboneliğin ayrılacağı abonelik noktasını belirten bir dizgidir.

Bu özellik yinelenmemelidir. Bir < Topic> belirtilirse ya da <RegOpt> içinde DeregAll belirtilirse bu atlanabilir. Bu özelliği çıkarırsanız, aşağıdakiler gerçekleşir:

- If you do **değil** specify DeregAll, subscriptions matching the < Topic> property (and the < Süzgeç > property, if present) are deregistered from the default subscription point.
- DeregAll seçeneğini belirlerseniz, tüm abonelikler (varsa < Topic> ve < Filter > özellikleriyle eşleştirilir), tüm abonelik noktalarından kaydı kaldırılır.

Varsayılan abonelik noktasını açık bir şekilde belirleyemeyeniz için dikkat edin. Bu nedenle, yalnızca bu abonelik noktasından tüm abonelikleri kayıttan kaldırma yöntemi yoktur; konuları belirtmeniz gerekir.

### SubIdentity (MQPSC\_SUBSCRIPTION\_IDENTITY)

Bu, en çok 64 karakter uzunluğunda bir değişken uzunluklu bir dizilimdir. Bir uygulamayı, bir abonelik ile ilgilenecek şekilde temsil etmek için kullanılır. Kuyruk yöneticisi, her abonelik için bir abone kimlikleri kümesini tutar. Her abonelik, kimliğinin yalnızca tek bir kimliği ya da sınırsız sayıda kimliğe sahip olması için ayarlanmasına izin verebilir.

SubIdentity , abonelik için kimlik ayarında yer aldıysa, bu değer kümeden kaldırılır. Kimlik kümesi bunun sonucu olarak boş olursa, LeaveOnly değeri RegOpt özelliğinde bir değer olarak belirtilmedikçe, abonelik kuyruk yöneticisinden kaldırılır. Kimlik kümesi başka kimlikler içeriyorsa, abonelik kuyruk yöneticisinden kaldırılmaz ve yayın akışı kesintiye uğramaz.

If SubIdentity is specified, but the SubIdentity is not in the identity set for the subscription, then the **Deregister Subscriber** command fails with the return code *MQRC CF\_SUB\_IDENTITY\_ERROR*.

### Süzgeç Uygula (MQPSC\_FILTER)

Değer, kaydı kayıttan geçirilecek süzgeci belirten bir dizgidir. Vaka ve boşluklar da dahil olmak üzere, önceden kaydedilmiş bir abonelik süzgeci ile eşleşmesi gerekir.

Bu özellik, isteğe bağlı olarak birden fazla süzgecin kayıttan kaldırılacaksa yinelenebilir. Bir < Topic> belirtilirse ya da <RegOpt> içinde DeregAll belirtilirse bu atlanabilir.

Belirtilen süzgeçler, abonenin diğer süzgeçler için abonelikleri korumak istemesi durumunda kaydedilenlerin bir alt kümesi olabilir.

### RegOpt (MQPSC\_REGISTRATION\_OPTION)

Kayıt seçenekleri özelliği aşağıdaki değerleri alabilir:

## **DeregAll**

(MQPSC\_DEREGISTER\_ALL)

Bu abone için kayıtlı olan tüm eşleşen abonelikler kayıttan kaldırılacaktır.

DeregAll seçeneğini belirlerseniz:

- < Topic>, <SubPoint> ve < Süzgeç > atlanabilir.
- Gerekirse, < Topic> ve < Filter > yinelenebilir.
- <SubPoint> yinelenmemelidir.

If you do **değil** specify DeregAll:

- < Topic> belirtilmeli ve gerekirse yinelenabilir.
- <SubPoint> ve < Süzgeç > atlanabilir.
- <SubPoint> yinelenmemelidir.
- Gerekliyse, < Filter > yinelenabilir.

Konular ve süzgeçler yinelenirse, ikisinin tüm birleşimleri eşleştirilir ve tüm abonelikler kaldırılır. Örneğin, üç konuyu belirten bir **Deregister Subscriber** komutu ve üç süzgeç, dokuz aboneliği kaldırmayı deneyecektir.

## **CorrelAsTnt**

(MQPSC\_CORREL\_ID\_AS\_IDENTITY)

Aboneyi tanımlamak için, sıfır olmamalıdır, ileti tanımlayıcısındaki (MQMD) CorrelId kullanılır. Özgün abonelikte kullanılan CorrelId ile eşleşmesi gerekir.

## **FullResp**

(MQPSC\_FULL\_RESPONSE)

FullResp seçeneği belirlenirse, aboneliğin tüm öznitelikleri yanıt iletisinde döndürülür; komut başarısız olursa, bu ileti başarısız olur.

FullResp belirtildiğinde, **Deregister Subscriber** komutunda DeregAll kullanılmasına izin verilmez. Birden çok konu belirtmek de olanaklı değildir. Her iki durumda da komut *MQRCCF\_REG\_OPTIONS\_ERROR* dönüş kodu ile başarısız olur.

## **LeaveOnly**

(MQPSC\_LEAVE\_ONLY)

When you specify this with a SubIdentity which is in the identity set for the subscription the SubIdentity is removed from the identity set for the subscription. Sonuçta ortaya çıkan kimlik kümesi boş olsa da, abonelik kuyruk yöneticisinden kaldırılmaz. If the SubIdentity value is not in the identity set the command fails with return code *MQRCCF\_SUB\_IDENTITY\_ERROR*.

LeaveOnly değeri SubIdentityyle belirtilmişse, komut *MQRCCF\_REG\_OPTIONS\_ERROR* dönüş koduyla başarısız olur.

LeaveOnly ya da SubIdentity belirtilmiyorsa, abonelik için ayarlanan kimlik içeriğinden bağımsız olarak abonelik kaldırılır.

## **Yok**

(MQPSC\_NONE)

Tüm seçenekler varsayılan değerlerini alır. Bu, kayıt seçenekleri özelliğinin atlanması ile aynı etkiye sahiptir. Diğer seçenekler aynı zamanda belirtilirse, Yok yoksayılr.

## **VariableUserTanıtıcısı**

(MQPSC\_VARIABLE\_USER\_ID)

Abonenin kimliği (kuyruk, kuyruk yöneticisi ve correlid) belirtildiğinde, tek bir kullanıcı kimliği ile sınırlı değildir. Bu, kuyruk yöneticisinin, özgün kayıt iletisinin kullanıcı kimliğini abonenin kimliğiyle ilişkilendiren var olan davranışından ve bundan sonra o kimliği kullanan diğer kullanıcıların

herhangi bir kullanıcının engellediği bir davranışa göre farklılık gösterir. If a new subscriber tries to use the same identity, the return code *MQRCCF\_DUPLICATE\_SUBSCRIPTION* is returned.

Herhangi bir kullanıcı, uygun yetkiye sahip olduğunda aboneliği değiştirebilir ya da kayıttan kaldırır; kullanıcı kimliğinin özgün abonenin eşleşmesi için var olan denetlemelerden kaçınması gerekir.

Bu seçeneği varolan bir aboneliğe eklemek için, komutun aynı kullanıcı kimliği ile özgün aboneliğin kendisiyle gelmesi gerekir.

Kayıt dışı olacak aboneliğin *VariableUserTnt* kümesi varsa, bu, hangi aboneliğin kayıttan kaldırıldığını belirtmek için kayıt kaldırma saatinde ayarlanmalıdır. Ters durumda, aboneliği tanımlamak için **Deregister Subscriber** komutunun kullanıcı kimliği kullanılır. Bu değer, bir abonelik adı sağlanırsa, diğer abone tanıtıcılarıyla birlikte geçersiz kılınmaktadır.

Varsayılan değer, bu özellik atlanırsa, hiçbir kayıt seçeneği ayarlanmamaktadır.

#### **QMgrName (MQPSC\_Q\_MGR\_NAME)**

Değer, abone kuyruğu için kuyruk yöneticisi adıdır. Özgün abonelikte kullanılan *QMgrName* ile eşleşmesi gerekir.

Bu özellik atılırsa, varsayılan değer, ileti tanımlayıcısında (*MQMD*) *ReplyToQMgr* adıdır. Sonuçtaki ad boş bırakılırsa, varsayılan değer olarak kuyruk yöneticisinin adı kullanılır.

#### **QName (MQPSC\_Q\_NAME)**

Değer, abone kuyruğun adıdır. Bu, özgün abonelikte kullanılan *QName* ile eşleşmelidir.

Bu özellik atılırsa, varsayılan değer, ileti tanımlayıcısındaki (*MQMD*) *ReplyToQ* adının boş bırakılmamasını sağlar.

#### **SubName (MQPSC\_XX\_ENCODE\_CASE\_ONE subscription\_name)**

If you specify *SubName* on a **Deregister Subscriber** command the *SubName* value takes precedence over all other identifier fields except the *userid*, unless *VariableUserTnt* is set on the subscription itself. *VariableUserTnt* belirlenmezse, **Deregister Subscriber** komutu yalnızca, komut iletisi kullanıcı değilse, dönüş kodu *MQRCCF\_DUPLICATE\_IDENTITY* ile başarısız olursa, komut iletisinin kullanıcı kimliği aboneliğin eşiyile eşleşirse başarılı olur.

If a subscription exists that matches the traditional identity of this command but has no *SubName* the **Deregister Subscriber** command fails with return code *MQRCCF\_XX\_ENCODE\_CASE\_ONE sub\_name\_error*. Geleneksel kimlikle eşleşen bir komut iletisi kullanılarak *SubName* içeren bir aboneliğin kaydını kaldırma girişiminde bulunulursa, ancak hiçbir *SubName* komutu başarılı olursa, komut başarılı olur.

#### **SubUserVerileri (MQPSC\_XX\_ENCODE\_CASE\_ONE subscription\_user\_data)**

Bu değişken uzunluklu bir metin dizilimidir. Değer, abonelikte kuyruk yöneticisi tarafından saklanır, ancak yayının aboneye teslim edilmesi üzerinde herhangi bir etkisi yoktur. Değer, yeni bir değerle aynı aboneliğe yeniden kayıt yaptırılarak değiştirilebilir. Bu öznelik, uygulamanın kullanımı içindir.

*SubUserVerileri* varsa, abonelik için *Meta Konu* bilgisinde (*MQCACF\_REG\_SUZ\_USER\_DATA*) *SubUserVerileri* döndürülür.

## **Örnek**

Burada, bir **Deregister Subscriber** komut iletisi için bir *NameValueData* örneği yer alıyor. Bu örnekte, örnek uygulama, tüm eşleşmeler için en son puanı içeren konulara aboneliğini derlemektedir. The subscriber's identity, including the *CorrelId*, is taken from the defaults in the *MQMD*.

```
<psc>
  <Command>DeregSub</Command>
  <RegOpt>CorrelAsId</RegOpt>
  <Topic>Sport/Soccer/State/LatestScore/#</Topic>
</psc>
```



## İletiyi yayınla

**Publish** komut iletisi, belirtilen bir konu ya da konu üzerinde bilgi yayınlamak için bir kuyruğa ya da bir kuyruk yöneticisinden aboneye yerleştirilir.

Belirli bir konu ya da konu başlıklarına ilişkin bilgi yayınlama yetkisi, bir kuyruğa ileti koyma yetkisi ve yetki.

Kullanıcının bazı konularda bilgi yayınlama yetkisi varsa, ancak tümü, konular, yalnızca bu konular yayınlamak için kullanılır; bir uyarı yanıtı, yayınlanmamak için hangi konuların kullanılmadığını gösterir.

Bir abonenin eşleşen abonelikleri varsa, kuyruk yöneticisi **Publish** iletisini karşılık gelen **Register Subscriber** komut iletilerinde tanımlanan abone kuyruklarına iletir.

Kuyruk yöneticisine bir komut iletisi gönderirken gereken ileti tanımlayıcı (MQMD) parametrelerinin ayrıntıları için bkz. [Kuyruk Yöneticisi Yanıt iletisi](#) ve kuyruk yöneticisi bir yayını bir aboneye iletirken kullanılır.

Kuyruk yöneticisi, yerel bir yayın değilse, **Publish** iletisini, abonelikleri eşleşen ağdaki diğer kuyruk yöneticilerine iletir; bu ileti yerel bir yayın değilse, bu ileti ağındaki diğer kuyruk yöneticilerine iletilir.

Yayın verileri, varsa, iletinin gövdesine eklenir. Veriler, MQRFH2 üstbilgisinin NameValueData alanındaki bir <mcd> klasöründe tanımlanabiliyor.

## Özellikler

### Komut (MQPSC\_COMMAND)

Değer Yayınla (MQPSC\_PUBLISH).

Bu özellik belirtilmeli.

### Konu (MQPSC\_KONUSU)

Değer, bu yayını kategorize eden bir konu içeren bir dizgidir. Genel arama karakterlerine izin verilmez.

Konuyu NAMElist SYSTEM.QPUBSUB.QUEUE.NAMELIST, bu görevin tamamlamaya ilişkin yönergeler için [Akış eklenmesi](#) başlıklı konuya bakın.

Bu özellik belirtilmeli ve isteğe bağlı olarak, gerektiği kadar konu için yinelenebilir.

### SubPoint (MQPSC\_SUBSCRIPTION\_POINT)

Yayının yayınlandığı abonelik noktası.

WebSphere Event Broker 6.0 içinde, <SubPoint> özelliğinin değeri, yayınlamayı işleyen Publication düğümünün Subscription Point özneliğinin değeridir.

IBM WebSphere MQ 7.0.1' ta, <SubPoint> özelliğinin değeri, bir abonelik noktasının adıyla eşleşmelidir. Bkz. [Abonelik noktası eklenmesi](#).

### PubOpt (MQPSC\_PUBLICATION\_OPTION)

Yayın seçenekleri özelliği aşağıdaki değerleri alabilir:

#### RetainPub

(MQPSC\_RETAIN\_PUB)

Kuyruk yöneticisi, yayının bir kopyasını saklayacaktır. Bu seçenek ayarlanmazsa, kuyruk yöneticisi yayını tüm yürürlükteki abonelerine gönderdiyse, yayın silinir.

#### IsRetainedPub

(MQPSC\_IS\_RETAINED\_PUB)

(Yalnızca bir kuyruk yöneticisi tarafından ayarlanabilirler.) Bu yayın kuyruk yöneticisi tarafından korundu. Kuyruk yöneticisi, bu yayının daha önce yayınlandığını ve aboneliğin InformIfAtained seçeneğiyle kaydedildiğini belirterek bir aboneyi bilgilendirmek için bu seçeneği ayarlar. Yalnızca bir Aboneyi Kaydet ya da Güncelleme İste komut iletisinde ayarlanır. Doğrudan abonelere gönderilen tutulan yayınların bu aksam ayarına sahip olmayanlardan yoktur.

## Yerel

(MQPSC\_LOCAL)

Bu seçenek, kuyruk yöneticisine bu yayının diğer kuyruk yöneticilerine gönderilmemesi gerektiğini bildirir. Bu kuyruk yöneticisinde kayıtlı olan tüm aboneler, abonelikleri eşleştirdiklerinde bu yayını alırlar.

## YalnızcaOtherSubs

(YALNIZCA MQPSC\_OTHER\_SUBS\_ONLY)

Bu seçenek, bir yayıncının aynı konuya abone olduğu, konferans tipi uygulamaların daha basit işlenmesini sağlar. Kuyruk yöneticisine, aboneliği eşleşen bir aboneliğe sahip olsa bile yayıncının abone kuyruğuna göndermemeyi belirtir. Yayıncının abone kuyruğu, aşağıdaki listede açıklandığı şekilde QMgrName, QNameve isteğe bağlı CorrelIdbileşenlerinden oluşur.

## CorrelAsTnt

(MQPSC\_CORREL\_ID\_AS\_IDENTITY)

MQMD 'deki CorrelId (sıfır olmamalıdır), yayıncının abone olduğu uygulamalarda, yayıncının abone kuyruğunun bir parçasıdır.

## Yok

(MQPSC\_NONE)

Tüm seçenekler varsayılan değerlerini alır. Bu, yayın seçenekleri özelliğinin atlanması ile aynı etkiye sahiptir. Diğer seçenekler aynı zamanda belirtilirse, Yok yoksayılır.

Ek <PubOpt> öğelerini tanıtarak birden çok yayın seçeneğiniz olabilir.

Varsayılan değer, bu özellik atlanırsa, hiçbir yayın seçeneği ayarlanmaz.

## PubTime (MQPSC\_PUBLISH\_TIMESTAMP)

Değer, yayıncı tarafından ayarlanan isteğe bağlı bir yayınlama zaman damgasıdır. Biçimi şu olan 16 karakter uzunluğundadır:

```
YYYYMMDDHHMSSSTH
```

Universal Time 'ı kullanma. Bu bilgiler, abonelere gönderilmeden önce kuyruk yöneticisi tarafından denetlenmez.

## SeqNum (MQPSC\_SEQUENCE\_NUMBER)

Değer, yayıncı tarafından ayarlanan isteğe bağlı bir sıra numarasıdır.

Her yayınla birlikte 1 artırılmalıdır. Ancak bu, yalnızca bu bilgileri abonelere ileten kuyruk yöneticisi tarafından denetlenmez.

Aynı konudaki yayınlar farklı birbirine bağlı kuyruk yöneticilerine yayınlandıysa, bu, kullanılıyorsa, sıra sayılarının anlamlı olmasını sağlamak için yayıncıların sorumluluğundadır.

## QMgrName (MQPSC\_Q\_MGR\_NAME)

Bu değer, yayıncının abone olduğu uygulamalarda, yayıncının abone kuyruğu için kuyruk yöneticisinin adını içeren bir dizgidir (yalnızca bkz. OtherSubsYalnızca ).

Bu özellik atılırsa, varsayılan değer, ileti tanımlayıcısında (MQMD) ReplyToMgr adıdır. Sonuçtaki ad boş bırakılırsa, varsayılan değer olarak kuyruk yöneticisinin adı kullanılır.

## QName (MQPSC\_Q\_NAME)

Değer, yayıncının abone olduğu uygulamalarda, yayıncının abone kuyruğunun adını içeren bir dizgidir (yalnızca bkz. OtherSubsYalnızca ).

Bu özellik atılırsa, YalnızcaOtherSubs ileti tanımlayıcısındaki (MQMD) ReplyToS varsayılan değeri,blankayarlandıysa boş bırakılmamalı.

## Örnek

Burada, bir **Publish** komut iletisi için bazı *NameValueData* örnekleri yer alıyor.

İlk örnek, örnek uygulamada eşleşme benzetimcisi tarafından bir eşleşmenin başladığını göstermek için gönderilen bir yayının yayınıdır.

```
<psc>
  <Command>Publish</Command>
  <Topic>Sport/Soccer/Event/MatchStarted</Topic>
</psc>
```

İkinci örnek, alıkona bir yayında yer alan bir yayındır. Team1 ile Team2 arasındaki eşleşmede en son puan yayınlanır.

```
<psc>
  <Command>Publish</Command>
  <PubOpt>RetainPub</PubOpt>
  <Topic>Sport/Soccer/State/LatestScore/Team1 Team2</Topic>
</psc>
```

## Abone iletisini kaydet

The **Register Subscriber** command message is sent to a queue manager by a subscriber, or by another application on behalf of a subscriber, to indicate that it wants to subscribe to one or more topics at a subscription point. Bir ileti içeriği süzgeci de belirtilebilir.

Yayınlama/abone olma süzgeci ifadelerinde, iç içe yerleştirme parantezleri, performansın katlanarak azalmasına neden olur. Parantezleri 6 'dan büyük bir derinliğe iç içe yerleştirmekten kaçınınız.

İleti, SYSTEM.BROKER.CONTROL.QUEUE, kuyruk yöneticisinin denetim kuyruğudur. Bu kuyruğa bir ileti koyma yetkisi (kuyruk yöneticisinin sistem denetimcisi tarafından belirlenen) erişim yetkisine ek olarak, konu ya da abonelikte konular için gereklidir.

Kullanıcı bazı konularda yetkisi varsa, ancak tümü değil, yalnızca yetkili kişiler üzerinde kayıt yaptıysa; bir uyarı yanıtı kayıtlı olmayan kişiler olduğunu gösterir.

Kuyruk yöneticisine bir komut iletisi gönderirken gerekli olan ileti tanımlayıcı (MQMD) parametrelerinin ayrıntıları için [“Komut iletisinde kuyruk yöneticisine MQMD ayarları” sayfa 882](#) başlıklı konuya bakınız.

Kuyruğun yanıtı geçici bir dinamik kuyruksa, kuyruk kapatıldığında abonelik kuyruk yöneticisi tarafından otomatik olarak kayıttan kaldırılır.

## Özellikler

### Komut (MQPSC\_COMMAND)

Değer RegSub (MQPSC\_REGISTER\_SUBSCRIBER) değeridir. Bu özellik belirtilmeli.

### Konu (MQPSC\_KONUSU)

Abonenin yayınları almak istediği konu. Genel arama karakterleri, konunun bir parçası olarak belirtilebilir.

If you use the MQSC command **display sub** to examine the subscription created in this way, the value of the < Topic> tag is shown as the TOPICSTR property of the subscription.

Bu özellik gereklidir ve isteğe bağlı olarak, gerektiği kadar konu için yinelenabilir.

### SubPoint (MQPSC\_SUBSCRIPTION\_POINT)

Değer, aboneliğin bağlı olduğu abonelik noktasıdır.

Bu özellik atılırsa, varsayılan abonelik noktası kullanılır.

WebSphere Event Broker 6.0' ta, <SubPoint> özelliğinin değeri, abone olunan Publication düğümlerinin Abonelik Noktası özneliği değeriyle eşleşmelidir.

IBM WebSphere MQ 7.0.1' ta, <SubPoint> özelliğinin değeri, bir abonelik noktasının adıyla eşleşmelidir. Bkz. [Abonelik noktası eklenmesi](#).

### **Süzgeç Uygula (MQPSC\_FILTER)**

Değer, yayın iletilerinin içeriğine süzgeç olarak kullanılan bir SQL ifadesidir. Belirtilen konuyla ilgili bir yayın süzgeçle eşleşiyorsa, bu, aboneye gönderilir. Bu özellik, MQSUB ve MQOP çağrılarında kullanılan Seçim Dizgisine karşılık gelir. Daha fazla bilgi için İletinin içeriğini seçme başlıklı konuya bakın.

Bu özellik atılırsa, içerik süzme işlemi gerçekleşmez.

### **RegOpt (MQPSC\_REGISTRATION\_OPTION)**

Bu Kayıt Seçenekleri özelliği aşağıdaki değerleri alabilir:

#### **AddName**

(MQPSC\_ADD\_NAME)

Var olan bir abonelik için bu Kayıt Aboneliği komutunun geleneksel kimliğiyle eşleşen, ancak geçerli SubName değeri olmayan bir abonelik için belirtildiğinde, bu komutta belirtilen SubName , aboneliğe eklenir.

AddName belirtilirse, SubName alanı zorunludur, aksi takdirde MQRCCF\_REG\_OPTIONS\_ERROR döndürülür.

#### **CorrelAsTnt**

(MQPSC\_CORREL\_ID\_AS\_IDENTITY)

İleti tanımlayıcısındaki (MQMD) CorrelId , eşleşen yayınları abone kuyruğuna gönderirken kullanılır. CorrelId sıfır olmamalıdır.

#### **FullResp**

(MQPSC\_FULL\_RESPONSE)

Belirtilen yanıt iletilerinde, aboneliğin tüm öznitelikleri yanıt iletilerinde döndürüldüğünde, komut başarısız olur.

FullResp , yalnızca komut iletileri tek bir aboneliğe başvuruda bulunduğunda geçerlidir.

Bu nedenle, komutta yalnızca bir konuya izin verilir; tersi durumda, komut MQRCCF\_REG\_OPTIONS\_ERROR dönüş koduyla başarısız olur.

#### **InformIfRet**

(MQPSC\_INFORM\_IF\_ALIKOYMA)

Kuyruk yöneticisi, bir **Register Subscriber** ya da **Request Update** komut iletilerine yanıt olarak bir Yayınlama iletileri gönderdiğinde bir yayının alınması durumunda aboneyi bilgilendirir. Kuyruk yöneticisi, iletide IsRetainedPub yayın seçeneğini de ekleyerek bunu yapar.

#### **JoinExcl**

(MQPSC\_JOIN\_EXCLUSIVE)

Bu seçenek, belirtilen SubIdentity ' in, abonelik için belirlenen kimlik kümesinin özel üyesi olarak ekleneceğini ve kümeye başka hiçbir kimliğin eklenebileceğini belirtir.

Kimlik önceden 'shared' birleştirdiyse ve kümedeki tek girişse, küme bu kimlik tarafından tutulan dışlayıcı bir kilitle değiştirilir. Ters durumda, abonelik şu anda kimlik kümesinde (paylaşılan erişime sahip) başka kimlikler varsa, komut MQRCCF\_SUBSCRIPTION\_IN\_USED dönüş koduyla başarısız olur.

#### **JoinShared**

(MQPSC\_JOIN\_SHARED)

Bu seçenek, belirtilen SubIdentity (Alt Kimlik) aboneliğinin kimlik kümesine ekleneceğini belirtir.

Abonelik şu anda özel olarak kilitlendiyse ( JoinExcl seçeneğini kullanarak), aboneliği kilitleyen kimlik bu komut iletilerine aynı kimlik olduğu sürece, komut MQRCCF\_SUBSCRIPTION\_LOCKED dönüş koduyla başarısız olur. Bu durumda, kilit otomatik olarak paylaşılan bir kilitle değiştirilir.

#### **Yerel**

(MQPSC\_LOCAL)

Abonelik yereldir ve ağıdaki diğer kuyruk yöneticilerine dağıtılmaz. Diğer kuyruk yöneticilerindeki yayınlar da karşılık gelen bir genel aboneliğe sahip değilse, bu aboneye teslim edilmez.

### **YalnızcaNewPubs**

(MQPSC\_NEW\_PUBS\_ONLY)

Aboneliğin kaydedildiği tarihte var olan tutulan yayınlar aboneye gönderilmez; yalnızca yeni yayınlar gönderilir.

Bir abone yeniden kayıt yaptıysa ve bu seçeneği artık ayarlanmıyorsa, önceden gönderilmiş olan bir yayın yeniden gönderilebilir.

### **NoAlter**

(MQPSC\_NO\_ALTER)

Var olan bir eşleşen aboneliğin öznitelikleri değiştirilmez.

Bir abonelik oluşturulduğunda, bu seçenek yoksayılr. Belirtilen diğer tüm seçenekler yeni abonelik için geçerli olur.

Bir SubIdentity , birleştirme seçeneklerinden birine de sahipse ( JoinExcl ya da JoinShared ) belirtilen, tanıtıcı, NoAlter ' un belirtilip belirtilmediğine bakılmaksızın kimlik kümesine eklenir.

### **Yok**

(MQPSC\_NONE)

Tüm kayıt seçenekleri varsayılan değerlerini alır.

Abone zaten kayıtlı ise, seçenekleri varsayılan değerlerine sıfırlanır (bu, kayıt seçenekleri özelliğini atlayarak aynı etkileye sahip olmadığını unutmayın) ve abonelik süre bitimi, **Register Subscriber** iletisinin MQMD ' den güncellenmektedir.

Aynı anda başka kayıt seçenekleri belirlendiyse, Yok yoksayılr.

### **NonPers**

(MQPSC\_NON\_PERSISTENT)

Bu abonelikte eşleşen yayınlar, aboneye kalıcı olmayan iletiler olarak teslim edilir.

### **Pers**

(MQPSC\_PERSISTENT)

Bu abonelikte eşleşen yayınlar, aboneye kalıcı iletiler olarak teslim edilir.

### **PersAsPub**

(MQPSC\_PERSSTENT\_AS\_PUBLICE)

Bu abonelikte eşleşen yayınlar, yayıncı tarafından belirtilen kalıcılık ile aboneye teslim edilir. Bu varsayılan davranıştır.

### **PersAsKuyruğu**

(MQPSC\_PERSISTENT\_AS\_Q)

Bu abonelikte eşleşen yayınlar, abone kuyruğunda belirtilen kalıcılıkla aboneye teslim edilir.

### **PubOnReqOnly**

(MQPSC\_PUB\_ON\_REQUEST\_ONLY)

Kuyruk yöneticisi, bir **Request Update** komut iletisine yanıt olarak, yayınlarını aboneye göndermez.

### **VariableUserTanıtıcısı**

(MQPSC\_VARIABLE\_USER\_ID)

Abonenin kimliği (kuyruk, kuyruk yöneticisi ve correlid) belirtildiğinde, tek bir kullanıcı kimliği ile sınırlı değildir. Bu, kuyruk yöneticisinin, özgün kayıt iletisinin kullanıcı kimliğini abonenin kimliğiyle ilişkilendiren var olan davranışından ve bundan sonra o kimliği kullanan diğer kullanıcıların

herhangi bir kullanıcının engellediği bir davranışa göre farklılık gösterir. If a new subscriber tries to use the same identity *MQRCCF\_DUPLICATE\_SUBSCRIPTION* is returned.

Bu, kullanıcının uygun yetkiye sahip olması durumunda, herhangi bir kullanıcının aboneliğin kaydını değiştirmesini ya da kaydını kaldırmasını sağlar. Bu nedenle, kullanıcı kimliğinin özgün aboneden eşleşip eşleşmediği denetlenmesine gerek yoktur.

Bu seçeneği varolan bir aboneliğe eklemek için, komutun aynı kullanıcı kimliği ile özgün aboneliğin kendisiyle gelmesi gerekir.

**Request Update** komutunun aboneliği `VariableUserId` değerine ayarlanmışsa, bu, hangi aboneliğin gönderileceğini belirtmek için istek güncelleme sırasında ayarlanmalıdır. Ters durumda, aboneliği tanımlamak için **Request Update** komutunun kullanıcı kimliği kullanılır. Bu değer, bir abonelik adı sağlanırsa, diğer abone tanıtıcılarıyla birlikte geçersiz kılınmaktadır.

Bu seçenek belirlenmeden bir **Register Subscriber** komut iletisi, bu seçenek ayarına sahip var olan bir aboneliğe başvuruyorsa, bu seçenek bu abonelikten kaldırılır ve aboneliğin kullanıcı kimliği artık düzeltilir. If there already exists a subscriber which has the same identity (queue, queue manager and correlation identifier) but with a different user ID associated to it, the command fails with return code *MQRCCF\_DUPLICATE\_IDENTITY* because there can only be one userid associated with a subscriber identity.

Kayıt seçenekleri özelliği atlanırsa ve abone zaten kayıtlı ise, kayıt seçenekleri değiştirilmez ve abonelik süre bitimi, **Register Subscriber** iletisinin `MQMD` ' den güncellenir.

Abone zaten kayıtlı değilse, varsayılan değerlerini alan tüm kayıt seçenekleri ile yeni bir abonelik yaratılır.

Varsayılan değerler `PersAsPub` ' dir ve başka bir seçenek ayarlanmaz.

#### **QMgrName (MQPSC\_Q\_MGR\_NAME)**

Bu değer, kuyruk yöneticisi tarafından eşleşen yayınların gönderildiği, abone kuyruğuna ilişkin kuyruk yöneticisinin adıdır.

Bu özellik atılırsa, varsayılan değer, ileti tanımlayıcısında (`MQMD`) `ReplyToQMgr` adıdır. Sonuçtaki ad boş bırakılırsa, varsayılan değer olarak kuyruk yöneticisinin `QMgrName` kullanılır.

#### **QName (MQPSC\_Q\_NAME)**

Bu değer, kuyruk yöneticisi tarafından eşleşen yayınların gönderileceği abone kuyruğunun adıdır.

Bu özellik atılırsa, varsayılan değer, ileti tanımlayıcısındaki (`MQMD`) `ReplyToQ` adının bu durumda boş bırakılmamasını sağlar.

Kuyruk, geçici bir dinamik kuyruksa, yayınların kalıcı olmayan teslimi ( `NonPers` ) `<RegOpt>` özelliğinde belirtilmelidir.

Kuyruk geçici bir dinamik kuyruksa, kuyruk kapatıldığında abonelik, kuyruk yöneticisi tarafından otomatik olarak kaydı kaldırılır.

#### **SubName (MQPSC\_XX\_ENCODE\_CASE\_ONE subscription\_name)**

Bu, belirli bir aboneliğe verilen bir addır. Bir aboneliği belirtmek için kuyruk yöneticisi, kuyruk ve isteğe bağlı `correlId` yerine bunu kullanabilirsiniz.

Bu **SubName** ile bir abonelik zaten varsa, aboneliğin diğer öznitelikleri (Konu, `QMgrName`, `QName`, `CorrelId`, `UserId`, `RegOpts`, `UserSubData` ve `Expiry`), belirtildiyse, yeni `Aboneyi` Kaydet komut iletisinde geçirilen özniteliklerle geçersiz kılır. Ancak, **SubName** belirtilen `QName` alanı belirtilmediyse ve `MQMD` üstbilgisinde bir `ReplyToQ` belirtilirse, abone kuyruğu `ReplyToQ` olacak şekilde değiştirilir.

Bu komutun geleneksel kimliğiyle eşleşen bir abonelik zaten varsa, ancak **SubName** yoksa, **AddName** seçeneği belirlenmedikçe, Kayıt komutu *MQRCCF\_DUPLICATE\_SUBSCRIPTION* dönüş kodu ile başarısız olur.

Aynı **SubName** değerini belirten başka bir `Aboneyi` Kaydet komutunu kullanarak, var olan adlandırılmış bir aboneliği değiştirmeye çalışırsanız ve yeni komuttaki Konu, `QMgrName`, `QName` ve `CorrelId` değerleri, `SubName` ile tanımlanmış farklı bir abonelikte eşleşirse, komut

*MQRCCF\_DUPLICATE\_SUBSCRIPTION* dönüş kodu ile başarısız olur. Bu, aynı abonelik ile ilgili iki abonelik adını önler.

### **SubIdentity (MQPSC\_SUBSCRIPTION\_IDENTITY)**

Bu dizgi, bir aboneliği olan bir uygulamayı bir abonelik ile temsil etmek için kullanılır. Uzunluk üst sınırı 64 karakter olan değişken uzunluklu bir karakter dizilimidir ve isteğe bağlıdır. Kuyruk yöneticisi, her abonelik için bir abone kimlikleri kümesini tutar. Her abonelik, kimlik kümesine yalnızca bir kimlik ya da sınırsız sayıda kimlik içermesine izin verebilir ( **JoinShared** ve **JoinExcl** seçeneklerine bakın).

A subscribe command that specifies the **JoinShared** or **JoinExcl** option adds the **SubIdentity** to the subscription's identity set, if it is not already there and if the existing set of identities allows such an action; that is, no other subscriber has joined exclusively or the identity set is empty.

Bir **SubIdentity** işleminin belirtildiği bir Abonenin `Kaydı` komutunun sonucu olarak aboneliğin özniteliklerinin değiştirilmesi, yalnızca bu abonelik için kimlik kümesinin tek üyesi olması durumunda başarılı olur. Ters durumda, komut *MQRCCF\_SUBSCRIPTION\_IN\_USE* dönüş koduyla başarısız olur. Bu, bir aboneliğin özniteliklerinin diğer ilgili abonelerin farkında olmadan değişmesini önler.

64 karakterden uzun bir karakter dizisi belirtirseniz, komut *MQRCCF\_SUB\_IDENTITY\_ERROR* dönüş kodu ile başarısız olur.

### **SubUserVerileri (MQPSC\_XX\_ENCODE\_CASE\_ONE subscription\_user\_data)**

Bu değişken uzunluklu bir metin dizilimidir. Değer, abonelik ile kuyruk yöneticisi tarafından saklanır, ancak aboneye yayın teslimi üzerinde hiçbir etkisi yoktur. Değer, yeni bir değerle aynı aboneliğe yeniden kayıt yaptırılarak değiştirilebilir. Bu öznitelik, uygulamanın kullanımı için burada yer alıyor.

**SubUserVerileri** , varsa, abonelik için Meta konu bilgilerinde (*MQCACF\_REG\_SUB\_USER\_DATA*) döndürülür.

If you specify more than one of the registration option values `NonPers` , `PersAsPub` , `PersAsQueue` , and `Pers` , then only the last one is used. Bu seçenekleri tek bir abonelikte birleştiremezsiniz.

## **Örnek**

Burada, bir **Register Subscriber** komut iletisi için bir `NameValueData` örneği yer alıyor. Örnek uygulamada, sonuç hizmeti bu iletiyi, 'Yayınlar olarak kalıcı' seçeneği ayarlı olarak, tüm eşleşmelerdeki en son puanları içeren konulara abone olarak kaydettirmek için kullanır. The subscriber's identity, including the `CorrelId`, is taken from the defaults in the MQMD.

```
<psc>
  <Command>RegSub</Command>
  <RegOpt>PersAsPub</RegOpt>
  <RegOpt>CorrelAsId</RegOpt>
  <Topic>Sport/Soccer/State/LatestScore/#</Topic>
</psc>
```

## **Güncelleme İletisi İste**

**Request Update** komut iletisi bir aboneden kuyruk yöneticisine gönderilir; belirtilen konu ve abonelik noktası için yürürlükteki alıkonan yayınların verili (isteğe bağlı) süzgeçle eşleşen yürürlükteki yayınlarını istemek için.

Bu ileti *SYSTEM.BROKER.CONTROL.QUEUE* (Kuyruk yöneticisi), kuyruk yöneticisinin denetim kuyruğu. İstek güncellemesinde konuya ilişkin erişim yetkisine ek olarak, bu kuyruğa bir ileti koyma yetkisi gereklidir; bu, kuyruk yöneticisinin sistem yöneticisi tarafından ayarlanır.

Bu komut, abonenin kayıt yaptıklarında `PubOnReqOn1y` seçeneğini belirlediğinde kullanılır. Kuyruk yöneticisinin alıkonan yayınları varsa, bu yayınlar aboneye gönderilir. Kuyruk yöneticisinin eşleşen bir yayını yok ise, istek dönüş kodu *MQRCCF\_NO\_RETAINED\_MSG* ile başarısız olur. İstekte bulunanın önceden aynı Konu, `SubPoint` ve `Süzgeç` değerlerine sahip bir abonelik kaydettirmiş olması gerekir.

## Özellikler

### Komut (**MQPSC\_COMMAND**)

Değer: ReqUpdate (**MQPSC\_REQUEST\_UPDATE**). Bu özellik belirtilmeli.

### Konu (**MQPSC\_KONUSU**)

Değer, abonenin istekte olduğu konudur; genel arama karakterlerine izin verilir.

Bu özellik belirtilmeli, ancak bu iletide yalnızca bir oluşuma izin verilir.

### SubPoint (**MQPSC\_SUBSCRIPTION\_POINT**)

Değer, aboneliğin bağlı olduğu abonelik noktasıdır.

Bu özellik atılırsa, varsayılan abonelik noktası kullanılır.

### Süzgeç Uygula (**MQPSC\_FILTER**)

Değer, yayın iletilerinin içeriğine süzgeç olarak kullanılan bir ESQL ifadesidir. Belirtilen konuyla ilgili bir yayın süzgeçle eşleşiyorsa, bu, aboneye gönderilir.

< Süzgeç > özelliği, şimdi güncelleme talep eden özgün abonelikte belirtilen değerle aynı değere sahip olmalıdır.

Bu özellik atılırsa, içerik süzme işlemi gerçekleşmez.

### RegOpt (**MQPSC\_REGISTRATION\_OPTION**)

Kayıt seçenekleri özelliği aşağıdaki değeri alabilir:

#### CorrelAsTnt

(**MQPSC\_CORREL\_ID\_AS\_IDENTITY**)

The CorrelId in the message descriptor (MQMD), which must not be zero, is used when sending matching publications to the subscriber queue.

#### Yok

(**MQPSC\_NONE**)

Tüm seçenekler varsayılan değerlerini alır. Bu, <RegOpt> özelliğinin atlanması ile aynı etkiye sahiptir. Diğer seçenekler aynı zamanda belirtilirse, Yok yoksayılır.

### VariableUserTanıtıcısı

(**MQPSC\_VARIABLE\_USER\_ID**)

Abonenin kimliği (kuyruk, kuyruk yöneticisi ve correlid) belirtildiğinde, tek bir kullanıcı kimliği ile sınırlı değildir. Bu, kuyruk yöneticisinin, özgün kayıt iletilerinin kullanıcı kimliğini abonenin kimliğiyle ilişkilendiren var olan davranışından ve bundan sonra o kimliği kullanan diğer kullanıcıların herhangi bir kullanıcının engellediği bir davranışa göre farklılık gösterir. Yeni bir abone aynı kimliği kullanmaya çalışırsa, komut **MQRCCF\_DUPLICATE\_SUBSCRIPTION** dönüş kodu ile başarısız olur.

Bu, herhangi bir kullanıcının uygun yetkiye sahip olduğunda aboneliği değiştirmesine ya da kayıttan kaldırmasına olanak sağlar. Bu nedenle, kullanıcı kimliğinin özgün aboneden eşleşip eşleşmediği denetlenmesine gerek yoktur.

Bu seçeneği varolan bir aboneliğe eklemek için, komutun özgün abonelikte aynı kullanıcı kimliğiyle gelmesi gerekir.

**Request Update** komutunun aboneliği **VariableUserId** değerine ayarlanmışsa, bu, hangi aboneliğin gönderileceğini belirtmek için istek güncelleme sırasında ayarlanmalıdır. Ters durumda, aboneliği tanımlamak için **Request Update** komutunun kullanıcı kimliği kullanılır. Bu değer, bir abonelik adı sağlanırsa, diğer abone tanıtıcılarıyla birlikte geçersiz kılınmaktadır.

Varsayılan değer, bu özellik atlanırsa, hiçbir kayıt seçeneği ayarlanmamaktadır.

### QMgrName (**MQPSC\_Q\_MGR\_NAME**)

Bu değer, kuyruk yöneticisi tarafından eşleşen alıkonan yayının gönderildiği, abone kuyruğuna ilişkin kuyruk yöneticisinin adıdır.

Bu özellik atılırsa, varsayılan değer, ileti tanımlayıcısında (MQMD) ReplyToQMGr adıdır. Sonuçtaki ad boş bırakılırsa, varsayılan değer olarak kuyruk yöneticisinin QMGrName kullanılır.



### QName (MQPSC\_Q\_NAME)

Değer, kuyruk yöneticisi tarafından eşleşen alıkonan yayınının gönderildiği abone kuyruğunun adıdır.

Bu özellik atılırsa, varsayılan değer, ileti tanımlayıcısındaki (MQMD) ReplyToQ adının bu durumda boş bırakılmamasını sağlar.

### SubName (MQPSC\_XX\_ENCODE\_CASE\_ONE subscription\_name)

Bu, belirli bir aboneliğe verilen bir addır. If specified on a **Request Update** command the SubName value takes precedence over all other identifier fields except the userid, unless VariableUserTnt is set on the subscription itself. VariableUserId belirlenmezse, *Request Update* komutu, yalnızca komut iletilisinin kullanıcı kimliği aboneliğin eşyle eşleşiyorsa başarılı olur. Komut iletilisinin kullanıcı kimliği, aboneliğin tanıtıcısı ile eşleşmiyorsa, komut *MQRCCF\_DUPLICATE\_IDENTITY* dönüş koduyla başarısız olur.

VariableUserId değeri ayarlandıysa ve kullanıcı kimliği, abonelikten farklıysa, yeni komut iletilisi kullanıcı kimliğinin akış kuyruğuna göz atma ve aboneliğin abone kuyruğuna konması için yetkisi varsa, komut başarılı olur. Ters durumda, komut *MQRCCF\_NOT\_YETKILID* dönüş kodu ile başarısız olur.

Bu komutun geleneksel kimliğiyle eşleşen bir abonelik varsa, ancak SubName yoksa, **Request Update** komutu *MQRCCF\_XX\_ENCODE\_CASE\_ONE sub\_name\_errord* dönüş koduyla başarısız olur.

Geleneksel kimlikle eşleşen bir komut iletilisi kullanarak SubName içeren bir abonelik için güncelleme isteğinde bulunmaya çalışılırsa, ancak SubName belirtilmez, komut başarılı olur.

## Örnek

Burada, bir **Request Update** komut iletilisi için bir NameValueData örneği yer alıyor. Örnek uygulamada, sonuçlar hizmeti bu iletiyi, tüm ekiplerin en son puanlarını içeren alıkonan yayınları istemek için kullanır. The subscriber's identity, including the CorrelId, is taken from the defaults in the MQMD.

```
<psc>
  <Command>ReqUpdate</Command>
  <RegOpt>CorrelAsId</RegOpt>
  <Topic>Sport/Soccer/State/LatestScore/#</Topic>
</psc>
```

## Kuyruk Yöneticisi Yanıt İletisi

A **Queue Manager Response** message is sent from a queue manager to the ReplyToS of a publisher or a subscriber, to indicate the success or failure of a command message received by the queue manager if the command message descriptor specified that a response is required.

Yanıt iletilisi, bir <psc> klasöründe bulunan MQRFH2 üstbilgisinin NameValueVerileri alanında bulunur.

In the case of a warning or error, the response message contains the <psc> folder from the command message as well as the <psc> folder. İleti verisi (varsa), kuyruk yöneticisi yanıt iletilisinde yer almıyorsa. Bir hata durumunda, bir hataya neden olan iletilinin hiçbiri işlenmedi; bir uyarı durumunda, iletilinin bir kısmı başarıyla işlenmiş olabilir.

Yanıt gönderilirken bir hata ortaya çıktı:

- Yayın iletileri için, kuyruk yöneticisi, MQPUT başarısız olursa, yanıtı IBM MQ ölü-mektup kuyruğuna göndermeyi dener. Bu, gönderinin yayıncıya geri gönderilememesi durumunda da yayının abonelere gönderilmesini sağlar.
- Diğer iletiler için ya da yayın yanıtı ölü-mektup kuyruğuna gönderilemezse, bir hata günlüğe kaydedilir ve komut iletilisi normal olarak geri alınır. Bu durum, MQInput düğümünün nasıl yapılandırılıp yapılandırılmadığına bağlıdır.

## Özellikler

### Tamamlanma (MQPSCR\_COMPLETION)

Üç değerden birini alabilen tamamlama kodu:

**tamam**

Komut başarıyla tamamlandı

**uyarı**

Komut tamamlandı, ancak uyarı ile

**hata**

Komut başarısız oldu

**Yanıt (MQPSCR\_RESPONSE)**

The response to a command message, if that command produced a completion code of **uyarı** or **hata**. Bir < Reason> özelliği içerir ve uyarının ya da hatanın nedenini gösteren diğer özellikleri içerebilir.

Bir ya da daha çok hata durumunda, yalnızca ilk hatanın nedenini belirten tek bir yanıt klasörü vardır. Bir ya da daha çok uyarı durumunda, her uyarı için bir yanıt klasörü vardır.

**Neden (MQPSCR\_REASON)**

Tamamlanma kodu, tamamlanma kodu **uyarı** ya da **hata** ise, tamamlanma kodunu niteleyen neden koddur. Bu, aşağıdaki örnekte listelenen hata kodlarından birine ayarlanır. < Reason> özelliği, bir < Response> klasöründe yer alır. The reason code can be followed by any valid property from the <psc> folder (for example, a topic name), indicating the cause of the error or warning. Neden kodu? ???, Verilerin doğru olup olmadığını denetleyin; örneğin, eşleşen köşeli ayraçlar (< >).

**Örnekler**

Burada, **Queue Manager Response** iletisinde bazı NameValueData örnekleri yer alıyor. Başarılı bir yanıt aşağıdaki gibi olabilir:

```
<pscr>
  <Completion>ok</Completion>
</pscr>
```

Aşağıda bir hata yanıtı örneği yer alıyor; hata, bir süzgeç hatasıdır. İlk NameValueData dizgisi, yanıtı içerir; ikinci komut ise özgün komutu içerir.

```
<pscr>
  <Completion>error</Completion>
  <Response>
    <Reason>3150</Reason>
  </Response>
</pscr>

<psc>
  ...
  command message (to which
  the queue manager is responding)
  ...
</psc>
```

Aşağıda bir uyarı yanıtı örneği yer alıyor (yetkisiz konular nedeniyle). İlk NameValueData dizgisi yanıtı içerir; ikinci NameValueData dizgisi özgün komutu içerir.

```
<pscr>
  <Completion>warning</Completion>
  <Response>
    <Reason>3081</Reason>
    <Topic>topic1</Topic>
  </Response>
  <Response>
    <Reason>3081</Reason>
    <Topic>topic2</Topic>
  </Response>
</pscr>

<psc>
  ...
  command message (to which
```

```
the queue manager is responding)
```

```
</psc>
```

## Yayınlama/abone olma neden kodları

Bu neden kodları, yayınlama/abone olma yanıt <psc> klasörünün Neden alanında döndürülebilirler. C ya da C++ programlama dillerinde bu kodları temsil etmek için kullanılacak sabitler de listelenir.

MQRCCF\_ sabitleri, IBM MQ cmqc . h üstbilgi dosyasını gerektirir. The MQRCCF\_ constants require the IBM MQ cmqc.fc . h header file (apart from MQRCCF\_FILTER\_HATASI and MQRCCF\_HATA\_KULLANICISI, which require the cmqpsc . h header file).

Neden kodu ve metin	Açıklama	Sertifikayı Veren
2336 MQRCCF_RFH_COMMAND_ERROR	Bir <psc> klasörünün < Komut> alanı için geçerli değerler şunlardır: RegSub, DeregSub, Publish, DeletePubve ReqUpdate. Diğer değerler, bu hata kodunun yayınlanmakta olmasına neden olur.	Herhangi bir komut
2337 MQRCCF_RFH_PARM_ERROR	<psc> ve <mcd> klasörlerinde her ikisinin de içinde belirtilebilecek geçerli bir parametre kümesi vardır. Bu klasörlerin tanımlarını denetleyin ve yanlış değerler belirtmediğinizden emin olun.	Herhangi bir komut
2338 MQRCCF_RFH_DUPLICATE_PARM	Bir <psc> klasörü içindeki bazı parametreler (Konu, Konu) yinelenebilir, ancak diğerleri (örneğin, Komut) yinelenemez. Yinelenemez bir parametre yinelenmediğinizi denetleyin.	Herhangi bir komut
2339 MQRCCF_RFH_PARM_MISSING	<psc> ya da <mcd> klasörlerindeki bazı parametreler isteğe bağlıdır ve atlanabilir; bazıları zorunludur ve atlanmamalıdır. Check that you have included all mandatory parameters within your <psc> and <mcd> folders.	Herhangi bir komut
2551 MQRCCF_SELECTION_NOT_AVAM	Belirtilen bir süzgeçle hangi abonelerin yayını alması gerektiğini belirlemek için kullanılabilir genişletilmiş ileti seçimi sağlayıcısı yok.	Yayınla, Aboneyi Kaydet ve Güncelleme İste
	Belirtilen aboneye ilişkin süzgeci işlemek için kullanılacak genişletilmiş ileti seçimi sağlayıcısı yok.	Aboneyi ve İstek Güncellemesini Kaydet
2554 MQRCCF_İÇERİĞİ	Genişletilmiş ileti seçimi sağlayıcısı, geçerli ya da alıkonan yayında bir hata buldu.	Yayınlama ve Güncelleme İste

<b>Neden kodu ve metin</b>	<b>Açıklama</b>	<b>Sertifikayı Veren</b>
3008 MQRCCF_COMMAND_FAILED	Komutun doğru yürütülmesini engelleyen bir iç hata ortaya çıktı. Komut yeniden verilirse hata oluşabilir. Kuyruk yöneticisine ilişkin sistem olay günlüğü, sorunu IBM' e bildirirken kullanılması gereken bilgileri içerir.	Herhangi bir komut
3072 MQRCCF_TOPIC_ERROR	Konu parametresi için belirttiğiniz değerlerden biri ya da daha fazlası yanlış. Konu değerlerinizin belirtilen kısıtlamalara uygun olup olmadığını denetleyin.	Herhangi bir komut
3073 MQRCCF_NOT_REGISTERED	The combination of SubPoint, Topic, and Filter that you specified on your DeregSub or ReqUpdate command was either not a combination with which you had previously registered or, for the DeregSub command if the DeregAll option was specified, one of the SubPoint, Topic, or Filter properties was not used to deregister any subscription.	Abone ve İstek Güncelleme komutlarının kaydı kaldırılıyor
3074 MQRCCF_Q_MGR_NAME_ERROR	Belirlenen kuyruk yöneticisi geçerli değil ya da kuyruk yöneticisi kullanılamıyor ya da yok.	Aboneyi, Aboneyi, Aboneyi Kaydet ve Güncelleme İste komutlarını kayıttan çıkar
3076 MQRCCF_Q_NAME_ERROR	Belirlenen kuyruk adı geçerli değil ya da kuyruk belirtilen kuyruk yöneticisinde yok.	Aboneyi, Aboneyi, Aboneyi Kaydet ve Güncelleme İste komutlarını kayıttan çıkar
3077 MQRCCF_NO_RETAINED_MSG	Belirttiğiniz konu için saklanan ileti yok. Bu durum, uygulama programınızın tasarımına bağlı olarak bir hata olabilir ya da olmayabilir.	Güncelleme İste komutu
3079 MQRCCF_INCORRECT_Q	RegSub, DeregSubve ReqUpdate komutları, her zaman, amaçlanan kuyruk yöneticisinin SYSTEM.BROKER.CONTROL.QUEUE kuyruğuna gönderilir. Yayınlama ve Silme Yayın komutlarının, amaçlanan yayınlama/abone olma ileti akışı için giriş kuyruğuna gönderilir; bu, ileti akışı tasarlandığında belirlenir. Bir komut yanlış kuyruğa gönderilirse bu hata kodu döndürülür.	Herhangi bir komut

<b>Neden kodu ve metin</b>	<b>Açıklama</b>	<b>Sertifikayı Veren</b>
3080 MQRCCF_CORREL_ID_ERROR	You have specified CorrelAsId as one of your RegOpt parameters. Ancak, MQMD ' nin CorrelId alanı geçerli bir ilinti tanıtıcısı (yani, MQCI_NONE olarak ayarlıdır) içermiyor.	Aboneyi Kaydet ve Abone Kaydet komutlarını kaydet
3081 MQRCCF_NOT_YETKILI	İstenen işlemi gerçekleştirme yetkiniz yok. Kuyruk yöneticisine ilişkin yetki ayarları, Konular Sıradüzeni düzenleyicisi kullanılarak sistem denetimcisi tarafından işlenir.	Abone komutlarını yayınlama ve kaydet
3083 MQRCCF_REG_OPTIONS_ERROR	RegSub ya da DeregSub komutunuzu içeren <psc> klasöründe tanınmayan bir RegOpt parametresi belirtmiş olmanız.	Aboneyi Kaydet ve Abone Kaydet komutlarını kaydet
3084 MQRCCF_PUB_OPTIONS_ERROR	Publish komutunuzu içeren <psc> klasöründe tanınmayan bir PubOpt parametresi belirtmiş olmanız.	Yayınlama komutu
3087 MQRCCF_DEL_OPTIONS_ERROR	You have specified an unrecognized DelOpt parameter in the <psc> folder that contains your DeletePub command.	Publication komutunu sil
3150 MQRCCF_FILTER_HATASI	Süzgeç değıştirgesi için belirtilen değer geçerli değil. Süzgeç ifadelerine ilişkin geçerli sözdizimini açıklayan ve ifadenizin uyumlu olduğundan emin olmak için bölümü işaretleyin.	Aboneyi, Aboneyi Kaydet ve Güncelleme İste komutlarını kayıttan çıkar
3151 MQRCCF_HATA_KULLANICISI	Belirtilen bir abonelik eşleşen bir abonelik zaten var; ancak, farklı bir kullanıcı tarafından kaydedildi. Bir abonelik, yalnızca ilk kayıt yaptıran kullanıcı tarafından değıştirilebilir ya da kayıttan kaldırılabilir.	Aboneyi, Aboneyi Kaydet ve Güncelleme İste komutlarını kayıttan çıkar
3152 MQRCCF_DUPLICATE_SUBSCRIPTION	Farklı bir abonelik adıyla eşleşen bir abonelik zaten var.	
3153 MQRCCF_XX_ENCODE_CASE_ONE sub_name_error	Abonelik adı biçimi geçerli değil ya da abonelik adı olmayan bir eşleşen abonelik zaten var.	
3154 MQRCCF_SUB_IDENTITY_ERROR	Abonelik kimliği parametresi hatalı. Sağlanan değer izin verilen uzunluk üst sınırını aşılıyor ya da abonelik kimliği şu anda aboneliğin kimlik kümesinin bir üyesi değil ve bir Katılım kaydı seçeneği belirtilmemiş.	
3155 MQRCCF_SUBSCRIPTION_IN_USE	Bu kümenin tek üyesi olmayınca, bir aboneliği değıştirme ya da kayıttan kaldırma girişimi, kimlik kümesinin bir üyesi tarafından denendi.	

Neden kodu ve metin	Açıklama	Sertifikayı Veren
3156 MQRCCF_SUBSCRIPTION_LOCKED	Abonelik şu anda özel olarak başka bir kimlik tarafından kilitlenmiş durumda.	
3157 MQRCCF_ALREADY_JOINED	Bir Birleştirme kaydı seçeneği belirtildi, ancak abone kimliği zaten aboneliğin kimlik kümesinin bir üyesiydi.	

## Komut iletilerinde kuyruk yöneticisine MQMD ayarları

Kuyruk yöneticisine komut iletileri gönderen uygulamalar, ileti tanımlayıcısında (MQMD) aşağıdaki alan ayarlarını kullanır. Varsayılan değer olarak bırakılan ya da olağan şekilde geçerli herhangi bir değere ayarlanabilen alanlar burada listelenmez.

### Rapor

Bkz. MsgType ve CorrelId.

### MsgType

MsgType , *MQMT\_REQUEST* ya da *MQMT\_DATAGRAM* olarak ayarlanmalıdır. MsgType bu değerlerden birine ayarlanmazsa, *MQRC\_MSG\_TYPE\_ERROR* döndürülecek.

Bir yanıt her zaman gerekiyorsa, bir komut iletilisi için *MsgType* , bir komut iletilisi için *MQMT\_REQUEST* değerine ayarlanmalıdır. Rapor alanındaki *MQRO\_PAN* ve *MQRO\_NAN* işaretleri, bu durumda önemli değil.

MsgType , *MQMT\_DATAGRAM* olarak ayarlandıysa, yanıtlar Rapor alanında *MQRO\_PAN* ve *MQRO\_NAN* işaretlerinin ayarına bağlıdır:

- Yalnızca *MQRO\_PAN*, kuyruk yöneticisinin yalnızca komut başarılı olursa yanıt göndereceği anlamına gelir.
- Yalnızca *MQRO\_NAN*, kuyruk yöneticisinin yalnızca komut başarısız olduğunda yanıt göndereceği anlamına gelir.
- Bir komut uyarıyla tamamlanırsa, *MQRO\_PAN* ya da *MQRO\_NAN* ayarlandıysa bir yanıt gönderilir.
- *MQRO\_PAN* + *MQRO\_NAN*, kuyruk yöneticisinin komutun başarılı olup olmadığı ya da başarısız olursa bir yanıt göndereceği anlamına gelir. This has the same effect from the queue manager's perspective as setting MsgType to *MQMT\_REQUEST*.
- Ne *MQRO\_PAN*, ne de *MQRO\_NAN* ayarlandıysa, hiçbir yanıt gönderilmez.

### Biçim

*MQFMT\_RF\_HEADER\_2* olarak ayarla

### MsgId

Bu alan olağan durumda *MQMI\_NONE* olarak ayarlanır; böylece kuyruk yöneticisi benzersiz bir değer oluşturur.

### CorrelId

Bu alan herhangi bir değere ayarlanabilir. Gönderenin kimliği bir *CorrelId* içeriyorsa, bu değer, kuyruk yöneticisi tarafından gönderene gönderilen tüm yanıt iletilerinde ayarlandığından emin olmak için Rapor alanında *MQRO\_PASS\_COREL\_ID* ile birlikte bu değeri belirtin.

### ReplyToQ

Bu alan, yanıtların (varsa) gönderileceği kuyruğu tanımlar. Bu, gönderenin kuyruğu olabilir; bu, QName parametresinin iletiden atanabileceği avantajına sahiptir. Ancak, yanıtlar farklı bir kuyruğa gönderilecekse, QName değiştirilmesi gereklidir.

### ReplyToQMgr

Bu alan, yanıtlar için kuyruk yöneticisini tanımlar. Bu alanı boş bırakırsanız (varsayılan değer), yerel kuyruk yöneticisi bu alana kendi adını koyar.

## Kuyruk yöneticisi tarafından iletilen yayınlara ilişkin MQMD ayarları

Kuyruk yöneticisi, bir aboneye bir yayın gönderdiğinde, bu alan ayarlarını ileti tanımlayıcısında (MQMD) kullanır. MQMD 'deki diğer tüm alanlar varsayılan değerlerine ayarlanır.

### Rapor

Rapor , MQRO\_NONE olarak ayarlıdır.

### MsgType

MsgType , MQMT\_DATAGRAM olarak ayarlıdır.

### Son kullanma tarihi

Süre Bitimi is set to the value in the Yayınla message received from the publisher. Alıkonan bir ileti olması durumunda, bekleyen süre, iletinin kuyruk yöneticisinde olduğu yaklaşık saatle azaltılır.

### Biçim

Biçim , MQFMT\_RF\_HEADER\_2olarak ayarlıdır.

### MsgId

MsgId , benzersiz bir değere ayarlıdır.

### CorrelId

CorrelId abonenin kimliğinin bir parçasıysa, bu değer, kayıt kaydedilirken abonenin belirlediği değerdir. Ters durumda, bu değer, kuyruk yöneticisi tarafından seçilmeyen sıfır dışında bir değerdir.

### Öncelik

Öncelik , yayınlayıcı tarafından ayarlanan değeri alır ya da yayınlayıcı tarafından belirtilen MQPRI\_PRIORİY\_AS\_Q\_DEF değerini belirlediyse çözülür.

### Kalıcılık

Persistence (Kalıcılık) yayıncı tarafından ayarlanan değeri alır ya da yayınlayıcı MQPER\_PERSISTENCE\_AS\_Q\_DEF belirtildiyse, bu yayının gönderilmekte olduğu aboneye ilişkin Aboneyi Kaydet iletisinde aksi belirtilmedikçe çözümlenir.

### ReplyToQ

ReplyToQ boşluk karakterlerine ayarlanır.

### ReplyToQMgr

ReplyToQMgr , kuyruk yöneticisinin adı olarak ayarlanır.

### UserIdentifier

UserIdentifier is the subscriber's user identifier, as set when the subscriber registered.

### AccountingToken

AccountingToken is the subscriber's accounting token, as set when the subscriber first registered.

### ApplIdentityVerileri

ApplIdentityVerileri is the subscriber's application identity data, as set when the subscriber first registered.

### PutApplTipi

PutApplTipi , MQAT\_BROKER değerine ayarlıdır.

### PutApplAdı

PutApplAd , kuyruk yöneticisi adının ilk 28 karakterine ayarlanır.

### PutDate

PutDate , iletinin konulduğu tarihtir.

### PutTime

PutTime is the time when the message was put.

### ApplOriginVerileri

ApplOriginVeriler boşluk olarak ayarlandı.

## Kuyruk yöneticisi yanıt iletilerinde MQMD ayarları

Kuyruk yöneticisi, bir yayın iletisine yanıt gönderirken ileti tanımlayıcısında (MQMD) alanların bu ayarlarını kullanır. MQMD 'deki diğer tüm alanlar varsayılan değerlerine ayarlanır.

**Rapor**

Rapor , tüm sıfırlara ayarlanır.

**MsgType**

MsgType , MQMT\_REPLY olarak ayarlanır.

**Biçim**

Biçim , MQFMT\_RF\_HEADER\_2olarak ayarlıdır.

**MsgId**

MsgId ayarı, özgün komut iletisinde Rapor seçeneklerinin belirlenmesine bağlıdır. Kuyruk yöneticisinin benzersiz bir değer oluşturabilmesi için, varsayılan olarak MQMI\_NONE değerine ayarlanır.

**CorrelId**

CorrelId ayarı, özgün komut iletisinde Report (Rapor) seçeneklerine bağlıdır. Varsayılan olarak bu, CorrelId değerinin, komut iletisinin MsgId ile aynı değere ayarlandığı anlamına gelir. Bu, komutları yanıtlarıyla ilintilendirmek için kullanılabilir.

**Öncelik**

Öncelik , özgün komut iletisinde olduğu gibi aynı değere ayarlıdır.

**Kalıcılık**

Persistence (Kalıcılık), özgün komut iletisinde ayarlanan değere ayarlanır.

**Son kullanma tarihi**

Süre Bitimi , kuyruk yöneticisi tarafından alınan özgün komut iletisinde olduğu gibi aynı değere ayarlıdır.

**PutApplTipi**

PutApplTipi , MQAT\_BROKER değerine ayarlıdır.

**PutApplAdı**

PutApplAd , kuyruk yöneticisi adının ilk 28 karakterine ayarlanır.

Diğer bağlam alanları, MQPMO\_PASS\_IDENTITY\_CONTEXT ile oluşturulduysa olarak ayarlanır.

## Makine kodlamaları

Bu bölümde, ileti tanımlayıcısındaki *Encoding* alanının yapısı açıklanmaktadır.

Yapıdaki alanların bir özeti için bkz. [“MQMD-İleti tanımlayıcı” sayfa 416](#) .

*Encoding* alanı, dört ayrı alt alana bölünen 32 bitlik bir tamsayıdır; bu alt alanlar şunlardır:

- İkili tamsayılar için kullanılan kodlama
- Paketlenmiş ondalık tamsayılar için kullanılan kodlama
- Kayan noktalı sayılar için kullanılan kodlama
- Ayrılmış bit

Her alt alan, alt alana karşılık gelen konumlarda 1 bit ve başka bir yerde 0 bit içeren bir bit maskesiyle tanımlanır. Bitler, 0 bit 'in en önemli bit, bit 31 'i ise en az anlamlı bit gibi numaralandırılmıştır. Aşağıdaki maskeleri tanımlıyor:

**MQENC\_INTEGER\_MASK**

İkili tamsayı kodlaması maskesi.

Bu alt alan, *Encoding* alanı içinde 28-31 arasında bit konumlarını kaplar.

**MQENC\_DECIMAL\_MASK**

Paketlenmiş ondalık-tamsayı kodlaması için maske.

Bu alt alan, *Encoding* alanı içinde 24-27 arasında bit konumlarını kaplar.

**MQENC\_FLOAT\_MASK**

Kayan noktalı kodlama için maske.

Bu alt alan, *Encoding* alanı içinde 20-23 arasında bit konumlarını kaplar.



## **MQENC\_RESERVE\_MASK**

Ayrılmış bitler için maske.

Bu alt alan, *Encoding* alanı içinde 0-19 arasındaki konumları bit olarak kaplar.

## **İkili tamsayı kodlaması**

İkili tamsayı kodlaması için geçerli olan değerler şunlardır:

### **MQENC\_INTEGER\_UNDEFIN**

İkili tamsayılar, tanımsız bir kodlama kullanılarak temsil edilir.

### **MQENC\_INTEGER\_NORMAL**

İkili tamsayılar geleneksel şekilde gösterilir:

- Sayıdaki en az önemli bayt, sayıdaki herhangi bir baytın en yüksek adresine sahiptir; en önemli bayt en düşük adrese sahiptir.
- Her bir baytta en az önemli bit, sonraki daha yüksek adresle bitişik olur; her bir baytın en önemli biti, bir sonraki alt adresle bitişik olan baytın bitişikidir.

### **MQENC\_INTEGER\_TERS**

İkili tamsayılar, MQENC\_INTEGER\_NORMAL ile aynı şekilde, ancak ters sırada düzenlenmiş byte 'lar ile gösterilir. Her bir baytın içindeki bitler, MQENC\_INTEGER\_NORMAL ile aynı şekilde düzenlenir.

## **Paketlenmiş-ondalık-tamsayı kodlaması**

Aşağıdaki değerler paketlenmiş ondalık-tamsayı kodlaması için geçerlidir:

### **MQENC\_DECIMAL\_UNDEFINED**

Paketlenmiş ondalık tamsayılar, tanımsız bir kodlama kullanılarak gösterilir.

### **MQENC\_DECIMAL\_NORMAL**

Paketlenmiş onlu tamsayılar geleneksel şekilde gösterilir:

- Sayının yazdırılabilir biçimindeki her bir ondalık basamağı, X' 0 'ile X' 9' aralığındaki tek bir onaltılı sayı ile paketlenmiş onlu sayıyla temsil edilir. Her onaltılı sayı dört biti kaplar ve paketlenmiş onlu sayıdaki her bayt, sayının yazdırılabilir biçiminde iki ondalık basamağı temsil eder.
- Paketli onlu sayıdaki en az önemli bayt, en az önemli ondalık basamağı içeren byte 'tır. Bu bayt içinde, en önemli dört bit, en az anlamlı ondalık basamak içerir ve en az önemli dört bit, işareti içerir. İşaret, X'C '(artı), X 'D' (negatif) ya da X'F ' (işaretsiz) olur.
- Sayıdaki en az önemli bayt, sayıdaki herhangi bir bayttan en yüksek adrese sahiptir; en önemli bayt en düşük adrese sahiptir.
- Her bir bayta ilişkin en az önemli bit, sonraki daha yüksek adresle bitişik baytın bitişikidir; her bir baytın en önemli biti, bir sonraki alt adresle bitişik baytın bitişikidir.

### **MQENC\_DECIMAL\_TERDI**

Paketlenmiş onlu tamsayılar, MQENC\_DECIMAL\_NORMAL ile aynı şekilde, ancak ters sırada düzenlenmiş byte 'lar ile gösterilir. Her bir baytın içindeki bitler, MQENC\_DECIMAL\_NORMAL ile aynı şekilde düzenlenir.

## **Kayan noktalı kodlama**

Kayan noktalı kodlama için geçerli olan değerler şunlardır:

### **MQENC\_FLOAT\_UNDEFIN**

Kayan noktalı sayılar, tanımlanmamış bir kodlama kullanılarak gösterilir.

### **MQENC\_FLOAT\_IEEE\_NORMAL**

Kayan noktalı sayılar, standart IEEE kullanılarak temsil edilir<sup>4</sup>Kayan noktalı biçim, aşağıda gösterildiği gibi, byte 'lar:

<sup>4</sup> Elektrik ve Elektronik Mühendisleri Enstitüsü

- mantissa 'da en az önemli bayt, sayıdaki herhangi bir baytın en yüksek adresine sahip; üsküdar' ı içeren bayt en düşük adrese sahip.
- Her bir baytta en az önemli bit, sonraki daha yüksek adresle bitişik olur; her bir baytın en önemli biti, bir sonraki alt adresle bitişik olan baytın bitişikidir.

IEEE kayar noktalı kodlama kodlamasının ayrıntıları IEEE Standard 754 'te bulunabilir.

### **MQENC\_FLOAT\_IEEEE\_TERS**

Kayan noktalı sayılar, MQENC\_FLOAT\_IEEE\_NORMAL ile aynı şekilde, ancak ters sırada düzenlenmiş byte 'lar ile gösterilir. Her bir baytın içindeki bitler, MQENC\_FLOAT\_IEEE\_NORMAL ile aynı şekilde düzenlenir.

### **MQENC\_FLOAT\_S390**

Kayan noktalı sayılar, standart System/390 kayan noktalı biçimi kullanılarak gösterilir; bu değer System/370 tarafından da kullanılır.

## **Kodlamalar oluşturuluyor**

MQMD ' de *Encoding* alanı için bir değer oluşturmak üzere, gerekli kodlamaları açıklayan ilgili sabitler bir araya eklenebilir (aynı değişmezi bir kereden fazla eklemeyin) ya da bit ya da işlemi kullanarak birleştirilebilir (programlama dili, bit işlemleri destekliyse).

Hangi yöntem kullanılırsa kullanılırsa, MQENC\_TEMIMAL\_ \* kodlamalarından biri ve MQENC\_FLOAT\_ \* kodlamalarından biri olan MQENC\_INTEGER\_ \* kodlamalarından yalnızca birini birleştirin.

## **Kodlamalar çözümleniyor**

*Encoding* alanı alt alanlar içerir; bunun nedeni, tamsayıyı, paketlenmiş onlu ya da kayar noktalı sayı kodlamasını incelemek için gereken uygulamaların, tanımlanan tekniklerden birini kullanması gerekir.

## **Bit işlemlerinin kullanılması**

Programlama dili, bit işlemlerini destekliyse, aşağıdaki adımları gerçekleştirin:

1. Gereken kodlama tipine göre, aşağıdaki değerlerden birini seçin:

- İkili tamsayı kodlamasına ilişkin MQENC\_INTEGER\_MASK
- Paketlenmiş onlu tamsayı kodlamasına ilişkin MQENC\_DECIMAL\_MASK
- Kayar noktalı kodlama için MQENC\_FLOAT\_MASK

Adeğerini arayın.

2. Combine the *Encoding* field with A using the bitwise AND operation; call the result B.

3. B , gereken kodlamadır ve bu kodlama tipi için geçerli olan her bir değer için eşitlik için test edilebilir.

## **Aritmetik kullanılması**

Programlama dili *bunu yapmaz* bit işlemleri destekliyse, tamsayı aritmetiği kullanarak aşağıdaki adımları gerçekleştirin:

1. Gereken kodlama tipine göre, aşağıdaki değerlerden birini seçin:

- İkili tamsayı kodlaması için 1
- Paketlenmiş onlu tamsayı kodlaması için 16
- 256 kayan noktalı kodlama için

Adeğerini arayın.

2. *Encoding* alanının değerini A ile bölün; Bsonucunu çağırın.

3. Divide B by 16; call the result C.

4. C ile 16 arasındaki çarpı ve B ' tan çıkarma; Dsonucunu çağırın.

5. Multiply D by A ; call the result E.
6. E , gereken kodlamadır ve bu kodlama tipi için geçerli olan her bir değerin her biriyle eşitlik için test edilebilir.

## Makine mimarisi kodlamalarının özeti

Makine mimarilerine ilişkin kodlamalar [Çizelge 632 sayfa 887](#) içinde gösterilir.

<i>Çizelge 632. Makine mimarilerine ilişkin kodlamaların özeti</i>			
<b>Makine mimarisi</b>	<b>İkili tamsayı kodlaması</b>	<b>Paketlenmiş onlu tamsayı kodlaması</b>	<b>Kayar noktalı kodlama</b>
IBM i	normal	normal	IEEE olağan
Intel x86	Tersine Çevrilmiş	Tersine Çevrilmiş	IEEE ters
PowerPC	normal	normal	IEEE olağan
System/390	normal	normal	System/390

## Rapor seçenekleri ve ileti işaretleri

Bu kısımda, MQGET, MQPUT ve MQPUT1 çağrılarında belirtilen MQMD ileti tanımlayıcısı MQMD ' nin bir parçası olan *Report* ve *MsgFlags* alanları açıklanmaktadır.

Bu kısımdaki konular şunlardır:

- Rapor alanının yapısı ve kuyruk yöneticisi tarafından nasıl işlem yapılma
- Bir uygulama, rapor alanını analiz eder
- İletinin yapısı-ışaretler alanı

MQMD ileti tanımlayıcısına ilişkin daha fazla bilgi için bkz. "[MQMD-İleti tanımlayıcı](#)" sayfa 416.

### Rapor alanının yapısı

Bu bilgiler, rapor alanının yapısını açıklar.

*Report* alanı, üç ayrı alt alana bölünen 32 bitlik bir tamsayıdır. Bu alt alanlar şunları tanımlar:

- Yerel kuyruk yöneticisi tarafından tanınmadığında reddedilen rapor seçenekleri
- Yerel kuyruk yöneticisi bunları tanımıyorsa bile her zaman kabul edilen rapor seçenekleri
- Yalnızca belirli bazı koşullar karşılanırsa kabul edilen rapor seçenekleri

Her alt alan, alt alana karşılık gelen konumlarda 1 bit ve başka bir yerde 0 bit içeren bir bit maskesiyle tanımlanır. Bir alt alandaki bitler tam olarak bitişik değildir. Bitler, 0 bit 'in en önemli bit, bit 31 'i ise en az anlamlı bit gibi numaralandırılmıştır. Alt alanları tanımlamak için aşağıdaki maskeleri tanımlar:

#### **MQRO\_REJECT\_UNSUP\_MASK**

Bu maske, yerel kuyruk yöneticisi tarafından desteklenmeyen rapor seçeneklerinin MQPUT ya da MQPUT1 çağrısının tamamlanma kodu MQCC\_FAILED ve neden kodu MQRC\_REPORT\_OPTIONS\_ERROR nedeniyle başarısız olmasına neden olan rapor seçeneklerinin bulunduğu *Report* alanı içindeki bit konumlarını tanıtır.

Bu alt alan, 3, 11 ile 13 arasında bit pozisyonları kaplıyor.

#### **MQRO\_ACCEPT\_UNSUP\_MASK**

This mask identifies the bit positions within the *Report* field where report options that are not supported by the local queue manager are nevertheless accepted on the MQPUT or MQPUT1 calls. Bu durumda, tamamlanma kodu MQCC\_UYARI, neden kodu MQRC\_UNKNOWN\_REPORT\_OPTION döndürüyor.

Bu alt alan, 0-2, 4-10 ve 24-31 arasında bit pozisyonları içerir.

Bu alt alana aşağıdaki rapor seçenekleri dahil edilir:

- MQRO\_ACTIVITY
- MQRO\_COPY\_MSG\_ID\_TO\_CORREL\_ID
- MQRO\_DEAD\_LETTER\_Q
- MQRO\_DISCARD\_MSG
- MQRO\_EXCEPTION
- MQRO\_EXCEPTION\_WITH\_DATA
- MQRO\_EXCEPTION\_WITH\_FULL\_DATA
- MQRO\_EXPIRATION
- MQRO\_EXPIRATION\_WITH\_DATA
- MQRO\_EXPIRATION\_WITH\_FULL\_DATA
- MQRO\_NAN
- MQRO\_NEW\_MSG\_ID
- MQRO\_NONE
- MQRO\_PAN
- MQRO\_PASS\_COREL\_ID
- MQRO\_PASS\_MSG\_ID

#### **MQRO\_ACCEPT\_UNSUP\_IF\_XMIT\_MASK**

This mask identifies the bit positions within the *Report* field where report options that are not supported by the local queue manager are nevertheless accepted on the MQPUT or MQPUT1 calls *sağlandı* that both of the following conditions are satisfied:

- İletinin kaderi uzak bir kuyruk yöneticisine ait.
- Uygulama iletiyi doğrudan yerel bir iletim kuyruğuna koymuyor (yani, MQOPER ya da MQPUT1 çağrısında belirlenen nesne tanımlayıcısındaki *ObjectQMGrName* ve *ObjectName* alanları tarafından tanımlanan kuyruk, yerel bir iletim kuyruğu değil).

Bu koşullar yerine getirilirse, MQRC\_UNKNOWN\_REPORT\_OPTION neden koduyla MQCC\_UYARI tamamlanma kodu döndürülür; değilse, MQRC\_REPORT\_OPTIONS\_ERROR neden koduyla MQCC\_BEKNIURD.

Bu alt alan 14-23 arasında bit pozisyonları kaplıyor.

Bu alt alana aşağıdaki rapor seçenekleri dahil edilir:

- MQRO\_COA
- MQRO\_CO\_WITH\_DATA
- MQRO\_COA\_WITHL\_FULL\_DATA
- MQRO\_COD
- MQRO\_COD\_WITH\_DATA
- MQRO\_COD\_WITH\_FULL\_DATA

If any options are specified in the *Report* field that the queue manager does not recognize, the queue manager checks each subfield in turn by using the bitwise AND operation to combine the *Report* field with the mask for that subfield. Bu işlemin sonucu sıfır değilse, daha önce açıklanan tamamlanma kodu ve neden kodları döndürülür.

MQCC\_UYARI döndürülürse, diğer uyarı koşulları varsa, bu neden kodlarının döndürüldüğü tanımlanmaz.

Yerel kuyruk yöneticisi tarafından tanınmayan rapor seçeneklerini belirleme ve kabul etme yeteneği, bir *uzak* kuyruk yöneticisi tarafından tanınan ve işlenen bir rapor seçeneği içeren bir ileti gönderirken yararlı olur.

## Rapor alanının çözümleniyor

*Report* alanı alt alanlar içerir; bunun nedeniyle, iletiyi gönderenin belirli bir rapor talep edip etmeyeceğini kontrol etmesi gereken uygulamalar, açıklanan tekniklerden birini kullanmalıdır.

### Bit işlemlerinin kullanılması

Programlama dili, bit işlemlerini destekliyorsa, aşağıdaki adımları gerçekleştirin:

1. Denetlenecek rapor tipine göre, aşağıdaki değerlerden birini seçin:

- COA raporu için MQRO\_COA\_WITHULL\_DATA
- COD raporu için MQRO\_COD\_WITH\_FULL\_DATA
- MQRO\_EXCEPTION\_WITH\_FULL\_DATA kural dışı durum raporu için
- MQRO\_EXPIRATION\_WITH\_FULL\_DATA süre sonu raporu

Adeğerini arayın.

z/OS üzerinde, MQRO\_\*\_WIT\_\_FULL\_DATA değerleri yerine MQRO\_\*\_WITH\_DATA değerlerini kullanın.

2. Combine the *Report* field with A using the bitwise AND operation; call the result B.

3. Bu rapor türü için mümkün olan her bir değerle eşitlik için B 'yi test edin.

For example, if A is MQRO\_EXCEPTION\_WITH\_FULL\_DATA, test B for equality with each of the following to determine what was specified by the sender of the message:

- MQRO\_NONE
- MQRO\_EXCEPTION
- MQRO\_EXCEPTION\_WITH\_DATA
- MQRO\_EXCEPTION\_WITH\_FULL\_DATA

Testler, uygulama mantığı için en uygun sırayla gerçekleştirilebilecek şekilde gerçekleştirilebilir.

MQRO\_PASS\_MSG\_ID ya da MQRO\_PASS\_COREL\_ID seçenekleri için sınama yapmak üzere benzer bir yöntem kullanın; bu iki değişmezden hangisi uygunsa, A değeri olarak seçin ve daha sonra açıklandığı gibi devam edin.

### Aritmetik kullanılması

Programlama dili *bunu yapmaz* bit işlemleri destekliyorsa, tamsayı aritmetiği kullanarak aşağıdaki adımları gerçekleştirin:

1. Denetlenecek rapor tipine göre, aşağıdaki değerlerden birini seçin:

- COA raporu için MQRO\_COA
- COD raporu için MQRO\_COD
- Kural dışı durum raporu için MQRO\_EXCEPTION
- MQRO\_EXPIRATION Süre Sonu Raporu

Adeğerini arayın.

2. *Report* alanını A ile bölün; Bsonucunu çağırın.

3. Divide B by 8 ; call the result C.

4. C 'u 8 ile çarpın ve B ' dan çıkarma; Dsonucunu çağırın.

5. Multiply D by A ; call the result E.

6. Bu rapor türü için mümkün olan her bir değerle eşitlik için E 'yi test edin.

Örneğin, A MQRO\_EXCEPTION ise, iletinin gönderici tarafından nelerin belirtildiğini saptamak için, aşağıdakilerin her biriyle eşitlik için E sınıadı:

- MQRO\_NONE
- MQRO\_EXCEPTION
- MQRO\_EXCEPTION\_WITH\_DATA
- MQRO\_EXCEPTION\_WITH\_FULL\_DATA

Testler, uygulama mantığı için en uygun sırayla gerçekleştirilebilecek şekilde gerçekleştirilebilir.

Aşağıdaki sözde kod, kural dışı durum raporu iletileri için bu tekniği gösterir:

```
A = MQRO_EXCEPTION
B = Report/A
C = B/8
D = B - C*8
E = D*A
```

Use a similar method to test for the MQRO\_PASS\_MSG\_ID or MQRO\_PASS\_CORREL\_ID options; select as the value A whichever of these two constants is appropriate, and then proceed as described previously, but replacing the value 8 in the previous steps by the value 2.

## İletinin yapısı-işaretler alanı

Bu bilgiler, ileti işaretlerinin yapısını açıklar.

*MsgFlags* alanı, üç ayrı alt alana bölünen 32 bitlik bir tamsayıdır. Bu alt alanlar şunları tanımlar:

- Yerel kuyruk yöneticisi tarafından tanınmadığında reddedilen ileti işaretleri
- Yerel kuyruk yöneticisi bunları tanımıyorsa bile, her zaman kabul edilen ileti işaretleri
- Yalnızca belirli bazı koşullar karşılanırsa kabul edilen ileti işaretleri

**Not:** All subfields in *MsgFlags* are reserved for use by the queue manager.

Her alt alan, alt alana karşılık gelen konumlarda 1 bit ve başka bir yerde 0 bit içeren bir bit maskesiyle tanımlanır. Bitler, 0 bit 'in en önemli bit, bit 31 'i ise en az anlamlı bit gibi numaralandırılmıştır. Alt alanları tanımlamak için aşağıdaki maskeleri tanımlar:

### MQMF\_REJECT\_UNSUP\_MASK

Bu maske, yerel kuyruk yöneticisi tarafından desteklenmeyen ileti işaretlerinin MQPUT ya da MQPUT1 çağrısının tamamlanma kodu MQCC\_FAILED ve neden kodu MQRC\_MSG\_FLAGS\_ERROR nedeniyle başarısız olmasına neden olan ileti işaretlerinin bulunduğu *MsgFlags* alanı içindeki bit konumlarını tanıtır.

Bu alt alan, 20 ile 31 arasında bit konumlarını kapsar.

Bu alt alana aşağıdaki ileti işaretleri dahil edilir:

- MQMF\_LAST\_MSG\_IN\_GROUP
- MQMF\_LAST\_SEGMENT
- MQMF\_MSG\_IN\_GROUP
- MQMF\_SEGMENT
- MQMF\_SEGMENTATION\_ALLOW
- MQMF\_SEGMENTATION\_ENGELLENDI

### MQMF\_ACCEPT\_UNSUP\_MASK

Bu maske, *MsgFlags* alanında, yerel kuyruk yöneticisi tarafından desteklenmeyen ileti işaretlerinin bulunduğu bit konumlarını tanıtır. Yine de, MQPUT ya da MQPUT1 çağrılarında kabul edilir. Tamamlanma kodu MQCC\_OK olur.

Bu alt alan 0 ile 11 arasında bit pozisyonları kaplıyor.

## MQMF\_ACCEPT\_UNSUP\_IF\_XMIT\_MASK

This mask identifies the bit positions within the *MsgFlags* field where message flags that are not supported by the local queue manager are nevertheless accepted on the MQPUT or MQPUT1 calls *sağlandı* that both of the following conditions are satisfied:

- İletin kaderi uzak bir kuyruk yöneticisine ait.
- Uygulama iletiyi doğrudan yerel bir iletim kuyruğuna koymuyor (yani, MQOPER ya da MQPUT1 çağrısında belirlenen nesne tanımlayıcısındaki *ObjectQMGrName* ve *ObjectName* alanları tarafından tanımlanan kuyruk, yerel bir iletim kuyruğu değil).

Bu koşullar karşılanırsa, tamamlanma kodu MQCC\_OK döndürülür ve MQCC\_FAILED neden kodu ile MQRC\_MSG\_FLAGS\_ERROR neden ile başarısız oldu.

Bu alt alan 12 ile 19 arasındaki pozisyonları kaplıyor.

*MsgFlags* alanında, kuyruk yöneticisinin tanımadığı işaretler varsa, kuyruk yöneticisi, *MsgFlags* alanını o alt alanın maskesiyle birleştirmek için bit ve işlemi kullanarak sırayla her alt alanı denetler. Bu işlemin sonucu sıfır değilse, daha önce açıklanan tamamlanma kodu ve neden kodları döndürülür.

## Veri dönüştürme çıkışı

Bu konu derlemi, veri dönüştürme çıkışıyla ilgili arabirimi ve veri dönüştürme gerektiğinde kuyruk yöneticisi tarafından gerçekleştirilen işlemleri açıklar.

Veri dönüştürme hakkında daha fazla bilgi için bkz. *Data Conversion under IBM MQ* ( <https://www.ibm.com/support/pages/node/317869>).

Uygulama ileti verilerini alan uygulamanın gerektirdiği gösteriye dönüştürmek için, MQGET çağrısının işlenmesinin bir parçası olarak, veri-dönüştürme çıkışı çağrılır. Uygulama ileti verilerinin dönüştürülmesi isteğe bağlıdır; MQGET çağrısında MQGMO\_CONVERTER seçeneğinin belirtilmesini gerektirir.

Aşağıdaki konular anlatılır:

- MQGMO\_CONVERT seçeneğine yanıt olarak kuyruk yöneticisi tarafından gerçekleştirilen işleme; bkz. [“Dönüştürme işlemi” sayfa 891](#).
- Yerleşik bir biçim işlenirken kuyruk yöneticisi tarafından kullanılan işleme kuralları; kullanıcı tarafından yazılan çıkışlar için de bu kurallar önerilir. Bkz. [“İşleme kuralları” sayfa 893](#).
- Rapor iletilerini dönüştürmeye ilişkin özel bilgiler; bkz. [“Rapor iletilerinin dönüştürülmesi” sayfa 896](#).
- Veri dönüştürme çıkışa aktarılan parametreler; bkz. [“MQ\\_DATA\\_CONV\\_EXIT-Veri dönüştürme çıkışı” sayfa 909](#).
- Karakter verilerini farklı gösterimler arasında dönüştürmek için çıkıştan kullanılacak bir çağrı; bkz. [“MQXCNVK-Karakterleri dönüştür” sayfa 903](#).
- Çıkışa özel veri yapısı parametresi; bkz. [“MQDXP-Data-conversion çıkış değiştirgesi” sayfa 897](#).

## Dönüştürme işlemi

Bu bilgiler, MQGMO\_CONVERT seçeneğine yanıt olarak kuyruk yöneticisi tarafından gerçekleştirilen işleme açıklar.

MQGET çağrısında MQGMO\_CONVERT seçeneği belirtilirse ve uygulamaya döndürülecek bir ileti varsa, kuyruk yöneticisi aşağıdaki işlemleri gerçekleştirir:

1. Aşağıdakilerden biri ya da birkaçı doğruysa, dönüştürme gerekli değildir:

- İleti verileri, MQGET çağrısını yayınlayan uygulamanın gerektirdiği karakter kümesiyle ve kodlamalarında zaten var. Uygulama, çağrıyı yayınlamadan önce, MQGET çağrısının **MsgDesc** parametresindeki *CodedCharSetId* ve *Encoding* alanlarını gerekli değerlere ayarlamalıdır.
- İleti verilerinin uzunluğu sıfır.
- MQGET çağrısına ilişkin **Buffer** değiştirgesinin uzunluğu sıfır.

Bu durumda, MQGET çağrısını yayınlayan uygulamaya dönüştürme yapılmadan ileti döndürülür; **MsgDesc** parametresindeki *CodedCharSetId* ve *Encoding* değerleri, iletteki denetim bilgilerindeki değerlere ayarlanır ve arama, aşağıdaki tamamlanma kodu ve neden kodlarından biriyle tamamlanır:

Çizelge 633. Tamamlanma kodu ve neden kodu birleşimleri

Tamamlanma kodu	Neden Kodu
MQCC_OK	MQRC_NONE
MQCC_UYARI	MQRC_TRUNCATED_MSG_KABUL edildi
MQCC_UYARI	MQRC_TRUNCATED_MSG_FAILED

Aşağıdaki adımlar yalnızca, ileti verilerinin karakter kümesi ya da kodlaması **MsgDesc** parametresindeki karşılık gelen değerden farklıysa ve dönüştürülecek veriler varsa gerçekleştirilir:

- İletideki denetim bilgilerindeki *Format* alanı MQFMT\_NONE değerine sahipse, ileti dönüştürülmez olarak döndürülür; tamamlanma kodu MQCC\_UYARI ve neden kodu MQRC\_FORMAT\_ERROR döndürülür.

Diğer tüm durumlarda dönüştürme işlemleri devam eder.

- İleti kuyruktan kaldırılır ve **Buffer** parametresiyle aynı boyutta olan geçici bir arabelleğe yerleştirilir. Göz atma işlemleri için, ileti kuyruktan kaldırılma yerine geçici arabelleğe kopyalanır.

- İletinin, arabelleğe sığması için kesilmesi gerekiyorsa, aşağıdaki yapılı:

- MQGMO\_ACCEPT\_TRUNCATED\_MSG seçeneği belirtilmediyse, ileti dönüştürülemedi, tamamlanma kodu MQCC\_UYARI ve neden kodu MQRC\_TRUNCATED\_MSG\_FAILED olur.
- If the MQGMO\_ACCEPT\_TRUNCATED\_MSG option : specified, the completion code is set to MQCC\_WARNING, the reason code is set to MQRC\_TRUNCATED\_MSG\_ACCEPTED, and conversion processing continues.

- İleti, arabellekte kesilmeden sığabiliyorsa ya da MQGMO\_ACCEPT\_TRUNCATED\_MSG seçeneği belirtildiyse, aşağıdaki yapılı:

- Biçim yerleşik bir biçimse, arabellek kuyruk yöneticisinin veri dönüştürme hizmetine geçirilir.
- Biçim yerleşik bir biçim değilse, arabellek, biçimle aynı adı taşıyan bir kullanıcı tarafından yazılan çıkışa geçirilir. Çıkış bulunamazsa, ileti dönüştürülemez olarak döndürülür; tamamlanma kodu MQCC\_UYARI ve neden kodu MQRC\_FORMAT\_ERROR döndürülür.

Hiçbir hata oluşmazsa, veri dönüştürme hizmetindeki ya da kullanıcı tarafından yazılan çıkıştan alınan çıkış, dönüştürülen iletidir ve MQGET çağrısını yayınlayan uygulamaya döndürülecek tamamlanma kodu ve neden kodu.

- Dönüştürme başarılı olursa, kuyruk yöneticisi dönüştürülen iletiyi uygulamaya geri döndürür. Bu durumda, MQGET çağrısının döndürdüğü tamamlanma kodu ve neden kodu, aşağıdaki birleşimlerden biridir:

Çizelge 634. Tamamlanma kodu ve neden kodu birleşimleri

Tamamlanma kodu	Neden Kodu
MQCC_OK	MQRC_NONE
MQCC_UYARI	MQRC_TRUNCATED_MSG_KABUL edildi

Ancak, dönüştürme, kullanıcı tarafından yazılan bir çıkış tarafından gerçekleştirilirse, dönüştürme başarılı olduğunda da diğer neden kodları geri döndürülebilir.

Dönüştürme başarısız olursa, kuyruk yöneticisi, **MsgDesc** parametresindeki *CodedCharSetId* ve *Encoding* alanları, iletteki denetim bilgilerindeki değerlere ve tamamlanma kodu MQCC\_UYARI ile uygulamaya dönüştürülmemiş iletiyi uygulamaya döndürür.



## İşleme kuralları

Yerleşik bir biçimi dönüştürürken, kuyruk yöneticisi açıklanan işleme kurallarını izler.

Bu durum, kuyruk yöneticisi tarafından zorlanmasa da, kullanıcı tarafından yazılan çıkışlar da bu kurallara uymalıdır. Kuyruk yöneticisi tarafından dönüştürülen yerleşik biçimler şunlardır:

- MQFMT\_ADMIN
- MQFMT\_CICS (yalnızca z/OS )
- MQFMT\_COMMAND\_1
- MQFMT\_COMMAND\_2
- MQFMT\_DEAD\_LETTER\_HEADER
- MQFMT\_DIST\_HEADER
- MQFMT\_EVENT sürüm 1
- MQFMT\_EVENT sürüm 2
- MQFMT\_IMS
- MQFMT\_IMS\_VAR\_STRING
- MQFMT\_MD\_EXTENSION
- MQFMT\_PCF
- MQFMT\_REF\_MSG\_HEADER
- MQFMT\_RF\_HEADER
- MQFMT\_RF\_HEADER\_2
- MQFMT\_STRING
- MQFMT\_TETIKLEYICISI
- MQFMT\_WORK\_INFO\_HEADER (yalnızca z/OS )
- MQFMT\_XMIT\_Q\_HEADER

1. İleti dönüştürme sırasında genişler ve **Buffer** parametresinin boyutunu aşarsa, aşağıdaki yapılır:

- MQGMO\_ACCEPT\_TRUNCATED\_MSG seçeneği belirtilmediyse, ileti, tamamlanma kodu MQCC\_UYARI ve neden kodu MQRC\_CONVERTED\_MSG\_TOO\_BIG ile dönüştürülemez olarak döndürülür.
- If the MQGMO\_ACCEPT\_TRUNCATED\_MSG option : specified, the message is truncated, the completion code is set to MQCC\_WARNING, the reason code is set to MQRC\_TRUNCATED\_MSG\_ACCEPTED, and conversion processing continues.

2. Kesme gerçekleşirse (dönüştürme öncesinde ya da dönüştürülürken), **Buffer** değıştirgesinde döndürülen geçerli byte sayısı arabelleğin uzunluğundan az olabilir.

Bu durum, örneğin, 4 baytlık bir tamsayı ya da bir DBCS karakter arabelleğindeki arabelleğin sonunu getirirse ortaya çıkabilir. Bilgilerin tamamlanmamış ögesi dönüştürülmedi ve döndürülen iletide bu byte 'lar geçerli bilgiler içermiyor. Dönüştürme sırasında dönüştürme daraltılmadan önce kesilmiş bir ileti varsa bu durum da oluşabilir.

Döndürülen geçerli byte sayısı arabelleğin uzunluğundan azsa, arabelleğin sonundaki kullanılmayan byte 'lar boş değere ayarlıdır.

3. Bir dizi ya da dizgi arabelleğin sonunu getirirse, olabildiğince çok veri dönüştürülür; yalnızca eksik olan belirli bir dizi ögesi ya da DBCS (DBCS) karakteri dönüştürülmez; önceki dizi öğeleri ya da karakterler dönüştürülür.

4. Kesme gerçekleşirse (dönüştürme sırasında ya da dönüştürülürken), **DataLength** parametresi için döndürülen uzunluk, dönüştürülen iletinin uzunluğunun kesilmeden önce uzunluğunun uzunluğunu sağlar.

5. Dizgiler, SBCS (tek baytlık karakter takımları), çift baytlık karakter takımları (DBCS) ya da çok baytlı karakter kümeleri (MBCS) arasında dönüştürüldüğünde, dizgiler genişletebilir ya da daraltabilir.

- PCF biçimlerinde, MQFMT\_ADMIN, MQFMT\_EVENT ve MQFMT\_PCF biçimlerinde, MQCFST ve MQCFSL yapılarındaki dizgiler, dönüştürmenin ardından dizgiyi sığdırmak için gereken şekilde genişletilecek ya da sözleşmeye başlanacak şekilde daraltabilir.

MQCFSL dizgi listesi yapısı için, listedeki dizgiler farklı miktarlara göre genişletebilir ya da daraltabilirler. Bu gerçekleşirse, kuyruk yöneticisi, dönüştürmeden sonraki en uzun dizgiyle aynı uzunlukta olacak şekilde, daha kısa dizgileri de boşluk karakteriyle destekler.

- In the format MQFMT\_REF\_MSG\_HEADER, the strings addressed by the SrcEnvOffset, SrcNameOffset, DestEnvOffset, and DestNameOffset fields expand or contract as necessary to accommodate the strings after conversion.
- In the format MQFMT\_RF\_HEADER, the NameValueString field expands or contracts as necessary to accommodate the name-value pairs after conversion.
- Değişmez alan büyüklükleriyle yapılan yapılarda, kuyruk yöneticisi önemli bir bilgi kaybolmaması koşuluyla, dizgilerin sabit alanları içinde genişletmesine ya da sözleşmeye izin verir. Bu bakımdan, alandaki ilk boş karakteri izleyen sondaki boşluk ve karakterler önemsiz olarak değerlendirilir.
  - Dizgi genişleirse, ancak yalnızca önemsiz karakterlerin, dönüştürülen dizgiyi alana sığdırmak için atılması gerekiyorsa, dönüştürme başarılı olur ve arama, MQCC\_OK ile tamamlanır ve neden kodu MQRC\_NONE (başka hata bulunmadığını varsayarlar).
  - Dizgi genişleirse, ancak dönüştürülen dizgi, alana sığması için önemli karakterler kullanılmasını gerektiriyorsa, ileti dönüştürülmez olarak döndürülür ve çağrı MQCC\_UYARI ile tamamlanır ve neden kodu MQRC\_CONVERTED\_STRING\_TOO\_BÜYÜK ile tamamlanır.
- **Not:** Neden kodu MQRC\_CONVERTED\_STRING\_TOO\_BÜYÜK, bu durumda MQGMO\_ACCEPT\_TRUNCATED\_MSG seçeneğinin belirlenip belirlenmemesine neden olur.
- Dizgi sözleşmeleri ise, kuyruk yöneticisi dizeyi, alanın uzunluğuna sahip olacak şekilde boşluklarla doldurur.

6. Kullanıcı verilerinin izlediği bir ya da daha çok MQ üstbilgisi yapısından oluşan iletiler için, bir ya da daha fazla başlık yapısı dönüştürülebilirken, iletinin geri kalan kısmı da dönüştürülemez. Ancak, her üstbilgi yapısındaki *CodedCharSetId* ve *Encoding* alanları, her zaman üstbilgi yapısından sonra gelen verilerin karakter kümesini ve kodlamasını doğru bir şekilde belirtir.

Bu iki kural dışı durum, MQCIH ve MQIIH yapılarıdır; bu yapılar, bu yapılardaki *CodedCharSetId* ve *Encoding* alanlarındaki değerlerin anlamlı olmamalarıdır. Bu yapılar için, yapıdan sonraki veriler, MQCIH ya da MQIIH yapısının kendisi ile aynı karakter kümesinde ve kodlamadır.

7. Alınmakta olan iletinin denetim bilgilerindeki *CodedCharSetId* ya da *Encoding* alanları ya da **MsgDesc** değiştirilmesinde, tanımsız ya da desteklenmeyen değerler belirtin; tanımsız ya da desteklenmeyen değerlerin ileti dönüştürülmesinde kullanılması gerekmiyorsa, kuyruk yöneticisi hatayı yoksayabilir.

Örneğin, iletteki *Encoding* alanı desteklenmeyen bir kayar noktalı sayı kodlamasını belirtiyorsa, ancak ileti yalnızca tamsayı verileri içeriyorsa ya da dönüştürme gerektirmeyen kayar noktalı veri içeriyorsa (kaynak ve hedef kayar noktalı kodlamalar özdeş olduğu için), hata saptanmayabilir.

Hata tanılınca, ileti dönüştürülmez olarak döndürülür; tamamlanma kodu MQCC\_UYARI ve MQRC\_SOURCE\_\*\_ERROR ya da MQRC\_TARGET\_\*\_ERROR neden kodlarından (uygun olduğu şekilde); **MsgDesc** parametresindeki *CodedCharSetId* ve *Encoding* alanları, iletteki denetim bilgilerindeki değerlere ayarlanır.

Hata saptanmazsa ve dönüştürme başarılı bir şekilde tamamlanırsa, **MsgDesc** parametresindeki *CodedCharSetId* ve *Encoding* alanlarında döndürülen değerler, MQGET çağrısını yayınlayan uygulama tarafından belirtilenlerdir.

8. Tüm durumlarda, ileti uygulamaya döndürülürse, tamamlanma kodu, tamamlanma kodu MQCC\_UYARI olarak ayarlanır ve **MsgDesc** parametresindeki *CodedCharSetId* ve *Encoding* alanları, dönüştürülenmemiş veriler için uygun değerlere ayarlanır. Bu, MQFMT\_NONE için de yapılır.

**Reason** parametresi, iletinin kesilmesini de gerekmediği sürece, dönüştürmenin neden gerçekleştirilemediğini gösteren bir kod olarak ayarlanır; kesme ile ilgili neden kodları, dönüştürme ile ilgili neden kodlarına göre önceliklidir. (Kesilen bir iletinin dönüştürülmüş olup olmadığını saptamak

için, **MsgDesc** parametresindeki *CodedCharSetId* ve *Encoding* alanlarında döndürülen değerleri denetleyin.)

Bir hata saptandığında, belirli bir neden kodu döndürülür ya da genel neden kodu MQRC\_NOT\_CONVERTED. Döndürülen neden kodu, temeldeki veri-dönüştürme hizmetinin tanılama yeteneklerine bağlıdır.

9. Tamamlanma kodu MQCC\_UYARI döndürülürse ve birden fazla neden kodu uygun ise, öncelik sırası aşağıdaki gibidir:

a. Aşağıdaki nedenler diğerlerinden öncelikli olarak ortaya çıkar; bu grubun nedenlerinden yalnızca biri ortaya çıkabilir:

- MQRC\_SIGNAL\_REQUEST\_ACCEPTED
- MQRC\_TRUNCATED\_MSG\_KABUL edildi

b. Geri kalan neden kodlarına göre öncelik sırası tanımlanmadı.

10. MQGET çağrısının tamamlandığı tarih:

• Aşağıdaki neden kodu, iletinin başarıyla dönüştürülmüş olduğunu gösterir:

– MQRC\_NONE

• The following reason codes indicate that the message *Olabilir* have been converted successfully (check the *CodedCharSetId* and *Encoding* fields in the **MsgDesc** parameter to find out):

– MQRC\_MSG\_MARKED\_BROWSE\_CO\_OP

– MQRC\_TRUNCATED\_MSG\_KABUL edildi

• Diğer tüm neden kodları, iletinin dönüştürülmediğini gösterir.

Aşağıdaki işlemler yerleşik biçimler için özgüdür; kullanıcı tanımlı biçimler için geçerli değildir:

11. Aşağıdaki biçimlerde kural dışı durum oluştu:

- MQFMT\_ADMIN
- MQFMT\_COMMAND\_1
- MQFMT\_COMMAND\_2
- MQFMT\_OLAY
- MQFMT\_IMS\_VAR\_STRING
- MQFMT\_PCF
- MQFMT\_STRING

Yerleşik biçimlerin hiçbiri, kuyruk adlarında geçerli olan karakterler için SBCS karakterleri olmayan karakter kümelerinden ya da bu biçimlerden dönüştürülemez. Böyle bir dönüştürmeyi gerçekleştirmek için bir girişimde bulunulursa, ileti dönüştürülmez olarak döndürülür; tamamlanma kodu MQCC\_UYARIL ve neden kodu MQRC\_SOURCE\_CCSD\_ERROR ya da MQRC\_TARGET\_CCSD\_ERROR.

Unicode karakter takımı UTF-16 , kuyruk adlarında geçerli olan karakterler için SBCS karakterleri olmayan bir karakter takımı örneğidir.

12. If the message data for a built-in format is truncated, fields within the message that contain lengths of strings, or counts of elements or structures, are not adjusted to reflect the length of the data actually returned to the application; the values returned for such fields within the message data are the values applicable to the message *kesmeden önce*.

Kısaltılmış bir MQFMT\_ADMIN iletisi gibi iletileri işlerken, uygulamanın döndürülen verilerin sonuna kadar verilere erişme girişiminde bulunmayacağına dikkat edin.

13. Biçim adı MQFMT\_DEAD\_LETTER\_HEADER ise, ileti verileri bir MQDLH yapısıyla başlar ve büyük olasılıkla sıfır ya da daha fazla sayıda uygulama iletisi verisi izlemektedir. Uygulama iletisi verilerinin biçimi, karakter kümesi ve kodlaması, iletinin başlangıcındaki MQDLH yapısındaki Format, CodedCharSetId ve Encoding alanları tarafından tanımlanır. MQDLH yapısı ve uygulama iletisi

verilerinin farklı karakter kümeleri ve kodlamaları, bir, diğer ya da her ikisi de MQDLH yapısı ve uygulama iletisi verilerinin dönüştürme gerektirebileceği.

Kuyruk yöneticisi, önce MQDLH yapısını gerektiği gibi dönüştürür. Dönüştürme başarılı olursa ya da MQDLH yapısı dönüştürme gerektirmiyorsa, kuyruk yöneticisi, MQDLH yapısındaki CodedCharSetId ve Encoding alanlarını, uygulama iletisi verilerinin dönüştürülmesinin gerekli olup olmadığını görmek için denetler. Dönüştürme gerekiyorsa, kuyruk yöneticisi kullanıcı tarafından yazılan çıkışı, MQDLH yapısındaki Format alanı tarafından belirtilen adla çağırır ya da dönüştürmeyi gerçekleştirir (Format, yerleşik bir biçimin adıdır).

MQGET çağrısı, MQCC\_UYARI için bir tamamlanma kodu döndürürse ve neden kodu, dönüştürmenin başarısız olduğunu gösteren kodlardan biriye, aşağıdakilerden biri geçerli olur:

- MQDLH yapısı dönüştürülemedi. Bu durumda, uygulama iletisi verileri de dönüştürülmez.
- MQDLH yapısı dönüştürüldü, ancak uygulama iletisi verileri dönüştürülmedi.

The application can examine the values returned in the CodedCharSetId and Encoding fields in the **MsgDesc** parameter, and those in the MQDLH structure, in order to determine which of the previously applies.

14. Biçim adı MQFMT\_XMIT\_Q\_HEADER ise, ileti verileri bir MQXQH yapısıyla başlar; bu yapıyla birlikte, sıfır ya da daha fazla bayt ek veri izlemektedir. Bu ek veriler genellikle uygulama iletisi verileridir (sıfır uzunluklu olabilir), ancak ek verilerin başında, bir ya da daha fazla MQ üstbilgi yapısı var.

MQXQH yapısı, kuyruk yöneticisinin karakter kümesinde ve kodlamasında olmalıdır. MQXQH yapısını izleyen verilerin biçimi, karakter kümesi ve kodlaması, MQXQH içinde yer alan MQMD yapısındaki Format, CodedCharSetId ve Encoding alanları tarafından verilir. Sonraki her MQ üstbilgi yapısı için, yapıdaki Format, CodedCharSetId ve Encoding alanları, o yapıyı izleyen verileri tanımlar; bu veriler başka bir MQ üstbilgi yapısı ya da uygulama iletisi veridir.

MQFMT\_XMIT\_Q\_HEADER iletisi için MQGMO\_CONVERT seçeneği belirtilirse, uygulama iletisi verileri ve MQ üstbilgi yapılarının belirli bir değeri dönüştürülür *ancak, MQXQH yapısındaki veriler dönüştürülür*. Bu nedenle, MQGET çağrısından geri dönün:

- **MsgDesc** parametresindeki Format, CodedCharSetId ve Encoding alanlarının değerleri, uygulama iletisi verilerinde değil, MQXQH yapısındaki verileri tanımlar; dolayısıyla, bu değerler MQGET çağrısını yayınlayan uygulama tarafından belirtilenler ile aynı değildir.

The effect of this is that an application that repeatedly gets messages from a transmission queue with the MQGMO\_CONVERT option specified must reset the CodedCharSetId and Encoding fields in the **MsgDesc** parameter to the values required for the application message data, before each MQGET call.

- Son MQ üstbilgi yapısındaki Format, CodedCharSetId ve Encoding alanlarının değerleri, uygulama iletisi verilerini açıklar. Başka bir MQ üstbilgi yapısı yoksa, uygulama iletisi verileri bu alanlar tarafından MQXQH yapısındaki MQMD yapısıyla tanımlanır. Dönüştürme başarılı olursa, değerler, MQGET çağrısını yayınlayan uygulama tarafından **MsgDesc** parametresindeki belirtilerle aynı olur.

İleti bir dağıtım listesi iletirse, MQXQH yapısının ardından bir MQDH yapısı (artı MQOR ve MQPMR kayıtları dizileri) gelir ve bu, sıfır ya da daha fazla ek MQ üstbilgi yapısıyla ve sıfır ya da daha fazla sayıda uygulama iletisi verisi içeren bir MQDH yapısı tarafından izlenebilir. MQXQH yapısı gibi, MQDH yapısı kuyruk yöneticisinin karakter kümesinde ve kodlamasında olmalı ve MQGET çağrısına dönüştürülmez, ancak MQGMO\_CONVERT seçeneği belirtilmişse bile bu işlem MQGET çağrısına dönüştürülmez.

Daha önce açıklanan MQXQH ve MQDH yapılarının işlenmesi öncelikli olarak ileti kanalı araçları tarafından iletim kuyruklarından ileti aldıklarında kullanılmak üzere tasarlanmıştır.

## Rapor iletilerinin dönüştürülmesi

Genel olarak bir rapor iletisi, özgün iletinin göndericisinin belirttiği rapor seçeneklerine göre değişen miktarlarda uygulama iletisi verisi içerebilir. Ancak, bir etkinlik raporu veri içerebilir, ancak değişmez olarak \* \_WITH\_DATA ' yı belirtmeden önce rapor seçeneği yoktur.

Belirli bir durumda, bir rapor iletisi aşağıdaki gibi olabilir:

1. Uygulama iletisi verisi yok
2. Özgün iletiden bazı uygulama iletisi verileri

Bu durum, özgün iletiyi gönderen MQRO\_\*\_WITH\_DATA belirttiğinde ve iletinin 100 byte 'tan uzun olduğunu belirtir.

3. Özgün iletiden gelen tüm uygulama iletisi verileri

Bu durum, özgün iletiyi gönderen MQRO\_\*\_WITH\_FULL\_DATA belirtildiğinde ya da MQRO\_\*\_WITH\_DATA belirtildiğinde ve iletinin 100 byte ya da daha kısa olduğunu belirttiğinde ortaya çıkar.

When the queue manager or message channel agent generates a report message, it copies the format name from the original message into the *Format* field in the control information in the report message. Bu nedenle, rapor iletisinde biçim adı, rapor iletisinde gerçekte var olan uzunluktan farklı bir veri uzunluğu anlamına gelebilir (önceki durumlar 1 ve 2.).

Rapor iletisi alındığında MQGMO\_CONVERT seçeneği belirtilirse:

- Daha önce 1 numaralı vaka için, veri dönüştürme çıkışı çağrılmaz (rapor iletisine veri girmediğinden).
- Daha önce 3. vaka için, biçim adı ileti verilerinin uzunluğunu doğru biçimde ifade eder.
- Ancak, daha önce 2. vaka için, veri dönüştürme çıkışı, biçim adının belirttiği uzunluktan *daha kısa* olan bir iletiyi dönüştürmek için çağrılır.

Buna ek olarak, çıkışa geçirilen neden kodu genellikle MQRC\_NONE olur (yani, neden kodu iletinin kesildiğini belirtmez). Bu durum, ileti verilerinin, rapor iletisinin *gönderen* tarafından kesildiğinden ve MQGET çağrısına yanıt olarak alıcının kuyruk yöneticisi tarafından kesilmediği için ortaya çıktı.

Bu olasılıklardan dolayı, veri-dönüştürme çıkışı, aktarılan verilerin uzunluğunu nasıl algılayabilmek için biçim adını kullanmamalıdır; bunun yerine, çıkış, sağlanan verilerin uzunluğunu denetlememeli ve biçim adının belirttiği uzunluktan daha az veri dönüştürmeye hazır olmalıdır. Veriler başarılı bir şekilde dönüştürülebiliyorsa, tamamlanma kodu MQCC\_OK ve neden kodu MQRC\_NONE çıkış tarafından döndürülmelidir. The length of the message data to be converted is passed to the exit as the **InBufferLength** parameter.

## Ürüne duyarlı programlama arabirimi

### MQDXP-Data-conversion çıkış değiştirgesi

MQDXP yapısı, kuyruk yöneticisinin, ileti verilerini MQGET çağrısının işlenmesinin bir parçası olarak dönüştürmek için çıkış çağrıldığında veri dönüştürme çıkışa geçtiği bir parametredir. Veri dönüştürme çıkışa ilişkin ayrıntılar için MQ\_DATA\_CONV\_EXIT çağrısının açıklamasına bakın.

MQDXP ' deki karakter verileri, yerel kuyruk yöneticisinin karakter takımında yer alıyor; bu, **CodedCharSetId** kuyruk yöneticisi özneliği tarafından verilir. MQDXP ' deki sayısal veriler yerel makine kodlamasıdır; bu, MQENC\_NATIVE tarafından verilir.

MQDXP ' de yalnızca *DataLength*, *CompCode*, *Reason* ve *ExitResponse* alanları, çıkışta değiştirilebilir; diğer alanlardaki değişiklikler yoksaılır. Ancak dönüştürülmekte olan ileti, mantıksal iletinin yalnızca bir kısmını içeren bir bölümse, *DataLength* alanı değiştirilemez.

Denetim çıkış kuyruğundan kuyruk yöneticisine geri döndüğünde, kuyruk yöneticisi MQDXP ' de döndürülen değerleri denetler. If the values returned are not valid, the queue manager continues processing as though the exit had returned MQXDR\_CONVERSION\_FAILED in *ExitResponse* ; however, the queue manager ignores the values of the *CompCode* and *Reason* fields returned by the exit in this case, and uses instead the values those fields had on *giriş* to the exit. MQDXP ' de aşağıdaki değerler, bu işlemin gerçekleşmesine neden olur:

- *ExitResponse* alanı MQXDR\_OK değil ve MQXDR\_CONVERSION\_FAILED değil
- *CompCode* alanı MQCC\_OK değil ve MQCC\_UYARI değil

- Dönüştürülmekte olan ileti bir mantıksal iletinin yalnızca bir kısmını içeren bir bölümse, *DataLength* alanı sıfırdan küçük ya da *DataLength* alanı değişmektedir.

Aşağıdaki çizelge yapıdaki alanları özetlemektedir.

Çizelge 635. MQDXP ' deki alanlar		
Alan	Tanım	Konu
<i>StrucId</i>	Yapı tanıtıcısı	<a href="#">StrucId</a>
<i>Version</i>	Yapı sürüm numarası	<a href="#">S\u00fcr\u00fcm</a>
<i>AppOptions</i>	Uygulama seçenekleri	<a href="#">AppOptions</a>
<i>Encoding</i>	Uygulama için sayısal kodlama gerekiyor	<a href="#">Kodlama</a>
<i>CodedCharSetId</i>	Uygulama için karakter kümesi gerekli	<a href="#">CodedCharSetId</a>
<i>DataLength</i>	İleti verilerinin bayt cinsinden uzunluğu	<a href="#">DataLength</a>
<i>CompCode</i>	Tamamlanma kodu	<a href="#">CompCode</a>
<i>Reason</i>	Neden kodu ön eleme <i>CompCode</i>	<a href="#">Neden</a>
<i>ExitResponse</i>	Çıkıştan yanıt	<a href="#">ExitResponse</a>
<i>Hconn</i>	Bağlantı tanıtıcı	<a href="#">Hconn</a>
<i>pEntryPoints</i>	MQIEP yapısının adresi	<a href="#">pEntryNoktaları</a>

## Alanlar

MQDXP yapısı aşağıdaki alanları içerir; alanlar alfabetik sırayla anlatılır.

### AppOptions

Tip: MQUZE

Bu, MQGET çağrısını yayınlayan uygulama tarafından belirtilen MQGMO yapısının *Options* alanının bir kopyasıdır. Çıkışta, MQGMO\_ACCEPT\_TRUNCATED\_MSG seçeneğinin belirlenip belirlenmediğini saptamak için bunları incelemeniz gerekebilir.

Bu, çıkışa giriş alanıdır.

### CodedCharSetId

Tip: MQUZE

Bu, MQGET çağrısını yayınlayan uygulama tarafından gerekli olan karakter kümesinin kodlanmış karakter takımı tanıtıcısıdır; ek ayrıntılar için MQMD yapısındaki *CodedCharSetId* alanına bakın. Uygulama, MQGET çağrısında MQCCSI\_Q\_MGR özel değerini belirtiyorsa, kuyruk yöneticisi bu değeri, çıkış çağrılmadan önce kuyruk yöneticisi tarafından kullanılan karakter kümesinin gerçek karakter kümesi tanıtıcısında değiştirir.

Dönüştürme başarılı olursa, çıkışta bu değeri ileti tanımlayıcısındaki *CodedCharSetId* alanına kopyalaması gerekir.

Bu, çıkışa giriş alanıdır.

### CompCode

Tip: MQUZE

Çıkış çağrıldığında, çıkışta hiçbir şey yapmazsa, MQGET çağrısını yayınlayan uygulamaya döndürülen tamamlanma kodunu içerir. Her zaman MQCC\_UYARI olur; çünkü ya ileti kesildi ya da ileti dönüştürme gerektiriyor ve bu henüz yapılamadı.

Çıkıştan çıkışta, bu alan MQGET çağrısının **CompCode** parametresindeki uygulamaya döndürülebilme için tamamlanma kodunu içerir; yalnızca MQCC\_OK ve MQCC\_UYARI geçerli olur. Çıkışta bu alanı nasıl ayarlayabileceğiyle ilgili öneriler için *Reason* alanının açıklamasına bakın.

Bu, çıkışa giriş/çıkış alanıdır.

### **DataLength**

Tip: MQUZE

Çıkış çağrıldığında, bu alanda uygulama iletisi verilerinin özgün uzunluğu yer alır. İleti, uygulama tarafından sağlanan arabelleğe sığması için kesildiyse, çıkışa sağlanan iletinin boyutu, *DataLength* değerinden *daha küçük* değerine sahip olur. Çıkışa sağlanan iletinin boyutu, her zaman çıkışa ilişkin **InBufferLength** parametresi tarafından verilir; bu parametre, oluşan kesilmeden bağımsız olarak verilir.

Kesme, çıkışa girişte MQRC\_TRUNCATED\_MSG\_ACCEPTED değerinin kabul edildiği *Reason* alanı tarafından gösterilir.

Çoğu dönüştürmenin bu uzunluğu değiştirmesi gerekmez, ancak gerekirse bir çıkış işlemi yapabilir; çıkışa göre ayarlanan değer, MQGET çağrısının **DataLength** parametresindeki uygulamaya döndürülür. Ancak, dönüştürülmekte olan ileti, mantıksal iletinin yalnızca bir kısmını içeren bir bölümse bu uzunluk değiştirilemez. Bunun nedeni, uzunluğun değiştirilmesi, mantıksal iletinin sonraki bölümlerinin yanlış olarak belirlenmesine neden olur.

Çıkış, verilerin uzunluğunu değiştirmek isterse, kuyruk yöneticisinin *dönüştürülemeyen* verilerin uzunluğuna dayalı olarak, ileti verilerinin uygulamanın arabelleğiyle uyup uymadığını önceden kararlaştıracağını unutmayın. Bu karar, iletinin kuyruktan kaldırılıp kaldırılmayacağını (ya da bir göz atma isteği için imleci hareket ettirerek) belirler ve dönüştürmenin neden olduğu veri uzunluğunda yapılan herhangi bir değişiklikten etkilenmez. Bu nedenle, dönüştürme çıkışlarının, uygulama iletisi verilerinin uzunluğunda bir değişikliğe neden olmamasını önermektedir.

Karakter dönüştürmesi bir uzunluk değişikliği anlamına geldiyse, bir dizgi byte olarak aynı uzunluğa sahip başka bir dizgiye dönüştürülebilir, sondaki boşlukları kesilebilir ya da gerektiği gibi boşluklarla doldurulabilir.

İleti herhangi bir uygulama iletisi verisi içermiyorsa çıkış çağrılmaz; *DataLength* her zaman sıfırdan büyük olur.

Bu, çıkışa giriş/çıkış alanıdır.

### **Encoding**

Tip: MQUZE

Uygulama için sayısal kodlama gerekiyor.

Bu, MQGET çağrısını yayınlayan uygulamanın gerektirdiği sayısal kodlamadır; ek ayrıntılar için MQMD yapısındaki *Encoding* alanına bakın.

Dönüştürme başarılı olursa, çıkış bunu ileti tanımlayıcısındaki *Encoding* alanına kopyalar.

Bu, çıkışa giriş alanıdır.

### **ExitOptions**

Tip: MQUZE

Bu ayrılmış bir alandır; değeri 0' dir.

### **ExitResponse**

Tip: MQUZE

Çıkıştan yanıt. Bu, başarıyı ya da dönüştürmenin başka bir şekilde belirtilmesini sağlamak için çıkışa göre ayarlanır. Aşağıdakilerden biri olmalıdır:

## **MQXDR\_OK**

Dönüştürme başarılı oldu.

Çıkış bu değeri belirtiyorsa, kuyruk yöneticisi, MQGET çağrısını yayınlayan uygulamaya aşağıdaki gibi döner:

- Çıkışta çıkışta bulunan *CompCode* alanının değeri
- Çıkışta çıkışta bulunan *Reason* alanının değeri
- Çıkışta çıkışta bulunan *DataLength* alanının değeri
- The contents of the exit's output buffer *OutBuffer*. Döndürülen bayt sayısı, çıkışın **OutBufferLength** parametresinden daha düşük ve çıkışta çıkış üzerindeki *DataLength* alanının değeri olur.

If the *Encoding* and *CodedCharSetId* fields in the exit's message descriptor parameter are *her ikisi* unchanged, the queue manager returns:

- Çıkışa *girişte* ilişkin MQDXP yapısındaki *Encoding* ve *CodedCharSetId* alanlarının değeri.  
Çıkışta, ileti tanımlayıcısı değiştirilmesindeki *Encoding* ve *CodedCharSetId* alanlarının biri ya da her ikisi değiştirildiyse, kuyruk yöneticisi şunları döndürür:
- Çıkıştaki çıkışta çıkış iletisi tanımlayıcısı parametresindeki *Encoding* ve *CodedCharSetId* alanlarının değeri

## **MQXDR\_CONVERSION\_FAILED**

Dönüştürme başarısız oldu.

Çıkış bu değeri belirtiyorsa, kuyruk yöneticisi, MQGET çağrısını yayınlayan uygulamaya aşağıdaki gibi döner:

- Çıkışta çıkışta bulunan *CompCode* alanının değeri
- Çıkışta çıkışta bulunan *Reason* alanının değeri
- Çıkışa *girişte* ilişkin *DataLength* alanının değeri
- The contents of the exit's input buffer *InBuffer*. Döndürülen bayt sayısı **InBufferLength** parametresiyle verilir.

Çıkış *InBuffer*değiştirildiyse, sonuçlar tanımsız olur.

*ExitResponse* , çıkışa ait bir çıkış alanıdır.

## **Hconn**

Tip: MQHCONN

Bu, MQXCNVC çağrısında kullanılabilecek bir bağlantı tanıtıcısıdır. Bu tanıtıcı değeri, MQGET çağrısını yayınlayan uygulama tarafından belirlenen tanıtıcı ile aynı değil.

## **pEntryPoints**

Tip: PMQIEP

MQI ve DCI çağrılarının yapılabileceği MQIEP yapısının adresi.

## **Reason**

Tip: MQUZE

Neden kodu ön eleme *CompCode*.

Çıkış çağrıldığında, bu çıkış MQGET çağrısını yayınlayan uygulamaya döndürülen neden kodunu (çıkış hiçbir şey yapmayı seçerse) içerir. Olası değerler arasında MQRC\_TRUNCATED\_MSG\_ACCEPTED, iletinin, uygulama tarafından sağlanan arabelleğe sığması için kesildiğini ve MQRC\_NOT\_CONVERTED iletisinin, iletinin dönüştürülmesini gerektirdiğini, ancak bunun henüz gerçekleştirilmediğini gösterir.

Çıkışta, bu alan, MQGET çağrısının **Reason** parametresindeki uygulamaya döndürülmesinin nedenini içerir; aşağıdaki durum önerilir:

- Çıkışa *girişte* MQRC\_TRUNCATED\_MSG\_ACCEPTED değeri *Reason* ise, dönüştürmenin başarılı olup olmadığına bakılmaksızın *Reason* ve *CompCode* alanlarının değiştirilmemesi gerekir.



( *CompCode* alanı MÖCC\_OK değilse, iletiyi alan uygulama, ileti tanımlayıcısındaki döndürülen *Encoding* ve *CodedCharSetId* değerlerini istenen değerlerle karşılaştırarak bir dönüştürme hatasını tanımlayabilir; buna karşılık, uygulama, kesilen bir iletiyi, arabelleği yerleştiren bir iletiden ayırt edemez. Bu nedenle, dönüştürme başarısızlığını belirten herhangi bir nedenden dolayı, MÖRC\_TRUNCATED\_MSG\_ACCEPTED döndürülmelidir.

• *Reason* ' un çıkışa giriş üzerinde başka bir değeri varsa:

- Dönüştürme başarılı olursa, *CompCode* MÖCC\_OK değerine ayarlanmalıdır ve *Reason* MÖRC\_NONE olarak ayarlanmalıdır.
- Dönüştürme başarısız olursa ya da iletinin genişlemesi ve arabelleğe sığması için kesilmesi gerekiyorsa, *CompCode* MÖCC\_UYARI (ya da değiştirilmeden sola) değerine ayarlanmalı ve *Reason* , başarısızlığın doğasını göstermek için listelenen değerlerden birine ayarlanmalıdır.

Dönüştürmenin ardından gelen iletinin arabellek için çok büyük olduğunu unutmayın; ancak, MÖGET çağrısını yayınlayan uygulama MÖGMO\_ACCEPT\_TRUNCATED\_MSG seçeneğini belirttiyse kesilmelidir:

- Bu seçenek belirtilirse, MÖRC\_TRUNCATED\_MSG\_ACCEPTED neden döndürülmedi.
- Bu seçenek belirlenmezse, ileti dönüştürülmeden geri döndürülür; neden kodu MÖRC\_CONVERTED\_MSG\_TOO\_BÜYÜK değeri döndürülür.

Dönüştürmenin başarısız olmasının nedenini belirtmek için, listelenen neden kodları, çıkışta kullanılması önerilir; ancak, çıkış, uygun görüldüğünde MÖRC\_\* kodları kümesinden başka değerler döndürebilir. Ayrıca, çıkışta MÖGET çağrısını veren uygulamayla iletişim kurmak istediği koşulları belirtmek için çıkış tarafından MÖRC\_APPL\_FIRST ile MÖRC\_APPL\_FIRST değerleri ayrılarak kullanılır.

**Not:** İleti başarıyla dönüştürülemezse, kuyruk yöneticisinin dönüştürülenmemiş iletiyi döndürmesi için, çıkış *ExitResponse* alanında MÖXDR\_CONVERSION\_FAILED değerini döndürmelidir. Bu, *Reason* alanında döndürülen neden kodundan bağımsız olarak geçerlidir.

#### **MÖRC\_APPL\_FIRST**

(900, X'384 ') Uygulama tanımlı neden kodu için En Düşük değeri.

#### **MÖRC\_APPL\_SON**

(999, X'3E7') Uygulama tanımlı neden kodlarına ilişkin en yüksek değer.

#### **MÖRC\_CONVERTED\_MSG\_TOO\_BÜYÜK**

(2120, X'848 ') Dönüştürülen veriler arabellek için çok büyük.

#### **MÖRC\_NOT\_CONVERTED**

(2119, X'847 ') İleti verileri dönüştürülmedi.

#### **MÖRC\_SOURCE\_CCSD\_ERROR**

(2111, X'83F') Kaynak kodlanmış karakter takımı tanıtıcısı geçerli değil.

#### **MÖRC\_SOURCE\_DECIMAL\_ENC\_ERROR**

(2113, X'841 ') İletide paketli ondalık kodlama tanınmadı.

#### **MÖRC\_SOURCE\_FLOATENC\_ERROR**

(2114, X'842 ') İletide kayan noktalı kodlama tanınmadı.

#### **MÖRC\_SOURCE\_INTEGER\_ENC\_ERROR**

(2112, X'840 ') Kaynak tamsayı kodlaması tanınmadı.

#### **MÖRC\_TARGET\_CCSD\_HATASI**

(2115, X'843 ') Hedef kodlanmış karakter takımı tanıtıcısı geçerli değil.

#### **MÖRC\_TARGET\_DECIMALIM\_ERROR**

(2117, X'845 ') Packed-receiver tarafından belirtilen paket-ondalık kodlama tanınmıyor.

#### **MÖRC\_TARGET\_FLOATENC\_ERROR**

(2118, X'846 ') Alıcı tarafından belirtilen kayan noktalı kodlama tanınmıyor.

#### **MÖRC\_TARGET\_INTEGER\_ENC\_ERROR**

(2116, X'844 ') Hedef tamsayı kodlaması tanınmadı.

#### **MÖRC\_TRUNCATED\_MSG\_KABUL edildi**

(2079, X'81F') Kesilen ileti döndürüldü (işleme tamamlandı).

Bu, çıkışa giriş/çıkış alanıdır.

### StrucId

Tip: MQCHAR4

Yapı tanıtıcısı.Değer şu olmalıdır:

### MQDXP\_STRUC\_ID

Veri dönüştürme çıkış parametresi yapısıyla ilgili tanıtıcı.

C programlama dili için, sabit MQDXP\_STRUC\_ID\_ARRAY değişmezi de tanımlanır; bu, MQDXP\_STRUC\_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisi olur.

Bu, çıkışa giriş alanıdır.

### Version

Tip: MQUZE

Yapı sürüm numarası.Değer şu olmalıdır:

### MQDXP\_VERSION\_1

Veri dönüştürme çıkış parametresi yapısı için sürüm numarası.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

### MQDXP\_CURRENT\_VERSION

Veri dönüştürme çıkış değıştirgesi yapısının yürürlükteki sürümü.

**Not:** Bu yapının yeni bir sürümü kullanıma sunulduğunda, var olan parçanın düzeni değıştirilmez. Bu nedenle, çıkışta, *Version* alanının, çıkışa gereken alanları içeren en düşük sürüme eşit ya da bu sürümden büyük olduğunu denetlemeniz gerekir.

Bu, çıkışa giriş alanıdır.

## C bildirim

```
typedef struct tagMQDXP MQDXP;
struct tagMQDXP {
    MQCHAR4  StrucId;           /* Structure identifier */
    MQLONG   Version;          /* Structure version number */
    MQLONG   ExitOptions;      /* Reserved */
    MQLONG   AppOptions;       /* Application options */
    MQLONG   Encoding;         /* Numeric encoding required by
                               application */
    MQLONG   CodedCharSetId;   /* Character set required by application */
    MQLONG   DataLength;       /* Length in bytes of message data */
    MQLONG   CompCode;         /* Completion code */
    MQLONG   Reason;           /* Reason code qualifying CompCode */
    MQLONG   ExitResponse;     /* Response from exit */
    MQHCONN  Hconn;           /* Connection handle */
    PMQIEP   pEntryPoints;     /* Address of the MQIEP structure */
};
```

## COBOL bildirim (yalnızca IBM i )

```
** MQDXP structure
10 MQDXP.
** Structure identifier
15 MQDXP-STRUCID PIC X(4).
** Structure version number
15 MQDXP-VERSION PIC S9(9) BINARY.
** Reserved
15 MQDXP-EXITOPTIONS PIC S9(9) BINARY.
** Application options
15 MQDXP-APPOPTIONS PIC S9(9) BINARY.
** Numeric encoding required by application
15 MQDXP-ENCODING PIC S9(9) BINARY.
** Character set required by application
15 MQDXP-CODEDCHARSETID PIC S9(9) BINARY.
** Length in bytes of message data
```

```

15 MQDXP-DATALength PIC S9(9) BINARY.
** Completion code
15 MQDXP-COMPCODE PIC S9(9) BINARY.
** Reason code qualifying COMPCODE
15 MQDXP-REASON PIC S9(9) BINARY.
** Response from exit
15 MQDXP-EXITRESPONSE PIC S9(9) BINARY.
** Connection handle
15 MQDXP-HCONN PIC S9(9) BINARY.

```

## System/390 çevirici bildirimi

```

MQDXP          DSECT
MQDXP_STRUCID DS CL4 Structure identifier
MQDXP_VERSION DS F   Structure version number
MQDXP_EXITOPTIONS DS F   Reserved
MQDXP_APPOPTIONS DS F   Application options
MQDXP_ENCODING DS F   Numeric encoding required by application
MQDXP_CODEDCHARSETID DS F   Character set required by application
MQDXP_DATALength DS F   Length in bytes of message data
MQDXP_COMPCODE DS F   Completion code
MQDXP_REASON DS F   Reason code qualifying COMPCODE
MQDXP_EXITRESPONSE DS F   Response from exit
MQDXP_HCONN DS F   Connection handle
*
MQDXP_LENGTH EQU *-MQDXP
MQDXP_AREA   ORG MQDXP
            DS CL(MQDXP_LENGTH)

```

## MQXCNCV-Karakterleri dönüştür

MQXCNCV çağrısı, C programlama dilini kullanarak karakterleri bir karakter kümesinden diğerine dönüştürür.

Bu çağrı, IBM MQ Framework arabirimlerinden biri olan DCI ( IBM MQ Data Conversion Interface; Veri Dönüştürme Arabirimi) ' nin bir parçasıdır.

Not: Bu çağrı hem uygulamadan, hem de veri dönüştürme çıkış ortamlarından kullanılabilir.

## Sözdizimi

MQXCNCV (*Hconn*, *Options*, *SourceCCSID*, *SourceLength*, *SourceBuffer*, *TargetCCSID*, *TargetLength*, *TargetBuffer*, *DataLength*, *CompCode*, *Reason*)

## Parametreler

### Hconn

Tip: MQHCONN-input

Bu tanıtıcı, kuyruk yöneticisiyle olan bağlantıyı gösterir.

In a data-conversion exit, Hconn is normally the handle that is passed to the data-conversion exit in the Hconn field of the MQDXP structure; this handle is not necessarily the same as the handle specified by the application which issued the MQGET call.

 IBM üzerinde, Hconni'nin aşağıdaki özel değer belirtilebilir:

### MQHC\_DEF\_HCONN

Varsayılan bağlantı tanıtıcısı.

Bir CICS TS 3.2 ya da üstü bir uygulama çalıştırırsanız, MQXCNCV çağrısını çağıran karakter dönüştürme çıkış programının OPENAPI olarak tanımlandığından emin olun. Bu tanımlama, yanlış bir bağlantıdan kaynaklanan 2018 MQRC\_HCONN\_ERROR hatasını önler ve MQGET ' nin tamamlanmasını sağlar.

## Seçenekler

Tip: MQHOME-girişi

MQXCNCV işlemini denetleyen seçenekler.

Tanımlanan seçeneklerden sıfır ya da daha fazlası belirtilebilir. birden çok seçenek belirtmek için, değerleri bir araya ekleyin (aynı değişmezi bir kereden fazla eklemeyin) ya da bit ya da işlemi kullanarak değerleri birleştirin (programlama dili, bit işlemleri destekliyse).

**Default-conversion option**(Varsayılan dönüştürme seçeneği): Aşağıdaki seçenek, varsayılan karakter dönüştürmenin kullanımını denetler:

### MQDC\_DEFAULT\_CONVERSION

Varsayılan dönüştürme.

Bu seçenek, çağrıda belirtilen karakter kümelerinden biri ya da her ikisi desteklenmiyorsa, varsayılan karakter dönüştürmenin kullanılabilmesini belirtir. Bu, kuyruk yöneticisinin dizeyi dönüştürürken belirtilen karakter kümesine yaklaşan, kuruluş tarafından belirlenmiş bir varsayılan karakter kümesi kullanmasını sağlar.

**Not:** Dizgiyi dönüştürmek için yaklaşık bir karakter kümesi kullanılmasının sonucu, bazı karakterlerin yanlış dönüştürülebileceğini ifade eder. Yalnızca, hem belirlenen karakter takımı, hem de varsayılan karakter kümesi için ortak olan karakterler dizgi içinde kullanılmaktan kaçınılabılır.

Varsayılan karakter kümeleri, kuyruk yöneticisi kurulu ya da yeniden başlatıldığında bir yapılandırma seçeneği tarafından tanımlanır.

MQDCC\_DEFAULT\_CONVERSION belirtilmediyse, kuyruk yöneticisi dizgiyi dönüştürmek için yalnızca belirtilen karakter kümelerini kullanır ve karakter kümelerinden biri ya da her ikisi desteklenmiyorsa çağrı başarısız olur.

Bu seçenek aşağıdaki ortamlarda desteklenir:

-  AIX
-  IBM i
-  Linux
-  Windows

**Doldurma seçeneği:** Aşağıdaki seçenek, dönüştürülen dizginin hedef arabelleğe sığmasını sağlamak için, kuyruk yöneticisinin dönüştürülen dizeyi boşluklarla doldurmasına ya da önemsiz sondaki karakterleri atmasına olanak sağlar.

### MQDCC\_FILL\_TARGET\_BUFFER

Hedef arabelleği doldur.

Bu seçenek, hedef arabelleğin tam olarak doldurulduğunu böyle bir şekilde dönüştürmenin gerçekleşmesini ister:

- Dönüştürüldüğünde dizgi sözleşmeleri varsa, hedef arabelleği doldurmak için sondaki boşluklar eklenir.
- Dizgi dönüştürüldüğünde genişlerse, dönüştürülen dizginin hedef arabelleğe sığmasını sağlamak için anlamlı olmayan sondaki karakterler atılır. Bu işlem başarıyla gerçekleştirilebilecekse, çağrı MQCC\_OK ile tamamlanır ve neden kodu MQRC\_NONE olur.

Çok az sayıda önemsiz sonda karakter varsa, dizginin sığabileceği kadar dizgi hedef arabelleğe yerleştirilir ve çağrı MQCC\_UYARI ile tamamlanır ve neden kodu MQRC\_CONVERTED\_MSG\_TOO\_BÜYÜK ile tamamlanır.

Önemsiz karakterler şunlardır:

- Sondaki boşluklar
- Dizgideki ilk boş karakteri izleyen karakterler (ilk boş karakterin kendisi dışında)

- Dizgi, TargetCCSIDve TargetLength , hedef arabellek geçerli karakterlerle tam olarak ayarlanamazsa, çağrı MQCC\_FAILED ve neden kodu MQRC\_TARGET\_LENGTH\_ERROR ile başarısız olur. TargetCCSID salt DBCS karakter kümesi ( UTF-16gibi) olduğunda bu durum oluşabilir; ancak TargetLength , tek sayıda byte olan bir uzunluk belirtir.
- TargetLength , SourceLengthdeğerinden küçük ya da daha büyük olabilir. MQXCNCV ' den dönüşte DataLength , TargetLengthhile aynı değere sahiptir.

Bu seçenek belirlenmezse:

- Dizilimin, hedef arabellek içinde gerektiği şekilde sözleşmesine ya da genişletmesine izin verilir. Önemli sondaki karakterler eklenmez ya da atılmaz.

Dönüştürülen dizgi hedef arabelleğe sığarsa, çağrı MQCC\_OK ile tamamlanır ve neden kodu MQRC\_NONE olur.

Dönüştürülen dizgi hedef arabellek için çok büyükse, dizginin büyük bölümü hedef arabelleğe yerleştirilirse ve çağrı MQCC\_UYARI ile tamamlanır ve neden kodu MQRC\_CONVERTED\_MSG\_TOO\_BÜYÜK ile tamamlanır. Bu durumda TargetLength byte 'tan az not döndürülebileceğini unutmayın.

- TargetLength , SourceLengthdeğerinden küçük ya da daha büyük olabilir. MQXCNCV ' den dönüşte DataLength , TargetLengthdeğerinden küçük ya da ona eşit.

Bu seçenek aşağıdaki ortamlarda desteklenir:

-  AIX
-  IBM i
-  Linux
-  Windows

**Kodlama seçenekleri:** Tanımlanan seçenekler, kaynak ve hedef dizgilerinin tamsayı kodlamasını belirtmek için kullanılabilir. İlgili kodlama yalnızca, ilgili karakter takımı tanıtıcısı, ana saklama alanında belirlenen karakter kümesinin ikili tamsayılar için kullanılan kodlamaya bağımlı olduğunu gösterdiğinde kullanılır. Bu, yalnızca bazı çok baytlı karakter kümelerini (örneğin, UTF-16 karakter kümeleri gibi) etkiler.

Karakter kümesi tek baytlık karakter takımı (SBCS) ise ya da ana saklama alanında, tamsayı kodlamasına bağımlı olmayan bir çok baytlı karakter kümesi ise, kodlama yoksayılr.

MQDC\_SOURCE\_ \* değerlerinden yalnızca biri belirtilmelidir; MQDC\_TARGET\_ \* değerlerinden biriyle birleştirilmelidir.

#### **MQDCC\_SOURCE\_ENC\_NATIVE**

Kaynak kodlama, ortam ve programlama dili için varsayılan değerdir.

#### **MQDCC\_SOURCE\_ENC\_NORMAL**

Kaynak kodlama olağandır.

#### **MQDCC\_SOURCE\_ENC\_RECYD**

Kaynak kodlama tersine çevrildi.

#### **MQDCC\_SOURCE\_ENC\_UNDEFINED**

Kaynak kodlama tanımsız.

#### **MQDCC\_TARGET\_ENC\_NATIVE**

Hedef kodlama, ortam ve programlama dili için varsayılan değerdir.

#### **MQDC\_TARGET\_ENC\_NORMAL**

Hedef kodlama normal.

#### **MQDCC\_TARGET\_ENC\_REVERSED**

Hedef kodlama tersine çevrildi.

#### **MQDCC\_TARGET\_ENC\_UNDEFED**

Hedef kodlama tanımsız.

Daha önce tanımlanan kodlama değerleri doğrudan Options alanına eklenmelidir. Ancak, kaynak ya da hedef kodlama, MQMD ' de ya da başka bir yapıdaki Encoding alanından alınır, aşağıdaki işlem gerçekleştirilmelidir:

1. The integer encoding must be extracted from the Encoding field by eliminating the float and packed-decimal encodings; see “Kodlamalar çözümleniyor” sayfa 886 for details of how to do this.
2. 1. adımdan kaynaklanan tamsayı kodlaması, Options alanına eklenmeden önce uygun etkenle çarpılmalıdır. Bu etkenler şunlardır:
  - Kaynak kodlamaya ilişkin MQDCC\_SOURCE\_ENC\_FACTOR
  - Hedef kodlama için MQDCC\_TARGET\_ENC\_FACTOR

Aşağıdaki örnek kod, C programlama dilinde bunun nasıl kodlanabileceğini göstermektedir:

```
Options = (MsgDesc.Encoding & MQENC_INTEGER_MASK)
          * MQDCC_SOURCE_ENC_FACTOR
        + (DataConvExitParms.Encoding & MQENC_INTEGER_MASK)
          * MQDCC_TARGET_ENC_FACTOR;
```

Belirtilmezse, kodlama seçenekleri varsayılan olarak tanımsız (MQDCC\_\* \_ENC\_UNDEFINED) olarak varsayılan değer olarak belirlenir. Çoğu durumda bu, MQXCNCV çağrısının başarıyla tamamlanmadığını etkilemez. Ancak, ilgili karakter kümesi, kodlamaya (örneğin, bir UTF-16 karakter takımı gibi) bağlı olarak, çok baytlı bir karakter kümesi ise, çağrı neden kodu MQRC\_SOURCE\_INTEGER\_ENC\_ERROR ya da MQRC\_TARGET\_INTEGER\_ENC\_ERROR neden ile başarısız olur.

Kodlama seçenekleri aşağıdaki ortamlarda desteklenir:

-  AIX
-  IBM i
-  Linux
-  Windows
-  z/OS

**Varsayılan seçenek:** Önceden açıklanan seçeneklerden hiçbiri belirtilmediyse, aşağıdaki seçenek kullanılabilir:

#### **MQDCC\_NONE**

Seçenek belirtilmedi.

MQDCC\_NONE, program belgelerine yardımcı olmak için tanımlıdır. Bu seçeneğin başka biriyle birlikte kullanılması amaçlanmaz, ancak değeri sıfır olduğu için, bu tür bir kullanım algılanamaz.

#### **SourceCCSID**

Tip: MQHOME-girişi

Bu, SourceBuffer içindeki giriş dizgisinin kodlanmış karakter takımı tanıtıcısıdır.

#### **SourceLength**

Tip: MQHOME-girişi

Bu, SourceBuffer içindeki giriş dizgisinin bayt cinsinden uzunluğudur; Sıfır ya da daha büyük olmalıdır.

#### **SourceBuffer**

Tip: MQCHAR x SourceLength -giriş

Bu, bir karakter kümesinden diğerine dönüştürülecek dizeyi içeren arabelleğidir.

#### **TargetCCSID**

Tip: MQHOME-girişi

Bu, SourceBuffer ' un dönüştürüleceği karakter kümesinin kodlanmış karakter takımı tanıtıcısıdır.

### **TargetLength**

Tip: MQHOME-girişi

This is the length in bytes of the output buffer TargetBuffer ; it must be zero or greater. SourceLengthdeğerinden küçük ya da daha büyük olabilir.

### **TargetBuffer**

Tip: MQCHAR x TargetLength -çıkış

Bu, TargetCCSIDtarafından tanımlanan karakter kümesine dönüştürüldükten sonra dizgidir. Dönüştürülen dize, dönüştürülenmemiş dizgiden daha kısa ya da daha uzun olabilir. **DataLength** parametresi, döndürülen geçerli baytların sayısını gösterir.

### **DataLength**

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

This is the length of the string returned in the output buffer TargetBuffer. Dönüştürülen dize, dönüştürülenmemiş dizgiden daha kısa ya da daha uzun olabilir.

### **CompCode**

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Bu, aşağıdakilerden biridir:

#### **MQCC\_OK**

İşlem başarıyla tamamlandı.

#### **MQCC\_UYARI**

Uyarı (kısmi tamamlama).

#### **MQCC\_FAILED**

Arama başarısız oldu.

### **Neden**

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Neden kodu ön eleme CompCode.

CompCode MQCC\_OK ise:

#### **MQRC\_NONE**

(0, X'000 ') Raporlamak için bir neden yok.

CompCode MQCC\_UYARI ise:

#### **MQRC\_CONVERTED\_MSG\_TOO\_BÜYÜK**

(2120, X'848 ') Dönüştürülen veriler arabellek için çok büyük.

CompCode MQCC\_FAILED ise:

#### **MQRC\_DATA\_LENGTH\_ERROR**

(2010, X'7DA') Veri uzunluğu parametresi geçerli değil.

#### **MQRC\_DBCS\_HATASI**

(2150, X'866 ') DBCS dizgisi geçerli değil.

#### **MQRC\_HCONN\_ERROR**

(2018, X'7E2') Bağlantı tanıtıcısı geçerli değil.

#### **MQRC\_OPTIONS\_ERROR**

(2046, X'7FE') Seçenekler geçerli değil ya da tutarlı değil.

#### **MQRC\_RESOURCE\_SORUNU**

(2102, X'836 ') Yetersiz sistem kaynakları var.

#### **MQRC\_SOURCE\_BUFFER\_ERROR**

(2145, X'861 ') Kaynak arabellek parametresi geçerli değil.

#### **MQRC\_SOURCE\_CCID\_ERROR**

(2111, X'83F') Kaynak kodlanmış karakter takımı tanıtıcısı geçerli değil.

**MQRC\_SOURCE\_INTEGER\_ENC\_ERROR**

(2112, X'840 ') Kaynak tamsayı kodlaması tanınmadı.

**MQRC\_SOURCE\_LENGTH\_ERROR**

(2143, X'85F') Kaynak uzunluğu parametresi geçerli değil.

**MQRC\_STORAGE\_NOT\_AVAM**

(2071, X'817 ') Kullanılabilir bellek yetersiz.

**MQRC\_TARGET\_BUFF\_ERROR**

(2146, X'862 ') Hedef arabellek parametresi geçerli değil.

**MQRC\_TARGET\_CCSDID\_HATASI**

(2115, X'843 ') Hedef kodlanmış karakter takımı tanıtcısı geçerli değil.

**MQRC\_TARGET\_INTEGER\_ENC\_ERROR**

(2116, X'844 ') Hedef tamsayı kodlaması tanınmadı.

**MQRC\_TARGET\_LENGTH\_ERROR**

(2144, X'860 ') Hedef uzunluk parametresi geçerli değil.

**MQRC\_UNEXPECTED\_ERROR**

(2195, X'893 ') Beklenmeyen bir hata oluştu.

Bu kodlarla ilgili ayrıntılı bilgi için [İletiler ve neden kodları](#) başlıklı konuya bakın.

## C çağırma

```
MQXCNCV (Hconn, Options, SourceCCSID, SourceLength, SourceBuffer,
        TargetCCSID, TargetLength, TargetBuffer, &DataLength,
        &CompCode, &Reason);
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
MQHCONN  Hconn;          /* Connection handle */
MQLONG   Options;        /* Options that control the action of
MQXCNCV */
MQLONG   SourceCCSID;    /* Coded character set identifier of string
before conversion */
MQLONG   SourceLength;  /* Length of string before conversion */
MQCHAR   SourceBuffer[n]; /* String to be converted */
MQLONG   TargetCCSID;   /* Coded character set identifier of string
after conversion */
MQLONG   TargetLength;  /* Length of output buffer */
MQCHAR   TargetBuffer[n]; /* String after conversion */
MQLONG   DataLength;    /* Length of output string */
MQLONG   CompCode;      /* Completion code */
MQLONG   Reason;        /* Reason code qualifying CompCode */
```

## COBOL bildirimi (yalnızca IBM i)

IBM i

```
CALL 'MQXCNCV' USING HCONN, OPTIONS, SOURCECCSID, SOURCELENGTH,
                    SOURCEBUFFER, TARGETCCSID, TARGETLENGTH,
                    TARGETBUFFER, DATALENGTH, COMPCODE, REASON.
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
** Connection handle
01 HCONN          PIC S9(9) BINARY.
** Options that control the action of MQXCNCV
01 OPTIONS        PIC S9(9) BINARY.
** Coded character set identifier of string before conversion
01 SOURCECCSID   PIC S9(9) BINARY.
** Length of string before conversion
01 SOURCELENGTH  PIC S9(9) BINARY.
** String to be converted
```



```

01 SOURCEBUFFER PIC X(n).
** Coded character set identifier of string after conversion
01 TARGETCCSID PIC S9(9) BINARY.
** Length of output buffer
01 TARGETLENGTH PIC S9(9) BINARY.
** String after conversion
01 TARGETBUFFER PIC X(n).
** Length of output string
01 DATALENGTH PIC S9(9) BINARY.
** Completion code
01 COMPCODE PIC S9(9) BINARY.
** Reason code qualifying COMPCODE
01 REASON PIC S9(9) BINARY.

```

## S/390 çevirici bildirimi

```

CALL MQXCNCV, (HCONN, OPTIONS, SOURCECCSID, SOURCELENGTH,          X
              SOURCEBUFFER, TARGETCCSID, TARGETLENGTH, TARGETBUFFER, X
              DATALENGTH, COMPCODE, REASON)

```

Değiştirgeleri aşağıdaki gibi bildirin:

HCONN	DS	F	Connection handle
OPTIONS	DS	F	Options that control the action of MQXCNCV
SOURCECCSID	DS	F	Coded character set identifier of string before conversion
*			
SOURCELENGTH	DS	F	Length of string before conversion
SOURCEBUFFER	DS	CL(n)	String to be converted
TARGETCCSID	DS	F	Coded character set identifier of string after conversion
*			
TARGETLENGTH	DS	F	Length of output buffer
TARGETBUFFER	DS	CL(n)	String after conversion
DATALENGTH	DS	F	Length of output string
COMPCODE	DS	F	Completion code
REASON	DS	F	Reason code qualifying COMPCODE

## MQ\_DATA\_CONV\_EXIT-Veri dönüştürme çıkışı

MQ\_DATA\_CONV\_EXIT çağrısı, veri-dönüştürme çıkışa geçirilen değiştirgeleri tanımlar.

Kuyruk yöneticisi tarafından MQ\_DATA\_CONV\_EXIT adlı bir giriş noktası sağlanmadı (bkz. kullanım notu 11).

This definition is part of the IBM MQ Data Conversion Interface (DCI), which is one of the IBM MQ framework interfaces.

## Sözdizimi

MQ\_DATA\_CONV\_EXIT (*DataConvExitParms*, *MsgDesc*, *InBufferLength*, *InBuffer*, *OutBufferLength*, *OutBuffer*)

## Parametreler

### DataConvExitParms

Tip: MQDXP-giriş/çıkış

Bu yapı, çıkışa ilişkin çağrıyla ilgili bilgileri içerir. Çıkış, dönüştürmenin sonucunu belirtmek için bu yapıdaki bilgileri ayarlar. Bu yapıdaki alanların ayrıntıları için bkz. [“MQDXP-Data-conversion çıkış değiştirgesi”](#) sayfa 897 .

### MsgDesc

Tip: MQMD-giriş/çıkış

Çıkışa girişte, bu, **InBuffer** parametresindeki çıkışa aktarılan ileti verileriyle ilişkili ileti tanımlayıcıdır.

**Not:** Çıkışa aktarılan **MsgDesc** parametresi her zaman, çıkışı çağırın kuyruk yöneticisi tarafından desteklenen en son MQMD sürümüdür. Çıkışta farklı ortamlar arasında taşınabilir olması

amaçlandıysa, çıkış, çıkışa erişmek için gereken alanların yapımda var olduğunu doğrulamak için `MsgDesc` içindeki `Version` alanını denetler.

Aşağıdaki ortamlarda, çıkışa bir `version-2 MQMD` geçirilir:

-  AIX
-  IBM i
-  Linux
-  Windows

Veri dönüştürme çıkışını destekleyen diğer tüm ortamlarda, çıkış bir `version-1 MQMD` ' si geçirilir.

Çıkışta, çıkış `Encoding` ve `CodedCharSetId` alanlarını, dönüştürme başarılı olursa, uygulama tarafından istenen değerlere çevrilir; bu değişiklikler uygulamaya geri yansıtılır. Çıkışta yapılan diğer değişiklikler yoksayılır; bu değişiklikler uygulamaya geri yansıtılmaz.

Çıkış `MQDXP` yapısının `ExitResponse` alanında `MQXDR_OK` değerini döndürürse, ancak ileti tanımlayıcısındaki `Encoding` ya da `CodedCharSetId` alanlarını değiştirmezse, kuyruk yöneticisi bu alanlar için, `MQDXP` yapısındaki ilgili alanların çıkışa giriş için sahip olduğu değerleri döndürür.

### **InBufferLength**

Tip: `MQHOME`-girişi

Length in bytes of `InBuffer`.

This is the length of the input buffer `InBuffer`, and specifies the number of bytes to be processed by the exit. `InBufferLength`, dönüştürme işleminden önce ileti verilerinin uzunluğunun küçüktür ve `MQGET` çağrısında uygulama tarafından sağlanan arabelleğin uzunluğudur.

Değer her zaman sıfırdan büyük olur.

### **InBuffer**

Tip: `MQBYTEInBufferLength` -giriş

Dönüştürülemez iletiyi içeren arabellek.

Bu, dönüştürmeden önce ileti verilerini içerir. Çıkış, verileri dönüştüremiyorsa, kuyruk yöneticisi, çıkış tamamlandıktan sonra bu arabelleğin içeriğini uygulamaya geri döndürür.

**Not:** Çıkış `InBuffer`; değiştirmemelidir; bu parametre değiştirilirse, sonuçlar tanımsız olur.

C programlama dilinde bu parametre, bir işaretçi (pointer-to-void) olarak tanımlanır.

### **OutBufferLength**

Tip: `MQHOME`-girişi

Length in bytes of `OutBuffer`.

This is the length of the output buffer `OutBuffer`, and is the same as the length of the buffer provided by the application on the `MQGET` call.

Değer her zaman sıfırdan büyük olur.

### **OutBuffer**

Tip: `MQBYTEOutBufferLength` -çıkış

Dönüştürülen iletiyi içeren arabellek.

Çıkıştan çıkışta, dönüştürme başarılı olursa (**`DataConvExitParms`** parametresinin `ExitResponse` alanındaki `MQXDR_OK` değeri ile belirtildiği gibi), `OutBuffer` istenen gösterimde uygulamaya teslim edilecek ileti verilerini içerir. Dönüştürme başarısız olduysa, çıkışta bu arabelleğe ilişkin yapılan değişiklikler yoksayılır.

C programlama dilinde bu parametre, bir işaretçi (pointer-to-void) olarak tanımlanır.

## Kullanım notları

1. Veri dönüştürme çıkışı, bir MQGET çağrısının işlenmesi sırasında denetimi alan, kullanıcı tarafından yazılmış bir çıkışıdır. Veri dönüştürme çıkışı tarafından gerçekleştirilen işlem, çıkışa ilişkin sağlayıcı tarafından tanımlanır; ancak çıkış, burada açıklanan kurallara ve MQDXP ' nin ilişkili değiştirgeyle uyumlu olmalıdır.

Veri dönüştürme çıkışı için kullanılabilir programlama dilleri ortam tarafından belirlenir.

2. Çıkış yalnızca aşağıdaki tüm deyimlerin doğru olması durumunda çağrılır:

- MQGET çağrısında MQGMO\_CONVERT seçeneği belirtildi.
- İleti tanımlayıcısındaki Format alanı MQFMT\_NONE değil.
- İleti gerekli gösterimde değil; yani, iletinin CodedCharSetId ve Encoding bir ya da her ikisi, MQGET çağrısında belirtilen ileti tanımlayıcısında uygulama tarafından belirtilen değerden farklı.
- Kuyruk yöneticisi dönüştürmeyi başarıyla yapmamış
- Uygulamanın arabelleğindeki uzunluk sıfırdan büyük.
- İleti verilerinin uzunluğu sıfırdan büyük.
- MQGET işlemi sırasında şu ana kadar neden kodu MQRC\_NONE ya da MQRC\_TRUNCATED\_MSG\_ACCEPTED olur.

3. Bir çıkış yazılmakta olduğunda, çıkışta, kesilen iletilerin dönüştürülmesine izin veren bir şekilde kodlamayı düşünün. Kısaltılmış iletiler aşağıdaki şekillerde ortaya çıkabilir:

- Alma uygulaması, iletiden küçük bir arabellek sağlar, ancak MQGET çağrısında MQGMO\_ACCEPT\_TRUNCATED\_MSG seçeneğini belirtir.

Bu durumda, çıkışa girişteki **DataConvExitParms** parametresindeki Reason alanının MQRC\_TRUNCATED\_MSG\_ACCEPTED değeri var.

- İletiyi gönderen, göndermeden önce kesmiş olur. Bu, rapor iletileriyle olabilir, örneğin (daha ayrıntılı bilgi için bkz. "[Rapor iletilerinin dönüştürülmesi](#)" sayfa 896 ).

Bu durumda, çıkışa girilen **DataConvExitParms** parametresindeki Reason alanı, MQRC\_NONE değerine sahiptir (alan uygulama ileti için yeterince büyük bir arabellek sağladıysa).

Dolayısıyla, çıkışa giriş üzerindeki Reason alanının değeri, iletinin kesilip kesilmediğine karar vermek için her zaman kullanılamaz.

Kısaltılmış bir iletinin ayırt edici özelliği, **InBufferLength** parametresindeki çıkışa sağlanan uzunluğun, ileti tanımlayıcısındaki Format alanında bulunan biçim adının belirttiği uzunluktan az olması. Bu nedenle, çıkışta veri dönüştürme girişiminde bulunmadan önce **InBufferLength** değeri denetlenmelidir; çıkış, biçim adına göre örtük olarak belirtilen tam veri miktarının sağlandığını varsaymamalıdır.

If the exit has not been written to convert truncated messages, and **InBufferLength** is less than the value expected, the exit will return MQXDR\_CONVERSION\_FAILED in the **ExitResponse** field of the **DataConvExitParms** parameter, with the **CompCode** and **Reason** fields set to MQCC\_WARNING and MQRC\_FORMAT\_ERROR.

Çıkış, kesilmiş iletileri dönüştürmek için yazıldıysa, çıkış mümkün olduğu kadar veri (sonraki kullanım notuna bakın) dönüştürülecek ve verileri **InBuffer**' in sonunun ötesine dönüştürmeye ya da dönüştürmeye çalışmamaya özen gösterecektir. Dönüştürme başarıyla tamamlanırsa, çıkış **DataConvExitParms** parametresindeki Reason alanını değiştirmeden bırakır. İleti, alıcının kuyruk yöneticisi tarafından kesildiyse, MQRC\_TRUNCATED\_MSG\_KABUL edilir ve ileti, iletiyi gönderen tarafından kesildiyse MQRC\_NONE değerini döndürür.

Ayrıca, bir iletinin dönüştürme sırasında genişletmesi, **OutBuffer** değerinden daha büyük olduğu bir noktaya kadar genişletilir. In this case the exit must decide whether to truncate the message; the **AppOptions** field in the **DataConvExitParms** parameter indicates whether the receiving application specified the MQGMO\_ACCEPT\_TRUNCATED\_MSG option.

4. Genel olarak, InBuffer içindeki çıkışa sağlanan iletteki tüm veriler dönüştürülür ya da hiçbirinin hiçbiri değildir. Ancak, bu durumda ileti kesilirse, dönüştürmeden önce ya da dönüştürme sırasında bir kural dışı durum oluşur; bu durumda, arabelleğin sonunda tamamlanmamış bir öge olabilir (örneğin: çift byte karakterinin 1 byte 'ı ya da 4 baytlık bir tamsayıya ilişkin 3 byte). Bu durumda, eksik ögeyi atlamayı düşünün ve OutBuffer ' da kullanılmayan baytları boş değere ayarlayın. Ancak, bir dizi ya da dizedeki tüm ögeler ya da karakterler dönüştürülmelidir.
5. İlk kez bir çıkış gerektiğinde, kuyruk yöneticisi, biçimiyle aynı adı taşıyan bir nesneyi (uzantılar dışında) yüklemeyi dener. Yüklenen nesne, o biçim adını taşıyan iletleri işleyen çıkışı içermelidir. Çıkış adını ve tüm ortamların bunu gerektirmemesine rağmen, çıkışı aynı olan nesnenin adını da aynı şekilde yapmayı düşünebilirsiniz.
6. Bir uygulama kuyruk yöneticisine bağlı uygulamadan bu yana Format ' u kullanan ilk iletiyi almayı denediğinde çıkışa ilişkin yeni bir kopya yüklenir. CICS ya da IMS uygulamaları için bu, kuyruk yöneticisine bağlı CICS ya da IMS altsistemine ne zaman bağlandığını belirtir. Kuyruk yöneticisi önceden yüklenmiş bir kopyayı atmışsa, başka zamanlarda yeni bir kopya da yüklenebilir. Bu nedenle, çıkışı bir çıkıştan diğerine iletmek için statik depolamayı kullanma girişiminde bulunulmamalı. Bu, iki çağırma arasında bir çıkışa boşaltılabilir.
7. Kuyruk yöneticisi tarafından desteklenen yerleşik biçimlerden biriyle aynı adı taşıyan bir kullanıcı tarafından sağlanan bir çıkış varsa, kullanıcı tarafından sağlanan çıkış yerleşik dönüştürme yordamlarından birini değiştirmez. Bu tür bir çıkışa çağrılan tek koşul şunlardır:
  - Yerleşik dönüştürme yordamı CodedCharSetId ya da Encoding ilişkisinden ya da involved' e dönüştürmeleri işleyemiyorsa ya da
  - Yerleşik dönüştürme yordamı verileri dönüştüremediye (örneğin, dönüştürülemeyen bir alan ya da karakter olduğu için).
8. Çıkışa ilişkin kapsam ortamdan bağımsız olur. Diğer biçimlerle çakışma riskini en aza indirmek için Format adları seçilmeli. Biçim adını tanımlayan uygulamayı tanıtan karakterlerle başlamayı göz önünde bulundurun.
9. Veri dönüştürme çıkışı, MQGET çağrısını yayınlayan programdaki gibi bir ortamda çalışır; ortam, adres alanını ve kullanıcı tanımını içerir (geçerli olduğu yerlerde). Bu program, ileti kanalı aracısının ileti dönüştürmesini desteklemeyen bir hedef kuyruk yöneticisine ileti göndermesine yol açabilirdi. Çıkış, kuyruk yöneticisinin ortamında çalıştırılmadığından, kuyruk yöneticisinin bütünlüğünden ödün veremiyor.
10. Çıkış tarafından kullanılacak tek MQI çağrısı MQXCNCV; diğer MQI çağrılarını kullanma girişimi sırasında neden kodu MQRC\_CALL\_IN\_PROGRESS ya da öngörülemez diğer hatalar var.
11. Kuyruk yöneticisi tarafından MQ\_DATA\_CONV\_EXIT adlı bir giriş noktası sağlanmadı. Ancak, C programlama dilinde MQ\_DATA\_CONV\_EXIT adı için bir typedef değeri verilir; parametrelerin doğru olduğundan emin olmak için, kullanıcı tarafından yazılan çıkışı bildirmek için bu kullanılabilir. Çıkışa ilişkin ad, tüm ortamlarda gerekli olmasa da, bu adın biçim adıyla (MQMD ' de Format alanında bulunan ad) aynı olması gerekir.

The following example illustrates how the exit that processes the format MYFORMAT can be declared in the C programming language:

```
#include "cmqc.h"
#include "cmqxc.h"

MQ_DATA_CONV_EXIT MYFORMAT;

void MQENTRY MYFORMAT(
    PMQDXP  pDataConvExitParms, /* Data-conversion exit parameter
                                block */
    PMQMD   pMsgDesc,           /* Message descriptor */
    MQLONG  InBufferLength,     /* Length in bytes of InBuffer */
    PMQVOID pInBuffer,         /* Buffer containing the unconverted
                                message */
    MQLONG  OutBufferLength,    /* Length in bytes of OutBuffer */
    PMQVOID pOutBuffer)        /* Buffer containing the converted
                                message */
{
```

```
    /* C language statements to convert message */  
  }
```

12. **z/OS** z/OS üzerinde, bir API geçiş çıkışı da zorda olursa, veri dönüştürme işleminden sonra bu çıkış da çağrılır.

## C çağırma

```
exitname (&DataConvExitParms, &MsgDesc, InBufferLength,  
          InBuffer, OutBufferLength, OutBuffer);
```

Çıkışa geçirilen parametreler aşağıdaki gibi bildirilir:

```
MQDXP  DataConvExitParms; /* Data-conversion exit parameter block */  
MQMD   MsgDesc;          /* Message descriptor */  
MQLONG InBufferLength;   /* Length in bytes of InBuffer */  
MQBYTE InBuffer[n];      /* Buffer containing the unconverted  
                           message */  
MQLONG OutBufferLength;  /* Length in bytes of OutBuffer */  
MQBYTE OutBuffer[n];     /* Buffer containing the converted  
                           message */
```

## COBOL bildirimi (yalnızca IBM i)

IBM i

```
CALL 'exitname' USING DATACONVEXITPARMS, MSGDESC, INBUFFERLENGTH,  
                     INBUFFER, OUTBUFFERLENGTH, OUTBUFFER.
```

Çıkışa geçirilen parametreler aşağıdaki gibi bildirilir:

```
** Data-conversion exit parameter block  
01 DATACONVEXITPARMS.  
   COPY CMQDXPV.  
** Message descriptor  
01 MSGDESC.  
   COPY CMQMDV.  
** Length in bytes of INBUFFER  
01 INBUFFERLENGTH PIC S9(9) BINARY.  
** Buffer containing the unconverted message  
01 INBUFFER PIC X(n).  
** Length in bytes of OUTBUFFER  
01 OUTBUFFERLENGTH PIC S9(9) BINARY.  
** Buffer containing the converted message  
01 OUTBUFFER PIC X(n).
```

## System/390 çevirici bildirimi

```
CALL EXITNAME, (DATACONVEXITPARMS, MSGDESC, INBUFFERLENGTH, X  
               INBUFFER, OUTBUFFERLENGTH, OUTBUFFER)
```

Çıkışa geçirilen parametreler aşağıdaki gibi bildirilir:

```
DATACONVEXITPARMS CMQDXPA , Data-conversion exit parameter block  
MSGDESC           CMQMDA , Message descriptor  
INBUFFERLENGTH   DS      F Length in bytes of INBUFFER  
INBUFFER         DS      CL(n) Buffer containing the unconverted  
*               message  
OUTBUFFERLENGTH  DS      F Length in bytes of OUTBUFFER  
OUTBUFFER        DS      CL(n) Buffer containing the converted  
*               message
```

## MQRFH2 öğeleri olarak belirtilen özellikler

İleti dışı tanımlayıcı özellikleri, MQRFH2 üstbilgi klasörlerinde öge olarak belirtilebilir. Özellik olarak belirtilmekte olan MQRFH2 öğelerine genel bakış.

Bu, IBM MQ JMS ve XMS istemcilerinin önceki sürümleriyle uyumluluğu korur. This section describes how to specify properties in MQRFH2 headers.

Özellikler olarak MQRFH2 öğelerini kullanmak için, öğeleri [Using IBM MQ classes for Java](#) içinde açıklandığı gibi belirtin. Bu bilgiler, "MQRFH2 -Kurallar ve biçimlendirme üstbilgisi 2" sayfa 520 içinde açıklanan bilgileri tamamlar.

## Özellik veri tiplerinin MQRFH2 veri tiplerine eşlenmesi

Bu konu, karşılık gelen MQRFH2 veri tipleriyle eşlenen ileti özelliği tiplerine ilişkin bilgi sağlar.

İleti özelliği tipi	MQRFH2 veri tipi
MQBYTE []	bin.hex
MQBOOL	boole
MQINT8	i1
MQINT16	i2
MQINT32	i4
MQINT64	i8
MQFLOAT32	r4
MQFLOAT64	r8
MQCHAR []	dizgi

Veri tipi olmayan herhangi bir öğenin "dizgi" tipinde olduğu varsayılır.

An MQRFH2 data type of `int`, meaning an integer of unspecified size, is treated as if it were an `i8`.

Boş değer, `xsi:nil='true'` öge özneliyle gösterilir. Boş değer olmayan değerler için `xsi:nil='false'` özneliğini kullanmayın.

Örneğin, aşağıdaki özelliğin boş bir değeri vardır:

```
<NullProperty xsi:nil='true'></NullProperty>
```

Bir bayt ya da karakter dizilimi özelliği boş bir değere sahip olabilir. Bu, sıfır uzunluklu bir öge değerine sahip bir MQRFH2 ögesi tarafından temsil edilir.

Örneğin, aşağıdaki özelliğin boş bir değeri vardır:

```
<EmptyProperty></EmptyProperty>
```

## Desteklenen MQRFH2 klasörleri

İleti tanımlayıcı alanlarının özellikler olarak kullanımına genel bakış.

The folders `<jms>`, `<mcd>`, `<mqext>`, and `<usr>` are described in MQRFH2 üstbilgisi ve JMS. `<usr>` klasörü, bir iletiyle ilişkilendirilmiş herhangi bir JMS uygulama tanımlı özelliği taşımak için kullanılır. Groups are not allowed in the `<usr>` folder.

MQRFH2 üstbilgisi ve JMS aşağıdaki ek klasörleri destekler:

- <mq>

Bu klasör, IBM MQ tarafından kullanılan MQ tanımlı özellikler için kullanılır ve ayrılmıştır.

- <mq\_usr>

This folder can be used to transport any application-defined properties that are not exposed as JMS user-defined properties, as the properties might not meet the requirements of a JMS property. Bu klasör, <usr> klasörünün yapamadığı grupları içerebilir.

- content='properties' öznitelikle işaretlenen herhangi bir klasör.

Böyle bir klasör, içeriğindeki <mq\_usr> klasörünün eşdeğeridir.

- <mqps>

Bu klasör, IBM MQ yayınlama/abone olma özellikleri için kullanılır.

IBM MQ , WAS/SIB tarafından önceden kullanımda olan şu klasörleri de destekler:

- <sib>

Bu klasör, JMS özellikleri olarak gösterilmeyen ya da JMS\_IBM\_\* özellikleriyle eşlenmiş olan WAS/SIB sistem iletişi özellikleri için kullanılır ve ayrılır; ancak WAS/SIB uygulamalarına gösterilir; bunlar arasında ileri ve ters yönlendirme yolları özellikleri bulunur.

Bunlar byte dizileri olduğundan en az bazıları JMS özellikleri olarak gösterilemez. Uygulamanız bu klasöre özellikler eklediyseniz, değer yoksayılar ya da kaldırılır.

- <sib\_usr>

Bu klasör, desteklenen tiplerden olmadıkları için JMS kullanıcı özellikleri olarak gösterilemeyen WAS/SIB kullanıcı iletişi özellikleri için kullanılır ve ayrılmıştır; WAS/SIB uygulamalarına maruz kaldıkları için kullanılır.

Bunlar, SIMessage arabirimi aracılığıyla alabileceğiniz ya da ayarlanabileceğiniz kullanıcı özellikleridir, ancak bayt dizisinin içeriği, gerekli özellik değeriyle eşlenir.

IBM MQ uygulamanız klasöre isteğe bağlı bir bin.hex ögesi yazarsa, uygulamanın geri yüklenmesi beklenen biçimde olmadığı için uygulama büyük olasılıkla bir IOException alır. bin.hex ögesinden başka bir şey eklerseniz, ClassCastException adlı bir öge alırsınız.

Bu klasörü kullanarak, WAS/SIB için kullanılabilir özellikleri yapmaya çalışmayın; bunun yerine, bu amaçla <usr> klasörünü kullanıcı olarak kullanın.

- <sib\_context>

Bu klasör, WAS/SIB kullanıcı uygulamalarında ya da JMS özellikleri olarak gösterilmeyen WAS/SIB sistem iletişi özellikleri için kullanılır. Bunlar, web hizmetleri ve benzeri için kullanılan güvenlik ve işlemsel özellikleri içerir.

Uygulamanızın bu klasöre özellikler eklememesi gerekir.

- <mqema>

Bu klasör, <mqext> klasörü yerine WAS/SIB tarafından kullanıldı.

MQRFH2 klasör adları büyük/küçük harfe duyarlıdır.

Aşağıdaki klasörler, küçük ya da büyük harf karakterlerinin herhangi bir karışımında ayrılmıştır:

- Any folder prefixed by mq or wmq ; reserved for use by IBM MQ.
- Any folder prefixed by sib ; reserved for use by WAS/SIB.
- <Root> ve <Body> klasörleri; ayrılmış, ancak kullanılmıyorsa.

Aşağıdaki klasörler, ileti özellikleri içeren olarak tanınmıyor:

- <psc>

Yayınlama/abone olma komut iletilerini aracıya iletmek için IBM Integration Bus tarafından kullanılır.

- <pscr>

Yayınlama/abone olma komut iletilerine yanıt olarak, aracından bilgi içermek için IBM Integration Bus tarafından kullanılır.

- IBM tarafından tanımlı olmayan, content=' properties ' öznitelikle işaretlenmemiş herhangi bir klasör.

<psc> ya da <psc> klasörlerinde content=' properties ' belirtmeyin. Bunu yapmazsanız, bu klasörler özellik olarak işlem görür ve IBM Integration Bus beklendiği gibi işlev görmeyi durduracaktır.

If your application is building messages with properties, in MQRFH2 headers to be recognized as an MQRFH2 header containing properties, the header must be in the list of headers that can be chained at the head of the message.

The MQRFH2 can be preceded by any number of MQH standard headers, or an MQCIH, an MQDLH, an MQIIH, an MQTM, an MQTMC2, or an MQXQH. Zincirlenemeyecekleri için bir dize ya da MQCFH ayrıştırması sonlandırılıyor.

İletin, tüm ileti özelliklerini taşıyan birden çok MQRFH2 üstbilgisi içermesi mümkündür. Aynı adı taşıyan klasörler, başka bir şekilde kısıtlanmadıkça, örneğin WAS/SIB tarafından farklı üstbilgilerde de birlikte bulunabilir. Klasörler, önemli üstbilgilerde varsa, tek bir mantıksal klasör olarak işlem görür.

Önemli üstbilgilerdeki klasörler, önemli olmayan üstbilgilerdeki klasörlerle birleştirilemezken, önemli üstbilgiler içinde aynı adı taşıyan klasörler birleştirilebilir ve çakışan özelliklerin kaldırılabilir. Uygulamalarınız, kendi iletilerinde özelliklerin yerleşim düzenine bağlı olmamalıdır.

MQRFH2 grupları, kullanıcı tanımlı klasörlerdeki özellikler için ayrıştırılır; <wmq>, <jms>, <mcd>, <usr>, <mqext>, <sib>, <sib\_usr>, <sib\_context> ve <mqema> klasörlerinde yer alan özellikler yoktur.

Groups in the IBM-defined properties folders, except for the <wmq> and <mq> folders, are parsed for properties.

Bir MQRFH2 klasörü karma içerik içeremez; bir klasör ya da grup, grupları ya da özellikleri ya da bir değeri içerebilir, ancak ikisini birden içeremez.

Bir iletin ilk ya da sonraki kesimden biri, ileti tanımlayıcısındaki bu özellikler dışında, IBM MQ tanımlı özellikleri içeremez. Bu nedenle, bu tür özelliklerin bulunduğu bir iletiyi MQMF\_SEGMENT ya da MQMF\_SEGMENTATION\_ALLOWALLOWLU kümesi içeren bir ileti koymak, MQRC\_SEGMENTATION\_NOT\_ALLOWALIZY ile başarısız olmasına neden olur.


Ancak, ileti grupları IBM MQ tanımlı özellikleri içerebilir.

## Generation of MQRFH2 headers

IBM MQ , ileti özelliklerini MQRFH2 gösterimlerine dönüştürürse, iletiye MQRFH2 (sayfa adı) eklemelidir. Bu, MQRFH2 ' u ayrı bir üstbilgi olarak ekler ya da var olan bir üstbilgiyle birleştirir.

IBM MQ ' a göre yeni MQRFH2 üstbilgileri oluşturulması, bir iletide var olan üstbilgileri bozabilir. Üstbilgiler için ileti arabelleğindeki bir ileti arabelleğinden ayrıştıran uygulamalar, bir arabelleğindeki üstbilgilerin sayısının ve konumunun bazı durumlarda değişebileceğini dikkate almalıdır. IBM MQ , ileti özelliklerini var olan bir MQRFH2 üstbilgisinde birleştirerek, özellikleri bir iletiye ekleme etkisini en aza indirmeye çalışır. Ayrıca, oluşturulmuş bir MQRFH2 ' ı ileti arabelleğindeki diğer üstbilgilere göre sabit bir konuma yerleştirerek etkiyi en aza indirmeyi dener.

A generated MQRFH2 header is placed following the MQMD, and any number of MQXQH, MQRFH, and MQDLH headers, whatever order they are in. Oluşturulan MQRFH2 üstbilgisi, MQMD, MQXQH, MQDLH ya da MQRFH üstbilgisinde olmayan ilk üstbilginin hemen önüne yerleştirilir.

 z/OS sistemlerinde, oluşturulan MQRFH2 üstbilgisi, uygulamanın CCSID içinde oluşturulur. Bu, aşağıdaki şekilde tanımlıdır:

- For batch LE applications using the DLL interface, the CCSID is the CODESET associated with the current locale at the time **MQCONN** is issued (default value is 1047).
- For batch LE applications bound with one of the batch MQ stubs, the CCSID is the CODESET associated with the current locale at the time of the first MQI call issued after **MQCONN** (default value is 1047).



- For batch non-LE applications running on a z/OS UNIX System Services (z/OS UNIX) thread, the CCSID is the value of THLICCSID at the time of the first MQI call issued after **MQCONN** (default value is 1047).
- Diğer toplu uygulamalar için CCSID , kuyruk yöneticisinin CCSID ' i dir.

LE uygulamaları için, yerel ayar, setlocale() / CEESETL LE çağrılabilir hizmeti kullanılarak değiştirilebilir. For non-LE applications running on z/OS UNIX threads, the value of THLICCSID can be changed using z/OS UNIX mapping macro **BPXYTHLI**.

## Oluşturulan MQRFH2 birleştirmeye ilişkin kurallar

Aşağıdaki kurallar, oluşturulmuş bir MQRFH2 ' ı var olan bir MQRFH2 ile birleştirmek için geçerlidir. Oluşturulan MQRFH2 üstbilgisi, varsa, var olan bir MQRFH2 üstbilgisiyle birleştirilir:

1. Var olan MQRFH2 aynı konumda; IBM MQ , oluşturulmuş bir MQRFH2 ya da daha önceki bir önceki üstbilgi zincirine daha önce yer alıyor.
2. Üretilen özelliklerin CCSID değeri, varolan MQRFH2' in NameValueCCSID ' si ile aynı.

Tersi durumda, üretilen üstbilgi daha önce açıklanan konumda, arabelleğe ayrı olarak yerleştirilir.

## Var olan bir MQRFH2 içinde klasör birleştirmeye ilişkin kurallar

İleti özellikleri var olan bir MQRFH2 ile birleştirilirse, var olan MQRFH2 , ileti özellikleriyle eşleşen klasörler için taranır ve bunları birleştirir. Eşleşen bir klasör yoksa, var olan klasörlerin sonuna yeni bir klasör eklenirse. Eşleşen bir klasör varsa, klasör aranır. Eşleşen özelliklerin üzerine yazılır. Klasörün sonuna yeni eklenen öge sayısı eklenir.

## MQRFH2 klasör kısıtlamaları

MQRFH2 üstbilgilerindeki klasör kısıtlamalarına genel bakış

MQRFH2 kısıtlamaları aşağıdaki klasörler için geçerlidir:

- <usr> klasöründeki öge adları JMS önekiyle başlamamalıdır; bu tür özellik adları JMS tarafından kullanılmak üzere ayrılmıştır ve kullanıcı tanımlı özellikler için geçerli değildir.  
Böyle bir öge adı, MQRFH2 ayrıştırmanın başarısız olmasına neden olmaz, ancak IBM MQ ileti özelliği API ' ları için erişilebilir değildir.
- <usr> klasöründeki öge adları, alt ya da büyük harf, NULL, TRUE, FALSE, NOT, AND, OR, LIKE, IS ve ESCAPE gibi alt ya da büyük harf karışımında olmamalıdır. These names match SQL keywords and make parsing selectors harder, because <usr> is the default folder used when no folder is specified for a particular property in a selector.  
Böyle bir öge adı, MQRFH2 ayrıştırmanın başarısız olmasına neden olmaz, ancak IBM MQ ileti özelliği API ' ları için erişilebilir değildir.
- <usr> klasörünün içerik modeli aşağıdaki gibidir:
  - Geçerli bir XML adı, kolon içermemesini sağlayan bir öge adı olarak kullanılabilir.
  - Yalnızca yalın ögeler (iççe yerleşimli klasörler değil) izin verilir.
  - Bir dt="xxx" özniteliği tarafından değiştirilmedikçe, tüm ögeler varsayılan dizgi tipini alır.
  - Tüm ögeler isteğe bağlıdır, ancak bir klasörde bir kereden fazla oluşmalıdır.
- İleti özelliklerini içermek üzere kabul edilen herhangi bir klasördeki öge adları nokta (.) içermemelidir (Unicode karakteri U+002E), bunun nedeni, sıradüzenini göstermek üzere özellik adlarında kullanılır.  
Böyle bir öge adı, MQRFH2 ayrıştırmanın başarısız olmasına neden olmaz, ancak IBM MQ ileti özelliği API ' ları için erişilebilir değildir.

In general, MQRFH2 headers that contain valid XML-style data can be parsed by IBM MQ without failure, although certain elements of the MQRFH2 are not accessible through the IBM MQ message property APIs.

## MQRFH2 öge adı çakışmaları

MQRFH2 öge adları içindeki çakışmalara genel bakış.

Bir ileti özelliğine yalnızca bir değer eklenebilir. Bir özelliğe erişme girişimi bir değer çakışmasına yol açarsa, başka bir özelliğe ilişkin tercihte biri seçilir.

MQRFH2 öğelerine erişmek için kullanılan IBM MQ sözdizimi, bir ögenin benzersiz olarak tanımlanmasına olanak verir; bir klasörde aynı adı taşıyan hiçbir öge bulunmuyorsa. Bir klasör aynı adı taşıyan birden çok öge içeriyorsa, kullanılan özelliğin değeri, iletinin başına en yakın olan özelliğin değeri olur.

Aynı ileti içinde aynı adı içeren iki ya da daha çok klasör farklı önemli MQRFH2 üstbilgilerinde yer almıyorsa geçerlidir.

MQGET çağrısı, ileti dışı bir tanımlayıcı özelliği iki kez ayarlandıktan sonra işlendiğinde, hem MQSETMP çağrısı yoluyla hem de doğrudan işlenmemiş MQRFH2 üstbilgisinde bir çakışma oluşur.

Böyle bir durumda, bir API çağrısı tarafından iletiyle ilişkilendirilen özellik, ileti verilerinde bulunan, yani işlenmemiş MQRFH2 üstbilgisindeki bir tercihle ilişkilendirilmiş olan özelliktir. Bir çakışma oluşursa, ileti verilerinin önce mantıksal olarak gelmesi düşünülmektedir.

## Özellik adlarından MQRFH2 klasöründen ve öge adlarıyla eşleme

MQRFH2 üstbilgisindeki özellik adları ve öge adları arasındaki farklara genel bakış.

İleti özelliklerini (örneğin, MQ JMS) belirtmek için en sonunda MQRFH2 üstbilgileri oluşturan tanımlı API' lardan birini kullanırken, özellik adı MQRFH2 klasöründe yer alan bir öge adı değildir.

Bu nedenle, özellik adından MQRFH2 ögesine bir eşleme gerçekleşir ve hem ögeyi içeren klasör adını hem de öge adını dikkate alarak ters yönlü olarak bir eşleme gerçekleşir. Some examples from IBM MQ classes for JMS are already documented in [Using IBM MQ classes for Java](#).

Çizelge 637. Property names mapped to MQRFH2 folder and element names

Özellik adı	MQRFH2 klasör adı	MQRFH2 öge adı
JMSDestination	jms	Dst
JMSType	mcd	Type, Set, Fmt
xxx (kullanıcı tanımlı, burada xxx , JMSile başlamaz)	usr	xxx

Bu nedenle, bir JMS uygulaması JMSDestination özelliğine bu eşlemleri <jms> klasöründeki Dst ögesiyle eriştiğinde erişir.

Özellikleri MQRFH2 öğeleri olarak belirlerken IBM MQ , öğelerini aşağıdaki gibi tanımlar:

Çizelge 638. MQRFH2 klasörü, grubu ve öge adlarıyla eşlenen özellik adları

Özellik adı	MQRFH2 klasör adı	MQRFH2 grup adı	MQRFH2 öge adı
<Property>	<usr>	yok	<Property>
<folder>. <Property>	<folder>	yok	<Property>
<folder>. <group>. <Property>	<folder>	<group>	<Property>

Örneğin, bir IBM MQ uygulaması Property1 özelliğine erişmeyi denediğinde, bu, <usr> klasöründeki Property1 ögesiyle eşlenmektedir. wmq .Property2 özelliği, <wmq> klasöründeki Property2 özelliğiyle eşlenir.

Özellik adı birden çok öge içeriyorsa. Son olarak, kullanılan MQRFH2 öge adı için kullanılan öge adı kullanılır. character, and MQRFH2 groups are used to form a hierarchy; nested MQRFH2 groups are permitted.

<mcd>, <jms>ve <mqext> klasörlerinde bir MQRFH2 içinde yer alan JMS üstbilgisi ve sağlayıcısına özgü özellikler, IBM MQ classes for Javakomutunu kullanma içinde tanımlanan kısa adları kullanan bir IBM MQ uygulaması tarafından erişilir.

JMS kullanıcı tanımlı özelliklere <usr> klasöründen erişilir. Bir IBM MQ uygulaması, özelliğin JMS uygulamalarına kullanıcı tanımlı özelliklerden biri olarak görünmesi için kabul edilebilirse, uygulama özellikleri için <usr> klasörünü kullanabilir.

Kabul edilebilir değilse, başka bir klasör seçin; <wmq\_usr> klasörü, bu tür JMS dışı özellikler için standart bir konum olarak sağlanır.

Uygulamalarınız,well-definedklasöründe yer alan herhangi bir MQRFH2 klasörünü belirtebilir ve kullanabilir, “MQRFH2 öğeleri olarak belirtilen özellikler” sayfa 914 dosyasında belgelenmez; aşağıdakileri dikkate aldığınız:

1. Klasör, içinde bulunan özelliklere tanımsız erişim sağlayan başka bir uygulama tarafından kullanılıyor olabilir ya da ileride kullanılabilir; özellik adları için önerilen adlandırma kuralı için Özellik adları başlıklı konuya bakın.
2. Özellikler, yalnızca kullanıcı tanımlı özellikler için <usr> klasörüne erişilebilen IBM MQ classes for JMS ya da XMS istemcisinin önceki sürümleri için erişilebilir değil.
3. The folder must be marked with the attribute content with the value set to properties, for example, content='properties'.

“MQSETMP-İleti özelliğini ayarla” sayfa 767 , bu özniteliği gerektiği şekilde otomatik olarak ekler. Bu öznitelik, IBMtanımlı klasörlerden birine eklenmemelidir; örneğin, <jms> ve <usr>. Doing so, causes the message to be rejected by the IBM MQ classes for JMS client before IBM WebSphere MQ 7.0. with a MessageFormatException.

<usr> klasörü, <Property> sözdiziminin özellikleri için varsayılan konum, aynı adı kullanan aynı kullanıcı tanımlı özellik değerine erişmek için bir IBM MQ uygulaması ve bir JMS uygulaması olduğu için.

## Ayrılmış klasör adları

Birkaç ayrılmış klasör adı vardır. You cannot use such names as your folder prefixes; for example, Root . Property1 does not access a valid property because Root is reserved. Aşağıdaki listede ayrılmış klasör adları var:

- Kök
- Gövde
- Özellikler
- Ortam
- LocalEnvironment
- DestinationList
- ExceptionList
- InputBody
- InputRoot
- InputProperties
- InputLocalOrtamı
- InputDestinationListesi
- InputExceptionListesi
- OutputRoot
- OutputLocalOrtamı
- OutputDestinationListesi
- OutputExceptionListesi

## Özellik tanımlayıcı alanları MQRFH2 üstbilgilerine eşleniyor

Bir özellik MQRFH2 ögesine çevrildiğinde, özellik tanımlayıcısının önemli alanlarını belirtmek için aşağıdaki öge öznitelikleri kullanılır: Bu, MQPD alanlarının MQRFH2 öge özniteliklerine nasıl çevrildiği ele alınmıştır.

### Destek

Support (Destek) özelliği tanımlayıcısı alanı üç öge öznitelğine bölünür.

- **sr** ögesi özniteliği, MQP\_REJECT\_UNSUP\_MASK bit maskesindeki değerleri belirtir.
- **sa** ögesi özniteliği, MQPD\_ACCEPT\_UNSUP\_MASK bit maskesindeki değerleri belirtir.
- **sx** ögesi özniteliği, MQPD\_ACCEPT\_UNSUP\_IF\_XMIT\_MASK bit maskesindeki değerleri belirtir.

Bu öge öznitelikleri yalnızca < mq> klasöründe geçerlidir ve özellikler içeren diğer klasörlerde ayarlandıysa yoksayılır.

Çizelge 639. MQPD alanları MQRFH2 öge özniteliklerine eşlendi		
Destek değeri	MQRFH2 öge özniteliği	MQRFH2 öznitelik değeri
MQPD_SUPPORT_OPTIONAL	Ct	isteğe bağlı Bu varsayılan değerdir.
MQPD_SUPPORT_REQUIRY	sr	zorunlu
MQPD_SUPPORT_REQUIRED_IF_LOCAL	sx	yerel

### Bağlam

Bir özelliğin ait olduğu ileti bağlamını belirtmek için **context** ögesi öznitelğini kullanın. Yalnızca bir değer kullanın. Bu öge özniteliği, özellikleri içeren herhangi bir klasördeki bir özellik üzerinde geçerlidir.

Çizelge 640. Bağlam değerleri MQRFH2 öznitelik değerleriyle eşlendi	
Bağlam değeri	MQRFH2 öznitelik değeri
MQPD_NO_CONTEXT	yok Bu varsayılan değerdir.
MQPD_USER_BAĞLAMı	kullanıcı

### CopyOptions

Bir özelliğin kopyalanması gereken iletileri belirtmek için **copy** ögesi öznitelğini kullanın. Birden fazla değer kabul edilebilir; birden çok değeri virgülle ayırın. Örneğin **copy= 'reply'** ve **copy= 'publish, report'** her ikisi de geçerlidir. Bu öge özniteliği, özellikleri içeren herhangi bir klasördeki bir özellik üzerinde geçerlidir.

**Not:** Öznitelik tanımlamasında, tek tırnak işareti ya da çift tırnak işaretleri geçerli bir kullanıdır; örneğin, **copy= 'reply'** ya da **copy="report"**

Çizelge 641. CopyOption değerleri MQRFH2 öznitelik değerleriyle eşlendi	
CopyOption değeri	MQRFH2 öznitelik değeri
MQPD_COPY_FORWY	ileri
MQPD_COPY_REPLY	Yanıt
MQPD_COPY_REPORT	rapor

Çizelge 641. CopyOption değerleri MQRFH2 öznitelik değerleriyle eşlendi (devamı var)	
CopyOption değeri	MQRFH2 öznitelik değeri
MQPD_COPY_YAYINLAMA	yayınlamak
MQPD_COPY_ALL	tümü Bunu başka bir değerle belirtmeyin. Başka bir değerle kullanıldığında, bu değer <b>none</b> dışındaki herhangi bir değere göre öncelikli olur.
MQPD_COPY_XX_ENCODE_CASE_ONE default	varsayılan Bu varsayılan değerdir. Bu, MQCOPY_FORWALI, MQCOPY_REPORT ve MQCOPY_PATLADIN üç değerini belirlemekle eşdeğerdir. Bunu başka bir değerle belirtmeyin.
MQPD_COPY_NONE	yok Bunu başka bir değerle belirtmeyin. Başka bir değerle kullanıldığında, bu değer öncelikli olarak uygulanır.

### < mq> MQRFH2 klasörüne yönelik kısıtlamalar

Bir ileti bir kuyruğa konduğunda, iletinin MQtanımlı özelliklerine göre işlenebilmesi için bir < mq> klasörü aranır. MQtanımlı özelliklerin verimli bir şekilde ayrıştırılmasına izin vermek için, klasör için aşağıdaki kısıtlamalar geçerli olur:

- Yalnızca iletteki ilk önemli < mq> klasöründeki özellikler MQtarafından üzerinde işlem yapar; iletteki diğer herhangi bir < mq> klasöründeki özellikler yoksayılr.
- Klasör UTF-8' de yer aldıysa, klasörde yalnızca tek baytlık UTF-8 karakterlere izin verilir. Klasördeki çok byte 'lık bir karakter, ayrıştırmanın başarısız olmasına neden olabilir ve iletinin reddedilmesine neden olabilir.
- < mq> klasörüne MQRFH2 gruplarını eklemeyin. The presence of Unicode character U+003C in a property value will cause the message to be rejected.
- Klasör içinde çıkış dizgileri kullanmayın. Çıkış dizgisi, ögenin gerçek değeri olarak işlem görür.
- Yalnızca Unicode karakteri U+0020 , klasör içinde beyaz alan olarak işlenir. Diğer tüm karakterler önemli olarak ele alınır ve klasörün ayrıştırılmasına neden olabilir ve iletinin reddedilmesine neden olabilir.

< mq> klasörünün ayrıştırılması başarısız olursa ya da klasör bu kısıtlamaları gözlemiyorsa, ileti CompCode **MQCC\_FAILED** ve Neden **MQRC\_RFH\_RESTRICTED\_FORMAT\_ERR** ile reddedilir.

### Geçerli olmayan MQRFH2 üstbilgileri

Bir MQPUT, MQPUT1 ya da MQGET çağrı işlemleri sırasında, iletelerde bulunan tüm MQRFH2 üstbilgilerinin kısmi bir ayrıştırması, hangi klasörlerin dahil olduğunu denetlemek ve klasörlerin özellikleri içerip içermediğini saptamak için oluşabilir. Overview of MQRFH2 headers that are not valid.

Yapı geçerli olmadığı için iletinin kısmi ayrıştırması başarıyla tamamlanamazsa, örneğin, StructLength alanı çok küçük olduğundan, aşağıdaki gibi:

- MQPUT ya da MQPUT1 çağrısı, MQRC\_RFH\_ERROR neden koduyla başarısız olur; uygulamanın bazı IBM WebSphere MQ 7 seçeneğini içerdiği saptanması durumunda, var olan uygulamaların başarısız olması gerekir.
- MQGET çağrısı başarıyla geri döner ve hatayı içeren MQRFH2 , belirttiğiniz arabellekte döndürülür.

Kismi ayrıştırma başarısız olursa, belirli bir klasörün özellikler içerip içermediği saptanamazsa, örneğin, klasör <<jmsbaşlar, bu nedenle ayrıştırma başarısız olur ve klasör adı belirlenmeden önce ayrıştırma başarısız olur:

- MQPUT ya da MQPUT1 çağrısı, MQRC\_RFH\_FORMAT\_ERROR neden koduyla başarısız olur; uygulamanın bazı IBM WebSphere MQ 7 seçeneğini içerdiği saptanması durumunda, var olan uygulamaların başarısız olması gerekir.
- MQGET çağrısı başarıyla geri döner ve hatayı içeren MQRFH2 , belirttiğiniz arabellekte döndürülür.
- Kuyruk yöneticisi içinde dahili olarak, hatalı biçimlendirilmiş klasör nedeniyle ileti reddedilmez; ancak, klasör her zaman içinde hiçbir özellik içermediği gibi işlem görür.

İleti, kuyruk yöneticisi ağı üzerinden, böyle bir sözdizimi hatası içeren, ancak hiçbir zaman ayrıştırılmayan ve algılanan bir klasörle akış yapabilir; iletteki bir ya da daha çok klasör aşağıdaki gibi olur:

- Geçerli
- Başarıyla ayrıştırıldı
- İletinin işlenmesinde kullanılır

Bu nedenle, algılama garanti edilmez.

Uygulamalarınızdan biri bir özelliğe erişmek için “MQSETMP-İleti özelliğini ayarla” sayfa 767’ye da MQINQMP kullanıyorsa ve bu durumda bu işlem bir MQRFH2 klasörünün tam olarak ayrıştırılmasına neden olur; ancak, ayrıştırma işleminin tamamlanamaması gibi bir hata saptanırsa, API çağrısına uygun bir dönüş kodu bildirilir. Uygulamada kullanılacak hiçbir özellik, uygulama tarafından kullanılabilir kılınmaz.

Bir MQRFH2 klasörünü tam olarak ayrıştırmak için bir girişimde bulunulursa ve ayrıştırıcı, tanınmayan öge öznitelikleri ya da tanınmayan bir veri tipi bulursa, ayrıştırma devam eder ve hiçbir uyarı yayınlanmamasıyla başarıyla tamamlanır; bu bir ayrıştırma hatası oluşturmaz.

## Kod sayfası dönüşümü

Bu bölümde, kod kümesi adları ve CCSID ' ler, ulusal dil, z/OS dönüştürme, IBM i dönüştürme, ve Unicode dönüştürme desteği anlatılır.

Her ulusal dil bölümü aşağıdaki bilgileri listeler:

- Desteklenen yerel CCSID ' ler
- Desteklenmeyen kod sayfası dönüştürmeleri

Bilgilerde aşağıdaki terimler kullanılmıştır:

**AIX** **AIX**

IBM MQ for AIXdeğerini belirtir.

**Linux** **Linux**

Indicates IBM MQ for Linux for Intel and IBM MQ for Linux for zSeries.

**IBM i** **OS/400**

IBM MQ for IBM i değerini belirtir.

**Windows** **Windows**

IBM MQ for Windowsdeğerini belirtir.

**z/OS** **z/OS**

IBM MQ for z/OSdeğerini belirtir.

Veri dönüştürme için varsayılan değer, hedef (alma) sisteminde gerçekleştirilecek dönüştürmeye ilişkin varsayılan değerdir.

Kaynak ürün, dönüştürmeyi destekliyorsa, kanal özniteliği DÖN ' un kaynakta YES değerine ayarlanarak değiş tokuş edilen bir kanal kurulabilir ve veri değış tokuş edilir.

**Not:**

1. IBM MQ MQI client bilgileri için dönüştürme işlemi sunucuda gerçekleşir; bu nedenle, sunucunun istemci CCSID 'den sunucu CCSID' ye dönüşümü desteklimesi gerekir.
2. The conversion might include support added by CSD/PTF to the latest version of IBM MQ. Bu dönüştürmeyi etkinleştirmek için bir CSD/PTF kurup kurmanıza gerek olup olmadığını görmek için en son hizmet düzeyinin içeriğini denetleyin.
3. IBM MQ kuyruk yöneticisi CCSID 'si karma ya da SBCS olmalıdır.
4. İşletim sisteminin desteklemediği bazı CCSID ' ler (örneğin, AIXüzerinde850) uygulama tarafından kullanılabilir ve IBM MQ kuyruk yöneticisi CCSID 'si olarak da ayarlanabilir. Bu, yalnızca geriye doğru uyumluluk amacıyla izin verilir ve ilgili dönüştürme çizelgeleri kurulu değilse, dönüştürme işlemi başarısız olur.

CCSID numaralarının bazıları ile bazı endüstri kod kümesi adları arasında çapraz başvuru için [Çizelge 642 sayfa 923](#) ' e bakın.

**İlgili başvurular**

“Ulusal diller” sayfa 924

Bu bilgiler, IBM MQtarafından desteklenen dilleri içerir.

**Kod kümesi adları ve CCSID ' ler**

Her bir kod kümesi adı için kod kümesi adları ve ilgili CCSID ' ler.

**z/OS** IBM MQ for z/OS , dile özgü tablolarda listelenenden daha fazla dönüştürme sağlar. Dönüştürmelerin tam listesi için bkz. [Çizelge 675 sayfa 949](#).

<i>Çizelge 642. Kod kümesi adları ve CCSID ' ler</i>	
<b>Kod kümesi adları</b>	<b>CCSID ' ler</b>
ISO 8859-1	819
ISO 8859-2	912
ISO 8859-3	913
ISO 8859-5	915
ISO 8859-6	1089
ISO 8859-7	813
ISO 8859-8	916
ISO 8859-9	920
ISO 8859-13	921
ISO 8859-15 (euro)	923
big5	950
eucJP	954 5050 33722
eucKR	970
eucTW	964
eucCN	1383
PCK	943
GBK	1386
koi8-r	878

## Ulusal diller



Bu bilgiler, IBM MQ tarafından desteklenen dilleri içerir.

IBM MQ tarafından desteklenen diller şunlardır:

- US English - see topic [“ABD İngilizcesi” sayfa 924](#)
- Almanca-see topic [“Almanca” sayfa 925](#)
- Danca ve Norveççe-bkz. [“Danimarka ve Norveççe” sayfa 926](#)
- Fince ve İsveççe-bkz. [“Fince ve İsveççe” sayfa 926](#)
- İtalyanca-bkz. [“İtalyanca” sayfa 927](#)
- İspanyolca-bkz. [“İspanyolca” sayfa 928](#)
- İngiliz İngilizcesi/Gaelic-bkz. [“İngiliz İngilizcesi /Gaelic” sayfa 929](#)
- Fransızca-bkz. [“Fransızca” sayfa 929](#)
- Çok dilli-konuya bakın [“Çok dilli” sayfa 930](#)
- Portekizce-bkz. [“Portekizce” sayfa 931](#)
- İzlandaca-konuya bakın [“İzlandaca” sayfa 931](#)
- Doğu Avrupa dilleri-bkz. [“Doğu Avrupa dilleri” sayfa 932](#)
- Kiril-konuya bakın [“Kiril” sayfa 933](#)
- Estonca-bkz. [“Estonyaca” sayfa 934](#)
- Letonya ve Litvanca-see topic [“Letonca ve Litvanca” sayfa 935](#)
- Ukraynca-konuya bakın [“Ukraynaca” sayfa 936](#)
- Yunanca-see topic [“Yunanca” sayfa 937](#)
- Türkçe-bkz. [“Türkçe” sayfa 938](#)
- İbranice-bkz. [“İbranice” sayfa 938](#)
- Farsça-konuya bakın [“Farsça” sayfa 940](#)
- Urdu - see topic [“Urduca” sayfa 941](#)
- Thai - see topic [“Tay dili” sayfa 941](#)
- Lao - see topic [“Lao” sayfa 942](#)
- Vietnam dili-konu başlığı [“Vietnam dili” sayfa 942](#)
- Japonca Latin SBCS-bkz. [“Japonca Latin SBCS” sayfa 943](#)
- Japonca Katakana SBCS-bkz. [“Japonca Katakana SBCS” sayfa 944](#)
- Japonca Kanji/Latin Karma-see topic [“Japonca Kanji/Latin Karma” sayfa 945](#)
- Japonca Kanji/Katakana Karma-see topic [“Japonca Kanji/Katakana Karışık” sayfa 946](#)
- Kore dili-bkz. [“Korece” sayfa 947](#)
- Basitleştirilmiş Çince-see topic [“Yalınlaştırılmış Çince” sayfa 948](#)
- Geleneksel Çince-bkz. [“Geleneksel Çince” sayfa 948](#)

## ABD İngilizcesi

ABD İngilizcesi için CCSID 'lerin ve CCSID' lerin ayrıntıları.

<i>Çizelge 643. Desteklenen platformlarda ABD İngilizcesi için yerel CCSID ' ler</i>	
Altyapı	Yerel CCSID ' ler
 IBM i	37, 924, 1140
 z/OS	



Çizelge 643. Desteklenen platformlarda ABD İngilizcesi için yerel CCSID ' ler (devamı var)

Altyapı	Yerel CCSID ' ler
 AIX	819, 923, 5348
 Windows	437, 850, 1252, 5348, 858
 Linux	819, 923
Apple istemcisi	1275

Tüm istemci dışı altyapılar, aşağıdaki kural dışı durumlar dışında, kendi yerel CCSID 'leri ile diğer platformların yerel CCSID' leri arasındaki dönüşümü destekler.

## IBM i



Kod sayısı:

### 37

923, 858 kod sayfalarına dönüştürülmez

### 924

437, 858, 1051, 1140, 1252, 1275, 5348 kod sayfalarına dönüştürülmez






### 1140

924, 1051, 1275 kod sayfalarına dönüştürülmez

## Almanca

Almanca için CCSID 'lerin ve CCSID' lerin ayrıntıları.

Çizelge 644. Desteklenen platformlarda Almanca için yerel CCSID ' ler

Altyapı	Yerel CCSID ' ler
 IBM i	273, 924, 1141
 z/OS	
 AIX	819, 923, 5348
 Windows	437, 850, 858, 1252, 5348
 Linux	819, 923
Apple istemcisi	1275

Tüm istemci dışı altyapılar, aşağıdaki kural dışı durumlar dışında, kendi yerel CCSID 'leri ile diğer platformların yerel CCSID' leri arasındaki dönüşümü destekler.

## IBM i



Kod sayısı:

### 273

858, 923, 924, 1275 kod sayfalarına dönüştürülmez

**924**






273, 437, 858, 1051, 1141, 1252, 1275, 5348 kod sayfalarına dönüştürülmez

**1141**

924, 1051, 1275 kod sayfalarına dönüştürülmez

**Danimarka ve Norveççe**

Danca ve Norveççe için CCSID ' lerin ayrıntıları ve CCSID dönüştürmesi.

<i>Çizelge 645. Desteklenen platformlarda Danca ve Norveççe için yerel CCSID ' ler</i>	
<b>Altyapı</b>	<b>Yerel CCSID ' ler</b>
 IBM i	277, 924, 1142
 z/OS	
 AIX	819, 923, 5348
 Windows	850, 858, 865, 1252, 5348
 Linux	819, 923
Apple istemcisi	1275

Tüm istemci dışı altyapılar, aşağıdaki kural dışı durumlar dışında, kendi yerel CCSID 'leri ile diğer platformların yerel CCSID' leri arasındaki dönüştürmeyi destekler.

**IBM i**

Kod sayfası:

**277**

858, 923, 924, 1275 kod sayfalarına dönüştürülmez

**924**

277, 858, 865, 1051, 1142, 1252, 1275, 5348 kod sayfalarına dönüştürülmez

**1142**

924, 865, 1051, 1275 kod sayfalarına dönüştürülmez

**AIX**

Kod sayfası:

**819**

865 kod sayfasına dönüştürülmez

**Windows**

Kod sayfası:






**865**

1051, 1275 kod sayfalarına dönüştürülmez

**Fince ve İsveççe**

Fin ve İsveççe için CCSID ' lerin ayrıntıları ve CCSID dönüştürmesi.

Çizelge 646. Desteklenen platformlarda Fince ve İsveççe için yerel CCSID ' ler

Altyapı	Yerel CCSID ' ler
 IBM i	278, 924, 1143
 z/OS	
 AIX	819, 923, 5348
 Windows	437, 850, 858, 865, 1252, 5348
 Linux	819, 923
Apple istemcisi	1275

Tüm istemci dışı altyapılar, aşağıdaki kural dışı durumlar dışında, kendi yerel CCSID 'leri ile diğer platformların yerel CCSID' leri arasındaki dönüşümü destekler.

## IBM i



Kod sayfası:

### 278

858, 923, 924, 1275 kod sayfalarına dönüştürülmez

### 924

278, 437, 858, 865, 1051, 1143, 1252, 1275, 5348 kod sayfalarına dönüştürülmez

### 1143

865, 924, 1051, 1275 kod sayfalarına dönüştürülmez

## AIX



Kod sayfası:

### 819

865 kod sayfasına dönüştürülmez

### 850

865 kod sayfasına dönüştürülmez

## Windows



Kod sayfası:






### 865

1051, 1275 kod sayfalarına dönüştürülmez

## İtalyanca

CCSID ' lerin Ayrıntıları ve İtalyanca için CCSID dönüştürmesi.

Çizelge 647. Desteklenen platformlarda İtalyanca için yerel CCSID ' ler

Altyapı	Yerel CCSID ' ler
 IBM i	280, 924, 1144
 z/OS	
 AIX	819, 923, 5348
 Windows	437, 850, 858, 1252, 5348
 Linux	819, 923
Apple istemcisi	1275

Tüm istemci dışı altyapılar, aşağıdaki kural dışı durumlar dışında, kendi yerel CCSID 'leri ile diğer platformların yerel CCSID' leri arasındaki dönüştürmeyi destekler.

## IBM i



Kod sayısı:

### 280

858, 923, 924, 1275 kod sayfalarına dönüştürülmez

### 924

280, 437, 858, 1051, 1144, 1252, 1275, 5348 kod sayfalarına dönüştürülmez






### 1144

924, 1051, 1275 kod sayfalarına dönüştürülmez

## İspanyolca

CCSID ' lerin ayrıntıları ve İspanyolca için CCSID dönüştürmesi.

Çizelge 648. Desteklenen platformlarda İspanyolca için yerel CCSID ' ler

Altyapı	Yerel CCSID ' ler
 IBM i	284, 924, 1145
 z/OS	
 AIX	819, 923, 5348
 Windows	437, 850, 858, 1252, 5348
 Linux	819, 923
Apple istemcisi	1275

Tüm istemci dışı altyapılar, aşağıdaki kural dışı durumlar dışında, kendi yerel CCSID 'leri ile diğer platformların yerel CCSID' leri arasındaki dönüştürmeyi destekler.

## IBM i



Kod sayfası:

**284**

858, 923, 924, 1275 kod sayfalarına dönüştürülmez

**924**






284, 437, 858, 1051, 1145, 1252, 1275, 5348 kod sayfalarına dönüştürülmez

**1145**

924, 1051, 1275 kod sayfalarına dönüştürülmez

**İngiliz İngilizcesi /Gaelc**

İngiliz İngilizcesi/Gaelic için CCSID 'lerin ve CCSID' lerin ayrıntıları.

<i>Çizelge 649. Desteklenen platformlarda İngiltere İngilizcesi/Gaelic için yerel CCSID ' ler</i>	
<b>Altyapı</b>	<b>Yerel CCSID ' ler</b>
 IBM i	285, 924, 1146
 z/OS	
 AIX	819, 923, 5348
 Windows	437, 850, 858, 1252, 5348
 Linux	819, 923
Apple istemcisi	1275

Tüm istemci dışı altyapılar, aşağıdaki kural dışı durumlar dışında, kendi yerel CCSID 'leri ile diğer platformların yerel CCSID' leri arasındaki dönüştürmeyi destekler.

**IBM i**



Kod sayfası:

**285**

858, 923, 924, 1275 kod sayfalarına dönüştürülmez

**924**





285, 437, 858, 1051, 1146, 1252, 1275, 5348 kod sayfalarına dönüştürülmez


**1146**

924, 1051, 1275 kod sayfalarına dönüştürülmez

**Fransızca**

CCSID ' lerin ayrıntıları ve Fransızca için CCSID dönüştürmesi.

<i>Çizelge 650. Desteklenen platformlarda Fransızca yerel CCSID ' ler</i>	
<b>Altyapı</b>	<b>Yerel CCSID ' ler</b>
 IBM i	297, 924, 1147
 z/OS	
 AIX	819, 923, 5348
 Windows	437, 850, 858, 1252, 5348

<i>Çizelge 650. Desteklenen platformlarda Fransızca yerel CCSID ' ler (devamı var)</i>	
Altyapı	Yerel CCSID ' ler
 Linux	819, 923
Apple istemcisi	1275

Tüm istemci dışı altyapılar, aşağıdaki kural dışı durumlar dışında, kendi yerel CCSID 'leri ile diğer platformların yerel CCSID' leri arasındaki dönüştürmeyi destekler.

## IBM i



Kod sayısı:

### 297

858, 923, 924, 1275, 5348 kod sayfalarına dönüştürülmez

### 924






297, 437, 858, 1051, 1147, 1252, 1275, 5348 kod sayfalarına dönüştürülmez

### 1147

924, 1051, 1275 kod sayfalarına dönüştürülmez

## Çok dilli

CCSID ' lerin Ayrıntıları ve Çok Dilli için CCSID dönüştürmesi.

<i>Çizelge 651. Desteklenen platformlarda çok dilli dönüştürme için yerel CCSID ' ler</i>	
Altyapı	Yerel CCSID ' ler
 IBM i	500, 924, 1148
 z/OS	
 AIX	819, 923, 5348
 Windows	437, 850, 858, 1252, 5348
 Linux	819, 923
Apple istemcisi	1275

Tüm istemci dışı altyapılar, aşağıdaki kural dışı durumlar dışında, kendi yerel CCSID 'leri ile diğer platformların yerel CCSID' leri arasındaki dönüştürmeyi destekler.

## IBM i



Kod sayısı:

### 500

858, 923 kod sayfalarına dönüştürülmez

### 924






437, 858, 1051, 1148, 1252, 1275, 5348 kod sayfalarına dönüştürülmez

### 1148

924, 1051, 1275 kod sayfalarına dönüştürülmez

## Portekizce

Portekizce 'ye ilişkin CCSID 'lerin ve CCSID' lerin ayrıntıları.

Çizelge 652. Desteklenen platformlarda Portekizce için yerel CCSID ' ler	
Altyapı	Yerel CCSID ' ler
 IBM i	37, 500, 924, 1140
 z/OS	500, 924, 1140
 AIX	819, 923, 5348
 Windows	850, 858, 860, 1252, 5348
 Linux	819, 923
Apple istemcisi	1275

Tüm istemci dışı altyapılar, aşağıdaki kural dışı durumlar dışında, kendi yerel CCSID 'leri ile diğer platformların yerel CCSID' leri arasındaki dönüştürmeyi destekler.

### IBM i



Kod sayfası:

#### 37

858, 923, 1275 kod sayfalarına dönüştürülmez

#### 500

858, 923, 1275 kod sayfalarına dönüştürülmez

#### 924

858, 860, 1051, 1140, 1252, 1275, 5348 kod sayfalarına dönüştürülmez

#### 1140

860, 924, 1051, 1275 kod sayfalarına dönüştürülmez

### Windows






Kod sayfası:

#### 860

1051, 1275 kod sayfalarına dönüştürülmez

### İzlandaca

İzlandaca için CCSID ' lerin ve CCSID dönüştürmenin ayrıntıları.

Çizelge 653. Desteklenen platformlarda İzlandaca için yerel CCSID ' ler	
Altyapı	Yerel CCSID ' ler
 IBM i	871, 924, 1149
 z/OS	
 AIX	819, 923, 5348

Çizelge 653. Desteklenen platformlarda İzlandaca için yerel CCSID ' ler (devamı var)

Altyapı	Yerel CCSID ' ler
 Windows	850, 858, 861, 1252, 5348
 Linux	819, 923
Apple istemcisi	1275

Tüm istemci dışı altyapılar, aşağıdaki kural dışı durumlar dışında, kendi yerel CCSID 'leri ile diğer platformların yerel CCSID' leri arasındaki dönüşümü destekler.

## IBM i



Kod sayısı:

### 871

858, 923, 924, 1275, 5348 kod sayfalarına dönüştürülmez

### 924

858, 861, 871, 1051, 1149, 1252, 1275, 5348 kod sayfalarına dönüştürülmez

### 1149

924, 1051, 1275 kod sayfalarına dönüştürülmez

## Windows



Kod sayısı:






### 861

1051, 1275 kod sayfalarına dönüştürülmez

## Doğu Avrupa dilleri

Doğu Avrupa Dilleri için CCSID 'lerin ve CCSID' lerin ayrıntıları. Bu CCSID ' leri kullanan tipik diller arasında Arnavutça, Hırvatça, Çekçe, Macarca, Lehçe, Romence, Sırpça, Slovakça ve Slovence dilleri yer alır.

Çizelge 654. Desteklenen platformlarda Doğu Avrupa dilleri için yerel CCSID ' ler

Altyapı	Yerel CCSID ' ler
 IBM i	870, 1153
 z/OS	
 Windows	852, 1250, 5346, 9044
 AIX	912
 Linux	
Doğu Avrupa Apple istemcisi	1282
Romen Apple istemcisi	1285
Hırvat Apple istemcisi	1284



Tüm istemci dışı altyapılar, aşağıdaki kural dışı durumlar dışında, kendi yerel CCSID 'leri ile diğer platformların yerel CCSID' leri arasındaki dönüşümü destekler.

## z/OS



Kod sayısı:

### 870

1284, 1285 kod sayfalarına dönüştürülmez

### 1153

1250, 1284, 1285 kod sayfalarına dönüştürülmez

## IBM i



Kod sayısı:

### 870

1284, 1285, 5346, 9044 kod sayfalarına dönüştürülmez

### 1153

1282, 1284, 1285, 5346, 9044 kod sayfalarına dönüştürülmez

## , Linux



Kod sayısı:

### 912

1284, 1285 kod sayfalarına dönüştürülmez

## Windows



Kod sayısı:

### 852

1284, 1285 kod sayfalarına dönüştürülmez

### 1250

1284, 1285 kod sayfalarına dönüştürülmez

### 9044



912, 1282, 1284, 1285 kod sayfalarına dönüştürülmez

## Kiril

CCSID ' lerin ayrıntıları ve Kiril CCSID dönüştürmesi. Bu CCSID ' leri kullanan tipik diller arasında Belarus, Bulgarca, Makedonca, Rusça ve Sırpça yer alıyor.

Çizelge 655. Desteklenen platformlarda Kiril yerel CCSID ' ler	
Altyapı	Yerel CCSID ' ler
z/OS	1025
IBM i	880, 1025
Windows	855, 866, 1131, 1251, 5347

Çizelge 655. Desteklenen platformlarda Kiril yerel CCSID ' ler (devamı var)

Altyapı	Yerel CCSID ' ler
 AIX	915
 Linux	
Apple istemcisi	1283

Tüm istemci dışı altyapılar, aşağıdaki kural dışı durumlar dışında, kendi yerel CCSID 'leri ile diğer platformların yerel CCSID' leri arasındaki dönüştürmeyi destekler.

## IBM i



Kod sayfası:

### 880

855, 866, 878, 1131, 5347 kod sayfalarına dönüştürülmez

### 1025

878, 5347 kod sayfalarına dönüştürülmez

## Windows



Kod sayfası:

### 855

1131 kod sayfasına dönüştürülmez

### 866

1131 kod sayfasına dönüştürülmez




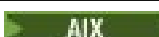

### 1131

855, 866, 880, 1283 kod sayfalarına dönüştürülmez

## Estonyaca

Estonyaca ilişkin CCSID 'lerin ve CCSID' lerin ayrıntıları.

Çizelge 656. Desteklenen platformlarda Estonya için yerel CCSID ' ler

Altyapı	Yerel CCSID ' ler
 IBM i	1122, 1157
 z/OS	
 Windows	902, 922, 1257, 5353, 9449
 AIX	902, 922
 Linux	

Tüm platformlar, yerel CCSID 'leri ile diğer platformların yerel CCSID' leri arasında, aşağıdaki istisnalar dışında dönüştürmeyi destekler.

## z/OS



Kod sayfası:

### 1122

902, 1157, 9449 kod sayfalarına dönüştürülmez

### 1157

922, 1122, 1257, 9449 kod sayfalarına dönüştürülmez

## IBM i



Kod sayfası:

### 1122

902, 5353, 9449 kod sayfalarına dönüştürülmez

### 1157

922, 5353, 9449 kod sayfalarına dönüştürülmez

## Linux



Kod sayfası:

### 902

922, 1122, 9449 kod sayfalarına dönüştürülmez

### 922

902, 1157, 9449 kod sayfalarına dönüştürülmez

## Windows



Kod sayfası:

### 5353

9449 kod sayfasına dönüştürülmez

### 9449

902, 922, 1122, 1157, 1257, 5353 kod sayfalarına dönüştürülmez

### 902



922, 1122, 9449 kod sayfalarına dönüştürülmez

## Letonca ve Litvanca

Letonya ve Litvanca için CCSID 'lerin ve CCSID' lerin ayrıntıları.

Çizelge 657. Desteklenen platformlarda Letonya ve Litvanyaca yerel CCSID ' ler	
Altyapı	Yerel CCSID ' ler
IBM i	1112, 1156
z/OS	
Windows	901, 921, 1257, 5353, 9449

Çizelge 657. Desteklenen platformlarda Letonya ve Litvanyaca yerel CCSID ' ler (devamı var)

Altyapı	Yerel CCSID ' ler
 AIX	901, 921
 Linux	

Tüm platformlar, yerel CCSID 'leri ile diğer platformların yerel CCSID' leri arasında, aşağıdaki istisnalar dışında dönüştürmeyi destekler.

## z/OS



Kod sayfası:

### 1112

901, 1156, 9449 kod sayfalarına dönüştürülmez

### 1156

901, 1156, 9449 kod sayfalarına dönüştürülmez

## IBM i



Kod sayfası:

### 1112

5353 kod sayfasına dönüştürülmez

### 1153

921, 5353, 9449 kod sayfalarına dönüştürülmez

## Linux



Kod sayfası:

### 902

921, 1112, 1257, 9449 kod sayfalarına dönüştürülmez

### 921

901, 1156, 9449 kod sayfalarına dönüştürülmez

## Windows



Kod sayfası:

### 901

921, 1112, 1257, 9449 kod sayfalarına dönüştürülmez

### 5355

9449 kod sayfasına dönüştürülmez






### 9449

901, 921, 1112, 1156, 1257 kod sayfalarına dönüştürülmez

## Ukraynaca

Ukraynaca için CCSID 'lerin ve CCSID' lerin ayrıntıları.

Çizelge 658. Desteklenen platformlarda Ukranian için yerel CCSID ' ler

Altyapı	Yerel CCSID ' ler
 IBM i  z/OS	1123
 Windows	1124, 1125, 1251, 5347
 AIX  Linux	1124

Tüm platformlar, yerel CCSID 'leri ile diğer platformların yerel CCSID' leri arasında, aşağıdaki istisnalar dışında dönüştürmeyi destekler.

### IBM i



Kod sayfası:

#### 1123

5347 kod sayfasına dönüştürülmez

### Windows



Kod sayfası:






#### 1125

1123 kod sayfasına dönüştürülmez

### Yunanca

GCCSID ' lerin ayrıntıları ve Yunanca için CCSID dönüştürmesi.

Çizelge 659. Desteklenen platformlarda Yunanca için yerel CCSID ' ler

Altyapı	Yerel CCSID ' ler
 IBM i  z/OS	875
 Windows	869, 1253, 5349
 AIX  Linux NCR	813
Apple istemcisi	1280
DOS istemcisi	737

Tüm istemci dışı altyapılar, aşağıdaki kural dışı durumlar dışında, kendi yerel CCSID 'leri (CCSID' ler), diğer altyapıların yerel CCSID ' leri (CCSID) arasındaki dönüştürmeyi destekler.

## IBM i



Kod sayfası:

**875**

5349 kod sayfasına dönüştürülmez

## Windows



Kod sayfası:

**1253**

737 kod sayfasına dönüştürülmez

**5349**

737 kod sayfasına dönüştürülmez

## Türkçe

CCSID ' lerin Ayrıntıları ve Türkçe için CCSID dönüştürmesi.

<i>Çizelge 660. Desteklenen platformlarda Türkçe için yerel CCSID ' ler</i>	
Altyapı	Yerel CCSID ' ler
IBM i z/OS	1026
Windows	857, 1254, 5350
AIX Linux	920
Apple istemcisi	1281

Tüm istemci dışı altyapılar, aşağıdaki kural dışı durumlar dışında, kendi yerel CCSID 'leri ile diğer platformların yerel CCSID' leri arasındaki dönüştürmeyi destekler.

## IBM i



Kod sayfası:

**1026**

5350 kod sayfasına dönüştürülmez

## İbranice

İbranice için CCSID ' lerin ayrıntıları ve CCSID dönüştürmesi.

<i>Çizelge 661. Desteklenen platformlarda İbranice için yerel CCSID ' ler</i>	
Altyapı	Yerel CCSID ' ler
z/OS	424, 803, 4899, 12712
IBM i	424

Çizelge 661. Desteklenen platformlarda İbranice için yerel CCSID ' ler (devamı var)

Altyapı	Yerel CCSID ' ler
 AIX	916, 9048
 Windows	1255, 5351
 Linux	916

Tüm platformlar, yerel CCSID 'leri ile diğer platformların yerel CCSID' leri arasında, aşağıdaki istisnalar dışında dönüştürmeyi destekler.

## z/OS



Kod sayfası:

### 424

867, 4899, 9048, 12712 kod sayfalarına dönüştürülemez

### 803

867, 4899, 5351, 9048, 12712 kod sayfalarına dönüştürülmez

### 4899

424, 803, 856, 862, 916, 1255 kod sayfalarına dönüştürülmez

### 12712

424, 803, 856, 916, 1255 kod sayfalarına dönüştürülmez

## IBM i



Kod sayfası:

### 424

803, 867, 4899, 5351, 9048, 12712 kod sayfalarına dönüştürülmez

424 kod sayfası da, 856 olan bir değişken olan CCSID 4952 'ye ve bu değere dönüştürülür.

## AIX



Kod sayfası:

### 916

867, 4899, 9048, 12712 kod sayfalarına dönüştürülemez

### 9048

424, 803, 856, 862, 916, 1255 kod sayfalarına dönüştürülmez

## Windows



Kod sayfası:

### 1255



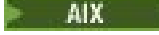


867, 4899, 9048, 12712 kod sayfalarına dönüştürülemez

### 5351

803 kod sayfasına dönüştürülmez

## Arapça

Arapça için CCSID ' lerin ve CCSID dönüştürmenin ayrıntıları

Çizelge 662. Desteklenen altyapılarda Arapça için yerel CCSID ' ler	
Altyapı	Yerel CCSID ' ler
 IBM i	420
 z/OS	
 AIX	1046, 1089
	1089 (nota bakın)
 Windows	720, 864, 1256, 5352
 Linux	1089

Tüm platformlar, yerel CCSID 'leri ile diğer platformların yerel CCSID' leri arasında, aşağıdaki istisnalar dışında dönüştürmeyi destekler.

### IBM i



Kod sayfası:

**420**

5352 kod sayfasına dönüştürülmez

### Linux Tru64



Kod sayfası:

**1089**

Kod sayfasına dönüştürülmez. 720

### Windows



Kod sayfası:

**720**



1089, 5352 kod sayfalarına dönüştürülmez

**5352**

Kod sayfasına dönüştürülmez. 720




## Farsça

CCSID ' lerin ayrıntıları ve Farsça için CCSID dönüştürmesi.

Çizelge 663. Desteklenen platformlarda Farsça için yerel CCSID ' ler	
Altyapı	Yerel CCSID ' ler
 IBM i	1097
 z/OS	



Çizelge 663. Desteklenen platformlarda Farsça için yerel CCSID ' ler (devamı var)

Altyapı	Yerel CCSID ' ler
 AIX	1098 (nota bakın)
 Linux	
 Windows	




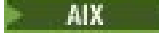

**Not:** Bu altyapılar için yerel CCSID değeri standartlaştırılmamış ve değiştirilebilir.

Tüm platformlar, yerel CCSID 'leri ile diğer platformların yerel CCSID' leri arasındaki dönüşümü destekler.

### Urduca

Urdu 'ya ilişkin CCSID 'lerin ve CCSID' lerin ayrıntıları.

Çizelge 664. Desteklenen platformlarda Urduca için yerel CCSID ' ler

Altyapı	Yerel CCSID ' ler
 IBM i	918
 z/OS	
 Windows	868
 AIX	1006
 Linux	

Tüm platformlar, yerel CCSID 'leri ile diğer platformların yerel CCSID' leri arasında, aşağıdaki istisnalar dışında dönüşümü destekler.

### IBM i



Kod sayfası:



#### 918

1006 kod sayfasına dönüştürülmez




### Tay dili

Thai için CCSID ' lerin ve CCSID dönüştürmenin ayrıntıları.

Çizelge 665. Desteklenen platformlarda Thai dilinde yerel CCSID ' ler

Altyapı	Yerel CCSID ' ler
 IBM i	838
 z/OS	

Çizelge 665. Desteklenen platformlarda Thai dilinde yerel CCSID ' ler (devamı var)

Altyapı	Yerel CCSID ' ler
 AIX	874 (nota bakın)
 Linux	
 Windows	






**Not:** Bu altyapılar için yerel CCSID değeri standartlaştırılmamış ve değiştirilebilir.

Tüm platformlar, yerel CCSID 'leri ile diğer platformların yerel CCSID' leri arasındaki dönüşümü destekler.

### Lao

Lao için CCSID 'lerin ve CCSID' lerin ayrıntıları.

Çizelge 666. Desteklenen platformlarda Lao için yerel CCSID ' ler




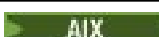

Altyapı	Yerel CCSID ' ler
 IBM i	1132
 z/OS	
 AIX	1133
 Linux	
 Windows	

Tüm platformlar, yerel CCSID 'leri ile diğer platformların yerel CCSID' leri arasındaki dönüşümü destekler.

### Vietnam dili

Vietnamca için CCSID ' lerin ayrıntıları ve CCSID dönüşümü.

Çizelge 667. Desteklenen platformlarda Vietnam dili için yerel CCSID ' ler

Altyapı	Yerel CCSID ' ler
 IBM i	1130
 z/OS	
 Windows	1258, 5354
 AIX	1129
 Linux	

Tüm platformlar, yerel CCSID 'leri ile diğer platformların yerel CCSID' leri arasında, aşağıdaki istisnalar dışında dönüşümü destekler.

### IBM i








Kod sayfası:

### 1130



1129, 5354 kod sayfalarına dönüştürülmez

## Japonca Latin SBCS

Japonca Latin SBCS için CCSID ' lerin ayrıntıları ve CCSID dönüştürmesi.

Çizelge 668. Desteklenen platformlarda Japonca Latin SBCS için yerel CCSID ' ler	
Altyapı	Yerel CCSID ' ler
 IBM i	1027
 z/OS	
 AIX	932, 5050, 33722 (bkz. Not 1)
 Windows	932, 943 (bkz. Not 2)
 Linux	943, 5050

### Not:

-  5050 ve 33722, AIX üzerinde 954 temel kod sayfasıyla ilgili CCSID'lerdir. İşletim sistemi tarafından bildirilen CCSID değeri 33722 'dir.
-  Windows NT , 932 kod sayfasını kullanır, ancak bu en iyi, 943 CCSID 'si tarafından temsil edilir. Ancak, IBM MQ 'un tüm altyapıları bu CCSID' yi desteklemiyor.

IBM MQ for Windows CCSID 932 üzerinde kod sayfası 932 'yi göstermek için kullanılır, ancak 943 'te kullanılan CCSID ' yi değiştiren . ./conv/table/ccsid.tbl kütüğüne yapılan bir değişiklik yapılabilir.

Tüm platformlar, yerel CCSID 'leri ile diğer platformların yerel CCSID' leri arasında, aşağıdaki istisnalar dışında dönüştürmeyi destekler.

## z/OS



Kod sayfası:

### 1027

932, 942, 943, 954, 5050, 33722 kod sayfalarına dönüştürülmez

## IBM i



Kod sayfası:

### 1027

932 kod sayfasına dönüştürülmez

## AIX



Kod sayfası:

### 932

1027 kod sayfasına dönüştürülmez

## 5050

1027 kod sayfasına dönüştürülmez

## 33722

1027 kod sayfasına dönüştürülmez

## Linux



Kod sayfası:

## 943

1027 kod sayfasına dönüştürülmez

## 5050

1027 kod sayfasına dönüştürülmez

## Japonca Katakana SBCS

Japonca Katakana SBCS için CCSID ' lerin ayrıntıları ve CCSID dönüştürmesi.

Çizelge 669. Desteklenen platformlarda Japonca (Katakana SBCS) yerel CCSID ' ler	
Altyapı	Yerel CCSID ' ler
IBM i	290
z/OS	
AIX	932, 5050, 33722 (bkz. Not 1)
Windows	932, 943 (bkz. Not 2)
Linux	943, 5050

### Not:

- AIX 5050 ve 33722, AIX üzerinde 954 temel kod sayfasıyla ilgili CCSID'lerdir. İşletim sistemi tarafından bildirilen CCSID değeri 33722 'dir.
- Windows NT , 932 kod sayfasını kullanır, ancak bu en iyi, 943 CCSID 'si tarafından temsil edilir. Ancak, IBM MQ 'un tüm altyapıları bu CCSID' yi desteklemiyor.  
IBM MQ for Windows CCSID 932 üzerinde kod sayfası 932 'yi göstermek için kullanılır, ancak 943 'te kullanılan CCSID ' yi değiştiren ../conv/table/ccsid.tbl kütüğüne yapılan bir değişiklik yapılabilir.
- IBM MQ , önceki dönüştürmelerin yanı sıra, aşağıdaki altyapılarda CCSID 897 'den CCSID ' ye, 37, 273, 277, 278, 280, 284, 285, 290, 297, 437, 500, 819, 850, 1027 ve 1252 'ye dönüştürmeyi destekler:

- AIX
- Linux

Tüm platformlar, yerel CCSID 'leri ile diğer platformların yerel CCSID' leri arasında, aşağıdaki istisnalar dışında dönüştürmeyi destekler.

## z/OS



Kod sayfası:

## 290

932, 943, 954, 5050, 33722 kod sayfalarına dönüştürülmez

## IBM i

IBM i

Kod sayısı:

## 290

932 kod sayfasına dönüştürülmez

## AIX

AIX

Kod sayısı:

## 932

290, 897 kod sayfalarına dönüştürülmez

## 5050

290, 897 kod sayfalarına dönüştürülmez

## 33722

290, 897 kod sayfalarına dönüştürülmez

## Linux

Linux

Kod sayısı:

## 943

290, 897 kod sayfalarına dönüştürülmez

## 5050

290, 897 kod sayfalarına dönüştürülmez

## Japonca Kanji/Latin Karma

Japonca Kanji/Latin Karma için CCSID ' lerin ayrıntıları ve CCSID dönüştürmesi.

Çizelge 670. Desteklenen platformlarda Japonca Kanji/Latin karma için yerel CCSID ' ler	
Altyapı	Yerel CCSID ' ler
IBM i IBM i z/OS z/OS	1399, 5035 (bkz. Not 1)
AIX AIX	932, 5050, 33722 (bkz. Not 2)
Windows Windows	932, 943 (bkz. Not 4)
Linux Linux	943, 5050

## Not:

- IBM i z/OS 5035, 939 kod sayfasıyla ilgili bir CCSID ' dir
- AIX 5050 ve 33722, AIX üzerinde 954 temel kod sayfasıyla ilgili CCSID'lerdir. İşletim sistemi tarafından bildirilen CCSID değeri 33722 'dir.

3. **Windows** Windows NT , 932 kod sayfasını kullanır, ancak bu en iyi, 943 CCSID 'si tarafından temsil edilir. Ancak, IBM MQ 'un tüm altyapıları bu CCSID' yi desteklemiyor.

IBM MQ for Windows CCSID 932 üzerinde kod sayfası 932 'yi göstermek için kullanılır, ancak 943 'te kullanılan CCSID ' yi değiştiren . ./conv/table/ccsid.tbl kütüğüne yapılan bir değişiklik yapılabilir.

Tüm platformlar, yerel CCSID 'leri ile diğer platformların yerel CCSID' leri arasında, aşağıdaki istisnalar dışında dönüştürmeyi destekler.

## z/OS

### z/OS

Kod sayfası:

#### 1399

954, 5035, 5050, 33722 kod sayfalarına dönüştürülmez

#### 5035

954, 1399, 5050, 33722 kod sayfalarına dönüştürülmez

## IBM i

### IBM i

Kod sayfası:

#### 1399

5039 kod sayfasına dönüştürülmez

#### 5035

5039 kod sayfasına dönüştürülmez

## Japonca Kanji/Katakana Karışık

Japonca Kanji/Katakana için CCSID ve CCSID dönüşümünün ayrıntıları Karışık.

Çizelge 671. Desteklenen platformlarda karıştırılmış Japonca Kanji/Katakana için yerel CCSID ' ler	
Hizmet olarak sunulan	Yerel CCSID ' ler
<b>z/OS</b> z/OS	1390, 5026 (bkz. Not 1)
<b>IBM i</b> IBM i	5026 (bkz. Not 1)
<b>AIX</b> AIX	932, 5050, 33722 (bkz. Not 2)
<b>Windows</b> Windows	932, 943 (bkz. Not 4)
<b>Linux</b> Linux	943, 5050

### Not:

1. **IBM i** **z/OS** EBCDIC 'de CCSID 1390 ve 5026 'nın tek baytlık kipi, farklı konumlarda, temel Latince için tipik/değişmez düzene göre küçük harfli karakterler içerir ve ileti verileri diğer CCSID' lere dönüştürülürken verilerin kaybolmaması için dikkatli olunmalıdır. Ayrıca, bu CCSID ' lerin kuyruk yöneticisinin varsayılan CCSID 'si olarak kullanılması, diğer kuyruk yöneticileriyle iletişim kurulurken sorunlara neden olabilir; örneğin, küçük harfli karakterler kullanan kanal adları uzak sistemde doğru şekilde yorumlanmayabilir. 5026, kod sayfası 930 ile ilişkili bir CCSID 'dir. CCSID 5026, Japonca Katakana (DBCS) özelliği seçildiğinde IBM i üzerinde bildirilen CCSID 'dir.

2. **AIX** 5050 ve 33722, AIX üzerinde temel kod sayfası 954 ile ilgili CCSID ' lerdir. İşletim sistemi tarafından bildirilen CCSID 33722 'dir.
3. **Windows** Windows NT , 932 kod sayfasını kullanır, ancak bu, en iyi 943 CCSID 'si ile gösterilir. Ancak, IBM MQ 'in tüm platformları bu CCSID 'yi desteklemez.

IBM MQ for Windows üzerinde, CCSID 932, kod sayfası 932 'yi göstermek için kullanılır, ancak . . / conv/table/ccsid.tbl dosyasında, kullanılan CCSID ' yi 943 olarak değiştiren bir değişiklik yapılabilir.

Tüm altyapılar, aşağıdaki kural dışı durumlar dışında, yerel CCSID 'leri ve diğer altyapıların yerel CCSID 'leri arasında dönüştürmeyi destekler.

## z/OS

### z/OS

Kod sayfası:

#### 1390

954, 5026, 5050, 33722 kod sayfalarına dönüştürülmez

Küçük harf karakterleri kabul etmez.

#### 5026

954, 1390, 5050, 33722 kod sayfalarına dönüştürülmez

## IBM i

### IBM i

Kod sayfası:

#### 5026

1390, 5039 kod sayfalarına dönüştürülmez

## Korece

Korece için CCSID ' lerin ve CCSID dönüştürmenin ayrıntıları.

Çizelge 672. Desteklenen platformlarda Korece için yerel CCSID ' ler	
Altyapı	Yerel CCSID ' ler
<b>IBM i</b> IBM i <b>z/OS</b> z/OS	933, 1364
<b>AIX</b> AIX <b>Linux</b> Linux	970
<b>Windows</b> Windows	949, 1363

Tüm platformlar, yerel CCSID 'leri ile diğer platformların yerel CCSID 'leri arasında, aşağıdaki istisnalar dışında dönüştürmeyi destekler.

## z/OS

### z/OS

Kod sayfası:

### 933




970 kod sayfasına dönüştürülmez

### 1364


970 kod sayfasına dönüştürülmez

## Yalınlaştırılmış Çince

Basitleştirilmiş Çince için CCSID ' lerin ayrıntıları ve CCSID dönüştürmesi.




Çizelge 673. Desteklenen platformlarda Yalınlaştırılmış Çince için yerel CCSID ' ler	
Altyapı	Yerel CCSID ' ler
 z/OS	935, 1388
 IBM i	935, 1388
 AIX	1383, 1386
 Windows	1381, 1386 (bkz. Not 2)
 Linux	1383

### Not:


1.  Windows , 936 kod sayfasını kullanır, ancak bu en iyi şekilde 1386 CCSID 'si tarafından temsil edilir. Ancak, IBM MQ 'un tüm altyapıları bu CCSID' yi desteklemiyor.

On IBM MQ for Windows CCSID 1381 is used to represent code page 936, but a change to file . . / conv/table/ccsid.tbl can be made which changes the CCSID used to 1386.

2. IBM MQ , Çince GB18030 standardını destekler.

   z/OS, Windows ve Linux üzerinde dönüştürme desteği, Unicode (UTF-8 ve UTF-16) ve CCSID 1388 ( GB18030 uzantıları olan EBCDIC), Unicode (UTF-8 ve UTF-16) ve CCSID 5488 (GB18030) ve CCSID 1388 ve CCSID 5488 arasında sağlanır.

### Not:

 IBM işletim sisteminde, işletim sistemi tarafından Unicode (UTF-8 ve UTF-16) ile CCSID 1388 ( GB18030 uzantıları ile EBCDIC) arasında dönüştürme için destek sağlanır.

Tüm platformlar, yerel CCSID 'leri ile diğer platformların yerel CCSID' leri arasında, aşağıdaki istisnalar dışında dönüştürmeyi destekler.

## z/OS



Kod sayfası:

### 935

1383 kod sayfasına dönüştürülmez

### 1388




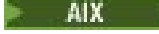

1383 kod sayfasına dönüştürülmez

## Geleneksel Çince

Geleneksel Çince için CCSID ' lerin ayrıntıları ve CCSID dönüştürmesi.



Çizelge 674. Desteklenen platformlarda Geleneksel Çince için Yerel CCSID ' ler

Altyapı	Yerel CCSID ' ler
 IBM i  z/OS	937
 Windows	950
 AIX  Linux	950, 964

Tüm platformlar, yerel CCSID 'leri ile diğer platformların yerel CCSID' leri arasında, aşağıdaki istisnalar dışında dönüştürmeyi destekler.

### z/OS



Kod sayfası:

#### 937

964 kod sayfasına dönüştürülmez

#### 1388

1383 kod sayfasına dönüştürülmez

### Linux



Kod sayfası:

#### 964

938 kod sayfasına dönüştürülmez



### z/OS dönüştürme desteği

Desteklenen CCSID dönüştürmelerinin bir listesi.

Çizelge 675. IBM MQ for z/OS CCSID dönüştürme desteği

CCSID	CCSID 'lere ve CCSID' ye dönüştürür
37	256, 273, 275, 277-278, 280, 284-285, 290, 297, 367, 420, 423-424, 437, 500, 720, 737, 775, 813, 819, 833, 836, 838, 850, 852, 855, 857-858, 860-866, 869-871, 874-875, 880, 897, 903-905, 912, 914-916, 920-924, 1009, 1025-1027, 1040-1043, 1047, 1051, 1088, 1097, 1100, 1112, 1114-1115, 1122, 1124, 1126, 1130-1132, 1137, 1140-1149, 1200, 1208, 1250-1255, 1257-1258, 1275, 1280-1281, 1283, 4386, 4909, 4929, 4932, 4934, 4946, 4948, 4951, 4953, 4960, 4970-4971, 5012, 5123, 5210-5211, 5346, 5348, 8229, 8482, 8612, 9025, 9030, 9044, 9049, 9056, 9061, 9066, 13121, 13488, 16804, 17248, 17584, 25473, 25479, 25480, 25617, 25619, 25664, 28709

Çizelge 675. IBM MQ for z/OS CCSID dönüştürme desteği (devamı var)

CCSID	CCSID 'lere ve CCSID' ye dönüştürür
256	37, 273, 277-278, 280, 284-285, 290, 297, 367, 420, 423-424, 437, 500, 737, 775, 819, 833, 836, 838, 850, 852, 857, 860-866, 869-871, 875, 880, 905, 1025-1027, 1112, 1122, 1200, 1208, 1251-1252, 1275, 4386, 4929, 4932, 4934, 4946, 4948, 4953, 4960, 4971, 5123, 8229, 8482, 8612, 9025, 9030, 9044, 9049, 9056, 9061, 13121, 13488, 16804, 17248, 17584, 28709
259	437, 808, 850-852, 855-858, 860-865, 867, 869, 872, 874, 899, 901-902, 915, 1098, 1161-1162, 1200, 1208, 1250-1258, 4946, 4948, 4951-4953, 4960, 4970, 5346, 5348, 9044, 9049, 9056, 9061, 9066, 13488, 17248, 17584
273	37, 256, 277-278, 280, 284-285, 290, 297, 367, 423, 437, 500, 737, 775, 813, 819, 833, 836, 838, 850, 852, 855-858, 860-865, 869-871, 874-875, 880, 897, 903, 912, 916, 920, 923-924, 1009, 1025-1027, 1040-1043, 1047, 1051, 1088, 1100, 1112, 1122, 1140-1149, 1200, 1208, 1250, 1252, 1275, 4386, 4909, 4929, 4932, 4934, 4946, 4948, 4951-4953, 4960, 4970-4971, 5012, 5123, 5346, 5348, 8229, 8482, 9025, 9030, 9044, 9049, 9056, 9061, 9066, 13121, 13488, 17248, 17584, 25473, 25479, 25617, 25619, 25664, 28709
274	500, 1047
275	37, 437, 500, 819, 850, 1047, 1200, 1208, 1252, 4946, 5348, 8229, 13488, 17584, 28709
277	37, 256, 273, 278, 280, 284-285, 290, 297, 367, 423, 437, 500, 737, 775, 813, 819, 833, 836, 838, 850, 852, 855, 857-858, 860-865, 869-871, 874-875, 880, 897, 903, 912, 916, 920, 923-924, 1009, 1025-1027, 1040-1043, 1047, 1051, 1088, 1100, 1112, 1122, 1140-1149, 1200, 1208, 1252, 1275, 4386, 4909, 4929, 4932, 4934, 4946, 4948, 4951, 4953, 4960, 4970-4971, 5012, 5123, 5348, 8229, 8482, 9025, 9030, 9044, 9049, 9056, 9061, 9066, 13121, 13488, 17248, 17584, 25473, 25479, 25617, 25619, 25664, 28709
278	37, 256, 273, 277, 280, 284-285, 290, 297, 367, 423, 437, 500, 737, 775, 813, 819, 833, 836, 838, 850, 852, 855, 857-858, 860-865, 869-871, 874-875, 880, 897, 903, 912, 916, 920, 923-924, 1009, 1025-1027, 1040-1043, 1047, 1051, 1088, 1100, 1112, 1122, 1140-1149, 1200, 1208, 1252, 1275, 4386, 4909, 4929, 4932, 4934, 4946, 4948, 4951, 4953, 4960, 4970-4971, 5012, 5123, 5348, 8229, 8482, 9025, 9030, 9044, 9049, 9056, 9061, 9066, 13121, 13488, 17248, 17584, 25473, 25479, 25617, 25619, 25664, 28709
280	37, 256, 273, 277-278, 284-285, 290, 297, 367, 423, 437, 500, 737, 775, 813, 819, 833, 836, 838, 850, 852, 855, 857-858, 860-865, 869-871, 874-875, 880, 897, 903, 912, 916, 920, 923-924, 1009, 1025-1027, 1040-1043, 1047, 1051, 1088, 1100, 1112, 1122, 1140-1149, 1200, 1208, 1252, 1275, 4386, 4909, 4929, 4932, 4934, 4946, 4948, 4951, 4953, 4960, 4970-4971, 5012, 5123, 5348, 8229, 8482, 9025, 9030, 9044, 9049, 9056, 9061, 9066, 13121, 13488, 17248, 17584, 25473, 25479, 25617, 25619, 25664, 28709
281	1047
282	500, 1047, 1200, 1208, 13488, 17584

Çizelge 675. IBM MQ for z/OS CCSID dönüştürme desteği (devamı var)

CCSID	CCSID 'lere ve CCSID' ye dönüştürür
284	37, 256, 273, 277-278, 280, 285, 290, 297, 367, 423, 437, 500, 737, 775, 813, 819, 833, 836, 838, 850, 852, 855, 857-858, 860-865, 869-871, 874-875, 880, 897, 903, 912, 916, 920, 923-924, 1009, 1025-1027, 1040-1043, 1047, 1051, 1088, 1100, 1112, 1122, 1140-1149, 1200, 1208, 1252, 1275, 4386, 4909, 4929, 4932, 4934, 4946, 4948, 4951, 4953, 4960, 4970-4971, 5012, 5123, 5348, 8229, 8482, 9025, 9030, 9044, 9049, 9056, 9061, 9066, 13121, 13488, 17248, 17584, 25473, 25479, 25617, 25619, 25664, 28709
285	37, 256, 273, 277-278, 280, 284, 290, 297, 423, 437, 500, 737, 775, 813, 819, 833, 836, 838, 850, 852, 855, 857-858, 860-865, 869-871, 874-875, 880, 897, 903, 912, 916, 920, 923-924, 1025-1027, 1040-1043, 1047, 1051, 1088, 1100, 1112, 1122, 1140-1149, 1200, 1208, 1252, 1275, 4386, 4909, 4929, 4932, 4934, 4946, 4948, 4951, 4953, 4960, 4970-4971, 5012, 5123, 5348, 8229, 8482, 9025, 9030, 9044, 9049, 9056, 9061, 9066, 13121, 13488, 17248, 17584, 25473, 25479, 25617, 25619, 25664, 28709
290	37, 256, 273, 277-278, 280, 284-285, 297, 367, 437, 500, 737, 775, 819, 833, 836, 850, 852, 855, 857, 860-865, 870-871, 895-897, 1009, 1025-1027, 1040-1043, 1047, 1088, 1112, 1122, 1139, 1200, 1208, 1252, 4386, 4929, 4932, 4946, 4948, 4951, 4953, 4960, 4992, 5123, 8229, 8482, 9025, 9044, 9049, 9056, 13121, 13488, 17248, 17584, 25473, 25617, 25619, 25664, 28709
293	1200, 1208, 13488, 17584
297	37, 256, 273, 277-278, 280, 284-285, 290, 367, 423, 437, 500, 737, 775, 813, 819, 833, 836, 838, 850, 852, 855, 857-858, 860-865, 869-871, 874-875, 880, 897, 903, 912, 916, 920, 923-924, 1009, 1025-1027, 1040-1043, 1047, 1051, 1088, 1100, 1112, 1122, 1140-1149, 1200, 1208, 1252, 1275, 4386, 4909, 4929, 4932, 4934, 4946, 4948, 4951, 4953, 4960, 4970-4971, 5012, 5123, 5348, 8229, 8482, 9025, 9030, 9044, 9049, 9056, 9061, 9066, 13121, 13488, 17248, 17584, 25473, 25479, 25617, 25619, 25664, 28709
300	301, 941, 1200, 1208, 1351, 4396, 8492, 13488, 16684, 17584
301	300, 941, 1200, 1208, 1351, 4396, 8492, 13488, 16684, 17584
367	37, 256, 273, 277-278, 280, 284, 290, 297, 500, 819, 833, 836, 850, 871, 875, 1009, 1026-1027, 1041, 1088, 1115, 1126, 1200, 1208, 4386, 4929, 4932, 4946, 4971, 5123, 5211, 8229, 8482, 9025, 13121, 13488, 17584, 25617, 25664, 28709
420	37, 256, 424, 437, 500, 720, 737, 775, 819, 850, 852, 857, 860-865, 1008, 1046, 1089, 1098, 1112, 1122, 1127, 1200, 1208, 1252, 1256, 4946, 4948, 4953, 4960, 5104, 5142, 5352, 8229, 8612, 9044, 9049, 9056, 9238, 13488, 16804, 17248, 17584, 28709
423	37, 256, 273, 277-278, 280, 284-285, 297, 437, 500, 737, 775, 813, 819, 838, 850-852, 857, 860-865, 869-871, 874-875, 880, 897, 903, 912, 916, 920, 1009, 1025-1027, 1041-1043, 1112, 1122, 1200, 1208, 1252-1253, 1280, 4909, 4934, 4946, 4948, 4953, 4960, 4970-4971, 5012, 5123, 8229, 9030, 9044, 9049, 9056, 9061, 9066, 13488, 17248, 17584, 25473, 25479, 25617, 25619, 28709

<i>Çizelge 675. IBM MQ for z/OS CCSID dönüştürme desteği (devamı var)</i>	
<b>CCSID</b>	<b>CCSID 'lere ve CCSID' ye dönüştürür</b>
424	37, 256, 420, 437, 500, 737, 775, 803, 819, 836, 850, 852, 856-857, 860-865, 916, 1112, 1122, 1200, 1208, 1252, 1255, 4932, 4946, 4948, 4952-4953, 4960, 5012, 5351, 8229, 8612, 9044, 9049, 9056, 13488, 16804, 17248, 17584, 28709
437	37, 256, 259, 273, 275, 277-278, 280, 284-285, 290, 297, 420, 423-424, 500, 737, 775, 813, 819, 833, 836, 838, 850, 852, 855, 857-858, 860-863, 865-866, 869-871, 874-875, 880, 897, 903, 905, 912, 914-916, 920-924, 1025-1027, 1040-1043, 1047, 1051, 1097, 1098, 1114-1115, 1126, 1140-1149, 1200, 1208, 1252, 1257, 1275, 1280-1281, 1283, 4386, 4909, 4929, 4932, 4934, 4946, 4948, 4951, 4953, 4970-4971, 5012, 5123, 5210-5211, 5348, 8229, 8482, 8612, 9025, 9030, 9044, 9049, 9061, 9066, 13121, 13488, 16804, 17584, 25473, 25479, 25617, 25619, 28709
500	37, 256, 273-275, 277-278, 280, 282, 284-285, 290, 297, 367, 420, 423-424, 437, 737, 775, 813, 819, 833, 836, 838, 850-852, 855-858, 860-866, 869-871, 874-875, 880, 891, 895, 897, 903-905, 912, 914-916, 920-924, 1004, 1009-1021, 1023, 1025-1027, 1040-1043, 1046-1047, 1051, 1088-1089, 1097, 1100-1107, 1112, 1114-1115, 1122, 1124-1126, 1129-1133, 1137, 1140-1149, 1200, 1208, 1250-1258, 1275, 1280-1283, 4386, 4909, 4929, 4932, 4934, 4946, 4948, 4951-4953, 4960, 4970-4971, 5012, 5123, 5142, 5210-5211, 5346, 5348, 8229, 8482, 8612, 9025, 9030, 9044, 9049, 9056, 9061, 9066, 9238, 13121, 13488, 16804, 17248, 17584, 25473, 25479, 25480, 25617, 25619, 25664, 28709
720	37, 420, 864, 1200, 1208, 1256, 4960, 8229, 8612, 9056, 13488, 16804, 17248, 17584, 28709
737	37, 256, 273, 277-278, 280, 284-285, 290, 297, 420, 423-424, 437, 500, 813, 833, 836, 838, 850, 869-871, 875, 880, 905, 1025-1027, 1097, 1200, 1208, 1252-1253, 1280, 4386, 4909, 4929, 4932, 4934, 4946, 4971, 5123, 8229, 8482, 8612, 9025, 9030, 9061, 13121, 13488, 16804, 17584, 28709
775	37, 256, 273, 277-278, 280, 284-285, 290, 297, 420, 423-424, 437, 500, 833, 836, 838, 850, 870-871, 875, 880, 905, 1025-1027, 1097, 1112, 1122, 1200, 1208, 1252, 1257, 4386, 4929, 4932, 4934, 4946, 4971, 5123, 8229, 8482, 8612, 9025, 9030, 13121, 13488, 16804, 17584, 28709
803	424, 819, 850, 856, 862, 916, 1200, 1208, 1252, 1255, 4946, 4952, 5012, 13488, 17584
806	1200, 1208, 13488, 17584
808	259, 858-859, 872, 923-924, 1140, 1148, 1153-1154, 1200, 1208, 5347, 5348, 13488, 17584
813	37, 273, 277-278, 280, 284-285, 297, 423, 437, 500, 737, 819, 838, 850, 852, 857, 860-861, 863, 869-871, 874-875, 880, 897, 903, 912, 916, 920, 1025-1027, 1041-1043, 1200, 1208, 1252-1253, 1280, 4909, 4934, 4946, 4948, 4953, 4970-4971, 5012, 5123, 5349, 8229, 9030, 9044, 9049, 9061, 9066, 13488, 17584, 25473, 25479, 25617, 25619, 28709

Çizelge 675. IBM MQ for z/OS CCSID dönüştürme desteği (devamı var)

CCSID	CCSID 'lere ve CCSID' ye dönüştürür
819	37, 256, 273, 275, 277-278, 280, 284-285, 290, 297, 367, 420, 423-424, 437, 500, 803, 813, 833, 836, 838, 850, 852, 855, 857-858, 860-861, 863-866, 869-871, 874-875, 880, 897, 903, 905, 912, 914-916, 920-924, 1004, 1025-1027, 1041-1043, 1047, 1051, 1088-1089, 1097, 1098, 1112, 1114, 1122-1123, 1126, 1130, 1132, 1137, 1140-1149, 1200, 1208, 1250-1255, 1257-1258, 1275, 1280-1281, 1283, 4386, 4909, 4929, 4932, 4934, 4946, 4948, 4951, 4953, 4960, 4970-4971, 5012, 5123, 5210, 5346, 5348, 8229, 8482, 8612, 9025, 9030, 9044, 9049, 9056, 9061, 9066, 13121, 13488, 16804, 17248, 17584, 25473, 25479, 25617, 25619, 25664, 28709
833	37, 256, 273, 277-278, 280, 284-285, 290, 297, 367, 437, 500, 737, 775, 819, 836, 850, 852, 855, 857, 860-865, 870-871, 891, 1009, 1025-1027, 1040-1043, 1088, 1112, 1122, 1126, 1200, 1208, 1252, 4386, 4929, 4932, 4946, 4948, 4951, 4953, 4960, 5123, 8229, 8482, 9025, 9044, 9049, 9056, 13121, 13488, 17248, 17584, 25617, 25619, 25664, 28709
834	926, 951, 1200, 1208, 1362, 4930, 9026, 13488, 17584
835	927, 947, 1200, 1208, 4931, 9027, 13488, 17584, 21427
836	37, 256, 273, 277-278, 280, 284-285, 290, 297, 367, 424, 437, 500, 737, 775, 819, 833, 850, 852, 855, 857, 870-871, 875, 903, 1009, 1025-1027, 1040-1043, 1088, 1112, 1114-1115, 1122, 1200, 1208, 1252, 4386, 4929, 4932, 4946, 4948, 4951, 4953, 4971, 5123, 5210-5211, 8229, 8482, 9025, 9044, 9049, 13121, 13488, 17584, 25479, 25617, 25619, 25664, 28709
837	928, 1200, 1208, 1380, 1385, 4933, 13488, 17584
838	37, 256, 273, 277-278, 280, 284-285, 297, 423, 437, 500, 737, 775, 813, 819, 850, 852, 857, 860-865, 869-871, 874-875, 880, 897, 903, 912, 916, 920, 1025-1027, 1041-1043, 1112, 1122, 1200, 1208, 1252, 4909, 4934, 4946, 4948, 4953, 4960, 4970-4971, 5012, 5123, 8229, 9030, 9044, 9049, 9056, 9061, 9066, 13488, 17248, 17584, 25473, 25479, 25617, 25619, 28709
848	924, 1148, 1158, 1200, 1208, 5347, 13488, 17584
849	924, 1148, 1154, 1200, 1208, 5347, 13488, 17584
850	37, 256, 259, 273, 275, 277-278, 280, 284-285, 290, 297, 367, 420, 423-424, 437, 500, 737, 775, 803, 813, 819, 833, 836, 838, 852, 855-858, 860-866, 869-871, 874-875, 880, 897, 903, 905, 912, 914-916, 920-924, 1004, 1025-1027, 1040-1043, 1047, 1051, 1088-1089, 1097, 1098, 1100, 1112, 1114, 1122, 1126, 1130, 1132, 1140-1149, 1200, 1208, 1250-1257, 1275, 1280-1281, 1283, 4386, 4909, 4929, 4932, 4934, 4946, 4948, 4951-4953, 4960, 4970-4971, 5012, 5123, 5210, 5346, 5348, 8229, 8482, 8612, 9025, 9030, 9044, 9049, 9056, 9061, 9066, 13121, 13488, 16804, 17248, 17584, 25473, 25479, 25617, 25619, 25664, 28709
851	259, 423, 500, 875, 1200, 1208, 4971, 13488, 17584

<i>Çizelge 675. IBM MQ for z/OS CCSID dönüştürme desteği (devamı var)</i>	
<b>CCSID</b>	<b>CCSID 'lere ve CCSID' ye dönüştürür</b>
852	37, 256, 259, 273, 277-278, 280, 284-285, 290, 297, 420, 423-424, 437, 500, 813, 819, 833, 836, 838, 850, 855, 857, 860-861, 863, 869-871, 874-875, 880, 897, 903, 905, 912, 916, 920, 1025-1027, 1040-1043, 1047, 1088, 1097, 1200, 1208, 1250, 1252, 1282, 4386, 4909, 4929, 4932, 4934, 4946, 4948, 4951, 4953, 4970-4971, 5012, 5123, 5346, 8229, 8482, 8612, 9025, 9030, 9044, 9049, 9061, 9066, 13121, 13488, 16804, 17584, 25473, 25479, 25617, 25619, 25664, 28709
855	37, 259, 273, 277-278, 280, 284-285, 290, 297, 437, 500, 819, 833, 836, 850, 852, 857, 866, 870-871, 878, 880, 912, 915, 1025-1027, 1040-1043, 1088, 1200, 1208, 1250-1252, 1283, 4386, 4929, 4932, 4946, 4948, 4951, 4953, 5123, 5346, 5347, 8229, 8482, 9025, 9044, 9049, 13121, 13488, 17584, 25617, 25619, 25664, 28709
856	259, 273, 424, 500, 803, 850, 862, 916, 1200, 1208, 1255, 4946, 4952, 5012, 5351, 13488, 17584
857	37, 256, 259, 273, 277-278, 280, 284-285, 290, 297, 420, 423-424, 437, 500, 813, 819, 833, 836, 838, 850, 852, 855, 860-861, 863, 869-871, 874-875, 880, 897, 903, 905, 912, 916, 920, 1025-1027, 1040-1043, 1088, 1097, 1200, 1208, 1252, 1254, 1281, 4386, 4909, 4929, 4932, 4934, 4946, 4948, 4951, 4953, 4970-4971, 5012, 5123, 5350, 8229, 8482, 8612, 9025, 9030, 9044, 9049, 9061, 9066, 13121, 13488, 16804, 17584, 25473, 25479, 25617, 25619, 25664, 28709
858	37, 259, 273, 277-278, 280, 284-285, 297, 437, 500, 808, 819, 850, 860-861, 865, 871-872, 901-902, 923-924, 1047, 1051, 1140-1149, 1153-1157, 1160-1162, 1164, 1200, 1208, 1252, 1275, 4946, 5348, 8229, 13488, 17584, 28709
859	808, 872, 901-902, 1153-1157, 1160-1162, 1164, 1200, 1208, 13488, 17584
860	37, 256, 259, 273, 277-278, 280, 284-285, 290, 297, 420, 423-424, 437, 500, 813, 819, 833, 838, 850, 852, 857-858, 861, 863, 865, 869-871, 874-875, 880, 897, 903, 905, 912, 916, 920, 923-924, 1025-1027, 1041-1043, 1097, 1140, 1145-1146, 1148, 1200, 1208, 1252, 4386, 4909, 4929, 4934, 4946, 4948, 4953, 4970-4971, 5012, 5123, 5348, 8229, 8482, 8612, 9025, 9030, 9044, 9049, 9061, 9066, 13121, 13488, 16804, 17584, 25473, 25479, 25617, 25619, 28709
861	37, 256, 259, 273, 277-278, 280, 284-285, 290, 297, 420, 423-424, 437, 500, 813, 819, 833, 838, 850, 852, 857-858, 860, 863, 869-871, 874-875, 880, 897, 903, 905, 912, 916, 920, 923-924, 1025-1027, 1041-1043, 1097, 1148, 1149, 1200, 1208, 1252, 4386, 4909, 4929, 4934, 4946, 4948, 4953, 4970-4971, 5012, 5123, 5348, 8229, 8482, 8612, 9025, 9030, 9044, 9049, 9061, 9066, 13121, 13488, 16804, 17584, 25473, 25479, 25617, 25619, 28709
862	37, 256, 259, 273, 277-278, 280, 284-285, 290, 297, 420, 423-424, 437, 500, 803, 833, 838, 850, 856, 870-871, 875, 880, 905, 916, 1025-1027, 1097, 1200, 1208, 1252, 1255, 4386, 4929, 4934, 4946, 4952, 4971, 5012, 5123, 5351, 8229, 8482, 8612, 9025, 9030, 12712, 13121, 13488, 16804, 17584, 28709

Çizelge 675. IBM MQ for z/OS CCSID dönüştürme desteği (devamı var)

CCSID	CCSID 'lere ve CCSID' ye dönüştürür
863	37, 256, 259, 273, 277-278, 280, 284-285, 290, 297, 420, 423-424, 437, 500, 813, 819, 833, 838, 850, 852, 857, 860-861, 865, 869-871, 874-875, 880, 897, 903, 905, 912, 916, 920, 1025-1027, 1041-1043, 1051, 1097, 1140-1149, 1200, 1208, 1252, 1275, 4386, 4909, 4929, 4934, 4946, 4948, 4953, 4970-4971, 5012, 5123, 5348, 8229, 8482, 8612, 9025, 9030, 9044, 9049, 9061, 9066, 13121, 13488, 16804, 17584, 25473, 25479, 25617, 25619, 28709
864	37, 256, 259, 273, 277-278, 280, 284-285, 290, 297, 420, 423-424, 500, 720, 819, 833, 838, 850, 870-871, 875, 880, 905, 918, 1008, 1025-1027, 1046, 1089, 1097, 1127, 1200, 1208, 1252, 1256, 4386, 4929, 4934, 4946, 4960, 4971, 5104, 5123, 5142, 5352, 8229, 8482, 8612, 9025, 9030, 9056, 9238, 13121, 13488, 16804, 17248, 17584, 28709
865	37, 256, 259, 273, 277-278, 280, 284-285, 290, 297, 420, 423-424, 437, 500, 819, 833, 838, 850, 858, 860, 863, 870-871, 875, 880, 905, 923-924, 1025-1027, 1097, 1142-1143, 1148, 1200, 1208, 1252, 4386, 4929, 4934, 4946, 4971, 5123, 5348, 8229, 8482, 8612, 9025, 9030, 13121, 13488, 16804, 17584, 28709
866	37, 256, 437, 500, 819, 850, 855, 870, 878, 880, 915, 1025, 1200, 1208, 1251-1252, 1283, 4946, 4951, 5347, 8229, 13488, 17584, 28709
867	259, 1153-1155, 1160, 1200, 1208, 4899, 5351, 9048, 12712, 13488, 17584
868	918, 1006, 1200, 1208, 13488, 17584
869	37, 256, 259, 273, 277-278, 280, 284-285, 297, 423, 437, 500, 737, 813, 819, 838, 850, 852, 857, 860-861, 863, 870-871, 874-875, 880, 897, 903, 912, 916, 920, 1025-1027, 1041-1043, 1200, 1208, 1252-1254, 1280, 4909, 4934, 4946, 4948, 4953, 4970-4971, 5012, 5123, 5349, 8229, 9030, 9044, 9049, 9061, 9066, 13488, 17584, 25473, 25479, 25617, 25619, 28709
870	37, 256, 273, 277-278, 280, 284-285, 290, 297, 423, 437, 500, 737, 775, 813, 819, 833, 836, 838, 850, 852, 855, 857, 860-866, 869, 871, 874-875, 880, 897, 903, 912, 915-916, 920, 1009, 1025-1027, 1040-1043, 1047, 1088, 1112, 1122, 1200, 1208, 1250, 1252, 1282, 4386, 4909, 4929, 4932, 4934, 4946, 4948, 4951, 4953, 4960, 4970-4971, 5012, 5123, 5346, 8229, 8482, 9025, 9030, 9044, 9049, 9056, 9061, 9066, 13121, 13488, 17248, 17584, 25473, 25479, 25617, 25619, 25664, 28709
871	37, 256, 273, 277-278, 280, 284-285, 290, 297, 367, 423, 437, 500, 737, 775, 813, 819, 833, 836, 838, 850, 852, 855, 857-858, 860-865, 869, 870, 874-875, 880, 897, 903, 912, 916, 920, 923-924, 1009, 1025-1027, 1040-1043, 1047, 1051, 1088, 1112, 1122, 1140-1149, 1200, 1208, 1252, 1275, 4386, 4909, 4929, 4932, 4934, 4946, 4948, 4951, 4953, 4960, 4970-4971, 5012, 5123, 5348, 8229, 8482, 9025, 9030, 9044, 9049, 9056, 9061, 9066, 13121, 13488, 17248, 17584, 25473, 25479, 25617, 25619, 25664, 28709
872	259, 808, 858-859, 923-924, 1140-1149, 1153-1155, 1200, 1208, 5347, 5348, 13488, 17584

<i>Çizelge 675. IBM MQ for z/OS CCSID dönüştürme desteği (devamı var)</i>	
<b>CCSID</b>	<b>CCSID 'lere ve CCSID' ye dönüştürür</b>
874	37, 259, 273, 277-278, 280, 284-285, 297, 423, 437, 500, 813, 819, 838, 850, 852, 857, 860-861, 863, 869-871, 875, 880, 897, 903, 912, 916, 920, 1025-1027, 1041-1043, 1200, 1208, 1252, 4909, 4934, 4946, 4948, 4953, 4970-4971, 5012, 5123, 8229, 9030, 9044, 9049, 9061, 9066, 13488, 17584, 25473, 25479, 25617, 25619, 28709
875	37, 256, 273, 277-278, 280, 284-285, 297, 367, 423, 437, 500, 737, 775, 813, 819, 836, 838, 850-852, 857, 860-865, 869-871, 874, 880, 897, 903, 912, 916, 920, 1009, 1025-1027, 1041-1043, 1047, 1088, 1112, 1122, 1200, 1208, 1252-1253, 1280, 4909, 4932, 4934, 4946, 4948, 4953, 4960, 4970-4971, 5012, 5123, 5349, 8229, 9030, 9044, 9049, 9056, 9061, 9066, 13488, 17248, 17584, 25473, 25479, 25617, 25619, 25664, 28709
878	855, 866, 880, 915, 1025, 1131, 1200, 1208, 1251, 1283, 4951, 5347, 13488, 17584
880	37, 256, 273, 277-278, 280, 284-285, 297, 423, 437, 500, 737, 775, 813, 819, 838, 850, 852, 855, 857, 860-866, 869-871, 874-875, 878, 897, 903, 912, 915-916, 920, 1009, 1025-1027, 1041-1043, 1112, 1122, 1200, 1208, 1251-1252, 1283, 4909, 4934, 4946, 4948, 4951, 4953, 4960, 4970-4971, 5012, 5123, 5347, 8229, 9030, 9044, 9049, 9056, 9061, 9066, 13488, 17248, 17584, 25473, 25479, 25617, 25619, 28709
891	500, 833, 1088, 1200, 1208, 4929, 9025, 13121, 13488, 17584, 25664
895	290, 500, 1027, 1041, 1200, 1208, 4386, 5123, 8482, 13488, 17584, 25617
896	290, 1027, 1041, 1200, 1208, 4386, 4992, 5123, 8482, 13488, 17584, 25617
897	37, 273, 277-278, 280, 284-285, 290, 297, 423, 437, 500, 813, 819, 838, 850, 852, 857, 860-861, 863, 869-871, 874-875, 880, 903, 912, 916, 920, 1025-1027, 1041-1043, 1200, 1208, 1252, 4386, 4909, 4934, 4946, 4948, 4953, 4970-4971, 5012, 5123, 8229, 8482, 9030, 9044, 9049, 9061, 9066, 13488, 17584, 25473, 25479, 25617, 25619, 28709
899	259
901	259, 858-859, 902, 923-924, 1140, 1148, 1156-1157, 1200, 1208, 5348, 5353, 13488, 17584
902	259, 858-859, 901, 923-924, 1140, 1148, 1156-1157, 1200, 1208, 5348, 5353, 13488, 17584
903	37, 273, 277-278, 280, 284-285, 297, 423, 437, 500, 813, 819, 836, 838, 850, 852, 857, 860-861, 863, 869-871, 874-875, 880, 897, 912, 916, 920, 1025-1027, 1041-1043, 1115, 1200, 1208, 1252, 4909, 4932, 4934, 4946, 4948, 4953, 4970-4971, 5012, 5123, 5211, 8229, 9030, 9044, 9049, 9061, 9066, 13488, 17584, 25473, 25479, 25617, 25619, 28709
904	37, 500, 1114, 1200, 1208, 5210, 8229, 13488, 17584, 25480, 28709
905	37, 256, 437, 500, 737, 775, 819, 850, 852, 857, 860-865, 920, 1026, 1112, 1122, 1200, 1208, 1252, 1254, 1281, 4946, 4948, 4953, 4960, 8229, 9044, 9049, 9056, 13488, 17248, 17584, 28709



<i>Çizelge 675. IBM MQ for z/OS CCSID dönüştürme desteği (devamı var)</i>	
<b>CCSID</b>	<b>CCSID 'lere ve CCSID' ye dönüştürür</b>
912	37, 273, 277-278, 280, 284-285, 297, 423, 437, 500, 813, 819, 838, 850, 852, 855, 857, 860-861, 863, 869-871, 874-875, 880, 897, 903, 916, 920, 1025-1027, 1041-1043, 1047, 1200, 1208, 1250, 1252, 1282, 4909, 4934, 4946, 4948, 4951, 4953, 4970-4971, 5012, 5123, 5346, 8229, 9030, 9044, 9049, 9061, 9066, 13488, 17584, 25473, 25479, 25617, 25619, 28709
914	37, 437, 500, 819, 850, 1200, 1208, 1252, 1257, 4946, 8229, 13488, 17584, 28709
915	37, 259, 437, 500, 819, 850, 855, 866, 870, 878, 880, 1025, 1131, 1200, 1208, 1251-1252, 1283, 4946, 4951, 5347, 8229, 13488, 17584, 28709
916	37, 273, 277-278, 280, 284-285, 297, 423-424, 437, 500, 803, 813, 819, 838, 850, 852, 856-857, 860-863, 869-871, 874-875, 880, 897, 903, 912, 920, 1025-1027, 1041-1043, 1200, 1208, 1252, 1255, 4909, 4934, 4946, 4948, 4952-4953, 4970-4971, 5012, 5123, 5351, 8229, 9030, 9044, 9049, 9061, 9066, 13488, 17584, 25473, 25479, 25617, 25619, 28709
918	864, 868, 1006, 1200, 1208, 4960, 9056, 13488, 17248, 17584
920	37, 273, 277-278, 280, 284-285, 297, 423, 437, 500, 813, 819, 838, 850, 852, 857, 860-861, 863, 869-871, 874-875, 880, 897, 903, 905, 912, 916, 1025-1026, 1200, 1208, 1252, 1254, 1281, 4909, 4934, 4946, 4948, 4953, 4970-4971, 5012, 5350, 8229, 9030, 9044, 9049, 9061, 9066, 13488, 17584, 25473, 25479, 28709
921	37, 437, 500, 819, 850, 922, 1112, 1122, 1200, 1208, 1252, 1257, 4946, 5353, 8229, 13488, 17584, 28709
922	37, 437, 500, 819, 850, 921, 1112, 1122, 1200, 1208, 1252, 1257, 4946, 5353, 8229, 13488, 17584, 28709
923	37, 273, 277-278, 280, 284-285, 297, 437, 500, 808, 819, 850, 858, 860-861, 865, 871-872, 901-902, 924, 1047, 1051, 1140-1149, 1153-1158, 1160-1162, 1164, 1200, 1208, 1252, 1275, 4946, 5348, 8229, 13488, 17584, 28709
924	37, 273, 277-278, 280, 284-285, 297, 437, 500, 808, 819, 848-850, 858, 860-861, 865, 871-872, 901-902, 923, 1047, 1051, 1140-1149, 1153-1157, 1160-1164, 1200, 1208, 1252, 1275, 4946, 5348, 8229, 13488, 17584, 28709
926	834, 951, 9026
927	835, 947, 1200, 1208, 4931, 9027, 13488, 17584, 21427
928	837, 1200, 1208, 1380, 13488, 17584
930	931-932, 939, 942-943, 1200, 1208, 1390, 1399, 5026, 5028, 5035, 5038-5039, 9122, 9124, 9131, 9135, 13218-13219, 13231, 13488, 17314, 17584, 25508, 25518, 29614, 33698-33700, 37796
931	930, 932, 939, 942-943, 1390, 1399, 5026, 5028, 5035, 5038-5039, 9122, 9124, 9131, 9135, 13218-13219, 13231, 17314, 25508, 25518, 29614, 33698-33700, 37796
932	930-931, 939, 942-943, 1200, 1208, 1390, 1399, 5026, 5028, 5035, 5038-5039, 9122, 9124, 9131, 9135, 13218-13219, 13231, 13488, 17314, 17584, 25508, 25518, 29614, 33698-33700, 37796

<i>Çizelge 675. IBM MQ for z/OS CCSID dönüştürme desteği (devamı var)</i>	
<b>CCSID</b>	<b>CCSID 'lere ve CCSID' ye dönüştürür</b>
933	934, 944, 949, 1200, 1208, 1363-1364, 5029, 5045, 5460, 9125, 9555, 13221, 13488, 13651, 17317, 17584, 25510, 25520, 25525, 29616, 29621, 33717, 37813
934	933, 949, 5029, 5045, 5460, 9125, 13221, 17317, 25510, 25525, 29621, 33717, 37813
935	936, 946, 1200, 1208, 1381, 1386, 1388, 5031, 5477, 5482, 5484, 9127, 13223, 13488, 17584, 25512
936	935, 946, 1381, 5031, 5477, 5484, 9127, 13223, 25512
937	938, 948, 950, 1200, 1208, 1370, 5033, 5046, 9142, 13488, 17584, 25514, 25524, 29620
938	937, 950, 1370, 5033, 5046, 9142, 25514
939	930-932, 942-943, 1200, 1208, 1390, 1399, 5026, 5028, 5035, 5038-5039, 9122, 9124, 9131, 9135, 13218-13219, 13231, 13488, 17314, 17584, 25508, 25518, 29614, 33698-33700, 37796
941	300-301, 1200, 1208, 1351, 4396, 8492, 13488, 16684, 17584
942	930-932, 939, 943, 1200, 1208, 1390, 1399, 5026, 5028, 5035, 5038-5039, 9122, 9124, 9131, 9135, 13218-13219, 13231, 13488, 17314, 17584, 25508, 25518, 29614, 33698-33700, 37796
943	930-932, 939, 942, 1200, 1208, 1390, 1399, 5026, 5028, 5035, 5038-5039, 9122, 9124, 9131, 9135, 13218-13219, 13231, 13488, 17314, 17584, 25508, 25518, 29614, 33698-33700, 37796
944	933, 949, 1200, 1208, 5029, 5045, 5460, 9125, 13221, 13488, 17317, 17584, 25520, 25525, 29616, 29621, 33717, 37813
946	935-936, 1200, 1208, 5031, 5484, 9127, 13223, 13488, 17584, 25512
947	835, 927, 1200, 1208, 4931, 9027, 13488, 17584, 21427
948	937, 950, 1200, 1208, 1370, 5033, 5046, 9142, 13488, 17584, 25524, 29620
949	933-934, 944, 1200, 1208, 1363-1364, 5029, 5045, 5460, 9125, 9555, 13221, 13488, 13651, 17317, 17584, 25510, 25520, 25525, 29616, 29621, 33717, 37813
950	937-938, 948, 1200, 1208, 1370, 5033, 5046, 9142, 13488, 17584, 25514, 25524, 29620
951	834, 926, 1200, 1208, 1362, 4930, 9026, 13488, 17584
1004	500, 819, 850, 1200, 1208, 4946, 13488, 17584
1006	868, 918, 1200, 1208, 13488, 17584
1008	420, 864, 1200, 1208, 4960, 5104, 8612, 9056, 13488, 16804, 17248, 17584
1009	37, 273, 277-278, 280, 284, 290, 297, 367, 423, 500, 833, 836, 870-871, 875, 880, 1025-1026, 1200, 1208, 4386, 4929, 4932, 4971, 8229, 8482, 9025, 13121, 13488, 17584, 28709
1010	500, 1200, 1208, 13488, 17584

<i>Çizelge 675. IBM MQ for z/OS CCSID dönüştürme desteği (devamı var)</i>	
<b>CCSID</b>	<b>CCSID 'lere ve CCSID' ye dönüştürür</b>
1011	500, 1200, 1208, 13488, 17584
1012	500, 1200, 1208, 13488, 17584
1013	500, 1140, 1200, 1208, 13488, 17584
1014	500, 1200, 1208, 13488, 17584
1015	500, 1200, 1208, 13488, 17584
1016	500, 1200, 1208, 13488, 17584
1017	500, 1200, 1208, 13488, 17584
1018	500, 1200, 1208, 13488, 17584
1019	500, 1200, 1208, 13488, 17584
1020	500
1021	500
1023	500
1025	37, 256, 273, 277-278, 280, 284-285, 290, 297, 423, 437, 500, 737, 775, 813, 819, 833, 836, 838, 850, 852, 855, 857, 860-866, 869-871, 874-875, 878, 880, 897, 903, 912, 915-916, 920, 1009, 1026-1027, 1040-1043, 1051, 1088, 1112, 1122, 1131, 1200, 1208, 1251-1252, 1283, 4386, 4909, 4929, 4932, 4934, 4946, 4948, 4951, 4953, 4960, 4970-4971, 5012, 5123, 5347, 8229, 8482, 9025, 9030, 9044, 9049, 9056, 9061, 9066, 13121, 13488, 17248, 17584, 25473, 25479, 25617, 25619, 25664, 28709
1026	37, 256, 273, 277-278, 280, 284-285, 290, 297, 367, 423, 437, 500, 737, 775, 813, 819, 833, 836, 838, 850, 852, 855, 857, 860-865, 869-871, 874-875, 880, 897, 903, 905, 912, 916, 920, 1009, 1025, 1027, 1040-1043, 1047, 1088, 1112, 1122, 1200, 1208, 1252, 1254, 1281, 4386, 4909, 4929, 4932, 4934, 4946, 4948, 4951, 4953, 4960, 4970-4971, 5012, 5123, 5350, 8229, 8482, 9025, 9030, 9044, 9049, 9056, 9061, 9066, 13121, 13488, 17248, 17584, 25473, 25479, 25617, 25619, 25664, 28709
1027	37, 256, 273, 277-278, 280, 284-285, 290, 297, 367, 423, 437, 500, 737, 775, 813, 819, 833, 836, 838, 850, 852, 855, 857, 860-865, 869-871, 874-875, 880, 895-897, 903, 912, 916, 1025-1026, 1040-1043, 1047, 1088, 1112, 1122, 1139, 1200, 1208, 1252, 4386, 4909, 4929, 4932, 4934, 4946, 4948, 4951, 4953, 4960, 4970-4971, 4992, 5012, 5123, 8229, 8482, 9025, 9030, 9044, 9049, 9056, 9061, 9066, 13121, 13488, 17248, 17584, 25473, 25479, 25617, 25619, 25664, 28709
1040	37, 273, 277-278, 280, 284-285, 290, 297, 437, 500, 833, 836, 850, 852, 855, 857, 870-871, 1025-1027, 1041-1043, 1088, 1200, 1208, 4386, 4929, 4932, 4946, 4948, 4951, 4953, 5123, 8229, 8482, 9025, 9044, 9049, 13121, 13488, 17584, 25617, 25619, 25664, 28709
1041	37, 273, 277-278, 280, 284-285, 290, 297, 367, 423, 437, 500, 813, 819, 833, 836, 838, 850, 852, 855, 857, 860-861, 863, 869-871, 874-875, 880, 895-897, 903, 912, 916, 1025-1027, 1040, 1042-1043, 1088, 1200, 1208, 1252, 4386, 4909, 4929, 4932, 4934, 4946, 4948, 4951, 4953, 4970-4971, 4992, 5012, 5123, 8229, 8482, 9025, 9030, 9044, 9049, 9061, 9066, 13121, 13488, 17584, 25473, 25479, 25617, 25619, 25664, 28709

<i>Çizelge 675. IBM MQ for z/OS CCSID dönüştürme desteği (devamı var)</i>	
<b>CCSID</b>	<b>CCSID 'lere ve CCSID' ye dönüştürür</b>
1042	37, 273, 277-278, 280, 284-285, 290, 297, 423, 437, 500, 813, 819, 833, 836, 838, 850, 852, 855, 857, 860-861, 863, 869-871, 874-875, 880, 897, 903, 912, 916, 1025-1027, 1040, 1041, 1043, 1088, 1200, 1208, 4386, 4909, 4929, 4932, 4934, 4946, 4948, 4951, 4953, 4970-4971, 5012, 5123, 8229, 8482, 9025, 9030, 9044, 9049, 9061, 9066, 13121, 13488, 17584, 25473, 25479, 25617, 25619, 25664, 28709
1043	37, 273, 277-278, 280, 284-285, 290, 297, 423, 437, 500, 813, 819, 833, 836, 838, 850, 852, 855, 857, 860-861, 863, 869-871, 874-875, 880, 897, 903, 912, 916, 1025-1027, 1040, 1041, 1042, 1088, 1114, 1200, 1208, 4386, 4909, 4929, 4932, 4934, 4946, 4948, 4951, 4953, 4970-4971, 5012, 5123, 5210, 8229, 8482, 9025, 9030, 9044, 9049, 9061, 9066, 13121, 13488, 17584, 25473, 25479, 25617, 25619, 25664, 28709
1046	420, 500, 864, 1089, 1127, 1200, 1208, 1256, 4960, 5142, 5352, 8612, 9056, 9238, 13488, 16804, 17248, 17584
1047	37, 273-275, 277-278, 280, 281, 282, 284-285, 290, 297, 437, 500, 819, 850, 852, 858, 870-871, 875, 912, 923-924, 1026-1027, 1140-1149, 1200, 1208, 1252, 1254, 4946, 4948, 5123, 8229, 8482, 13488, 17584, 28709
1051	37, 273, 277-278, 280, 284-285, 297, 437, 500, 819, 850, 858, 863, 871, 923-924, 1025, 1097, 1140-1149, 1200, 1208, 1252, 1275, 4946, 5348, 8229, 13488, 17584, 28709
1088	37, 273, 277-278, 280, 284-285, 290, 297, 367, 500, 819, 833, 836, 850, 852, 855, 857, 870-871, 875, 891, 1025-1027, 1040-1043, 1126, 1200, 1208, 4386, 4929, 4932, 4946, 4948, 4951, 4953, 4971, 5123, 8229, 8482, 9025, 9044, 9049, 13121, 13488, 17584, 25617, 25619, 25664, 28709
1089	420, 500, 819, 850, 864, 1046, 1127, 1200, 1208, 1256, 4946, 4960, 5142, 5352, 8612, 9056, 9238, 13488, 16804, 17248, 17584
1097	37, 437, 500, 737, 775, 819, 850, 852, 857, 860-865, 1051, 1098, 1112, 1122, 1200, 1208, 1252, 4946, 4948, 4953, 4960, 8229, 9044, 9049, 9056, 13488, 17248, 17584, 28709
1098	259, 420, 437, 819, 850, 1097, 1200, 1208, 1252, 4946, 8612, 13488, 16804, 17584
1100	37, 273, 277-278, 280, 284-285, 297, 500, 850, 4946, 8229, 28709
1101	500
1102	500
1103	500
1104	500
1105	500
1106	500
1107	500

<i>Çizelge 675. IBM MQ for z/OS CCSID dönüştürme desteği (devamı var)</i>	
<b>CCSID</b>	<b>CCSID 'lere ve CCSID' ye dönüştürür</b>
1112	37, 256, 273, 277-278, 280, 284-285, 290, 297, 420, 423-424, 500, 775, 819, 833, 836, 838, 850, 870-871, 875, 880, 905, 921-922, 1025-1027, 1097, 1122, 1200, 1208, 1252, 1257, 4386, 4929, 4932, 4934, 4946, 4971, 5123, 5353, 8229, 8482, 8612, 9025, 9030, 13121, 13488, 16804, 17584, 28709
1114	37, 437, 500, 819, 836, 850, 904, 1043, 1115, 1200, 1208, 4932, 4946, 5210-5211, 8229, 13488, 17584, 25480, 25619, 28709
1115	37, 367, 437, 500, 836, 903, 1114, 1200, 1208, 4932, 5210-5211, 8229, 13488, 17584, 25479, 28709
1122	37, 256, 273, 277-278, 280, 284-285, 290, 297, 420, 423-424, 500, 775, 819, 833, 836, 838, 850, 870-871, 875, 880, 905, 921-922, 1025-1027, 1097, 1112, 1200, 1208, 1252, 1257, 4386, 4929, 4932, 4934, 4946, 4971, 5123, 5353, 8229, 8482, 8612, 9025, 9030, 13121, 13488, 16804, 17584, 28709
1123	819, 1124-1125, 1148, 1200, 1208, 1251-1252, 1283, 5347, 13488, 17584
1124	37, 500, 1123, 1125, 1200, 1208, 1251, 1283, 5347, 8229, 13488, 17584, 28709
1125	500, 1123, 1124, 1200, 1208, 1251, 1283, 5347, 13488, 17584
1126	37, 367, 437, 500, 819, 833, 850, 1088, 1200, 1208, 1252, 4929, 4946, 8229, 9025, 13121, 13488, 17584, 25664, 28709
1127	420, 864, 1046, 1089, 1256, 4960, 5142, 8612, 9056, 9238, 16804, 17248
1129	500, 1130, 1200, 1208, 1258, 5354, 13488, 17584
1130	37, 500, 819, 850, 1129, 1200, 1208, 1252, 1258, 4946, 5354, 8229, 13488, 17584, 28709
1131	37, 500, 878, 915, 1025, 1200, 1208, 1251, 1283, 5347, 8229, 13488, 17584, 28709
1132	37, 500, 819, 850, 1133, 1200, 1208, 1252, 4946, 8229, 13488, 17584, 28709
1133	500, 1132, 1200, 1208, 13488, 17584
1137	37, 500, 819, 1200, 1208, 8229, 13488, 17584, 28709
1139	290, 1027, 4386, 5123, 8482
1140	37, 273, 277-278, 280, 284-285, 297, 437, 500, 808, 819, 850, 858, 860, 863, 871-872, 901-902, 923-924, 1013, 1047, 1051, 1141-1149, 1153-1157, 1160-1162, 1164, 1200, 1208, 1252, 1275, 4946, 5348, 8229, 13488, 17584, 28709
1141	37, 273, 277-278, 280, 284-285, 297, 437, 500, 819, 850, 858, 863, 871-872, 923-924, 1047, 1051, 1140, 1142-1149, 1153-1157, 1160-1162, 1200, 1208, 1252, 1275, 4946, 5348, 8229, 13488, 17584, 28709

<i>Çizelge 675. IBM MQ for z/OS CCSID dönüştürme desteği (devamı var)</i>	
<b>CCSID</b>	<b>CCSID 'lere ve CCSID' ye dönüştürür</b>
1142	37, 273, 277-278, 280, 284-285, 297, 437, 500, 819, 850, 858, 863, 865, 871-872, 923-924, 1047, 1051, 1140-1141, 1143-1149, 1153-1157, 1160-1162, 1200, 1208, 1252, 1275, 4946, 5348, 8229, 13488, 17584, 28709
1143	37, 273, 277-278, 280, 284-285, 297, 437, 500, 819, 850, 858, 863, 865, 871-872, 923-924, 1047, 1051, 1140-1142, 1144-1149, 1153-1157, 1160-1162, 1200, 1208, 1252, 1275, 4946, 5348, 8229, 13488, 17584, 28709
1144	37, 273, 277-278, 280, 284-285, 297, 437, 500, 819, 850, 858, 863, 871-872, 923-924, 1047, 1051, 1140-1143, 1145-1149, 1153-1157, 1160-1162, 1200, 1208, 1252, 1275, 4946, 5348, 8229, 13488, 17584, 28709
1145	37, 273, 277-278, 280, 284-285, 297, 437, 500, 819, 850, 858, 860, 863, 871-872, 923-924, 1047, 1051, 1140-1144, 1146-1149, 1153-1157, 1160-1162, 1200, 1208, 1252, 1275, 4946, 5348, 8229, 13488, 17584, 28709
1146	37, 273, 277-278, 280, 284-285, 297, 437, 500, 819, 850, 858, 860, 863, 871-872, 923-924, 1047, 1051, 1140-1145, 1147-1149, 1153-1157, 1160-1162, 1200, 1208, 1252, 1275, 4946, 5348, 8229, 13488, 17584, 28709
1147	37, 273, 277-278, 280, 284-285, 297, 437, 500, 819, 850, 858, 863, 871-872, 923-924, 1047, 1051, 1140-1146, 1148-1149, 1153-1157, 1160-1162, 1200, 1208, 1252, 1275, 4946, 5348, 8229, 13488, 17584, 28709
1148	37, 273, 277-278, 280, 284-285, 297, 437, 500, 808, 819, 848-850, 858, 860-861, 863, 865, 871-872, 901-902, 923-924, 1047, 1051, 1123, 1140-1147, 1149, 1153-1164, 1200, 1208, 1252, 1275, 4899, 4946, 5348, 5349, 8229, 12712, 13488, 17584, 28709
1149	37, 273, 277-278, 280, 284-285, 297, 437, 500, 819, 850, 858, 861, 863, 871-872, 923-924, 1047, 1051, 1140-1148, 1153-1157, 1160-1162, 1200, 1208, 1252, 1275, 4946, 5348, 8229, 13488, 17584, 28709
1153	808, 858-859, 867, 872, 923-924, 1140-1149, 1154-1157, 1160-1162, 1200, 1208, 5348, 9044, 13488, 17584
1154	808, 849, 858-859, 867, 872, 923-924, 1140-1149, 1153, 1155-1157, 1160-1162, 1200, 1208, 5347, 5348, 13488, 17584
1155	858-859, 867, 872, 923-924, 1140-1149, 1153-1154, 1156-1157, 1160-1162, 1200, 1208, 5348, 5350, 9049, 13488, 17584
1156	858-859, 901-902, 923-924, 1140-1149, 1153-1155, 1157, 1160, 1200, 1208, 5348, 5353, 12712, 13488, 17584
1157	858-859, 901-902, 923-924, 1140-1149, 1153-1156, 1160, 1200, 1208, 5348, 5353, 12712, 13488, 17584
1158	848, 923, 1148, 1200, 1208, 5347, 5348, 13488, 17584
1159	1148, 1200, 1208, 13488, 17584
1160	858-859, 867, 923-924, 1140-1149, 1153-1157, 1161-1162, 1200, 1208, 5348, 13488, 17584

<i>Çizelge 675. IBM MQ for z/OS CCSID dönüştürme desteği (devamı var)</i>	
<b>CCSID</b>	<b>CCSID 'lere ve CCSID' ye dönüştürür</b>
1161	259, 858-859, 923-924, 1140-1149, 1153-1155, 1160, 5348, 17584
1162	259, 858-859, 923-924, 1140-1149, 1153-1155, 1160, 5348, 17584
1163	924, 1148, 1164, 5354, 17584
1164	858-859, 923-924, 1140, 1148, 1163, 1200, 1208, 5348, 5354, 13488, 17584
1166	1200,1208,13488,17584
1200	37, 256, 259, 273, 275, 277-278, 280, 282, 284-285, 290, 293, 297, 300-301, 367, 420, 423-424, 437, 500, 720, 737, 775, 803, 806, 808, 813, 819, 833-838, 848-852, 855-872, 874-875, 878, 880, 891, 895-897, 901-905, 912, 914-916, 918, 920-924, 927-928, 930, 932-933, 935, 937, 939, 941-944, 946-951, 1004, 1006, 1008-1019, 1025-1027, 1040-1043, 1046-1047, 1051, 1088-1089, 1097-1098, 1112, 1114-1115, 1122-1126, 1129-1133, 1137, 1140-1149, 1153-1160, 1164, 1166, 1208, 1250-1258, 1275-1277, 1280-1285, 1351, 1362-1364, 1370-1371, 1374-1379, 1380-1381, 1385-1386, 1388, 1390, 1399, 4899, 4909, 4930, 4933, 4948, 4951-4952, 4960, 4971, 5012, 5039, 5104, 5123, 5142, 5210, 5346-5354, 8482, 8612, 9027, 9030, 9044, 9048-9049, 9056, 9061, 9066, 9238, 12712, 13121, 13218, 13488, 16684, 16804, 17248, 17584, 21427, 28709
1208	37, 256, 259, 273, 275, 277-278, 280, 282, 284-285, 290, 293, 297, 300-301, 367, 420, 423-424, 437, 500, 720, 737, 775, 803, 806, 808, 813, 819, 833-838, 848-852, 855-872, 874-875, 878, 880, 891, 895-897, 901-905, 912, 914-916, 918, 920-924, 927-928, 930, 932-933, 935, 937, 939, 941-944, 946-951, 1004, 1006, 1008-1019, 1025-1027, 1040-1043, 1046-1047, 1051, 1088-1089, 1097-1098, 1112, 1114-1115, 1122-1126, 1129-1133, 1137, 1140-1149, 1153-1160, 1164, 1166, 1200, 1250-1258, 1275-1277, 1280-1285, 1351, 1362-1364, 1370-1371, 1374-1379, 1380-1381, 1385-1386, 1388, 1390, 1399, 4899, 4909, 4930, 4933, 4948, 4951-4952, 4960, 4971, 5012, 5026, 5035, 5039, 5104, 5123, 5142, 5210, 5346-5354, 8482, 8612, 9027, 9030, 9044, 9048-9049, 9056, 9061, 9066, 9238, 12712, 13121, 13218, 13488, 16684, 16804, 17248, 17584, 21427, 28709
1250	37, 259, 273, 500, 819, 850, 852, 855, 870, 912, 1200, 1208, 1252, 1282, 4946, 4948, 4951, 5346, 8229, 9044, 13488, 17584, 28709
1251	37, 256, 259, 500, 819, 850, 855, 866, 878, 880, 915, 1025, 1123-1125, 1131, 1200, 1208, 1252, 1283, 4946, 4951, 5347, 8229, 13488, 17584, 28709
1252	37, 256, 259, 273, 275, 277-278, 280, 284-285, 290, 297, 420, 423-424, 437, 500, 737, 775, 803, 813, 819, 833, 836, 838, 850, 852, 855, 857-858, 860-866, 869-871, 874-875, 880, 897, 903, 905, 912, 914-916, 920-924, 1025-1027, 1041, 1047, 1051, 1097-1098, 1112, 1122-1123, 1126, 1130, 1132, 1140-1149, 1200, 1208, 1250-1251, 1254-1255, 1257, 1275, 1280-1281, 1283, 4386, 4909, 4929, 4932, 4934, 4946, 4948, 4951, 4953, 4960, 4970-4971, 5012, 5123, 5346, 5348, 8229, 8482, 8612, 9025, 9030, 9044, 9049, 9056, 9061, 9066, 13121, 13488, 16804, 17248, 17584, 25473, 25479, 25617, 28709
1253	37, 259, 423, 500, 737, 813, 819, 850, 869, 875, 1200, 1208, 1280, 4909, 4946, 4971, 5349, 8229, 9061, 13488, 17584, 28709

<i>Çizelge 675. IBM MQ for z/OS CCSID dönüştürme desteği (devamı var)</i>	
<b>CCSID</b>	<b>CCSID 'lere ve CCSID' ye dönüştürür</b>
1254	37, 259, 500, 819, 850, 857, 869, 905, 920, 1026, 1047, 1200, 1208, 1252, 1281, 4946, 4953, 5350, 8229, 9049, 9061, 13488, 17584, 28709
1255	37, 259, 424, 500, 803, 819, 850, 856, 862, 916, 1200, 1208, 1252, 1281, 4946, 4952, 5012, 5351, 8229, 13488, 17584, 28709
1256	259, 420, 500, 720, 850, 864, 1046, 1089, 1127, 1200, 1208, 4946, 4960, 5142, 5352, 8612, 9056, 9238, 13488, 16804, 17248, 17584
1257	37, 259, 437, 500, 775, 819, 850, 914, 921-922, 1112, 1122, 1200, 1208, 1252, 4946, 5353, 8229, 13488, 17584, 28709
1258	37, 259, 500, 819, 1129-1130, 1200, 1208, 5354, 8229, 13488, 17584, 28709
1275	37, 256, 273, 277-278, 280, 284-285, 297, 437, 500, 819, 850, 858, 863, 871, 923-924, 1051, 1140-1149, 1200, 1208, 1252, 4946, 5348, 8229, 13488, 17584, 28709
1276	1200, 1208, 13488, 17584
1277	1200, 1208, 13488, 17584
1280	37, 423, 437, 500, 737, 813, 819, 850, 869, 875, 1200, 1208, 1252-1253, 4909, 4946, 4971, 5349, 8229, 9061, 13488, 17584, 28709
1281	37, 437, 500, 819, 850, 857, 905, 920, 1026, 1200, 1208, 1252, 1254-1255, 4946, 4953, 5350, 8229, 9049, 13488, 17584, 28709
1282	500, 852, 870, 912, 1200, 1208, 1250, 4948, 5346, 9044, 13488, 17584
1283	37, 437, 500, 819, 850, 855, 866, 878, 880, 915, 1025, 1123-1125, 1131, 1200, 1208, 1251-1252, 4946, 4951, 5347, 8229, 13488, 17584, 28709
1284	1200, 1208, 13488, 17584
1285	1200, 1208, 13488, 17584
1351	300-301, 941, 1200, 1208, 4396, 8492, 13488, 16684, 17584
1362	834, 951, 1200, 1208, 4930, 9026, 13488, 17584
1363	933, 949, 1200, 1208, 1364, 5029, 5045, 5460, 9125, 9555, 13221, 13488, 13651, 17317, 17584, 25525, 29621, 33717, 37813
1364	933, 949, 1200, 1208, 1363, 5029, 5045, 5460, 9125, 9555, 13221, 13488, 13651, 17317, 17584, 25525, 29621, 33717, 37813
1370	937-938, 948, 950, 1200, 1208, 1371, 5033, 5046, 9142, 13488, 17584, 25514, 25524, 29620
1371	1200, 1208, 1370, 13488, 17584
1374	1200, 1208
1375	1200, 1208
1376	1200, 1208
1377	1200, 1208
1378	1200, 1208
1379	1200, 1208



<i>Çizelge 675. IBM MQ for z/OS CCSID dönüştürme desteği (devamı var)</i>	
<b>CCSID</b>	<b>CCSID 'lere ve CCSID' ye dönüştürür</b>
1380	837, 928, 1200, 1208, 1385, 4933, 13488, 17584
1381	935-936, 1200, 1208, 1386, 1388, 5031, 5477, 5482, 5484, 9127, 13223, 13488, 17584, 25512
1385	837, 1200, 1208, 1380, 4933, 13488, 17584
1386	935, 1200, 1208, 1381, 1388, 5031, 5477, 5482, 5484, 9127, 13223, 13488, 17584
1388	935, 1200, 1208, 1381, 1386, 5031, 5477, 5482, 5484, 5488, 9127, 13223, 13488, 17584
1390	930-932, 939, 942-943, 1200, 1208, 1399, 5026, 5028, 5035, 5038-5039, 5055, 9122, 9124, 9131, 9135, 13218-13219, 13231, 13488, 17314, 17584, 25508, 25518, 29614, 33698-33700, 37796
1399	930-932, 939, 942-943, 1200, 1208, 1390, 5026, 5028, 5035, 5038-5039, 5050, 9122, 9124, 9131, 9135, 13218-13219, 13231, 13488, 17314, 17584, 25508, 25518, 29614, 33698-33700, 37796
4386	37, 256, 273, 277-278, 280, 284-285, 290, 297, 367, 437, 500, 737, 775, 819, 833, 836, 850, 852, 855, 857, 860-865, 870-871, 895-897, 1009, 1025-1027, 1040-1043, 1088, 1112, 1122, 1139, 1252, 4929, 4932, 4946, 4948, 4951, 4953, 4960, 4992, 5123, 8229, 8482, 9025, 9044, 9049, 9056, 13121, 17248, 25473, 25617, 25619, 25664, 28709
4396	300-301, 941, 1351, 8492, 16684
4899	867, 1148, 1200, 1208, 5351, 9048, 12712, 13488, 17584
4909	37, 273, 277-278, 280, 284-285, 297, 423, 437, 500, 737, 813, 819, 838, 850, 852, 857, 860-861, 863, 869-871, 874-875, 880, 897, 903, 912, 916, 920, 1025-1027, 1041-1043, 1200, 1208, 1252-1253, 1280, 4934, 4946, 4948, 4953, 4970-4971, 5012, 5123, 5349, 8229, 9030, 9044, 9049, 9061, 9066, 13488, 17584, 25473, 25479, 25617, 25619, 28709
4929	37, 256, 273, 277-278, 280, 284-285, 290, 297, 367, 437, 500, 737, 775, 819, 833, 836, 850, 852, 855, 857, 860-865, 870-871, 891, 1009, 1025-1027, 1040-1043, 1088, 1112, 1122, 1126, 1252, 4386, 4932, 4946, 4948, 4951, 4953, 4960, 5123, 8229, 8482, 9025, 9044, 9049, 9056, 13121, 17248, 25617, 25619, 25664, 28709
4930	834, 951, 1200, 1208, 1362, 9026, 13488, 17584
4931	835, 927, 947, 9027, 21427
4932	37, 256, 273, 277-278, 280, 284-285, 290, 297, 367, 424, 437, 500, 737, 775, 819, 833, 836, 850, 852, 855, 857, 870-871, 875, 903, 1009, 1025-1027, 1040-1043, 1088, 1112, 1114-1115, 1122, 1252, 4386, 4929, 4946, 4948, 4951, 4953, 4971, 5123, 5210-5211, 8229, 8482, 9025, 9044, 9049, 13121, 25479, 25617, 25619, 25664, 28709
4933	837, 1200, 1208, 1380, 1385, 13488, 17584
4934	37, 256, 273, 277-278, 280, 284-285, 297, 423, 437, 500, 737, 775, 813, 819, 838, 850, 852, 857, 860-865, 869-871, 874-875, 880, 897, 903, 912, 916, 920, 1025-1027, 1041-1043, 1112, 1122, 1252, 4909, 4946, 4948, 4953, 4960, 4970-4971, 5012, 5123, 8229, 9030, 9044, 9049, 9056, 9061, 9066, 17248, 25473, 25479, 25617, 25619, 28709

Çizelge 675. IBM MQ for z/OS CCSID dönüştürme desteği (devamı var)

CCSID	CCSID 'lere ve CCSID' ye dönüştürür
4946	37, 256, 259, 273, 275, 277-278, 280, 284-285, 290, 297, 367, 420, 423-424, 437, 500, 737, 775, 803, 813, 819, 833, 836, 838, 850, 852, 855-858, 860-866, 869-871, 874-875, 880, 897, 903, 905, 912, 914-916, 920-924, 1004, 1025-1027, 1040-1043, 1047, 1051, 1088-1089, 1097-1098, 1100, 1112, 1114, 1122, 1126, 1130, 1132, 1140-1149, 1250-1257, 1275, 1280-1281, 1283, 4386, 4909, 4929, 4932, 4934, 4948, 4951-4953, 4960, 4970-4971, 5012, 5123, 5210, 5346, 5348, 8229, 8482, 8612, 9025, 9030, 9044, 9049, 9056, 9061, 9066, 13121, 16804, 17248, 25473, 25479, 25617, 25619, 25664, 28709
4948	37, 256, 259, 273, 277-278, 280, 284-285, 290, 297, 420, 423-424, 437, 500, 813, 819, 833, 836, 838, 850, 852, 855, 857, 860-861, 863, 869-871, 874-875, 880, 897, 903, 905, 912, 916, 920, 1025-1027, 1040-1043, 1047, 1088, 1097, 1200, 1208, 1250, 1252, 1282, 4386, 4909, 4929, 4932, 4934, 4946, 4951, 4953, 4970-4971, 5012, 5123, 5346, 8229, 8482, 8612, 9025, 9030, 9044, 9049, 9061, 9066, 13121, 13488, 16804, 17584, 25473, 25479, 25617, 25619, 25664, 28709
4951	37, 259, 273, 277-278, 280, 284-285, 290, 297, 437, 500, 819, 833, 836, 850, 852, 855, 857, 866, 870-871, 878, 880, 912, 915, 1025-1027, 1040-1043, 1088, 1200, 1208, 1250-1252, 1283, 4386, 4929, 4932, 4946, 4948, 4953, 5123, 5346, 5347, 8229, 8482, 9025, 9044, 9049, 13121, 13488, 17584, 25617, 25619, 25664, 28709
4952	259, 273, 424, 500, 803, 850, 856, 862, 916, 1200, 1208, 1255, 4946, 5012, 5351, 13488, 17584
4953	37, 256, 259, 273, 277-278, 280, 284-285, 290, 297, 420, 423-424, 437, 500, 813, 819, 833, 836, 838, 850, 852, 855, 857, 860-861, 863, 869-871, 874-875, 880, 897, 903, 905, 912, 916, 920, 1025-1027, 1040-1043, 1088, 1097, 1252, 1254, 1281, 4386, 4909, 4929, 4932, 4934, 4946, 4948, 4951, 4970-4971, 5012, 5123, 5350, 8229, 8482, 8612, 9025, 9030, 9044, 9049, 9061, 9066, 13121, 16804, 25473, 25479, 25617, 25619, 25664, 28709
4960	37, 256, 259, 273, 277-278, 280, 284-285, 290, 297, 420, 423-424, 500, 720, 819, 833, 838, 850, 864, 870-871, 875, 880, 905, 918, 1008, 1025-1027, 1046, 1089, 1097, 1127, 1200, 1208, 1252, 1256, 4386, 4929, 4934, 4946, 4971, 5104, 5123, 5142, 5352, 8229, 8482, 8612, 9025, 9030, 9056, 9238, 13121, 13488, 16804, 17248, 17584, 28709
4970	37, 259, 273, 277-278, 280, 284-285, 297, 423, 437, 500, 813, 819, 838, 850, 852, 857, 860-861, 863, 869-871, 874-875, 880, 897, 903, 912, 916, 920, 1025-1027, 1041-1043, 1252, 4909, 4934, 4946, 4948, 4953, 4971, 5012, 5123, 8229, 9030, 9044, 9049, 9061, 9066, 25473, 25479, 25617, 25619, 28709
4971	37, 256, 273, 277-278, 280, 284-285, 297, 367, 423, 437, 500, 737, 775, 813, 819, 836, 838, 850-852, 857, 860-865, 869-871, 874-875, 880, 897, 903, 912, 916, 920, 1009, 1025-1027, 1041-1043, 1047, 1088, 1112, 1122, 1200, 1208, 1252-1253, 1280, 4909, 4932, 4934, 4946, 4948, 4953, 4960, 4970, 5012, 5123, 5349, 8229, 9030, 9044, 9049, 9056, 9061, 9066, 13488, 17248, 17584, 25473, 25479, 25617, 25619, 25664, 28709
4992	290, 896, 1027, 1041, 4386, 5123, 8482, 25617

Çizelge 675. IBM MQ for z/OS CCSID dönüştürme desteği (devamı var)

CCSID	CCSID 'lere ve CCSID' ye dönüştürür
5012	37, 273, 277-278, 280, 284-285, 297, 423-424, 437, 500, 803, 813, 819, 838, 850, 852, 856-857, 860-863, 869-871, 874-875, 880, 897, 903, 912, 916, 920, 1025-1027, 1041-1043, 1200, 1208, 1252, 1255, 4909, 4934, 4946, 4948, 4952-4953, 4970-4971, 5123, 5351, 8229, 9030, 9044, 9049, 9061, 9066, 13488, 17584, 25473, 25479, 25617, 25619, 28709
5026	930-932, 939, 942-943, 1390, 1399, 5028, 5035, 5038-5039, 9122, 9124, 9131, 9135, 1208, 13218-13219, 13231, 17314, 25508, 25518, 29614, 33698-33700, 37796
5028	930-932, 939, 942-943, 1390, 1399, 5026, 5035, 5038-5039, 9122, 9124, 9131, 9135, 13218-13219, 13231, 17314, 25508, 25518, 29614, 33698-33700, 37796
5029	933-934, 944, 949, 1363-1364, 5045, 5460, 9125, 9555, 13221, 13651, 17317, 25510, 25520, 25525, 29616, 29621, 33717, 37813
5031	935-936, 946, 1381, 1386, 1388, 5477, 5482, 5484, 9127, 13223, 25512
5033	937-938, 948, 950, 1370, 5046, 9142, 25514, 25524, 29620
5035	930-932, 939, 942-943, 1390, 1399, 5026, 5028, 5038-5039, 9122, 9124, 9131, 9135, 1208, 13218-13219, 13231, 17314, 25508, 25518, 29614, 33698-33700, 37796
5038	930-932, 939, 942-943, 1390, 1399, 5026, 5028, 5035, 5039, 9122, 9124, 9131, 9135, 13218-13219, 13231, 17314, 25508, 25518, 29614, 33698-33700, 37796
5039	930-932, 939, 942-943, 1200, 1208, 1390, 1399, 5026, 5028, 5035, 5038, 9122, 9124, 9131, 9135, 13218-13219, 13231, 13488, 17314, 17584, 25508, 25518, 29614, 33698-33700, 37796
5045	933-934, 944, 949, 1363-1364, 5029, 5460, 9125, 9555, 13221, 13651, 17317, 25510, 25520, 25525, 29616, 29621, 33717, 37813
5046	937-938, 948, 950, 1370, 5033, 9142, 25514, 25524, 29620
5104	420, 864, 1008, 1200, 1208, 4960, 8612, 9056, 13488, 16804, 17248, 17584
5123	290, 367, 423, 437, 819, 1027, 1041, 1047, 1140-1149, 1156, 1157, 1160, 1200, 1208, 1252, 4948, 5348, 8482, 13488
5142	420, 500, 864, 1046, 1089, 1127, 1200, 1208, 1256, 4960, 5352, 8612, 9056, 9238, 13488, 16804, 17248, 17584
5210	37, 437, 500, 819, 836, 850, 904, 1043, 1114-1115, 1200, 1208, 4932, 4946, 5211, 8229, 13488, 17584, 25480, 25619, 28709
5211	37, 367, 437, 500, 836, 903, 1114-1115, 4932, 5210, 8229, 25479, 28709
5346	37, 259, 273, 500, 819, 850, 852, 855, 870, 912, 1200, 1208, 1250, 1252, 1282, 4946, 4948, 4951, 8229, 9044, 13488, 17584, 28709
5347	808, 848-849, 855, 866, 872, 878, 880, 915, 1025, 1123-1125, 1131, 1154, 1158, 1200, 1208, 1251, 1283, 4951, 13488, 17584

<i>Çizelge 675. IBM MQ for z/OS CCSID dönüştürme desteği (devamı var)</i>	
<b>CCSID</b>	<b>CCSID 'lere ve CCSID' ye dönüştürür</b>
5348	37, 259, 273, 275, 277-278, 280, 284-285, 297, 437, 500, 808, 819, 850, 858, 860-861, 863, 865, 871-872, 901-902, 923-924, 1051, 1140-1149, 1153-1158, 1160-1162, 1164, 1200, 1208, 1252, 1275, 4946, 8229, 13488, 17584, 28709
5349	813, 869, 875, 1148, 1200, 1208, 1253, 1280, 4909, 4971, 9061, 13488, 17584
5350	857, 920, 1026, 1155, 1200, 1208, 1254, 1281, 4953, 9049, 13488, 17584
5351	424, 856, 862, 867, 916, 1200, 1208, 1255, 4899, 4952, 5012, 9048, 12712, 13488, 17584
5352	420, 864, 1046, 1089, 1200, 1208, 1256, 4960, 5142, 8612, 9056, 9238, 13488, 16804, 17248, 17584
5353	901-902, 921-922, 1112, 1122, 1156-1157, 1200, 1208, 1257, 13488, 17584
5354	1129-1130, 1163, 1164, 1200, 1208, 1258, 13488, 17584
5460	933-934, 944, 949, 1363-1364, 5029, 5045, 9125, 9555, 13221, 13651, 17317, 25510, 25520, 25525, 29616, 29621, 33717, 37813
5477	935-936, 1381, 1386, 1388, 5031, 5482, 5484, 9127, 13223, 25512
5482	935, 1381, 1386, 1388, 5031, 5477, 5484, 9127, 13223
5484	935-936, 946, 1381, 1386, 1388, 5031, 5477, 5482, 9127, 13223, 25512
5488	1388
8229	37, 256, 273, 275, 277-278, 280, 284-285, 290, 297, 367, 420, 423-424, 437, 500, 720, 737, 775, 813, 819, 833, 836, 838, 850, 852, 855, 857-858, 860-866, 869-871, 874-875, 880, 897, 903-905, 912, 914-916, 920-924, 1009, 1025-1027, 1040-1043, 1047, 1051, 1088, 1097, 1100, 1112, 1114-1115, 1122, 1124, 1126, 1130-1132, 1137, 1140-1149, 1250-1255, 1257-1258, 1275, 1280-1281, 1283, 4386, 4909, 4929, 4932, 4934, 4946, 4948, 4951, 4953, 4960, 4970-4971, 5012, 5123, 5210-5211, 5346, 5348, 8482, 8612, 9025, 9030, 9044, 9049, 9056, 9061, 9066, 13121, 16804, 17248, 25473, 25479, 25480, 25617, 25619, 25664, 28709
8482	37, 256, 273, 277-278, 280, 284-285, 290, 297, 367, 437, 500, 737, 775, 819, 833, 836, 850, 852, 855, 857, 860-865, 870-871, 895-897, 1009, 1025-1027, 1040-1043, 1047, 1088, 1112, 1122, 1139, 1200, 1208, 1252, 4386, 4929, 4932, 4946, 4948, 4951, 4953, 4960, 4992, 5123, 8229, 9025, 9044, 9049, 9056, 13121, 13488, 17248, 17584, 25473, 25617, 25619, 25664, 28709
8492	300-301, 941, 1351, 4396, 16684
8612	37, 256, 420, 424, 437, 500, 720, 737, 775, 819, 850, 852, 857, 860-865, 1008, 1046, 1089, 1098, 1112, 1122, 1127, 1200, 1208, 1252, 1256, 4946, 4948, 4953, 4960, 5104, 5142, 5352, 8229, 9044, 9049, 9056, 9238, 13488, 16804, 17248, 17584, 28709

Çizelge 675. IBM MQ for z/OS CCSID dönüştürme desteği (devamı var)

CCSID	CCSID 'lere ve CCSID' ye dönüştürür
9025	37, 256, 273, 277-278, 280, 284-285, 290, 297, 367, 437, 500, 737, 775, 819, 833, 836, 850, 852, 855, 857, 860-865, 870-871, 891, 1009, 1025-1027, 1040-1043, 1088, 1112, 1122, 1126, 1252, 4386, 4929, 4932, 4946, 4948, 4951, 4953, 4960, 5123, 8229, 8482, 9044, 9049, 9056, 13121, 17248, 25617, 25619, 25664, 28709
9026	834, 926, 951, 1362, 4930
9027	835, 927, 947, 1200, 1208, 4931, 13488, 17584, 21427
9030	37, 256, 273, 277-278, 280, 284-285, 297, 423, 437, 500, 737, 775, 813, 819, 838, 850, 852, 857, 860-865, 869-871, 874-875, 880, 897, 903, 912, 916, 920, 1025-1027, 1041-1043, 1112, 1122, 1200, 1208, 1252, 4909, 4934, 4946, 4948, 4953, 4960, 4970-4971, 5012, 5123, 8229, 9044, 9049, 9056, 9061, 9066, 13488, 17248, 17584, 25473, 25479, 25617, 25619, 28709
9044	37, 256, 259, 273, 277-278, 280, 284-285, 290, 297, 420, 423-424, 437, 500, 813, 819, 833, 836, 838, 850, 852, 855, 857, 860-861, 863, 869-871, 874-875, 880, 897, 903, 905, 912, 916, 920, 1025-1027, 1040-1043, 1047, 1088, 1097, 1153, 1200, 1208, 1250, 1252, 1282, 4386, 4909, 4929, 4932, 4934, 4946, 4948, 4951, 4953, 4970-4971, 5012, 5123, 5346, 8229, 8482, 8612, 9025, 9030, 9049, 9061, 9066, 13121, 13488, 16804, 17584, 25473, 25479, 25617, 25619, 25664, 28709
9048	867, 1200, 1208, 4899, 5351, 12712, 13488, 17584
9049	37, 256, 259, 273, 277-278, 280, 284-285, 290, 297, 420, 423-424, 437, 500, 813, 819, 833, 836, 838, 850, 852, 855, 857, 860-861, 863, 869-871, 874-875, 880, 897, 903, 905, 912, 916, 920, 1025-1027, 1040-1043, 1088, 1097, 1155, 1200, 1208, 1252, 1254, 1281, 4386, 4909, 4929, 4932, 4934, 4946, 4948, 4951, 4953, 4970-4971, 5012, 5123, 5350, 8229, 8482, 8612, 9025, 9030, 9044, 9061, 9066, 13121, 13488, 16804, 17584, 25473, 25479, 25617, 25619, 25664, 28709
9056	37, 256, 259, 273, 277-278, 280, 284-285, 290, 297, 420, 423-424, 500, 720, 819, 833, 838, 850, 864, 870-871, 875, 880, 905, 918, 1008, 1025-1027, 1046, 1089, 1097, 1127, 1200, 1208, 1252, 1256, 4386, 4929, 4934, 4946, 4960, 4971, 5104, 5123, 5142, 5352, 8229, 8482, 8612, 9025, 9030, 9238, 13121, 13488, 16804, 17248, 17584, 28709
9061	37, 256, 259, 273, 277-278, 280, 284-285, 297, 423, 437, 500, 737, 813, 819, 838, 850, 852, 857, 860-861, 863, 869-871, 874-875, 880, 897, 903, 912, 916, 920, 1025-1027, 1041-1043, 1200, 1208, 1252-1254, 1280, 4909, 4934, 4946, 4948, 4953, 4970-4971, 5012, 5123, 5349, 8229, 9030, 9044, 9049, 9066, 13488, 17584, 25473, 25479, 25617, 25619, 28709
9066	37, 259, 273, 277-278, 280, 284-285, 297, 423, 437, 500, 813, 819, 838, 850, 852, 857, 860-861, 863, 869-871, 874-875, 880, 897, 903, 912, 916, 920, 1025-1027, 1041-1043, 1200, 1208, 1252, 4909, 4934, 4946, 4948, 4953, 4970-4971, 5012, 5123, 8229, 9030, 9044, 9049, 9061, 13488, 17584, 25473, 25479, 25617, 25619, 28709
9122	930-932, 939, 942-943, 1390, 1399, 5026, 5028, 5035, 5038-5039, 9124, 9131, 9135, 13218-13219, 13231, 17314, 25508, 25518, 29614, 33698-33700, 37796

<i>Çizelge 675. IBM MQ for z/OS CCSID dönüştürme desteği (devamı var)</i>	
<b>CCSID</b>	<b>CCSID 'lere ve CCSID' ye dönüştürür</b>
9124	930-932, 939, 942-943, 1390, 1399, 5026, 5028, 5035, 5038-5039, 9122, 9131, 9135, 13218-13219, 13231, 17314, 25508, 25518, 29614, 33698-33700, 37796
9125	933-934, 944, 949, 1363-1364, 5029, 5045, 5460, 9555, 13221, 13651, 17317, 25510, 25520, 25525, 29616, 29621, 33717, 37813
9127	935-936, 946, 1381, 1386, 1388, 5031, 5477, 5482, 5484, 13223, 25512
9131	930-932, 939, 942-943, 1390, 1399, 5026, 5028, 5035, 5038-5039, 9122, 9124, 9135, 13218-13219, 13231, 17314, 25508, 25518, 29614, 33698-33700, 37796
9135	930-932, 939, 942-943, 1390, 1399, 5026, 5028, 5035, 5038-5039, 9122, 9124, 9131, 13218-13219, 13231, 17314, 25508, 25518, 29614, 33698-33700, 37796
9142	937-938, 948, 950, 1370, 5033, 5046, 25514, 25524, 29620
9238	420, 500, 864, 1046, 1089, 1127, 1200, 1208, 1256, 4960, 5142, 5352, 8612, 9056, 13488, 16804, 17248, 17584
9555	933, 949, 1363-1364, 5029, 5045, 5460, 9125, 13221, 13651, 17317, 25525, 29621, 33717, 37813
12712	862, 867, 1148, 1156-1157, 1200, 1208, 4899, 5351, 9048, 13488, 17584
13121	37, 256, 273, 277-278, 280, 284-285, 290, 297, 367, 437, 500, 737, 775, 819, 833, 836, 850, 852, 855, 857, 860-865, 870-871, 891, 1009, 1025-1027, 1040-1043, 1088, 1112, 1122, 1126, 1200, 1208, 1252, 4386, 4929, 4932, 4946, 4948, 4951, 4953, 4960, 5123, 8229, 8482, 9025, 9044, 9049, 9056, 13488, 17248, 17584, 25617, 25619, 25664, 28709
13218	930-932, 939, 942-943, 1200, 1208, 1390, 1399, 5026, 5028, 5035, 5038-5039, 9122, 9124, 9131, 9135, 13219, 13231, 13488, 17314, 17584, 25508, 25518, 29614, 33698-33700, 37796
13219	930-932, 939, 942-943, 1390, 1399, 5026, 5028, 5035, 5038-5039, 9122, 9124, 9131, 9135, 13218, 13231, 17314, 25508, 25518, 29614, 33698-33700, 37796
13221	933-934, 944, 949, 1363-1364, 5029, 5045, 5460, 9125, 9555, 13651, 17317, 25510, 25520, 25525, 29616, 29621, 33717, 37813
13223	935-936, 946, 1381, 1386, 1388, 5031, 5477, 5482, 5484, 9127, 25512
13231	930-932, 939, 942-943, 1390, 1399, 5026, 5028, 5035, 5038-5039, 9122, 9124, 9131, 9135, 13218-13219, 17314, 25508, 25518, 29614, 33698-33700, 37796

Çizelge 675. IBM MQ for z/OS CCSID dönüştürme desteği (devamı var)

CCSID	CCSID 'lere ve CCSID' ye dönüştürür
13488	37, 256, 259, 273, 275, 277-278, 280, 282, 284-285, 290, 293, 297, 300-301, 367, 420, 423-424, 437, 500, 720, 737, 775, 803, 806, 808, 813, 819, 833-838, 848-852, 855-872, 874-875, 878, 880, 891, 895-897, 901-905, 912, 914-916, 918, 920-924, 927-928, 930, 932-933, 935, 937, 939, 941-944, 946-951, 1004, 1006, 1008-1019, 1025-1027, 1040-1043, 1046-1047, 1051, 1088-1089, 1097-1098, 1112, 1114-1115, 1122-1126, 1129-1133, 1137, 1140-1149, 1153-1160, 1164, 1166, 1200, 1208, 1250-1258, 1275-1277, 1280-1285, 1351, 1362-1364, 1370-1371, 1380-1381, 1385-1386, 1388, 1390, 1399, 4899, 4909, 4930, 4933, 4948, 4951-4952, 4960, 4971, 5012, 5039, 5104, 5123, 5142, 5210, 5346-5354, 8482, 8612, 9027, 9030, 9044, 9048-9049, 9056, 9061, 9066, 9238, 12712, 13121, 13218, 16684, 16804, 17248, 17584, 21427, 28709
13651	933, 949, 1363-1364, 5029, 5045, 5460, 9125, 9555, 13221, 17317, 25525, 29621, 33717, 37813
16684	300-301, 941, 1200, 1208, 1351, 4396, 8492, 13488, 17584
16804	37, 256, 420, 424, 437, 500, 720, 737, 775, 819, 850, 852, 857, 860-865, 1008, 1046, 1089, 1098, 1112, 1122, 1127, 1200, 1208, 1252, 1256, 4946, 4948, 4953, 4960, 5104, 5142, 5352, 8229, 8612, 9044, 9049, 9056, 9238, 13488, 17248, 17584, 28709
17248	37, 256, 259, 273, 277-278, 280, 284-285, 290, 297, 420, 423-424, 500, 720, 819, 833, 838, 850, 864, 870-871, 875, 880, 905, 918, 1008, 1025-1027, 1046, 1089, 1097, 1127, 1200, 1208, 1252, 1256, 4386, 4929, 4934, 4946, 4960, 4971, 5104, 5123, 5142, 5352, 8229, 8482, 8612, 9025, 9030, 9056, 9238, 13121, 13488, 16804, 17584, 28709
17314	930-932, 939, 942-943, 1390, 1399, 5026, 5028, 5035, 5038-5039, 9122, 9124, 9131, 9135, 13218-13219, 13231, 25508, 25518, 29614, 33698-33700, 37796
17317	933-934, 944, 949, 1363-1364, 5029, 5045, 5460, 9125, 9555, 13221, 13651, 25510, 25520, 25525, 29616, 29621, 33717, 37813
17584	37, 256, 259, 273, 275, 277-278, 280, 282, 284-285, 290, 293, 297, 300-301, 367, 420, 423-424, 437, 500, 720, 737, 775, 803, 806, 808, 813, 819, 833-838, 848-852, 855-872, 874-875, 878, 880, 891, 895-897, 901-905, 912, 914-916, 918, 920-924, 927-928, 930, 932-933, 935, 937, 939, 941-944, 946-951, 1004, 1006, 1008-1019, 1025-1027, 1040-1043, 1046-1047, 1051, 1088-1089, 1097-1098, 1112, 1114-1115, 1122-1126, 1129-1133, 1137, 1140-1149, 1153-1160, 1164, 1166, 1200, 1208, 1250-1258, 1275-1277, 1280-1285, 1351, 1362-1364, 1370-1371, 1380-1381, 1385-1386, 1388, 1390, 1399, 4899, 4909, 4930, 4933, 4948, 4951-4952, 4960, 4971, 5012, 5039, 5104, 5123, 5142, 5210, 5346-5354, 8482, 8612, 9027, 9030, 9044, 9048-9049, 9056, 9061, 9066, 9238, 12712, 13121, 13218, 13488, 16684, 16804, 17248, 21427, 28709
21427	835, 927, 947, 1200, 1208, 4931, 9027, 13488, 17584
25473	37, 273, 277-278, 280, 284-285, 290, 297, 423, 437, 500, 813, 819, 838, 850, 852, 857, 860-861, 863, 869-871, 874-875, 880, 897, 903, 912, 916, 920, 1025-1027, 1041-1043, 1252, 4386, 4909, 4934, 4946, 4948, 4953, 4970-4971, 5012, 5123, 8229, 8482, 9030, 9044, 9049, 9061, 9066, 25479, 25617, 25619, 28709

<i>Çizelge 675. IBM MQ for z/OS CCSID dönüştürme desteği (devamı var)</i>	
<b>CCSID</b>	<b>CCSID 'lere ve CCSID' ye dönüştürür</b>
25479	37, 273, 277-278, 280, 284-285, 297, 423, 437, 500, 813, 819, 836, 838, 850, 852, 857, 860-861, 863, 869-871, 874-875, 880, 897, 903, 912, 916, 920, 1025-1027, 1041-1043, 1115, 1252, 4909, 4932, 4934, 4946, 4948, 4953, 4970-4971, 5012, 5123, 5211, 8229, 9030, 9044, 9049, 9061, 9066, 25473, 25617, 25619, 28709
25480	37, 500, 904, 1114, 5210, 8229, 28709
25508	930-932, 939, 942-943, 1390, 1399, 5026, 5028, 5035, 5038-5039, 9122, 9124, 9131, 9135, 13218-13219, 13231, 17314, 25518, 29614, 33698-33700, 37796
25510	933-934, 949, 5029, 5045, 5460, 9125, 13221, 17317, 25525, 29621, 33717, 37813
25512	935-936, 946, 1381, 5031, 5477, 5484, 9127, 13223
25514	937-938, 950, 1370, 5033, 5046, 9142
25518	930-932, 939, 942-943, 1390, 1399, 5026, 5028, 5035, 5038-5039, 9122, 9124, 9131, 9135, 13218-13219, 13231, 17314, 25508, 29614, 33698-33700, 37796
25520	933, 944, 949, 5029, 5045, 5460, 9125, 13221, 17317, 25525, 29616, 29621, 33717, 37813
25524	937, 948, 950, 1370, 5033, 5046, 9142, 29620
25525	933-934, 944, 949, 1363-1364, 5029, 5045, 5460, 9125, 9555, 13221, 13651, 17317, 25510, 25520, 29616, 29621, 33717, 37813
25617	37, 273, 277-278, 280, 284-285, 290, 297, 367, 423, 437, 500, 813, 819, 833, 836, 838, 850, 852, 855, 857, 860-861, 863, 869-871, 874-875, 880, 895-897, 903, 912, 916, 1025-1027, 1040-1043, 1088, 1252, 4386, 4909, 4929, 4932, 4934, 4946, 4948, 4951, 4953, 4970-4971, 4992, 5012, 5123, 8229, 8482, 9025, 9030, 9044, 9049, 9061, 9066, 13121, 25473, 25479, 25619, 25664, 28709
25619	37, 273, 277-278, 280, 284-285, 290, 297, 423, 437, 500, 813, 819, 833, 836, 838, 850, 852, 855, 857, 860-861, 863, 869-871, 874-875, 880, 897, 903, 912, 916, 1025-1027, 1040-1043, 1088, 1114, 4386, 4909, 4929, 4932, 4934, 4946, 4948, 4951, 4953, 4970-4971, 5012, 5123, 5210, 8229, 8482, 9025, 9030, 9044, 9049, 9061, 9066, 13121, 25473, 25479, 25617, 25664, 28709
25664	37, 273, 277-278, 280, 284-285, 290, 297, 367, 500, 819, 833, 836, 850, 852, 855, 857, 870-871, 875, 891, 1025-1027, 1040-1043, 1088, 1126, 4386, 4929, 4932, 4946, 4948, 4951, 4953, 4971, 5123, 8229, 8482, 9025, 9044, 9049, 13121, 25617, 25619, 28709



Çizelge 675. IBM MQ for z/OS CCSID dönüştürme desteği (devamı var)	
CCSID	CCSID 'lere ve CCSID' ye dönüştürür
28709	37, 256, 273, 275, 277-278, 280, 284-285, 290, 297, 367, 420, 423-424, 437, 500, 720, 737, 775, 813, 819, 833, 836, 838, 850, 852, 855, 857-858, 860-866, 869-871, 874-875, 880, 897, 903-905, 912, 914-916, 920-924, 1009, 1025-1027, 1040-1043, 1047, 1051, 1088, 1097, 1100, 1112, 1114-1115, 1122, 1124, 1126, 1130-1132, 1137, 1140-1149, 1200, 1208, 1250-1255, 1257-1258, 1275, 1280-1281, 1283, 4386, 4909, 4929, 4932, 4934, 4946, 4948, 4951, 4953, 4960, 4970-4971, 5012, 5123, 5210-5211, 5346, 5348, 8229, 8482, 8612, 9025, 9030, 9044, 9049, 9056, 9061, 9066, 13121, 13488, 16804, 17248, 17584, 25473, 25479, 25480, 25617, 25619, 25664
29614	930-932, 939, 942-943, 1390, 1399, 5026, 5028, 5035, 5038-5039, 9122, 9124, 9131, 9135, 13218-13219, 13231, 17314, 25508, 25518, 33698-33700, 37796
29616	933, 944, 949, 5029, 5045, 5460, 9125, 13221, 17317, 25520, 25525, 29621, 33717, 37813
29620	937, 948, 950, 1370, 5033, 5046, 9142, 25524
29621	933-934, 944, 949, 1363-1364, 5029, 5045, 5460, 9125, 9555, 13221, 13651, 17317, 25510, 25520, 25525, 29616, 33717, 37813
33698	930-932, 939, 942-943, 1390, 1399, 5026, 5028, 5035, 5038-5039, 9122, 9124, 9131, 9135, 13218-13219, 13231, 17314, 25508, 25518, 29614, 33699-33700, 37796
33699	930-932, 939, 942-943, 1390, 1399, 5026, 5028, 5035, 5038-5039, 9122, 9124, 9131, 9135, 13218-13219, 13231, 17314, 25508, 25518, 29614, 33698, 33700, 37796
33700	930-932, 939, 942-943, 1390, 1399, 5026, 5028, 5035, 5038-5039, 9122, 9124, 9131, 9135, 13218-13219, 13231, 17314, 25508, 25518, 29614, 33698-33699, 37796
33717	933-934, 944, 949, 1363-1364, 5029, 5045, 5460, 9125, 9555, 13221, 13651, 17317, 25510, 25520, 25525, 29616, 29621, 37813
37796	930-932, 939, 942-943, 1390, 1399, 5026, 5028, 5035, 5038-5039, 9122, 9124, 9131, 9135, 13218-13219, 13231, 17314, 25508, 25518, 29614, 33698-33700
37813	933-934, 944, 949, 1363-1364, 5029, 5045, 5460, 9125, 9555, 13221, 13651, 17317, 25510, 25520, 25525, 29616, 29621, 33717

## IBM i IBM i dönüştürme desteği

CCSID 'lerin tam listesi ve IBM itarafından desteklenen dönüştürmeler, uygun IBM i yayınında bulunabilir.

Desteklenen kod sayfaları, [Desteklenen CCSID eşlemelerilistesinde](#) yer alıyor.

## Unicode dönüştürme desteği

Bazı platformlar, kullanıcı verilerinin Unicode kodlamasına ya da Unicode kodlamasından dönüştürülmesini destekler. Desteklenen iki Unicode kodlaması biçimi UTF-16 (CCSID 1200, 13488 ve 17584) ve UTF-8 (CCSID 1208) biçimleridir. Desteklenen en son Unicode sürümünü gösterdikleri için CCSID 1200 ya da 1208 kullanmalısınız.

UTF-16 vekil çiftleri (U + FFFF üzerinde bir Unicode kod noktasını gösteren X'D800' aralığındaki 2 baytlık UTF-16 karakter çifti) desteklenir. Hedef CCSID, UTF-16 yedek çifti tarafından gösterilen bir kod noktası için eşleme içermiyorsa, karakter çifti tek bir yerine koyma karakterine dönüştürülür.

IBM MQ, karakter sıralarını birleştirmeyi destekler. Bu, bazı durumlarda kaynak CCSID 'de önceden oluşturulmuş bir karakterin hedef CCSID' de ya da diğer bir şekilde bir araya getirilmiş karakter sırasına dönüştürüleceği anlamına gelir.

**Not:** IBM MQ , UTF-16 kuyruk yöneticisi CCSID ' leri desteklemediğinden ileti üstbilgisi verileri UTF-16biçiminde kodlanamıyor.

## Unicode için IBM MQ AIX desteği



IBM MQ for AIX dönüşümünde, aşağıdaki listede Unicode CCSID 'leri olmayan CCSID' ler için desteklenen Unicode CCSID ' leri (tercihen 1200 ya da 1208) desteklenir:

037  
273, 278, 280, 284, 285, 297  
423, 437  
500  
813, 819, 850, 852, 856, 857, 858, 860, 861, 865, 867, 869, 875, 878, 880  
901, 902, 912, 915, 916, 920, 923, 924, 932, 933, 935, 937, 938, 939, 942, 943, 948, 949, 950, 954, 964, 970  
1026, 1046, 1089  
1129, 1130, 1131, 1132, 1133, 1140, 1141, 1142, 1143, 1144, 1145, 1146, 1147, 1148, 1149, 1153, 1156, 1157  
1200, 1208, 1250, 1251, 1253, 1254, 1258, 1280, 1281, 1282, 1283, 1284, 1285  
1363, 1364, 1381, 1383, 1386, 1388  
4899  
5026, 5035, 5050, 5346, 5347, 5348, 5349, 5350, 5351, 5352, 5353, 5354, 5488  
9044, 9048, 9449  
12712  
13488  
17584  
33722

## IBM MQ for Windows ve Linux Unicode desteği



IBM MQ for Windows ve IBM MQ for Linux üzerinde, aşağıdaki listede Unicode CCSID 'leri olmayan CCSID' ler için desteklenen Unicode CCSID ' leri (tercihen 1200 ya da 1208) desteklenir:

037.  
277, 278, 280, 284, 285, 290, 297  
300, 301  
420, 424, 437  
500  
813, 819, 833, 835, 836, 837, 838, 850, 852, 855, 856, 857, 858, 860, 861, 862, 863, 864, 865, 866, 867, 868, 869, 870, 871, 874, 875, 878, 880, 891, 897  
901, 902, 903, 904, 912, 913<sup>“5” sayfa 975</sup>, 915, 916, 918, 920, 921, 922, 923, 924, 927, 928, 930, 931<sup>“1” sayfa 975</sup>, 932<sup>“2” sayfa 975</sup>, 933, 935, 937, 938<sup>“3” sayfa 975</sup>, 939, 941, 942, 943, 947, 948, 949, 950, 951, 954<sup>“4” sayfa 975</sup>, 964, 970  
1006, 1025, 1026, 1027, 1040, 1041, 1042, 1043, 1046, 1047, 1051, 1088, 1089, 1097, 1098  
1112, 1114, 1115, 1122, 1123, 1124, 1129, 1130, 1132, 1133, 1140, 1141, 1142, 1143, 1144, 1145, 1146, 1147, 1148, 1149, 1153, 1156, 1157

1200, 1208, 1250, 1251, 1252, 1253, 1254, 1255, 1256, 1257, 1258, 1275, 1280, 1281, 1282, 1283  
1363, 1364, 1374, 1375, 1376, 1377, 1378, 1379, 1380, 1381, 1383, 1386, 1388  
4899  
5050, 5346, 5347, 5348, 5349, 5350, 5351, 5352, 5353, 5354, 5488<sup>"5"</sup> sayfa 975  
9044, 9048, 9449  
12712  
13488  
17584  
33722<sup>"4"</sup> sayfa 975

#### Notlar:

1. 931, dönüştürme için 939 kullanır.
2. 932, dönüştürme için 942 kullanır.
3. 938, dönüştürme için 948 kullanır.
4. 954 ve 33722, dönüştürme için 5050 'yi kullanır.
5. Yalnızca Windows ve Linux üzerinde.

### Unicode için IBM i desteği



UNICODE desteğine ilişkin ayrıntılar için işletim sisteminizle ilgili uygun IBM i yayınına bakın.

### Unicode için IBM MQ for z/OS desteği



IBM MQ for z/OS dönüşümünde, aşağıdaki listede Unicode CCSID 'leri olmayan CCSID' ler için desteklenen Unicode CCSID 'leri (tercihen 1200 ya da 1208) desteklenir:

37  
256, 259, 273, 275, 277, 278, 280, 282, 284, 285, 290, 293, 297  
300, 301, 367  
420, 423, 424, 437  
500  
720, 737, 775.  
803, 806, 808, 813, 819, 833, 834, 835, 836, 837, 838, 848, 849, 850, 851, 852, 855, 856, 857, 858,  
859, 860, 861, 862, 863, 864, 865, 866, 867, 868, 869, 870, 871, 872, 874, 875, 878, 880, 891, 895,  
896, 897  
901, 902, 903, 904, 905, 912, 914, 915, 916, 918, 920, 921, 922, 923, 924, 927, 928, 930, 932, 933,  
935, 937, 939, 941, 942, 943, 944, 946, 947, 948, 949, 950, 951  
1004, 1006, 1008, 1009, 1010, 1011, 1012, 1013, 1014, 1015, 1016, 1017, 1018, 1019, 1025,  
1026, 1027, 1040, 1041, 1042, 1043, 1046, 1047, 1051, 1088, 1089, 1097, 1098  
1112, 1114, 1115, 1122, 1123, 1124, 1125, 1126, 1129, 1130, 1131, 1132, 1133, 1137, 1140,  
1141, 1142, 1143, 1144, 1145, 1146, 1147, 1148, 1149, 1153, 1154, 1155, 1156, 1157, 1158,  
1159, 1160, 1161, 1161, 1164  
1200, 1208, 1250, 1251, 1252, 1253, 1254, 1255, 1256, 1257, 1258, 1275, 1276, 1277, 1280,  
1281, 1282, 1283, 1284, 1285  
1351, 1362, 1363, 1364, 1370, 1371, 1380, 1381, 1385, 1386, 1388, 1390, 1399  
4899, 4909, 4930, 4933, 4948, 4951, 4952, 4960, 4971  
5012 5039 5104 5123 5142 5210 5346 5347 5348 5349 5350 5351 5352 5353 5354 5488  
8482 8612  
9027 9030 9044 9048 9049 9056 9061 9066 9238 9449  
1166

12712  
13121, 13218, 13488, 1374, 1375, 1376, 1377, 1378, 1379  
16684, 16804  
17248, 17584  
21427  
28709

## 64 bit altyapılarda kodlama standartları

64 bit altyapılarda ve tercih edilen veri tiplerinde kodlama standartlarıyla ilgili bilgi edinmek için bu bilgileri kullanın.

### Tercih edilen veri tipleri

Bu tipler hiçbir zaman boyut deęiřtirme ve hem 32 bit, hem de 64 bit IBM MQ altyapılarında kullanılabilir:

*Çizelge 676. Veri tipi adları ve uzunlukları*

Ad	Uzunluk
MQLONG	4 Byte
MQULONG	4 Byte
MQINT32	4 Byte
MQUINT32	4 Byte
MQINT64	8 Byte
MQUINT64	8 Byte

### AIX, Linux, and Windows üzerinde standart veri tipleri

32 bit AIX and Linux ve 64 bit AIX, Linux, and Windows uygulamalarındaki standart veri tipleri hakkında bilgi edinin.

### 32 bit AIX and Linux uygulamaları

*Çizelge 677. 32 bit AIX and Linux uygulamaları için veri tipi adları ve uzunlukları*

Ad	Uzunluk
DAMGA	1 Byte
kısa	2 Byte
int	4 Byte
uzun	4 Byte
kayan nokta	4 Byte
çift	8 Byte
uzun çift	8 Byte
iřaretçi	4 Byte
ptrdiff_t	4 Byte
size_t	4 Byte
tima_t	4 Byte

Çizelge 677. 32 bit AIX and Linux uygulamaları için veri tipi adları ve uzunlukları (devamı var)

Ad	Uzunluk
saat_t	4 Byte
wchar_t	4 Byte

**AIX** AIX a wchar\_t için 2 byte olduğunu unutmayın.

## 64 bit AIX and Linux uygulamaları

**Linux** **AIX**

Çizelge 678. 64 bit AIX and Linux uygulamaları için veri tipi adları ve uzunlukları

Ad	Uzunluk
DAMGA	1 Byte
kısa	2 Byte
int	4 Byte
uzun	8 Byte
kayan nokta	4 Byte
çift	8 Byte
uzun çift	8 Byte
işaretçi	8 Byte
ptrdiff_t	8 Byte
size_t	8 Byte
tima_t	8 Byte
saat_t	4 Byte
wchar_t	4 Byte

**AIX** AIX a wchar\_t için 2 byte olduğunu unutmayın.

## Windows 64 bit uygulamaları

**Windows**

Çizelge 679. Windows 64 bit uygulamaları için veri tipi adları ve uzunlukları

Ad	Uzunluk
DAMGA	1 Byte
kısa	2 Byte
int	4 Byte
uzun	4 Byte
kayan nokta	4 Byte
çift	8 Byte
uzun çift	8 Byte

Ad	Uzunluk
işaretçi	8 Byte Tüm işaretçilerin 8 bayt olduğunu unutmayın.
ptrdiff_t	8 Byte
size_t	8 Byte
tima_t	8 Byte
saat_t	4 Byte
wchar_t	2 Byte
Word	2 Byte
SÖZCÜK	4 Byte
Tnt	8 Byte
DOSYA	4 Byte

## Windowsüzerinde kodlama konuları

### Windows

#### HF;

Kullan

```
hf = CreateFile((LPCTSTR) FileName,  
               Access,  
               ShareMode,  
               xihSecAttsNTRestrict,  
               Create,  
               AttrAndFlags,  
               NULL);
```

Kullanma

```
HFILE hf;  
hf = (HFILE) CreateFile((LPCTSTR) FileName,  
                       Access,  
                       ShareMode,  
                       xihSecAttsNTRestrict,  
                       Create,  
                       AttrAndFlags,  
                       NULL);
```

bu şekilde bir hata ortaya çıkarması gerekir.

#### size\_t len fgets

Kullan

```
size_t len  
while (fgets(string1, (int) len, fp) != NULL)  
len = strlen(buffer);
```

Kullanma

```
int len;  
  
while (fgets(string1, len, fp) != NULL)  
len = strlen(buffer);
```

## printf

Kullan

```
printf("My struc pointer: %p", pMyStruc);
```

Kullanma

```
printf("My struc pointer: %x", pMyStruc);
```

Onaltılı çıkışa gereksinim duyarsanız, üst ve alt 4 byte 'ı ayrı olarak yazdırmalı.

## char \* ptr

Kullan

```
char * ptr1;  
char * ptr2;  
size_t buflen;  
  
buflen = ptr2 - ptr1;
```

Kullanma

```
char *ptr1;  
char *ptr2;  
UINT32 buflen;  
  
buflen = ptr2 - ptr1;
```

## alignBytes

Kullan

```
alignBytes = (unsigned short) ((size_t) address % 16);
```

Kullanma

```
void *address;  
unsigned short alignBytes;  
  
alignBytes = (unsigned short) ((UINT32) address % 16);
```

## UZUNLUK

Kullan

```
len = (UINT32) ((char *) address2 - (char *) address1);
```

Kullanma

```
void *address1;  
void *address2;  
UINT32 len;  
  
len = (UINT32) ((char *) address2 - (char *) address1);
```

## scanf

Kullan

```
MQLONG SBCSprt;
```

```
sscanf(line, "%d", &SBCSprt);
```

Kullanma

```
MQLONG SBCSprt;  
sscanf(line, "%1d", &SBCSprt);
```

%1d , 4 baytlık bir tip için 8 byte 'lık bir tip yerleştirmeyi dener; yalnızca gerçek bir long veri tipi ile ilgileniyorsanız %1 kullanın. MQLONG, UINT32 ve INT32 , tüm IBM MQ altyapılarında bir int ile aynı olmak üzere dört byte olarak tanımlanmalıdır:

IBM i

## IBM i Application Programming Reference (ILE/RPG)

IBM için uygulama programlama.

IBM için uygulama geliştirmenize yardımcı olması için bu bilgileri kullanın.

- [“IBM üzerindeki veri tipi açıklamaları” sayfa 981](#)
- [“Function calls on IBM i” sayfa 1230](#)
- [“IBM üzerindeki nesnelerin öznitelikleri” sayfa 1345](#)
- [“Uygulamalar” sayfa 1390](#)
- [“IBM i için dönüş kodları \(ILE RPG\)” sayfa 1403](#)
- [“IBM i için MQI seçeneklerinin geçerliliğini denetleme kuralları \(ILE RPG\)” sayfa 1404](#)
- [“Machine encodings on IBM i” sayfa 1407](#)
- [“IBM i' ta rapor seçenekleri ve ileti işaretleri” sayfa 1410](#)

### IBM üzerinde RPG ve COBOL uygulamaları için uyumluluk kipinin kullanımdan kaldırılması

IBM i

IBM MQ for IBM i 9.0' tan ürün artık, uyumluluk kipi olarak bilinen dinamik bağlantıyı kullanan RPG ya da COBOL uygulamaları için destek sağlamaz. MQSeries 5.1 öncesinde yazılan uygulamalar için bu işlem kipi gerekiyordu ve ürünün sonraki sürümleri bu uygulamalar için uyumlu bir çalıştırma zamanı ortamı sağlamış olsa da, bunları derlemek için gereken kopya kitaplar IBM WebSphere MQ 6.0' ta kaldırılmıştı. Dinamik bağlantı (uyumluluk kipi), aşağıdaki programlar tarafından QMQM kitaplığında yer alan ve IBM MQ for IBM i 9.0' da kaldırılmış olan programlar tarafından sağlanmıştır:

- AMBVSTUB
- AMQZKALINTI
- QMQM
- MQCLOSE
- MQCONN
- MQDISC
- MQGet
- MQINQ
- MQOPEN
- MQPUT
- MQPUT1
- MQSET

IBM MQ for IBM i 9.0' tan, bu uyumluluk kipini kullanan uygulamaların, LIBMQM ve LIBMQM\_R hizmet programları tarafından sağlanan statik bağlı MQ çağrılarını kullanmak için yeniden derlenmeleri gerekir.



AMQ3PUT4 ve AMQ3GET4 gibi örnek programlar, bu programlama modelini nasıl kullanacağını gösterir. Bu MQ çağrılarını kullanma hakkında daha fazla bilgi için bkz. [IBM i Application Programming Reference \(ILE/RPG\)](#).

#### Notlar:

- Şu anda CALL 'QMOM' arabirimini kullanarak uygulamaları yeniden kodlamak için LIBMQM hizmet programını kullanmanız gerekir.

Önceki listede yer alan program nesnelere ve hizmet programları; örneğin, QMOM, MQCONN, MQPUT, AMQVSTUB ve AMQZPRICI, IBM MQ for IBM i 9.0' ta kaldırılır ve uyumluluk kipini kullanmak için kodlanan uygulamalar çalışmaya son vermektedir.

- If applications are bound to the LIBMQM service program at IBM MQ for IBM i 8.0, you should not need to recompile or relink those applications at IBM MQ for IBM i 9.0 or later.
- Aynı bölüme IBM MQ for IBM i ' in birden çok sürümü kurulması mümkün değildir.

RPG ya da COBOL programınızın uyumluluk kipini kullanıp kullanmadığını öğrenmek için, uygulama programı tarafından çağrılan dış programları görüntülemek için **DSPPGMREF** (Program Başvurularını Görüntüle) komutunu kullanın. Bu bölümde listelenen programlara başvurular varsa, program IBM MQ for IBM i 9.0 ya da sonraki bir yayın düzeyinde çalıştırılmaz. Aşağıdaki **DSPPGMREF** çıkışı örneği, kullanımdan kaldırılmış üç program nesnesini gösterir: MQCONN, MQOPEN, MQCLOSE:

```
Program . . . . . : MYAPPPGM
Library . . . . . : MYLIB
Text 'description'. . . . . : ILE/COBOL SAMPLE PUT TO QUEUE (MQPUT)
Number of objects referenced . . . . . : 5
Object . . . . . : MQCONN
Library . . . . . : *LIBL
Object type . . . . . : *PGM
Object . . . . . : MQOPEN
Library . . . . . : *LIBL
Object type . . . . . : *PGM
Object . . . . . : MQCLOSE
Library . . . . . : *LIBL
Object type . . . . . : *PGM
```

Bu tür programlar, [Preparing COBOL programs in IBM i](#) içinde açıklanan Bound Procedural Call yöntemi kullanılarak yeniden derlenmelidir.

Uyumluluk kipi kullanan IBM MQ for IBM i 9.0 ya da sonraki bir yayın düzeyinde bir uygulama programı çalıştırma girişiminde bulunursanız, en sık görülen ilk hata, program MQCONN ya da QMOM ' yi çağrılmaya çalışılan bir MCH3401 hatasıdır.

#### İlgili görevler

[Uygulamaların geliştirilmesi](#)

## IBM i IBM üzerindeki veri tipi açıklamaları

Bu konu derlemi, IBM i programcısında kullanılan veri tiplerinin tanımlarını içerir.

### Veri tipleri tanımında kullanılan kurallar

Her temel veri tipi için bu bilgiler, programlama dilinden bağımsız bir biçimde, kullanımının bir tanımını verir. Bu, RPG programlama dilinin ILE sürümünde tipik bildirimlerden sonra gelir. Tutarlılığı sağlamak için temel veri tiplerinin tanımları burada yer alır. RPG, çalışma alanlarının gereksinim duyduğunuz öznitelikleri kullanarak bildirilebileceği ' D' belirtimlerini kullanır. Ancak, alanın kullanıldığı hesaplama belirtimlerinde bunu yapabilirsiniz.

Temel veri tiplerini kullanmak için şunları yaratmanız gerekir:

- Tüm veri tiplerini içeren bir /COPY üyesi ya da
- Tüm veri tiplerini içeren bir dış veri yapısı (PF). Daha sonra, 'LIKE' öznitelikleriyle çalışma alanlarınızı, uygun veri tipi alanı olarak belirtmeniz gerekir.

İkinci seçeneğin yararları, tanımların diğer IBM nesnelere için 'FIELD REFERENCE FILE' olarak kullanılabilmesinin sağlandığı. Bir IBM MQ veri tipi tanımlaması değişirse, bu nesnelere yeniden oluşturmak nispeten basit bir maddedir.

## Temel veri tipleri

Bu bölümde açıklanan diğer tüm veri tipleri doğrudan bu temel veri tipleriyle ya da bu temel veri tiplerinin (diziler ya da yapılar) toplanmalarına eşittir.

<i>Çizelge 680. Temel veri tipleri</i>	
<b>Veri türü</b>	<b>Gösterim</b>
MQBOOL	10 basamaklı işaretli tamsayı
MQBYTE	1 baytlık alfasayısal alan
MQBYTE16	16 baytlık alfasayısal alan
MQBYTE24	24 baytlık alfasayısal alan
MQBYTE32	32 baytlık alfasayısal alan
MQBYTE64	64 baytlık alfasayısal alan
MQCHAR	1 baytlık alfasayısal alan
MQCHAR4	4 baytlık alfasayısal alan
MQCHAR8	8 baytlık alfasayısal alan
MQCHAR12	12 baytlık alfasayısal alan
MQCHAR16	16 baytlık alfasayısal alan
MQCHAR20	20 baytlık alfasayısal alan
MQCHAR28	28 baytlık alfasayısal alan
MQCHAR32	32 baytlık alfasayısal alan
MQCHAR48	48 baytlık alfasayısal alan
MQCHAR64	64 baytlık alfasayısal alan
MQCHAR128	128 baytlık alfasayısal alan
MQCHAR256	256 baytlık alfasayısal alan
MQFLOAT32	4 baytlık kayan noktalı sayı
MQFLOAT64	8 baytlık kayan noktalı sayı
MQHCONFIG	Yapılandırma tutamacı
MQHCONN	10 basamaklı işaretli tamsayı
MQHMSG	İletiyeye erişim veren ileti tanıtıcısı
MQHOBJ	10 basamaklı işaretli tamsayı
MQINT8	8 bitlik işaretli tamsayı
MQINT16	16 bitlik işaretli tamsayı
MQINT32	32 bitlik işaretli tamsayı
MQINT64	64 bit imzalı tamsayı
MQLONG	32 bitlik işaretli tamsayı

Çizelge 680. Temel veri tipleri (devamı var)

Veri türü	Gösterim
MQPID	İşlem tanıtıcısı
MQPTR	İşaretçi
MQTID	İş parçacığı tanıtıcısı
MQUINT8	8 bitlik imzalanmamış tamsayı
MQUINT16	16 bit işaretli tamsayı
MQUINT32	32 bitlik imzalanmamış tamsayı
MQUINT64	64 bit işaretli tamsayı
MQULONG	32 bitlik imzalanmamış tamsayı
PMQACH	MQACH tipinde bir veri yapısına ilişkin gösterge
PMQAIR	MQAIR tipinde bir veri yapısına ilişkin gösterge
PMQAXC	MQAXC tipinde bir veri yapısına ilişkin gösterge
PMAXP	MAXP tipinde bir veri yapısına işaret eden gösterge
PMQBMHO	MQBMHO tipinde bir veri yapısına ilişkin gösterge
PMQBO	MQBO tipinde bir veri yapısına ilişkin gösterge
PMQBOOL	MQBOOL tipi veriye işaretçi
PMQBYTE	MMQBATY tipinde verilere ilişkin gösterge
PMQBYTEN	MQBYTEN tipinde verilere ilişkin gösterge
PMQCBC	İmlecin MQCBC tipinde bir veri yapısı
PMQCBD	MQCBD tipinde bir veri yapısına ilişkin gösterge
PMQCHAR	MQCHAR tipinde bir veri yapısına ilişkin gösterge
PMQCHARV	MQCHARV tipinde bir veri yapısına işaret eden gösterge
PMQCHARN	MQCHARn tipinde veri işaretçisi
PMQCIH	MQCIH tipinde bir veri yapısına ilişkin gösterge
PMQCMHO	MQCMHO tipinde bir veri yapısına ilişkin gösterge
PMQCNO	MQCNO tipinde bir veri yapısına ilişkin gösterge
PMQCSP	MQCSP tipinde bir veri yapısına işaret eden gösterge
PMQCTLO	MQCTLO tipinde bir veri yapısına ilişkin gösterge
PMQDH	MQDH tipinde bir veri yapısına ilişkin gösterge
PMQDHO	MQDHO tipinde bir veri yapısına ilişkin gösterge
PMQDLH	MQDLH tipinde bir veri yapısına ilişkin gösterge
PMQDMHO	MQDMHO tipinde bir veri yapısına ilişkin gösterge
PMQDMPO	MQDMPO tipinde bir veri yapısına işaret eden gösterge

Çizelge 680. Temel veri tipleri (devamı var)

Veri türü	Gösterim
PMQEPH	MQEPH tipinde bir veri yapısına işaretçi
PMQFLOAT32	MQFLOAT32tipi veriye ilişkin gösterge
PMQFLOAT64	MQFLOAT64tipi veriye ilişkin gösterge
PMQFUNC	İşleve ilişkin gösterge
PMQGM0	MQGM0 tipinde bir veri yapısına ilişkin gösterge
PMQHCONFIG	MQHCONFIG tipli verilere g " sterge
PMQHCONN	MQHCONN tipinde veri işaretçisi
PMQHMSG	MQHMSG tipli verilere g " sterge
PMQH0BJ	MQH0BJ tipindeki verileri gösteren gösterge
PMQIL	MQIIH tipinde bir veri yapısına ilişkin gösterge
PMQIMPO	MQIMPO tipinde bir veri yapısına ilişkin gösterge
PMQINT8	MQINT8tipli verilere gösterge
PMQINT16	MQINT16tipli verilere gösterge
PMQINT32	MQINT32tipli verilere gösterge
PMQINT64	MQINT64tipli verilere gösterge
PMQUZA	İmlecini veri tipi MQUBE
PMQMD	MQMD tipinde bir veri yapısına işaret eden gösterge
PMQMDE	MQMDE tipinde bir veri yapısına ilişkin gösterge
PMQMD1	MQMD1tipinde bir veri yapısına ilişkin gösterge
PMQMD2	MQMD2tipinde bir veri yapısına işaret eden gösterge
PMQMHB0	MQMHB0 tipinde bir veri yapısına işaret eden gösterge
PMQOD	MQOD tipinde bir veri yapısına işaretçi
PMQOR	MQOR tipinde bir veri yapısına ilişkin gösterge
PMQPD	MQPD tipinde bir veri yapısına işaret eden gösterge
PMQPID	İşlem tanıtıcısı MQPID ' e ilişkin gösterge
PMQPMO	MQPMO tipinde bir veri yapısına ilişkin gösterge
PMQPTR	MQPTR tipinde veriye ilişkin gösterge
PMQRFH	MQRFH tipinde bir veri yapısına işaret eden gösterge
PMQRFH2	MQRFH2tipinde bir veri yapısı göstergesi
PMQRMH	MQRMH tipinde bir veri yapısına ilişkin gösterge
PMQRR	MQRR tipinde bir veri yapısına ilişkin gösterge
PMQSCO	MQSCO tipinde bir veri yapısına ilişkin gösterge

Çizelge 680. Temel veri tipleri (devamı var)

Veri türü	Gösterim
PMQSD	MQSD tipinde bir veri yapısına işaret eden gösterge
PMQSMPO	MQSMPO tipinde bir veri yapısına işaret eden gösterge
PMQSRO	MQSRO tipinde bir veri yapısına ilişkin gösterge
PMQSTS	MQSTS tipinde bir veri yapısına işaret eden gösterge
PMQTID	Bir iş parçacığı tanıtıcısı MQTID ' ye işaretçi
PMQTM	MQTM tipinde bir veri yapısına işaret eden gösterge
PMQTM2	MQTM2tipinde bir veri yapısına işaret eden gösterge
PMQUINT8	MQUINT8tipinde veri işaretçisi
PMQUINT16	MQUINT16tipli verilere gösterge
PMQUINT32	MQUINT32tipindeki verilere ilişkin gösterge
PMQUINT64	MQUINT64tipindeki verilere ilişkin gösterge
PMQULONG	MQULONG tipinde veriye ilişkin gösterge
PMQVOID	İşaretçi
PMQWIH	MQWIH tipinde bir veri yapısına ilişkin gösterge
PMQXQH	MQXQH tipinde bir veri yapısına ilişkin gösterge

### IBM i **IBM üzerinde MQBOOL**

MQBOOL veri tipi bir boole değeri temsil eder. 0 değeri false değerini gösterir. Başka bir değer, true değerini gösterir.

MQHOOL, MQHOT veri tipi için aynı şekilde hizalanmalıdır.

### IBM i **IBM üzerinde MQBYTE**

MQBYTE veri tipi, tek bir byte veri tipini gösterir.

Belirli bir yorum byte 'a konmaz; ikili bir sayı ya da karakter olarak değil, bit dizgisi olarak kabul edilir. Özel hizalama gerekmez.

Bir MQBbayt dizisi bazen, kuyruk yöneticisi tarafından tanınmayan bir doğa ile ana bellek alanını göstermek için kullanılır. Örneğin, alan, uygulama iletisi verileri ya da bir yapı içerebilir. Bu alanın sınır hizalaması, içinde bulunan verilerin nitesiyile uyumlu olmalıdır.

### IBM i **MQBYTEN (String of n bytes) on IBM i**

Her MQBYTEN veri tipi  $n$  byte dizesini temsil eder.

Burada  $n$  , aşağıdaki değerlerden birini alabilir:

- 16, 24, 32 ya da 64.

Her bayt, MQBYTE veri tipi tarafından tanımlanır. Özel hizalama gerekmez.

Dizgideki veriler dizginin tanımlı uzunluğundan daha kısaysa, dizgiyi doldurmak için veriler boş değerler ile doldurulmalıdır.

Kuyruk yöneticisi uygulamaya bayt dizgileri döndürdüğünde (örneğin, MQGET çağrısında), kuyruk yöneticisi her zaman boş değer içeren boş değerleri dizginin tanımlı uzunluğuna sahip olur.

Bayt dizilimi alanlarının uzunluklarını tanımlayan deşiymezler kullanılabilir.

### **IBM i** **IBM üzerinde MQCHAR (karakter)**

MQCHAR veri tipi tek bir karakteri temsil eder.

The coded character set identifier of the character is that of the queue manager (see the **CodedCharSetId** attribute in topic [CodedCharSetId](#) ). Özel hizalama gerekmez.

**Not:** MQGET, MQPUT ve MQPUT1 çağrılarında belirtilen uygulama iletisi verileri, MQCHAR veri tipi değil, MQBYTE veri tipi tarafından tanımlanıyor.

### **IBM i** **MQCHARn (String of n characters) on IBM i**

Her bir MQCHARn veri tipi, bir *n* karakteri dizisini gösterir.

Burada *n* , aşağıdaki değerlerden birini alabilir:

- 4, 8, 12, 16, 20, 28, 32, 48, 64, 128 ya da 256

Her karakter MQCHAR veri tipi tarafından tanımlanır. Özel hizalama gerekmez.

Dizideki veriler, dizilimin tanımlı uzunluğundan daha kısaysa, dizgiyi doldurmak için verilerin boşluklarla doldurulması gerekir. Bazı durumlarda, boş bir karakter, boşlukları doldurmak için boşluklara; boşluk karakteri ve izleyen karakterler boşluk olarak, dizginin tanımlı uzunluğuna kadar boşluk olarak kabul edilir. Arama ve veri tipi tanımlamalarında boş değer kullanılabileceği yerler.

Kuyruk yöneticisi uygulamaya karakter dizgileri döndürdüğünde (örneğin, MQGET çağrısında), kuyruk yöneticisi her zaman, dizginin tanımlı uzunluğuna boşluk olarak boşluk yapar; kuyruk yöneticisi, dizgiyi sınırlamak için boş değerli karakteri kullanmaz.

Karakter dizgisi alanlarının uzunluklarını tanımlayan sabitler vardır.

### **IBM i** **IBM üzerindeMQFLOAT32**

MQFLOAT32 veri tipi, standart IEEE kayar noktalı biçim kullanılarak gösterilen, 32 bitlik bir kayan noktalı sayıdır.

Bir MQFLOAT32 , 4 baytlık bir sınır üzerinde hizalanmalıdır.

### **IBM i** **IBM üzerindeMQFLOAT64**

MQFLOAT64 veri tipi, standart IEEE kayar noktalı biçim kullanılarak gösterilen, 64 bitlik kayan noktalı bir sayıdır.

Bir MQFLOAT64 , 8 baytlık bir sınır üzerinde hizalanmalıdır.

### **MQHCONFIG-yapılanış tanıtıcısı**

MQHCONFIG veri tipi, belirli bir kurulabilir hizmet için yapılandırılmakta olan bir yapılanış tanıtıcısını (yani, bileşen) gösterir. Bir yapılandırma tanıtıcısı, doğal sınırı üzerinde hizalanmalıdır.

**Not:** Uygulamalar, yalnızca eşitlik için bu tipteki deęişkenleri test etmelidir.

### **IBM i** **IBM üzerinde MQHCONN (Baęlantı tanıtıcısı)**

MQHCONN veri tipi bir baęlantı tanıtıcısını, yani belirli bir kuyruk yöneticisiyle olan baęlantıyı gösterir.

Bir baęlantı tanıtıcısı, doğal sınırı üzerinde hizalanmalıdır.

**Not:** Uygulamalar, yalnızca eşitlik için bu tipteki deęişkenleri test etmelidir.

### **IBM i** **IBM üzerinde MQHMSG (İleti tanıtıcısı)**

MQHMSG veri tipi, bir iletiye erişim veren bir ileti tanıtıcısını temsil eder.

Bir ileti tanıtıcısı, 8 byte 'lık bir sınırdır hizalanmalıdır.

**Not:** Uygulamalar, yalnızca eşitlik için bu tipteki değişkenleri test etmelidir.

### **IBM i IBM üzerinde MQHOBJ (Nesne tanıtıcısı)**

MQHOBJ veri tipi, bir nesneye erişim veren bir nesne tanıtıcısını gösterir.

Bir nesne tanıtıcısı, doğal sınırları üzerinde hizalanmalıdır.

**Not:** Uygulamalar, yalnızca eşitlik için bu tipteki değişkenleri test etmelidir.

### **IBM i IBM üzerinde MQINT8 (8 bit imzalı tamsayı)**

The MQINT8 data type is an 8-bit signed integer that can take any value in the range -128 to +127, unless otherwise restricted by the context.

### **IBM i IBM üzerinde MQINT16 (16 bit imzalı tamsayı)**

MQINT16 veri tipi, bağlam tarafından başka bir kısıtlama getirmediği sürece, -32 768+32 767 aralığında herhangi bir değeri alabilen 16 bitlik bir işaretli tamsayıdır.

Bir MQINT16 , 2 baytlık bir sınır üzerinde hizalanmalıdır.

### **IBM i IBM üzerinde MQINT32 (32 bitlik tamsayı)**

MQINT32 veri tipi 32 bit imzalı bir tamsayıdır.

Bu, MQHOT ' a eşdeğerdir.

### **IBM i IBM üzerinde MQINT64 (64 bitlik tamsayı)**

The MQINT64 data type is a 64-bit signed integer that can take any value in the range -9 223 372 036 854 775 808 through +9 223 372 036 854 775 807, unless otherwise restricted by the context.

COBOL için geçerli değerler, +999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 999 numaralı 999 999 999 999 numaralı 999 999 999 999 numaralı 999 999 999 999 numaralı 999 8 baytlık bir sınır üzerinde bir MQINT64 hizalanmalıdır.

### **IBM i IBM üzerinde MQlong (Uzun tamsayı)**

MQlong veri tipi, bağlam tarafından doğal olarak sınırlanmadıkça, -2 147 483 648 ile + 2 147 483 647 aralığındaki herhangi bir değeri alabilen 32 bitlik imzalı bir ikili tamsayıdır.

### **MQPID-süreç tanıtıcısı**

IBM MQ işlem tanıtıcısı.

Bu tanıtıcı, IBM MQ izleme ve FFST dökümlerinde kullanılan tanıtıcıdır, ancak işletim sistemi işlem tanıtıcısından farklı olabilir.

### **MQPTR-işaretçi**

MQPTR veri tipi, herhangi bir tipteki verilerin adresidir. Bir işaretçinin doğal sınırları üzerinde hizalanması gerekir; bu, IBM i' da 16 baytlık bir sınırdır.

Bazı programlama dilleri yazılmış işaretçileri destekler; MQI da bunları birkaç durumda kullanır.

### **MQTID-iş parçacığı tanıtıcısı**

MQ iş parçacığı tanıtıcısı.

Bu tanıtıcı, MQ izleme ve FFST dökümlerinde kullanılan tanıtıcıyla aynıdır, ancak işletim sistemi iş parçacığı tanıtıcısından farklı olabilir.

### **IBM i IBM üzerindeMQUINT8 (8 bit işaretli tamsayı)**

MQUINT8 veri tipi, bağlam tarafından başka bir kısıtlama getirmediği sürece, 0-+255 aralığında herhangi bir değeri alabilen, 8 bitlik imzalanmamış bir tamsayıdır.

### **MQUINT16 -16 bit işaretli tamsayı**

MQUINT16 veri tipi, bağlam tarafından aksi belirtilmediği sürece, 0 ile +65 535 aralığında herhangi bir değeri alabilen, 16 bitlik imzalanmamış bir tamsayıdır.

Bir MQUINT16 , 2 baytlık bir sınır üzerinde hizalanmalıdır.

### **IBM i IBM üzerindeMQUINT32 (32 bit işaretli tamsayı)**

MQUINT32 veri tipi, 32 bitlik imzalanmamış bir tamsayıdır. Bu, MQULONG ' un eşdeğeridir.

### **MQUINT64 -64 bit işaretli tamsayı**

The MQUINT64 data type is a 64-bit unsigned integer that can take any value in the range 0 through +18 446 744 073 709 551 615 unless otherwise restricted by the context.

COBOL için geçerli aralık, 0 ile +999 999 999 999 999 999 numaralı 999 arasında bir değer ile sınırlanır. Bir MQUINT64 , 8 baytlık bir sınır üzerinde hizalanmalıdır.

### **MQULONG-32 bit işaretli tamsayı**

MQULONG veri tipi, bağlam tarafından aksi belirtilmediği sürece, 0-+ 4 294 967 294 aralığında herhangi bir değeri alabilen, 32 bitlik imzalanmamış bir ikili tamsayıdır.

Bir MQULONG, 4 baytlık bir sınır üzerinde hizalanmalıdır.

### **PMQACH-MQACH tipinde bir veri yapısına ilişkin gösterge**

MQACH tipinde bir veri yapısına ilişkin gösterge.

### **PMQAIR-MQAIR tipinde bir veri yapısı göstergesi**

MQAIR tipinde bir veri yapısına işaret eden gösterge.

### **PMQAXC-MQAXC tipinde bir veri yapısı göstergesi**

MQAXC tipinde bir veri yapısına işaret eden gösterge.

### **PMQAXP-MQAXP tipinde bir veri yapısı göstergesi**

MQAXP tipinde bir veri yapısına ilişkin gösterge.

### **PMQBMHO-MQBMHO tipinde bir veri yapısı göstergesi**

MQBMHO tipinde bir veri yapısına ilişkin gösterge.

### **PMQBO-işaretçi, MQBO tipinde bir veri yapısına**

MQBO tipinde bir veri yapısına ilişkin gösterge.



**PMQBOOL-MQBOOL tipinde veri işaretçisi**

MQBOOL tipi veriye ilişkin gösterge.

MQBOOL tipi veriye ilişkin gösterge.

**PMQBYTE-g "nderici veri tipi olarak MQBYTE g" stergesi**

MQBYTE veri tipine ilişkin gösterge.

**PMQBYTEN-impact tipi MQBYTEN tipinde bir veri yapısı**

MQBYTEN tipinde bir veri yapısına ilişkin gösterge; n 8, 12, 16, 24, 32, 40, 48 ya da 128 olabilir.

**PMQCBC-MQCBC tipinde bir veri yapısı göstergesi.**

MQCBC tipinde bir veri yapısına işaret eden gösterge.

**PMQCBD-MQCBD tipinde bir veri yapısı göstergesi**

MQCBD tipinde bir veri yapısına ilişkin gösterge.

**PMQCHAR-işaretçinin MQCHAR tipinde veriler**

MQCHAR tipinde veriye ilişkin gösterge.

**PMQCHARV-tip MQCHARV tipinde bir veri yapısı göstergesi**

MQCHARV tipinde bir veri yapısına ilişkin gösterge.

**PMQCHARn-MQCHARn veri tipinin göstergesi**

MQCHARn veri tipine ilişkin gösterge. Burada n 4, 8, 12, 20, 28, 32, 64, 128, 256, 264 olabilir.

**PMQCIH-MQCIH tipinde bir veri yapısı göstergesi**

MQCIH tipinde bir veri yapısına ilişkin gösterge.

**PMQCMHO-MQCMHO tipinde bir veri yapısına ilişkin gösterge**

MQCMHO tipinde bir veri yapısına ilişkin gösterge.

**PMQCNO-işaretçi, MQCNO tipinde bir veri yapısı**

MQCNO tipinde bir veri yapısına işaret eden gösterge.

**PMQCSP-tipi MQCSP tipinde bir veri yapısı**

MQCSP tipinde bir veri yapısına ilişkin gösterge.

**PMQCTLO-MQCTLO tipinde bir veri yapısı göstergesi**

MQCTLO tipinde bir veri yapısına ilişkin gösterge.

***PMQDH-tip MQDH tipinde bir veri yapısına gösterge***

MQDH tipinde bir veri yapısına ilişkin gösterge.

***PMQDHO-işaretçi, MQDHO tipinde bir veri yapısına işaret eder.***

MQDHO tipinde bir veri yapısına ilişkin gösterge.

***PMQDLH-MQDLH tipinde bir veri yapısı göstergesi***

MQDLH tipinde bir veri yapısına ilişkin gösterge.

***PMQDMHO-MQDMHO tipinde bir veri yapısı göstergesi***

MQDMHO tipinde bir veri yapısına ilişkin gösterge.

***PMQDMPO-mqDMPO tipinde bir veri yapısına işaret***

MQDMPO tipinde bir veri yapısına ilişkin gösterge.

MQDMPO tipinde bir veri yapısına ilişkin gösterge.

***PMQEPH-MQEPH tipinde bir veri yapısı göstergesi***

MQEPH tipinde bir veri yapısına ilişkin gösterge.

***PMQFLOAT32 - pointer to data of type MQFLOAT32***

MQFLOAT32tipindeki verilere ilişkin gösterge.

***PMQFLOAT64 - pointer to data of type MQFLOAT64***

MQFLOAT64tipindeki verilere ilişkin gösterge.

***PMQFUNC-işlevi gösteren gösterge***

Bir işleve işaretçi.

***PMQGM0-tipi MQGM0 tipinde bir veri yapısı göstergesi***

MQGM0 tipinde bir veri yapısına ilişkin gösterge.

***PMQHCONFIG-göstergesi MQHCONFIG veri tipini gösterir***

MQHCONFIG veri tipine ilişkin gösterge.

***PMQHCONN-MQHCONN veri tipinin işaretçisi***

MQHCONN veri tipine ilişkin gösterge.

**PMQHMSG-MQHMSG veri tipine ilişkin gösterge.**

MQHMSG veri tipine ilişkin gösterge.

**PMQHOBJ-İşaretçiyi MQHOBJ tipinde verilere**

MQSMPO tipine ilişkin veri göstergesi.

**PMQIIH-MQIIH tipinde bir veri yapısı göstergesi**

MQIIH tipinde bir veri yapısına ilişkin gösterge.

**PMQIMPO-MQIMPO tipinde bir veri yapısı göstergesi**

MQIMPO tipinde bir veri yapısına ilişkin gösterge.

**PMQINT8 - pointer to data of type MQINT8**

MQINT8tipine ilişkin bir veri göstergesi.

**PMQINT16 - pointer to data of type MQINT16**

MQINT16tipine ilişkin bir veri göstergesi.

**IBM i PMQINT32 ( IBM üzerinde MQINT32tipinde veri verileri)**

PMQINT32 veri tipi, MQINT32tipindeki verilere ilişkin bir göstergedir. Bu, PMQlong ile eşdeğerdir.

**IBM i PMQINT64 ( IBM üzerinde MQINT64tipinde veri tipi)**

PMQINT64 veri tipi, MQINT64tipindeki verilere ilişkin bir göstergedir.

**PMQUZ-işaretçisi, MQUZE tipinde veriler için**

MQlong tipi veriye ilişkin gösterge.

**PMQMD-tip MQMD ' nin yapısına ilişkin gösterge**

MQMD tipinde bir yapıya ilişkin gösterge.

**PMQMDE-MQMDE tipinde bir veri yapısına gösterge**

MQMDE tipinde bir veri yapısına ilişkin gösterge.

**PMQMDE-tipi MQMDI tipinde bir veri yapısı**

MQMDI tipinde bir veri yapısına ilişkin gösterge.

**PMQMD2 - MQMD2tipinde bir veri yapısı göstergesi.**

MQMD2tipinde bir veri yapısına ilişkin gösterge

***PMQMHBO-MQMHBO tipinde bir veri yapısı göstergesi***  
MQMHBO tipinde bir veri yapısına işaret eden gösterge.

***PMQOD-MQOD tipinde bir veri yapısı göstergesi***  
MQOD tipinde bir veri yapısına ilişkin gösterge.

***PMQOR-MQOR tipinde bir veri yapısına ilişkin gösterge***  
MQOR tipinde bir veri yapısına ilişkin gösterge.

***PMQPD-tip MQPD ' nin veri yapısına ilişkin gösterge***  
MQPD tipinde bir veri yapısına ilişkin gösterge.

***PMQPID-İşlem tanıtıcısını gösteren gösterge***  
Bir işlem tanıtıcısının işaretçisi.

***PMQPMO-tipindeki bir MQPMO tipinde bir veri yapısı göstergesi***  
MQPMO tipinde bir veri yapısına ilişkin gösterge.

***PMQPTR-tipindeki verilerin MQPTR tipine ilişkin gösterge***  
MQPTR tipinde veriye ilişkin gösterge.

***PMQRFH-MQRFH tipinde bir veri yapısı göstergesi***  
MQRFH tipinde bir veri yapısına ilişkin gösterge.

***PMQRFH2 - MQRFH2tipinde bir veri yapısı göstergesi.***  
MQRFH2tipinde bir veri yapısına ilişkin gösterge.

***PMQRMH-MQRMH tipinde bir veri yapısı göstergesi***  
MQRMH tipinde bir veri yapısına ilişkin gösterge.

***PMQRR-MQRR tipinde bir veri yapısı göstergesi***  
MQRR tipinde bir veri yapısına ilişkin gösterge.

***MQSCO tipindeki bir veri yapısına PMQSCO göstergesi***  
MQSCO tipinde bir veri yapısına ilişkin gösterge.

**PMQSD-tip MQSD veri yapısına ilişkin gösterge**

MQSD tipinde bir veri yapısına ilişkin gösterge.

**PMQSMPO-işaretçi, MQSMPO tipinde bir veri yapısına işaret ediyor**

MQSMPO tipinde bir veri yapısına işaret eden gösterge.

**PMQSRO-tipindeki bir veri yapısına ilişkin gösterge: MQSRO**

MQSRO tipinde bir veri yapısına ilişkin gösterge.

**PMQSTS-MQSTS tipinde bir veri yapısına gösterge**

MQSTS tipinde bir veri yapısına ilişkin gösterge.

**PMQTID-işaretçi, MQTID tipinde bir veri yapısına işaret ediyor**

MQTID tipinde bir veri yapısına ilişkin gösterge.

**PMQTM-tip MQTM ' nin veri yapısına ilişkin gösterge**

MQTM tipinde bir veri yapısına ilişkin gösterge.

**PMQTM C2 - MQTM C2 tipinde bir veri yapısı göstergesi.**

MQTM C2 tipinde bir veri yapısına ilişkin gösterge.

**PMQUINT8 - pointer to data of type MQUINT8**

MQUINT8 tipine ilişkin bir veri göstergesi.

**PMQUINT16 - pointer to data of type MQUINT16**

MQUINT16 tipindeki verilere ilişkin gösterge.

**IBM i PMQUINT32 ( IBM üzerinde MQUINT32 tipindeki veriler için işaretçi)**

PMQUINT32 veri tipi, MQUINT32 tipindeki verilere ilişkin bir göstergedir. Bu, PMQULONG ' un eşdeğeridir.

**IBM i PMQUINT64 ( IBM üzerinde MQUINT64 tipindeki veriler için işaretçi)**

PMQUINT64 veri tipi, MQUINT64 tipindeki verilere ilişkin bir göstergedir.

**PMQULONG-tipindeki MQULONG verilerine ilişkin gösterge**

MQULONG tipinde verilere ilişkin gösterge.

**PMQVOID-işaretçi**

Bir işaretçi.

## **PMQWIH-MQWIH tipinde bir veri yapısı göstergesi**

MQWIH tipinde bir veri yapısına işaret eden gösterge.

## **PMQXQH-MQXQH tipinde bir veri yapısına gösterge**

MQXQH tipinde bir veri yapısına ilişkin gösterge.

## **Dil bilgileri**

Bu konu, RPG programlama dilindeki MQI ' yi kullanmanıza yardımcı olacak bilgileri içerir.

Bu dillerden bazıları şunlardır:

- “Kütüklerin KOPYASI” sayfa 994
- “Çağrılar” sayfa 996
- “Arama değiştirgeleri” sayfa 996
- “Yapılar” sayfa 996
- “Adlandırılmış Değişmezler” sayfa 997
- “MQI yordamları” sayfa 997
- “Göz önünde bulundurulması gerekenler” sayfa 997
- “Kesinleştirme denetimi” sayfa 997
- “Bağlı çağrılar kodlanması” sayfa 997
- “Notasyonla ilgili kurallar” sayfa 999

## **Kütüklerin KOPYASI**

İleti kuyruklama olanağını kullanan RPG uygulama programlarının yazılmasına yardımcı olması için çeşitli COPY dosyaları sağlanır. Üç COPY dosyası kümesi vardır:

- COPY files with names ending with the letter *G* are for use with programs that use static linkage. Bu dosyalar, “Yapılar” sayfa 996’de belirtilen kural dışı durumlarla başlatılır.
- COPY files with names ending with the letter *H* are for use with programs that use static linkage, but are **değil** initialized.
- COPY files with names ending with the letter *E* are for use with programs that use dynamic linkage. Bu dosyalar, “Yapılar” sayfa 996’de belirtilen kural dışı durumlarla başlatılır.

KOPYA dosyaları, QMQM kitaplığında QRPGLSRC içinde bulunur.

Her bir COPY dosyası kümesi için, adlandırılmış değişmezleri içeren iki dosya ve her bir yapı için bir dosya vardır. Kopya dosyaları Çizelge 681 sayfa 994’de özetlenir.

Çizelge 681. RPG COPY dosyaları			
Dosya adı (statik bağlantı, ilk kullanıma hazırlandı, CMQ* G)	Dosya adı (durağan bağ, kullanıma hazırlanmadı, CMQ* H)	Dosya adı (dinamik bağlantı, ilk kullanıma hazırlanmış, CMQ* R)	İçindekiler
CMQBOG	CMQBOH	-	Seçenekler yapısını başlat
CMQCDG	CMQCDH	CMQCDR	Kanal tanımlama yapısı
CMQCFBG	CMQCFBH	-	PCF bit süzgeç değiştirgesi

Çizelge 681. RPG COPY dosyaları (devamı var)

Dosya adı (statik bağlantı, ilk kullanıma hazırlandı, CMQ* G)	Dosya adı (durağan bağ, kullanıma hazırlanmadı, CMQ* H)	Dosya adı (dinamik bağlantı, ilk kullanıma hazırlanmış, CMQ* R)	İçindekiler
CMQCFG	-	-	PCF ve olaylarla ilgili sabitler
CMQCFBSG	CMQCFBSH	-	PCF byte dizgisi
CMQCFGRG	CMQCFGRH	-	PCF grup parametresi
CMQCFIFG	CMQCFIFH	-	PCF tamsayı süzgeç değiştirgesi
CMQCFHG	CMQCFHH	-	PCF üstbilgisi
CMQCFIGG	CMQCFILH	-	PCF tamsayı listesi değiştirgesi yapısı
CMQCFING	CMQCFINH	-	PCF tamsayı değiştirgesi yapısı
CMQCFSG	CMQCFSH	-	PCF dizgi süzgeç değiştirgesi
CMQCFSLG	CMQCFSLH	-	PCF dizgi listesi değiştirgesi yapısı
CMQCFSTG	CMQCFSTH	-	PCF dizgi değiştirgesi yapısı
CMQCFXLG	CMQCFXLH	-	CFIL64için PCF kısa adı
CMQCFXNG	CMQCFXNH	-	CFIN64için PCF kısa adı
CMQCIHG	CMQCIHH	-	CICS bilgi üstbilgisi yapısı
CMQCNQG	CMQCNQH	-	Bağlantı seçenekleri yapısı
CMQCSPG	CMQCSPH	-	Güvenlik değiştirgeleri
CMQCXPG	CMQCXPH	CMQCXPR	Kanal çıkışı parametre yapısı
CMQDHG	CMQDHH	CMQDHR	Dağıtım üstbilgisi yapısı
CMQDLHG	CMQDLHH	CMQDLHR	Ölü harf üstbilgisi yapısı
CMQDXPG	CMQDXPH	CMQDXPR	Veri dönüştürme çıkış parametresi yapısı
CMQEPHG	CMQEPHH	-	Gömülü PCF üstbilgi yapısı
CMQG	-	CMQR	Ana MQI için adlandırılan sabitler
CMQGMQG	CMQGMQH	CMQGMOR	İleti seçenekleri yapısını al
CMQIHG	İHH ' dan CNIR	CMQIHR	IMS bilgi üstbilgisi yapısı
CMQMDEG	CMQMDEH	CMQAGER	İleti tanımlayıcı uzantısı yapısı
CMQMDG	CMQMDH	CMQMDR	İleti tanımlayıcı yapısı
CMQMD1G	CMQMD1H	CMQMD1R	İleti tanımlayıcı yapısı sürüm 1
CMQMD2G	CMQMD2H	-	İleti tanımlayıcı yapısı sürüm 2
CMQODG	CMQODH	CMQODR	Nesne tanımlayıcı yapısı
CMQORG	CMQORH	CMQORR	Nesne kaydı yapısı
CMQPMQG	CMQPMQH	CMQPMOR	İleti seçenekleri yapısını koy
CMQPSG	-	-	Yayınlama/abone olma değişmezleri

Çizelge 681. RPG COPY dosyaları (devamı var)

Dosya adı (statik bağlantı, ilk kullanıma hazırlandı, CMQ* G)	Dosya adı (durağan bağ, kullanıma hazırlanmadı, CMQ* H)	Dosya adı (dinamik bağlantı, ilk kullanıma hazırlanmış, CMQ* R)	İçindekiler
CMQRFHG	CMQRFHH	-	Kurallar ve biçimlendirme üstbilgi yapısı
CMQRFH2G	CMQRFH2H	-	Kurallar ve biçimlendirme üstbilgi 2 yapısı
CMQRMHG	CMQRMHH	CMQRMHR	Başvuru iletisi üstbilgi yapısı
CMQRRG	CMQRRH	CMQRRR	Yanıt kaydı yapısı
CMQTMCG	CMQTMCH	CMQTMCR	İleti yapısını tetikle (karakter biçimi)
CMQTMCG2G	CMQTMCG2H	CMQTMCG2R	İleti yapısını tetikle (karakter biçimi) sürüm 2
CMQTMG	CMQTMH	CMQTMR	İleti yapısını tetikle
CMQWIHG	CMQWIHH	-	İş bilgileri üstbilgi yapısı
CMQXG	-	CMQXR	Veri dönüştürme çıkışı için adlandırılan sabitler
CMQXQHG	CMQXQHH	CMQXQR	İletim kuyruğu üstbilgi yapısı

## Çağrılar

Çağrılar, tek tek adları kullanılarak tanımlanır.

## Arama değiştirgeleri

MQI ' ye geçirilen bazı değiştirgeler birden çok koşutzamanlı işleve sahip olabilir. Bunun nedeni, geçirilen tamsayı değerinin çoğu zaman alandaki tek tek bit ayarlarında sınanması ve toplam değerinin üzerinde olmamasıdır. Bu, birden çok işlevi bir araya getirmenizi ve tek bir parametre olarak geçirmenizi sağlar.

## Yapılar

Tüm IBM MQ yapıları, aşağıdaki kural dışı durumlar dışında, alanlar için başlangıç değerleriyle tanımlanır:

- H. soneki olan herhangi bir yapı
- MQTMC
- MQTMC2

Bu başlangıç değerleri, her bir yapı için ilgili çizelgede tanımlanır.

Yapı bildirimlerinde DS deyimi yoktur. This allows the application to declare either a single data structure or a multiple-occurrence data structure, by coding the DS statement and then using the /KOPYALA statement to copy in the remainder of the declaration:

```
D*..1.....2.....3.....4.....5.....6.....7
D* Declare an MQMD data structure with 5 occurrences
DMYMD      DS              5
D/COPY CMQMDR
```



## Adlandırılmış Değişmezler

Uygulama programınız ile kuyruk yöneticisi arasında veri değiş tokuşu sağlayan çok sayıda tamsayı ve karakter değeri vardır. Bu değerlerin kullanılması için daha okunabilir ve tutarlı bir yaklaşım kullanılmasını kolaylaştırmak için, bunlar için sabit değerler tanımlıdır. Program kaynak kodunun okunabilirliğini iyileştirdiği için, bu adlandırılan değişmezleri temsil ettikleri değerleri değil de kullanabilirsiniz.

Değişmezlerin tanımlanması için bir programa COPY (CMQG) dosyası eklendiğinde, RPG derleyicisi, program tarafından kullanılmayan sabitler için birçok önem düzeyi sıfır iletisi yayınlayacaktır; bu iletiler iyi huylu ve güvenli bir şekilde yoksayılabilir.

## MQI yordamları

ILE bağlı çağrılarını kullanırken, programınızı yaratırken MQI yordamlarına bağlanmanız gerekir. Bu yordamlar, aşağıdaki hizmet programlarından uygun şekilde ihraç edilir:

### QMQM/LIBMQM

Bu hizmet programı, sürüm 5.1 ve üstü için tek iş parçacıklı bağ tanımlarını içerir. İş parçacıklı uygulamalar yazılırken dikkat edilmesi gereken özel noktalar için aşağıdaki bölüme bakın.

### QMQM/LIBMQM\_R

Bu hizmet programı, sürüm 5.1 ve üstü için çok iş parçacıklı bağ tanımlarını içerir. İş parçacıklı uygulamalar yazılırken dikkat edilmesi gereken özel noktalar için aşağıdaki bölüme bakın.

### QMQM/LIBMQIC

Bu hizmet programı, iş parçacıklı olmayan istemci uygulamaları için bağ tanımlıdır.

### QMQM/LIBMQIC\_R

Bu hizmet programı, iş parçacıklı istemci uygulamaları için bağ tanımlıdır.

Programlarınızı yaratmak için CRTPGM komutunu kullanın. Örneğin, aşağıdaki komut, ILE bağlı çağrılarını kullanan tek iş parçacıklı bir program yaratır:

```
CRTPGM PGM(MYPROGRAM) BNDSRVPGM(QMQM/LIBMQM)
```

## Göz önünde bulundurulması gerekenler

IBM i için kullanılan RPG derleyicisi, WebSphere Development Toolset and WebSphere Development Studio for IBM i ' un bir parçasıdır ve ILE RPG IV Compiler olarak bilinir.

Genel olarak, RPG programları çok iş parçacıklı hizmet programlarını kullanmamalıdır. Kural dışı durumlar, ILE RPG IV Compiler kullanılarak yaratılan RPG programlarıdır ve denetim belirtiminde THREAD(\*SERIALIZE) anahtar sözcüğünü içerir. Ancak, bu programlar iş parçacığı güvenliği olsa da, genel uygulama tasarımına dikkatli bir şekilde dikkat edilmelidir; THREAD(\*SERIALIZE) , birim düzeyinde RPG yordamlarıyla diziselleştirilirken, bu durum genel performansa olumsuz etkilenebilir.

Where RPG programs are used as data-conversion exits, they must be made thread-safe, and should be recompiled using the version 4.4 ILE RPG compiler or above, with THREAD(\*SERIALIZE) specified in the control specification.

Threading ile ilgili daha fazla bilgi için *IBM i IBM MQ Development Studio: ILE RPG Reference* ve *IBM i IBM MQ Development Studio: ILE RPG Programmer's Guide* adlı yayına bakın.

## Kesinleştirme denetimi

MQI syncpoint işlevleri MQCMIT ve MQBACK işlevleri, olağan kipte çalışan ILE RPG programları için kullanılabilir; bu çağrılar, programın değişiklikleri MQ kaynaklarına kesinleştirmesine ve geri göndermesine olanak sağlar.

## Bağlı çağrılarının kodlanması

MQI ILE yordamları [Çizelge 682 sayfa 998](#) listesinde yer alıyor.

Çizelge 682. Her hizmet programı tarafından desteklenen ILE RPG bağlı çağruları

Arama adı	LIBMQM ve LIBMQM_R	LIBMQIC ve LIBMQIC_R
MQBACK	Y	Y
MQBEGIN	Y	Y
MQCMIT	Y	Y
MQCLOSE	Y	Y
MQCONN	Y	Y
MQCONNX	Y	Y
MQDISC	Y	Y
MQGet	Y	Y
MQINQ	Y	Y
MQOPEN	Y	Y
MQPUT	Y	Y
MQPUT1	Y	Y
MQSET	Y	Y
MQXCNVC	Y	Y

Bu yordamları kullanmak için aşağıdaki işlemleri yapmak gerekir:

1. 'D' belirtilerinizde dış yordamları tanımlayın. Bunların tümü, adı belirtilen değişmezleri içeren COPY dosya üyesi CMQG içinde kullanılabilir.
2. Yordamlarıyla birlikte yordamı çağırarak için CALLP işlem kodunu kullanın.

Örneğin, MQOPED çağırısı aşağıdaki kodu içermesini gerektirir:

```

D*****
D** MQOPEN Call -- Open Object (From COPY file CMQG) **
D*****
D*
D*.1....:....2....:....3....:....4....:....5....:....6....:....7..
DMQOPEN PR EXTPROC('MQOPEN')
D* Connection handle
D HCONN 10I 0 VALUE
D* Object descriptor
D OBJDSC 224A
D* Options that control the action of MQOPEN
D OPTS 10I 0 VALUE
D* Object handle
D HOBJ 10I 0
D* Completion code
D CMPCOD 10I 0
D* Reason code qualifying CMPCOD
D REASON 10I 0
D*

```

Yordamı çağırarak için, çeşitli deęiřtirgeleri kullanıma hazırladıktan sonra aşağıdaki kodu gerekir:

```

...+... 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7 ...+... 8
C CALLP MQOPEN(HCONN : MQOD : OPTS : HOBJ :
C CMPCOD : REASON)

```

Burada, MQOD yapısı, COPY üyesi CMQODG kullanılarak tanımlanıyor ve bu da bileşenleri bileşenlerine ayırır.

## Notasyonla ilgili kurallar

Bu bölümde yer alan sonraki konular aşağıdakileri nasıl gösterir:

- Çağrılar çağrılmalıdır
- Parametreler bildirilmelidir
- Çeşitli veri tipleri bildirilmelidir

Bir dizi durumda, parametreler sabit olmayan dizilerdir ya da karakter dizgileridir. Bunlar için küçük harf "n" , sayısal bir sabiti göstermek için kullanılır. When the declaration for that parameter is coded, the "n" must be replaced by the numeric value required.

## IBM i IBM üzerinde MQAIR (Kimlik doğrulama bilgileri kaydı)

MQAIR yapısı, kimlik doğrulama bilgileri kaydını temsil eder.

### Genel Bakış

**Amaç:** MQAIR yapısı, bir uygulamanın IBM MQ istemcisi olarak çalışan bir uygulamanın istemci bağlantısı için kullanılacak kimlik doğrulayıcıya ilişkin bilgileri belirtmesini sağlar. Yapı, MQCONNX çağrısına ilişkin bir giriş değiştirir.

**Karakter kümesi ve kodlama:** MQAIR ' deki veriler, ENNAT tarafından verilen yerel kuyruk yöneticisinin **CodedCharSetId** kuyruk yöneticisi özniteliği ve kodlaması tarafından belirtilen karakter kümesinde olmalıdır.

- [“Alanlar” sayfa 999](#)
- [“Başlangıçtaki değerler” sayfa 1001](#)
- [“RPG bildirim” sayfa 1001](#)

### Alanlar

MQAIR yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

#### ATICN (10 basamaklı işaretli tamsayı)

Bu ad, LDAP sunucusunun çalışmakta olduğu anasistemin adı ya da ağ adresi. Bunu, parantez içine alınmış isteğe bağlı bir kapı numarası takip edebilir.

Değer, alanın uzunluğundan kısaysa, değeri boş değerli bir karakterle sonlandırın ya da alanın uzunluğuna göre boşluklarla doldurur. Değer geçerli değilse, arama başarısız olur ve neden kodu RC2387 ile başarısız olur.

Varsayılan kapı numarası 389'dur.

Bu bir giriş alanıdır. Bu alanın uzunluğu LNAICN tarafından verilir. Bu alanın ilk değeri boşluk karakteridir.

#### AITYP (10 basamaklı işaretli tamsayı)

Bu, kayıta bulunan kimlik doğrulama bilgilerinin tipidir.

Değer şu olmalıdır:

##### AITLP

LDAP sunucusu kullanılarak sertifika iptal işlemi.

Değer geçerli değilse, çağrı neden kodu RC2386 ile başarısız olur.

Bu bir giriş alanıdır. Bu alanın ilk değeri AITLDP 'dir.

#### AIPW (10 basamaklı işaretli tamsayı)

Bu, LDAP CRL sunucusuna erişmek için gerekli olan paroladır.

Değer, alanın uzunluğundan kısaysa, değeri boş değerli bir karakterle sonlandırın ya da alanın uzunluğuna göre boşluklarla doldurur. LDAP sunucusu parola gerektirmiyorsa ya da LDAP kullanıcı adını atarsanız, *AIPW* boş değerli ya da boş olmalıdır. LDAP kullanıcı adını atarsanız ve *AIPW* boş değilse ya da boş değilse, arama başarısız olur ve neden kodu RC2390 olur.

Bu bir giriş alanıdır. Bu alanın uzunluğu LNLDPW tarafından verilir. Bu alan boş karakterlerinin ilk değeri.

#### **AILUL (imzalanmış 10 basamaklı tamsayı)**

Bu, *AILUP* ya da *AILUO* alanı tarafından adreslenen LDAP kullanıcı adının bayt cinsinden uzunluğudur. Değer, LNDISN ile sıfır aralığında olmalıdır. Değer geçerli değilse, arama işlemi neden kodu RC2389 ile başarısız olur.

İlgili LDAP sunucusu için bir kullanıcı adı gerekmiyorsa, bu alanı sıfır olarak ayarlayın.

Bu bir giriş alanıdır. Bu alanın ilk değeri 0 'tır.

#### **AILUO (10 basamaklı işaretli tamsayı)**

Bu, MQAIR yapısının başlangıcındaki LDAP kullanıcı adının bayt cinsinden görelidir.

Görelidir pozitif ya da negatif olabilir. *LDAPUserNameLength* sıfırsa, alan yoksayılır.

LDAP kullanıcı adını belirtmek için *LDAPUserNamePtr* ya da *LDAPUserNameOffset* seçeneğini kullanabilirsiniz, ancak her ikisi de değil; ayrıntılar için *LDAPUserNamePtr* alanının açıklamasına bakın.

Bu bir giriş alanıdır. Bu alanın ilk değeri 0 'tır.

#### **AILUP (10 basamaklı işaretli tamsayı)**

Bu, LDAP kullanıcı adıdır.

Bu, LDAP CRL sunucusuna erişmeyi deneyen kullanıcının Ayırt Edici Adı ile oluşur. If the value is shorter than the length specified by *AILUL*, terminate the value with a null character, or pad it with blanks to the length *AILUL*. *AILUL* sıfırsa, alan yoksayılır.

LDAP kullanıcı adını aşağıdaki iki yoldan birini kullanarak temin edebilirsiniz:

- *AILUP* işaretçi alanını kullanarak

Bu durumda uygulama, MQAIR yapısından ayrı bir dizgi bildirebilir ve *AILUP*, dizginin adresine ayarlıdır.

Farklı ortamlara (örneğin, C programlama dili gibi) taşınabilir bir şekilde işaretçi veri tipini destekleyen programlama dilleri için *AILUP* 'i kullanmayı düşünün.

- By using the offset field *AILUO*

Bu durumda, uygulamanın MQSCO yapısını içeren bir bileşik yapı bildirmesi gerekir; bunu izleyen MQAIR kayıtları dizisi ve ardından LDAP kullanıcı adı dizgileri gelir ve *AILUO* olarak, MQAIR yapısının başlangıcındaki uygun ad dizgisinin görelidir konuta ayarlanır. Bu değer doğru olduğundan ve bir MQLong (en kısıtlayıcı programlama dili COBOL) içinde konaklayabilecek bir değere sahip olduğundan emin olun (geçerli aralık -999 999 999 ile +999 999 999 aralığında).

İşaretçi veri tipini desteklemeyen programlama dilleri için *AILUO*, ya da işaretçi veri tipini farklı ortamlara (örneğin, COBOL programlama dili gibi) portatif olmayabilecek bir şekilde uygulamayı düşünün.

Hangi teknik seçiliyorsa, *AILUP* ve *AILUO* 'den yalnızca birini kullanın; Çağrı, RC2388 neden koduyla başarısız olur.

Bu bir giriş alanıdır. Bu alanın ilk değeri, göstergeleri destekleyen programlama dillerindeki boş değerli gösterge ve tersi durumda, boş değerli byte dizilimidir.

**Not:** Programlama dilinin işaretçi veri tipini desteklemediği altyapılarda, bu alan uygun uzunluğun byte dizgisi olarak bildirilir.

## AISID (10 basamaklı işaretli tamsayı)

Değer şu olmalıdır:

### AISIDV

Kimlik doğrulama bilgileri kaydına ilişkin tanıttıcı.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri AISIDV ' dir.

## AIVER (10 basamaklı işaretli tamsayı)

Değer şu olmalıdır:

### AIVER1

Version-1 kimlik doğrulama bilgileri kaydı.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

### AIVERC

Kimlik doğrulama bilgileri kaydının geçerli sürümü.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri AIVER1' dir.

## Başlangıçtaki değerler

Çizelge 683. MQAIR için MQAIR ' de alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
AISID	AISIDV	'AIR↵'
AIVER	AIVERC	1
AITYP	AITLP	1
AICN	Yok	Dizgi ya da boşluk boş değerli
AILUP	Yok	Boş değerli gösterge ya da boş değerli byte
AILUO	Yok	0
AILUL	Yok	0
AIPW	Yok	Dizgi ya da boşluk boş değerli

### Notlar:

1. ↵ simgesi tek bir boş karakteri temsil eder.

## RPG bildirimi

```
D*..1.....2.....3.....4.....5.....6.....7..
D* MQAIR Structure
D*
D* Structure identifier
D AISID 1 4 INZ('AIR ')
D* Structure version number
D AIVER 5 8I 0 INZ(1)
D* Type of authentication information
D AITYP 9 12I 0 INZ(1)
D* Connection name of CRL LDAP server
D AICN 13 276 INZ
D* Address of LDAP user name
D AILUP 277 292* INZ(*NULL)
D* Offset of LDAP user name from start of MQAIR structure
D AILUO 293 296I 0 INZ(0)
D* Length of LDAP user name
```

```
D AILUL          297    300I 0 INZ(0)
D* Password to access LDAP server
D AIPW          301    332    INZ
```

## IBM i IBM üzerinde MQBMHO (İleti işleme seçenekleri için arabellek)

Arabelleği ileti tutamacı seçeneklerine tanımlayan yapı.

### Genel Bakış

**Amaç:** MQBMHO yapısı, uygulamaların, ileti tutamaçlarının arabelleklerden nasıl üretildiğini denetleyen seçenekleri belirlemesine olanak sağlar. Yapı, MQBUFMH çağrısına ilişkin bir giriş değişikidir.

**Karakter kümesi ve kodlama:** MQBMHO içindeki veriler, uygulamanın (ENNAT) uygulamanın uygulama ve kodlamasının karakter kümesinde yer almalıdır.

- “Alanlar” sayfa 1002
- “Başlangıçtaki değerler” sayfa 1003
- “RPG bildirim” sayfa 1003

### Alanlar

MQBMHO yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

#### **BMSID (10 basamaklı işaretli tamsayı)**

Arabellek-ileti tanıtıcısı yapısı- StrucId alanı.

Bu, yapı tanıtıcısıdır. Değer şu olmalıdır:

#### **MMSIDV**

İleti tanıtıcısı yapısına arabellek için tanıtıcı.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri BMSIDV ' dir.

#### **BMVER (10 basamaklı işaretli tamsayı)**

Arabellek-ileti tanıtıcısı yapısı-Sürüm alanı.

Bu, yapı sürüm numarasıdır. Değer şu olmalıdır:

#### **BMVER1**

Arabellek için ileti tanıtıcısı yapısına ilişkin sürüm numarası.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

#### **BMVERVC**

Arabellek, ileti tanıtıcısı yapısına ilişkin yürürlükteki sürüm.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri BMVER1' dir.

#### **BMOPT (10 basamaklı işaretli tamsayı)**

Arabellek-ileti tanıtıcısı yapısı-Seçenekler alanı.

Değer şu şekilde olabilir:

#### **BMDLPR**

İleti tutamaçına eklenen özellikler arabellekten silinir. Arama başarısız olursa özellikler silinmez.

Varsayılan seçenekler: Açıklanmanız gerekmiyorsa, aşağıdaki seçeneği kullanın:

#### **BMYT**

Seçenek belirtilmedi.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri BMDLPR ' dir.

## Başlangıçtaki değerler

Çizelge 684. MQBMHO 'daki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
BMSID	MMSIDV	'BMHO'
BMVER	BMVER1	1
BMOPT	BYT	0

## RPG bildirim

```
D* MQBMHO Structure
D*
D*
D* Structure identifier
D BMSID 1 4 INZ('BMHO')
D*
D* Structure version number
D BMVER 5 8I 0 INZ(1)
D*
D* Options that control the action of MQBUFMH
D BMOPT 9 12I 0 INZ(1)
```

## IBM i MQBO (Begin options) on IBM i

MQBO yapısı, uygulamanın bir iş birimi yaratılmasıyla ilgili seçenekleri belirlemesine olanak sağlar.

## Genel Bakış

**Amaç:** Yapı, MQBEGIN çağrısındaki bir giriş/çıkış parametresidir.

**Karakter kümesi ve kodlama:** MQBO 'daki veriler, ENNAT tarafından verilen yerel kuyruk yöneticisinin **CodedCharSetId** kuyruk yöneticisi özniteliği ve kodlaması tarafından verilen karakter kümesinde olmalıdır.

- “Alanlar” sayfa 1003
- “Başlangıçtaki değerler” sayfa 1004
- “RPG bildirim” sayfa 1004

## Alanlar

MQBO yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

### BOOPT (10 basamaklı işaretli tamsayı)

MQBEGIN işlemini denetleyen seçenekler.

Değer şu olmalıdır:

#### BONONE

Seçenek belirtilmedi.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri BONONE 'tır.

### BOSID (4 baytlık karakter dizilimi)

Yapı tanıtıcısı.

Değer şu olmalıdır:

#### BOSDİS

Başlangıç seçenekleri yapısına ilişkin tanıtıcı.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri BOSIDV 'dir.

### OVER (10 basamaklı işaretli tamsayı)

Yapı sürüm numarası.

Değer şu olmalıdır:

#### BOVER1

Başlangıç seçenekleri yapısına ilişkin sürüm numarası.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

#### GENEL

Başlangıç seçenekleri yapısının yürürlükteki sürümü.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri BOVER1' dir.

### Başlangıçtaki değerler

Çizelge 685. MQBO 'daki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
BOSID	BOSDİS	'BO- -'
BOVER	BOVER1	1
BOOPT	BONONE	0

#### Notlar:

1. - simgesi tek bir boş karakteri temsil eder.

### RPG bildirim

```
D* .1.....2.....3.....4.....5.....6.....7..
D* MQBO Structure
D*
D* Structure identifier
D BOSID 1 4 INZ('BO ')
D* Structure version number
D BOVER 5 8I 0 INZ(1)
D* Options that control the action of MQBEGIN
D BOOPT 9 12I 0 INZ(0)
```

### IBM i MQCBC (Callback context) on IBM i

Geri çağırma yordamını tanımlayan yapı.

#### Genel Bakış

##### Amaç

MQCBC yapısı, geri bildirme işlevine geçirilen bağlam bilgilerini belirtmek için kullanılır.

Yapı, bir ileti tüketici yordamuna yapılan çağrıdaki bir giriş/çıkış parametresidir.

##### S\u00fcr\u00fcm

MQCBC 'nin yürürlükteki sürümü CBCV2' dir.

##### Karakter kümesi ve kodlama

Data in MQCBC is in the character set given by the **CodedCharSetId** queue manager attribute and encoding of the local queue manager given by ENNAT. Ancak, uygulama bir IBM MQ istemcisi olarak çalışıyorsa, yapı, istemcinin karakter kümesinde ve kodlamasında yer alıyor.

- [“Alanlar” sayfa 1005](#)



- “Başlangıçtaki değerler” sayfa 1010
- “RPG bildirim” sayfa 1010

## Alanlar

MQCBC yapısı aşağıdaki alanları içerir; alanlar alfabetik sırayla anlatılır:

### **CBCBUFFLEN (10 basamaklı işaretli tamsayı)**

Arabellek, tüketici için tanımlanan MaxMsgUzunluğu değerinden ve MQGMO 'daki ReturnedLength değerine göre daha büyük olabilir.

Geri bildirme bağlamı yapısı- BufferLength alanı.

Bu, bu işleve aktarılan ileti arabelleğindeki bayt cinsinden uzunluktur.

Gerçek ileti uzunluğu [DataLength](#) alanında sağlanır.

Uygulama, geri bildirme işlevi süresi boyunca tüm arabelleği kendi amaçları için kullanabilir.

Bu, ileti tüketicisi işlevine ilişkin bir giriş alanıdır; bir kural dışı durum işleyici işleviyle ilgili değildir.

### **CBCCALLBA (10 basamaklı işaretli tamsayı)**

Geri bildirme bağlamı yapısı- CallbackArea alanı.

Bu alan, geri bildirme işlevi için kullanılabilir bir alandır.

Kuyruk yöneticisi, bu alanın içeriğine dayalı olarak karar vermez ve geri bildirme işlevini tanımlamak için kullanılan MQCBB çağrısındaki bir parametre olan MQCBD yapısındaki [CBDCALLBA](#) alanından değiştirilmeden geçirilir.

[CBCCALLBA](#) üzerinde yapılan değişiklikler, bir [CBCHOB](#) için geri bildirme işlevine ilişkin çağrılar boyunca korunur. Bu alan, diğer çekme noktalarına ilişkin geri çağrı işlevleriyle paylaşılmaz.

Bu, geri bildirme işlevi için bir giriş/çıkış alanıdır. Bu alanın başlangıç değeri, boş değerli ya da boş değerli bayttır.

### **CBCCALLT (10 basamaklı işaretli tamsayı)**

Geri Çağırma Bağlamı yapısı- CallType alanı.

Bu işlevin neden çağrıldığı hakkında bilgi içeren alan. Aşağıdaki çağrı tipleri tanımlanır.

İleti teslim çağrısı tipleri: Bu çağrı tipleri bir iletiyle ilgili bilgi içerir. **CBCLLEN** ve **CBCBUFFLEN** parametreleri bu çağrı tipleri için geçerlidir.

#### **CCTMR**

İleti tüketici işlevi, nesne tanıtıcısından yok edici bir şekilde kaldırılmış bir iletiyle çağrıldı.

[CBCCC](#) değeri CCWARN ise, *Reason* alanının değeri RC2079 ya da bir veri dönüştürme sorununu belirten kodlardan birdir.

#### **CCTMN**

İleti tüketici işlevi, nesne tanıtıcısından henüz yok edici bir şekilde kaldırılmamış bir iletiyle çağrıldı. İleti, *MsgToken* komutunu kullanarak, nesne tanıtıcısından yok edici bir şekilde kaldırılabilir.

İleti şu nedenle kaldırılmamış olabilir:

- MQGMO seçenekleri için bir göz atma işlemi istendi, GMBR\*
- İleti, kullanılabilir arabelleğinden büyük ve MQGMO seçenekleri *gmtmbelirtmiyor*

[CBCCC](#) değeri CCWARN ise, *Reason* alanının değeri RC2080 ya da bir veri dönüştürme sorununu belirten kodlardan birdir.

Geri çağırma denetim çağrısı tipleri: Bu çağrı tipleri, geri bildirim denetime ilişkin bilgileri içerir ve bir iletiyle ilgili ayrıntıları içermez. Bu çağrı tipleri, MQCBD yapısındaki [CBDOPT](#) kullanılarak istenir.

**CBCLEN** ve **CBCBUFFLEN** parametreleri bu çağrı tipleri için geçerli değildir.

#### **CBCTRC**

Bu çağrı tipinin amacı, geri bildirme işlevinin bazı ilk ayarları gerçekleştirmesine izin vermesidir.

Geri bildirme işlevi, CBREG ' nin *Operation* alanı için bir değer kullanılarak MOCB çağrısından geri çağırıldıktan sonra, çağırıldığında çağrılır.

Bu arama tipi hem ileti tüketicileri hem de olay işleyicileri için kullanılır.

İstenirse, bu, geri çağırma işlevinin ilk çağrısıdır.

*CBCREA* alanının değeri RCNONE olur.

#### **CCTSC**

Bu çağrı tipinin amacı, geri bildirme işlevinin başlatıldığı zaman bazı ayarları gerçekleştirmesine izin vermesidir; örneğin, daha önce durdurulduğunda temizlenen kaynakları yeniden yürürlüğe almak.

Geri bildirme işlevi, bağlantı CTLSR ya da CTLSW kullanılarak başlatıldığında çağrılır.

Bir geri bildirme işlevi başka bir geri çağırma işlevinde kaydedilirse, bu çağrı tipi geri çağırma döndürdüğünde çağrılır.

Bu arama tipi yalnızca ileti kullanıcıları için kullanılır.

*CBCREA* alanının değeri RCNONE olur.

#### **CBCTTC**

Bu çağrı tipinin amacı, geri bildirme işlevinin bir süre durdurulduğunda temizleme işlemini gerçekleştirmesine izin vermesidir; örneğin, iletilerin tüketilmesi sırasında edinilen ek kaynakları temizlemesidir.

Geri bildirme işlevi, CTLSP *Operation* alanı için bir değer kullanılarak bir MQCTL çağrısı yayınlandığında çağrılır.

Bu arama tipi yalnızca ileti kullanıcıları için kullanılır.

*CBCREA* alanının değeri, durmanın nedenini gösterecek şekilde ayarlanır.

#### **CCTDC**

Bu çağrı tipinin amacı, geri bildirme işlevinin tüketmek işleminin sonunda son temizlemeyi gerçekleştirmesine izin vermesidir. Geri bildirme işlevi aşağıdaki durumlarda çağrılır:

- Geri bildirme işlevi, BCUNR ile bir MOCB çağrısı kullanılarak kayıttan kaldırılır.
- Kuyruk kapatıldı, örtülü bir kayıttan çıkarıcı neden oluyor. Bu örnekte, geri bildirme işlevi nesne tanıtıcısı olarak HOUNUH iletilidir.
- MQDISC çağrısı tamamlandı-örtülü kapanmaya neden oluyor ve bu nedenle bir kayıt kaldırma işlemi tamamlandı. Bu durumda, bağlantının bağlantısı kesilir ve devam eden hareket henüz kesinleştirilmedi.

Bu eylemlerden herhangi biri geri bildirme işlevinin kendisi içinde alınırsa, geri çağırma döndürdükten sonra işlem çağrılır.

Bu arama tipi hem ileti tüketicileri hem de olay işleyicileri için kullanılır.

İstenirse, bu, geri bildirme işlevinin son çağrısıdır.

*CBCREA* alanının değeri, durmanın nedenini gösterecek şekilde ayarlanır.

#### **CCTC**

##### **Olay işleyici işlevi**

Olay işleyici işlevi, aşağıdaki durumlarda bir ileti olmadan çağrıldı:

- CTLSP ' nin *Operation* alanı için bir değer içeren bir MQCTL çağrısı yayınlanır ya da
- Kuyruk yöneticisi ya da bağlantı durdurulur ya da susturulur.

Bu arama, tüm geri bildirme işlevleri için uygun işlemi yapmak için kullanılabilir.

- **Message consumer işlevi**

Nesne tanıtıcısı için belirli bir hata (CBCCC = CCFAIL) saptandığında ileti tüketici işlevi çağrılmadan çağrılır; örneğin, CBCREA kodu = RC2016 .

CBCREA alanının değeri, çağrımın nedenini gösterecek şekilde ayarlanır.

Bu bir giriş alanıdır. CBCTMR ve CMCTMN yalnızca ileti tüketici işlevleri için geçerlidir.

### **CBCCC (10 basamaklı işaretli tamsayı)**

Geri bildirme bağlamı yapısı- CompCode alanı.

Bu, tamamlanma kodudur. Bu ileti, iletiyi alan herhangi bir sorun olup olmadığını gösterir; bu, aşağıdakilerden biridir:

#### **CCOK**

Başarıyla tamamlandı

#### **CCWARN**

Uyarı (kısmi tamamlama)

#### **CCCFAIL**

Arama başarısız oldu

Bu bir giriş alanıdır. Bu alanın ilk değeri CCOK 'tır.

### **CBCCONNAREA (10 basamaklı işaretli tamsayı)**

Geri bildirme bağlamı yapısı- ConnectionArea alanı.

Bu alan, geri bildirme işlevi için kullanılabilir bir alandır.

Kuyruk yöneticisi, bu alanın içeriğine dayalı olarak hiçbir karar vermez ve geri bildirme işlevini denetlemek için kullanılan MQCTL çağrısındaki bir parametre olan MQCTLO yapısındaki ConnectionArea alanından değiştirilmeden geçirilir.

Geri bildirme işlevleri tarafından bu alanda yapılan değişiklikler, geri bildirme işlevi çağrıları boyunca korunur. Bu alan, tüm geri bildirme işlevleri tarafından paylaşılacak bilgileri geçirmek için kullanılabilir. *CallbackArea* 'in tersine, bu alan, bağlantı tanıtıcısı için tüm geri çağırılarda ortakdır.

Bu bir giriş ve çıkış alanıdır. Bu alanın başlangıç değeri, boş değerli ya da boş değerli bayttır.

### **CBCLLEN (10 basamaklı işaretli tamsayı)**

Bu, iletteki uygulama verilerinin bayt cinsinden uzunluğudur. Değer sıfırsa, iletinin uygulama verisi içermediği anlamına gelir.

CBLEN alanı, iletinin uzunluğunu içerir, ancak tüketiciye aktarılan ileti verilerinin uzunluğunun tam olarak uzunluğunu içermez. Bu, iletinin kısaltılması olabilir. Tüketiciye ne kadar veri aktarıldığını belirlemek için MQGMO 'daki GMRL alanını kullanın.

Neden kodu, iletinin kesildiğini gösteriyorsa, gerçek iletinin ne kadar büyük olduğunu belirlemek için CBCLLEN alanını kullanabilirsiniz. Bu, ileti verilerini barındırmak için gereken arabelleğin büyüklüğünü belirlemenizi ve sonra MQCBD 'deki CBDMML 'yi uygun değerle güncellemek için bir MQCB çağrısı yayınlayabilmenize olanak sağlar.

GMCONV seçeneği belirtilirse, dönüştürülen ileti, DataLength için döndürülen değerden daha büyük olabilir. Bu tür durumlarda, uygulamanın, MQCBD 'deki CBDMML 'i DataLength için kuyruk yöneticisi tarafından döndürülen değerden büyük olması için bir MQCB çağrısı yayınlamaya gerek vardır.

İleti kesme sorunlarını önlemek için, CBDFM olarak MaxMsgLength değerini belirtin. Bu, kuyruk yöneticisinin veri dönüştürmesinden sonra tam ileti uzunluğu için bir arabellek ayırmasına neden olur. Ancak, bu seçenek belirtilse bile, isteği doğru işlemek için yeterli depolama alanının kullanılmaması yine de mümkün olur. Uygulamalar her zaman döndürülen neden kodunu denetlemelidir. Örneğin, iletiyi dönüştürmek için yeterli bellek ayırmak mümkün değilse, iletiler uygulamaya dönüştürülmemiş olarak döndürülür.

Bu, ileti tüketicisi işlevine ilişkin bir giriş alanıdır; bir olay işleyici işleviyle ilgili değildir.

### **CBCFLG (10 basamaklı işaretli tamsayı)**

Bu tüketiciyle ilgili bilgileri içeren işaretler.

Aşağıdaki seçenek tanımlıdır:

#### **CCBCFBE**

COQSC seçeneğini kullanarak önceki bir MQCLOSE çağrısı başarısız olursa, bu işaret döndürülebilir. Neden kodu RC2458.

Bu kod, son okuma öncesinde okunan iletinin yayınlanmakta olduğunu ve arabelleğin artık boş olduğunu gösterir. Uygulama COQSC seçeneğini kullanarak başka bir MQCLOSE çağrısı yayınlarsa başarılı olur.

Bir uygulamanın, yürürlükteki seçim ölçütleriyle eşleşmeyen okuma tamamlama arabelleğindeki iletiler olabileceğinden, bir uygulamanın bu işaret kümesiyle bir ileti verilmesine garanti verilmemesini unutmayın. Bu örnekte, tüketici işlevi RC2019 neden koduyla çağrılır.

Okuma öncesinde okuma arabelleği boşsa, tüketici CBCFBE işaretiyle ve RC2518neden koduyla çağrılır.

Bu, ileti tüketicisi işlevine ilişkin bir giriş alanıdır; bir olay işleyici işleviyle ilgili değildir.

### **CBCHOBJ (10 basamaklı işaretli tamsayı)**

Geri bildirme bağlamı yapısı-CBCHOBJ alanı.

Bir ileti tüketicisi çağrılması için, ileti tüketicisi ile ilgili nesnenin tanıtıcı değeri budur.

Bir olay işleyicisi için bu değer HONONE değeridir.

Bir ileti kuyruktan kaldırılmamışsa iletiyi almak için, uygulama bu tanıtıcıyı ve İleti Alma Seçenekleri bloğunda ileti simgesini kullanabilir.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri HOUNUH değeridir.

### **CBCRCD (10 basamaklı işaretli tamsayı)**

**CBCRCD** , kuyruk yöneticisinin yeniden bağlanmayı denemeden önce ne kadar bekleyeceğini belirtir. Bu alan, gecikmeyi değiştirmek ya da tümüyle yeniden bağlantıyı durdurmak için bir olay işleyicisiyle değiştirilebilir.

Use the **CBCRCD** field only if the value of the **Reason** field in the Callback Context is RC2545.

On entry to the event handler the value of **CBCRCD** is the number of milliseconds the queue manager is going to wait before making a reconnection attempt. [Çizelge 686 sayfa 1008](#) , olay işleyicisinden dönüşte kuyruk yöneticisinin davranışını değiştirmek için ayarlayabileceğiniz değerleri listeler.

<i>Çizelge 686. CBCRCD değerler</i>	
<b>Değer</b>	<b>Tanım</b>
-1	Başka bir yeniden bağlantı kurma girişiminde bulunmaz. Uygulamaya bir hata döndürüldü.
0	Hemen yeniden bağlanmayı deneyin.
>0	Bağlantıyı yeniden denemeden önce bu milisaniyeleri bekleyin.

### **CBCREA (10 basamaklı işaretli tamsayı)**

Geri bildirme bağlamı yapısı-Neden alanı.

Bu, CBCCC' yi niteleyen neden kodudur.

Bu bir giriş alanıdır. Bu alanın ilk değeri RCNONE olur.

### **CBSTATE (10 basamaklı işaretli tamsayı)**

Mevcut tüketicinin durumuna ilişkin bir gösterge. Tüketici işlevine sıfır olmayan bir neden kodu geçirildiğinde, bu alan bir uygulamaya değer olarak en çok değer sağlar.

Bu alanı, her neden kodu için davranış kodlarına gerek kalmadığından uygulama programlamasını kolaylaştırmak için kullanabilirsiniz.

Bu bir giriş alanıdır. Bu alanın ilk değeri CSNONE olur.

<i>Çizelge 687. CBBSTATE değerleri ve sonuçtaki işlemler</i>		
<b>Durum</b>	<b>Kuyruk yöneticisi işlemi</b>	<b>Değişmez değer</b>
<i>CSNONE</i> Bu neden kodu, ek neden bilgisi olmayan olağan çağırıcıyı gösterir.	Hayır, bu normal bir işlemdir.	0
<i>CSSUST</i> Bu neden kodları geçici durumları gösterir.	Geri çağırma yordamı, koşulu raporlamak ve sonra askıya almak için çağırılır. Bir süre geçtikten sonra, sistem işlemi yeniden deneyebilir ve bu da aynı durumun yeniden yükseltilmesine neden olabilir.	1
<i>CSSUSU</i> Bu neden kodları, geri bildirmenin koşulu çözümlenmek için harekete geçeceği koşulları gösterir.	Tüketici askıya alınır ve geri çağırma yordamı durumu raporlamak için çağırılır. Geri çağırma yordamı, mümkünse koşulu çözmeli ve bağlantıyı devam ettirmeli ya da kapatılmalıdır.	2
<i>CSSUS</i> Bu neden kodları, daha fazla ileti geri bildirmesini önleyen hataları gösterir.	Kuyruk yöneticisi geri çağırma işlevini otomatik olarak askıya alır. Geri çağırma işlevi sürdürülürse, yine aynı neden kodu alma olasılığı vardır.	3
<i>CSSTOP</i> Bu neden kodları, ileti tüketiminin sona ermesini gösterir.	Kural dışı durum işleyicisine ve belirtilen CBDTC 'nin geri çağırılmasına teslim edildi. Başka ileti tüketilemez.	4

#### **CBCSID (10 basamaklı işaretli tamsayı)**

Geri bildirme bağlamı yapısı- StrucId alanı.

Bu, yapı tanıttıcısıdır; değer şöyle olmalıdır:

#### **CBCSI**

Geri çağırma bağlamı yapısına ilişkin tanıttıcı.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri CBCSI 'tır.

#### **CBCVER (10 basamaklı işaretli tamsayı)**

Geri bildirme bağlamı yapısı-Sürüm alanı.

Bu, yapı sürüm numarasıdır; değer şu olmalıdır:

#### **CBCV1**

Version-1 geri bildirme bağlamı yapısı.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

#### **CBCCV**

Geri çağırma bağlamı yapısının yürürlükteki sürümü.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri CBCV1' dir.

## Başlangıçtaki değerler

Çizelge 688. MQCBC 'deki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
CBCSID	CBCSI	'CBC¬'
CBCVER	CBCV1	1
CBCCALLT	Yok	0
CBCHOBJ	HOUNUH	-1
CBCCALLBA	Yok	Boş değerli gösterge ya da boş değerli byte
CBCCONNAREA	Yok	Boş değerli gösterge ya da boş değerli byte
CBCCC	CCOK	0
CBCREA	YOK	0
CBCSTATE	MCSNONE	0
CBCLLEN	Yok	0
CBCBUFFLEN	Yok	0
CBCFLG	Yok	0
CBCRCD	yok	0

### Not:

1. ¬ simgesi tek bir boş karakteri temsil eder.

## RPG bildirimi

```
D* MQCBC Structure
D*
D*
D* Structure identifier
D CBCSID          1      4      INZ('CBC ')
D*
D* Structure version number
D CBCVER          5      8I 0  INZ(1)
D*
D* Why Function was called
D CBCCALLT        9      12I 0  INZ(0)
D*
D* Object Handle
D CBCHOBJ         13     16I 0  INZ(-1)
D*
D* Callback data passed to the function
D CBCCALLBA       17     32*   INZ(*NULL)
D*
D* MQCTL Data area passed to the function
D CBCCONNAREA     33     48*   INZ(*NULL)
D*
D* Completion Code
D CBCCC           49     52I 0  INZ(0)
D*
D* Reason Code
D CBCREA          53     56I 0  INZ(0)
D*
D* Consumer State
D CBCSTATE        57     60I 0  INZ(0)
D*
D* Message Data Length
D CBCLLEN         61     64I 0  INZ(0)
```

```
D*
D* Buffer Length
D CBCUFFLEN          65      68I 0 INZ(0)
D*
** Flags containing information about
D* this consumer
D CBCFLG             69      72I 0 INZ(0)
D* Ver:1 **
D* Number of milliseconds before reconnect attempt
D CBCRCD             73      76I 0 INZ(0)
D* Ver:2 **
D*
```

## IBM i MQCBD (Callback descriptor) on IBM i

Geri bildirme işlevini belirten yapı.

### Genel Bakış

**Amaç:** MQCBD yapısı, bir geri çağırma işlevi ve kuyruk yöneticisi tarafından kullanımını denetleyen seçenekleri belirlemek için kullanılır.

Yapı, MQCB çağrısına ilişkin bir giriş değiştirgedir.

**Sürüm:** MQCBD 'nin yürürlükteki sürümü CBDV1' dir.

**Karakter kümesi ve kodlama:** MQCBD içindeki veriler, yerel kuyruk yöneticisinin karakter kümesinde ve kodlamasında olmalıdır; bunlar, **CodedCharSetId** kuyruk yöneticisi özniteliği ve ENNAT tarafından verilir. Ancak, uygulama bir IBM MQ MQI clientolarak çalışıyorsa, yapı istemcinin karakter kümesinde ve kodlamasında olmalıdır.

- [“Alanlar” sayfa 1011](#)
- [“Başlangıçtaki değerler” sayfa 1015](#)
- [“RPG bildirimi” sayfa 1015](#)

### Alanlar

MQCBD yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

#### CBDCALLBA (10 basamaklı işaretli tamsayı)

Bu alan, geri bildirme işlevi için kullanılabilir bir alandır.

Kuyruk yöneticisi, bu alanın içeriğine dayalı olarak karar vermez ve geri bildirme işlevi bildiriminde bir parametre olan MQCBD yapısındaki CBCCaLLBA alanından değiştirilmeden geçirilir.

Bu değer yalnızca, tanımlı bir geri çağırım olmadan CBREG değeri olan bir *Operation* ' ta kullanılır; önceki bir tanımlamayı değiştirmez.

Bu, geri bildirme işlevi için bir giriş ve çıkış alanıdır. Bu alanın başlangıç değeri, boş değerli ya da boş değerli bayttır.

#### CBDCALLBF (10 basamaklı işaretli tamsayı)

Geri bildirme işlevi, işlev çağrısı olarak çağrılır.

Geri bildirme işlevine ilişkin bir gösterge belirtmek için bu alanı kullanın.

*CallbackFunction* ya da *CallbackName* değerini belirtmeniz gerekir. Her ikisini de belirtirseniz, RC2486 neden kodunu da döndürerek geri döndür.

*CallbackName* ya da *CallbackFunction* ayarlanmadıysa, çağrı neden kodu RC2486 ile başarısız olur.

Bu seçenek aşağıdaki ortamlarda desteklenmez:

- CICS Değiştirildiği tarih/saat: z/OS
- İşlev göstergesi başvurularını desteklemeyen programlama dilleri ve derleyiciler

Bu tür durumlarda, arama, RC2486neden koduyla başarısız olur.

Bu bir giriş alanıdır. Bu alanın başlangıç değeri, boş değerli ya da boş değerli bayttır.

### **CBDSCALLBN (10 basamaklı işaretli tamsayı)**

Geri bildirme işlevi dinamik olarak bağlantılı bir program olarak çağrılır.

*CallbackFunction* ya da *CallbackName*değerini belirtmeniz gerekir. Her ikisini de belirtirseniz, RC2486 neden kodunu da döndürerek geri döndür.

*CallbackName* ya da *CallbackFunction* doğru değilse, arama başarısız olur ve neden kodu RC2486ile başarısız olur.

Kullanılacak ilk geri çağrı yordamı kayıtlı olduğunda modül yüklenir ve son geri bildirme yordamı derejisterleri kullanacak şekilde geri yüklendiğinde yüklenir.

Aşağıdaki metinde belirtilenler dışında, ad alanda sola hizalanmış, boşluk içermeyen bir ad; adın kendisi, alanın uzunluğuna kadar boşluklarla dolduruludur. Aşağıdaki açıklamalarda, köşeli ayraçlar ([]) isteğe bağlı bilgileri gösterir:

#### **IBMi**

Geri çağırma adı, aşağıdaki biçimlerden biri olabilir:

- Kitaplık "/" Program
- Library "/" ServiceProgram ("FunctionName")

Örneğin, MyLibrary/MyProgram(MyFunction).

Kitaplık adı \*LIBL olabilir. Kitaplık ve program adları en çok 10 karakterle sınırlanmıştır.

#### **AIX and Linux**

Geri bildirme adı, dinamik olarak yüklenebilir bir modülün ya da kitaplığın adı, o kitaplıkta bulunan bir işlevin adına sahip olan bir kitaplık adıdır. İşlev adının ayraç içine alınması gerekir. Kitaplık adı isteğe bağlı olarak bir dizin yolu ile önek olarak önek olarak kullanılabilir:

```
[path]library(function)
```

Yol belirlenmezse, sistem arama yolu kullanılır.

Ad en çok 128 karakterle sınırlanmıştır.

#### **Windows**

Geri çağırma adı, bir dinamik bağlantı kitaplığının adıdır, o kitaplıkta bulunan bir işlevin adına sahip olan bir alt düzeltmenin adıdır. İşlev adının ayraç içine alınması gerekir. Kitaplık adı, isteğe bağlı olarak bir dizin yolu ve sürücü öneki ile önek olarak kullanılabilir:

```
[d:][path]library(function)
```

Sürücü ve yol belirlenmezse, sistem arama yolu kullanılır.

Ad en çok 128 karakterle sınırlanmıştır.

#### **z/OS**

Geri çağırma adı, LINK ya da LOAD makrosu için EP parametresindeki belirtim için geçerli olan bir yükleme modülünün adıdır.

Ad en çok 8 karakterden oluşan bir değer ile sınırlanmıştır.

#### **z/OS CICS**

Geri bildirme adı, EXEC CICS LINK komut makrosu PROGRAM parametresindeki belirtim için geçerli olan bir yükleme modülünün adıdır.

Ad en çok 8 karakterden oluşan bir değer ile sınırlanmıştır.

Program, kurulu PROGRAM tanımının REMOTEYTEM seçeneği kullanılarak ya da dinamik yöneltme programı tarafından uzak olarak tanımlanabilir.



Program IBM MQ API çağrılarını kullanacaksa, uzak CICS bölgesi IBM MQ ' e bağlanmalıdır. Ancak, MQCBC yapısındaki CBCHOBJ alanının uzak bir sistemde geçerli olmadığı unutulmamaktadır.

*CallbackName* yükleme girişimi sırasında bir hata oluşursa, uygulamaya aşağıdaki hata kodlarından biri döndürülür:

- RC2495
- RC2496
- RC2497

Yükleme girişiminde bulunulan modülün adını ve işletim sisteminden hatalı neden kodunu içeren hata günlüğüne bir ileti de yazılır.

Bu bir giriş alanıdır. Bu alanın başlangıç değeri, boş değerli bir dizilim ya da boşluktur.

### **CBDCALLBT (10 basamaklı işaretli tamsayı)**

Bu, geri bildirme işlevinin tipidir. Değer aşağıdakilerden biri olmalıdır:

#### **CCBTMC**

Bu geri çağrıyı ileti tüketici işlevi olarak tanımlar.

Bir ileti tüketici geri bildirme işlevi, bir ileti, belirtilen seçim ölçütleriyle toplantı sırasında, bir nesne tanıtıcısında kullanılabilir olduğunda ve bağlantı başlatılmış olduğunda çağrılır.

#### **CTEH**

Bu geri çağrıyı zamanuyumsuz olay yordamı olarak tanımlar; bir tanıtıcı için iletileri tüketmek için yönlendirilmez.

Olay işleyicisini tanımlayan MQCB çağrısında *Hobj* gerekli değil ve belirtilirse yoksayılır.

Olay işleyici, tüm ileti tüketicisi ortamını etkileyen koşullar için çağrılır. Bir olay, kuyruk yöneticisi ya da bağlantı durdurma ya da susturma gibi bir olay gerçekleştiğinde, tüketici işlevi ileti olmadan çağrılır. Tek bir ileti tüketicisi için özel durumlar için çağrılmaz; örneğin, RC2016.

Aşağıdaki ortamlar dışında, olaylar, bağlantının başlatılıp başlatılmadığından bağımsız olarak, uygulamaya teslim edilir:

- z/OS üzerinde CICS ortamı
- iş parçacıklı uygulamalar

Çağırın bu değerlerden birini geçmezse, çağrı bir neden kodu RC2483 ile başarısız olur

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri CBTMC 'dir.

### **CBDMML (10 basamaklı işaretli tamsayı)**

Bu, tanıtıcıdan okunabilen ve geri çağırma yordamında okunabilen en uzun iletinin bayt cinsinden uzunluğudur. If a message has a longer length, the callback routine receives *MaxMsgLength* bytes of the message, and reason code:

- RC2080 ya da
- RC2079 , GMATM ' yi belirtmiş misiniz.

Gerçek ileti uzunluğu, MQCBC yapısının "CBCLLEN (10 basamaklı işaretli tamsayı)" sayfa 1007 alanında sağlanır.

Aşağıdaki özel değer tanımlıdır:

#### **CDFM**

Arabellek uzunluğu sistem tarafından, iletilerin kesilmeden döndürülmesi için ayarlanıyor.

İletiyi almak için arabellek ayırmak için yeterli bellek yoksa, sistem geri bildirme işlevini bir RC2071 neden koduyla çağırır.

Örneğin, veri dönüştürme isteğinde bulunsanız ve ileti verilerini dönüştürmek için yeterli bellek yoksa, dönüştürülemez ileti geri bildirme işlevine geçirilir.

Bu bir giriş alanıdır. *MaxMsgLength* alanının ilk değeri CBDFM ' dir.

### **CBDOPT (10 basamaklı işaretli tamsayı)**

Geri çağırma tanımlayıcısı yapısı-Seçenekler alanı.

Aşağıdakilerin herhangi biri ya da tümü belirlenebilir. Geçerli olmayanbirden çok seçenek belirtmek için, değerleri bir araya ekleyin (aynı değışmezi bir kereden fazla eklemeyin) ya da bit ya da işlemi kullanarak değerleri birleştirin (programlama dili, bit işlemleri destekliyorsa). Bileşimleri dikkat çekilir; diğer birleşimler geçerlidir.

#### **CBDFK**

Kuyruk yöneticisi susturma durumdaysa, MQCB çağırısı başarısız olur.

z/OS' ta bu seçenek, bağlantı ( CICS ya da IMS uygulaması için) susturulmuş durumdaysa, MQCB çağırısını başarısız olarak zorlar.

MQCB çağırısında geçirilen MQGMO seçeneklerinde, susturma sırasında ileti tüketicilerine bildirim yol açabilmek için GMFIQ değerini belirtin.

**Denetim seçenekleri:** Aşağıdaki seçenekler, tüketici değışikliklerinin durumu değıştiğinde, bir ileti olmadan geri çağırma işlevinin çağrılıp çağrılmayacağını denetler:

#### **CCBDRC**

Geri bildirme işlevi CBCTRC çağırısıyla çağrılır

#### **CBDSC**

Geri bildirme işlevi, CBCTSC çağrı tipiyle çağrılır.

#### **CBDTC**

Geri bildirme işlevi, CBCTTC çağrı tipiyle çağrılır.

#### **CCBDDC**

Geri bildirme işlevi CBCTDC (CBCTDC) çağırısıyla çağrılır.

Bu çağrı tipleriyle ilgili daha fazla ayrıntı için bkz. [“CBCCALLT \(10 basamaklı işaretli tamsayı\)” sayfa 1005](#).

**Varsayılan seçenek:** Açıklanılan seçeneklerden herhangi birine gereksiniminiz yoksa, aşağıdaki seçeneği kullanın:

#### **G CBDNO**

Başka bir seçenek belirtilmemeyi belirtmek için bu değeri kullanın; tüm seçenekler varsayılan değerlerini kabul eder.

CBDNO, program belgelerine yardımcı olmak üzere tanımlıdır; bu seçeneğin başka biriyle birlikte kullanılması amaçlanmaz, ancak değeri sıfır olduğu için, bu tür bir kullanım saptanamaz.

Bu bir giriş alanıdır. *Options* alanının başlangıç değeri CBDNO ' dur.

### **CBDSID (10 basamaklı işaretli tamsayı)**

Geri çağırma tanımlayıcısı yapısı- StrucId alanı.

Bu, yapı tanıttıcısıdır; değeri şöyle olmalıdır:

#### **CBDSI**

Geri çağırma tanımlayıcısı yapısına ilişkin tanıtıcı.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri CBDSI 'dir.

### **CBDVER (10 basamaklı işaretli tamsayı)**

Geri çağırma tanımlayıcısı yapısı-Sürüm alanı.

Bu, yapı sürüm numarasıdır; değeri şu olmalıdır:

#### **CBDV1**

Version-1 geri bildirim tanımlayıcısı yapısı.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

### CDCV V

Geri çağırma tanımlayıcısı yapısının yürürlükteki sürümü.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri CBDV1' dir.

## Başlangıçtaki değerler

Çizelge 689. MQCBD içindeki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
<i>StrucId</i>	CBDSI	' CBD↵ '
<i>Version</i>	CBDV1	1
<i>CallbackType</i>	CCBTMC	1
<i>Options</i>	GCBDNO	0
<i>CallbackArea</i>	Yok	Boş bayt
<i>CallbackFunction</i>	Yok	Boş bayt
<i>CallbackName</i>	Yok	Boşluklar
<i>MaxMsgLength</i>	CDFM	-1

### Not:

1. ↵ simgesi tek bir boş karakteri temsil eder.

## RPG bildirim

```
D* MQCBD Structure
D*
D*
D* Structure identifier
D  CBDSID          1      4    INZ('CBD ')
D*
D* Structure version number
D  CBDVER          5      8I 0 INZ(1)
D*
D* Callback function type
D  CBDCALLBT       9     12I 0 INZ(1)
D*
** Options controlling message
D* consumption
D  CBDOPT          13     16I 0 INZ(0)
D*
D* User data passed to the function
D  CBDCALLBA       17     32*
D*
D* FP: Callback function pointer
D  CBDCALLBF       33     48*
D*
D* Callback name
D  CBDCALLBN       49     176  INZ('\0')
D*
D* Maximum message length
D  CBDMML          177    180I 0 INZ(-1)
```

## IBM i IBM üzerinde MQCHARV (Değişken Uzunluğu Dizgisi)

Değişken uzunluklu bir dizilimi tanımlamak için MQCHARV yapısını kullanın.

## Genel Bakış

**Karakter kümesi ve kodlama:** MQCHARV içindeki veriler, ENNAT tarafından verilen yerel kuyruk yöneticisinin kodlamasında ve yapı içindeki VCHRC alanının karakter kümesiyle olmalıdır. Uygulama bir IBM MQ MQI clientolarak çalışıyorsa, yapının istemcinin kodlamasında olması gerekir. Bazı karakter kümelerinin, kodlamaya bağlı bir gösterimi vardır. VCHRC, bu karakter kümelerinden biriyse, kullanılan kodlama, MQCHARV içindeki diğer alanlarla aynı kodlamayla aynı kodda olur. VSCCSID ile tanımlanan karakter takımı, çift baytlık karakter takımı (DBCS) olabilir.

**Kullanım:** MQCHARV yapısı, onu içeren yapıyla ayrılabilir olabilecek verileri ele alır. Bu verileri ele almak için, işaretçi veri tipiyle bildirilen alanlar kullanılabilir.

- [“Alanlar” sayfa 1016](#)
- [“Başlangıçtaki değerler” sayfa 1017](#)
- [“RPG bildirim” sayfa 1017](#)
- [“CSAPL ' nin yeniden tanımlanması” sayfa 1018](#)

## Alanlar

MQCHARV yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

### VCHRC (10 basamaklı işaretli tamsayı)

Bu, VCHRP ya da VCHRO alanı tarafından adreslenen değişken uzunluklu dizilimin karakter takımı tanıtıcısıdır.

Bu alanın ilk değeri CSAPL ' dir. Bu, kuyruk yöneticisinin kuyruk yöneticisinin true karakter kümesi tanıtıcısına kuyruk yöneticisi tarafından değiştirilmesi gerektiğini belirtmek için IBM MQ tarafından tanımlanır. Bu, CSQM ' nin davranışı ile aynı şekilde. Sonuç olarak CSAPL değeri hiçbir zaman bir değişken uzunluğu dizisiyle ilişkilendirilmemiş. Bu alanın ilk değeri, uygulama programlama diliniz için uygun olan derleme biriminiz için sabit CSAPL değeri için farklı bir değer tanımlanarak değiştirilebilir.

### VCHRL (10 basamaklı işaretli tamsayı)

VCHRP ya da VCHRO alanı tarafından adreslenen değişken uzunluklu dizilimin bayt cinsinden uzunluğu.

Bu alanın ilk değeri 0 ' tır. Değer sıfırdan büyük ya da sıfıra eşit olmalı ya da şu özel değer tanınmalıdır:

### VSNTL

VSNTL belirlenmezse, dizilimin bir parçası olarak VCHRL byte ' ları eklenir. Boş değerli karakterler varsa, diziyi sınırlamazlar.

VSNTL belirtilirse, dizgi dizgide saptanan ilk boş değerle sınırlanır. Boş değer, o dizginin bir parçası olarak içerilmedi.

**Not:** VSNTL belirtilirse, bir dizgiyi sonlandırmak için kullanılan boş karakter, VCHRC tarafından belirlenen kod kümesinden boş değerde olur.

Örneğin, UTF-16 ' da (CCSID 1200, 13488 ve 17584), bu, 2 baytlık Unicode kodlamasıdır; burada boş değer, sıfırların 16 bitlik bir sayısıdır. UTF-16 ' da, karakterlerin bir parçası olan tüm sıfıra ayarlanmış tek baytların bulunması yaygındır (örneğin, 7 bitlik ASCII karakterleri), ancak çift byte sınırında iki ' sıfır ' byte bulunduğunda dizgiler boş olarak sonlandırılır. Geçerli karakterlerin her bir parçasıysa, tek bir sınır üzerinde iki ' sıfır ' byte elde etmek mümkündür. Örneğin, x '01' x '00' x '00' x '30', geçerli iki Unicode karakteri temsil eder ve boş olarak dizgiyi sonlandırmaz.

### VCHRO (10 basamaklı işaretli tamsayı)

MQCHARV ' nin başlangıcındaki değişken uzunluklu dizginin bayt cinsinden görel konumu ya da onu içeren yapı.

MQCHARV yapısı başka bir yapı içine yerleştirildiğinde, bu değer, bu MQCHARV yapısını içeren yapının başlangıcındaki değişken uzunluktaki dizginin bayt cinsinden görel konudur. MQCHARV yapısı başka

bir yapı içine yerleştirilmediyse, örneğin, bir işlev çağrısında parametre olarak belirtilirse, görelî konum MQCHARV yapısının başlangıcıyla görelî olur.

Görelî konum pozitif ya da negatif olabilir. Değişken uzunluklu dizgiyi belirtmek için VCHRP ya da VCHRO alanını kullanabilirsiniz, ancak her ikisini birden belirleyemez.

Bu alanın ilk değeri 0 'tır.

### VCHRP (işaretçi)

Bu, değişken uzunluklu dizilimin göstergesini gösterir.

Değişken uzunluklu dizgiyi belirtmek için VCHRP ya da VCHRO alanını kullanabilirsiniz, ancak her ikisini birden belirleyemez.

Bu alanın başlangıç değeri, boş değerli ya da boş değerli bayttır.

### VCHRS (10 basamaklı işaretli tamsayı)

VCHRP ya da VCHRO alanı tarafından adreslenen arabelleğin bayt cinsinden büyüklüğü.

Bir işlev çağrısında bir çıkış alanı olarak MQCHARV yapısı kullanıldığında, bu alan, sağlanan arabelleğin uzunluğuna göre ilk kullanıma hazırlanmalıdır. VCHRL değeri, VCHRS değerinden büyükse, arabelleğe çağrıya yalnızca VCHRS byte veri byte 'ları döndürülecektir.

Değer, sıfırdan büyük ya da bu değere eşit olmalı ya da şu özel değer tanınmalıdır:

### VSUML

VSUML belirtilirse, arabelleğin uzunluğu MQCHARV yapısındaki VCHRL alanından alınır. Bu özel değer, yapı bir çıkış alanı olarak kullanıldığında ve bir arabellek sağlandığında uygun değerdir. Bu, bu alanın ilk değeridir.

## Başlangıçtaki değerler

Çizelge 690. Sabitler için MQCHARV ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
VCHRP	Yok	Boş değerli gösterge ya da boş değerli byte.
VCHRO	Yok	0
VCHRS	VSUML	-1
VCHRL	Yok	0
VCHRC	CSAPL	-3

## RPG bildirimî

```
D* .1.....2.....3.....4.....5.....6.....7..
D* MQCHARV Structure
D*
D* Address of variable length string
D VCHRP          1      16*
D* Offset of variable length string
D VCHRO          17      20I 0
D* Size of buffer
D VCHRS          21      24I 0
D* Length of variable length string
D VCHRL          25      28I 0
D* CCSID of variable length string
D VCHRC          29      32I 0
```

## CSAPL ' nin yeniden tanımlanması

Diğer altyapılarda desteklenen programlama dillerinin tersine, RPG 'nin tanımlı bir değişmezi yeniden tanımlamanın bir yolu yoktur; bu nedenle, her VCHRC' yi CSAPL dışında bir değer kullanmak istiyorsanız özel olarak ayarlamalısınız.

## IBM i IBM üzerinde MQCIH (CICS bridge üstbilgisi)

MQCIH yapısı, CICS bridge aracılığıyla IBM MQ for z/OS' e gönderilen bir iletinin başlangıcında var olabilecek bilgileri açıklar.

### Genel Bakış

**Biçim adı:** FMCICS.

**Sürüm:** MQCIH 'nin yürürlükteki sürümü CIVER2' dir. Yalnızca yapının daha yeni sürümünde var olan alanlar, izleyen açıklamalarda olduğu gibi tanımlanır.

The COPY file provided contains the most recent version of MQCIH, with the initial value of the CIVER field set to CIVER2.

**Karakter kümesi ve kodlama:** Özel koşullar, MQCIH yapısı ve uygulama iletisi verileri için kullanılan karakter kümesi ve kodlama için geçerlidir:

- CICS bridge kuyruğunun sahibi olan kuyruk yöneticisine bağlanan uygulamalar, kuyruk yöneticisinin karakter kümesinde ve kodlamasında bulunan bir MQCIH yapısı sağlamalıdır. Bunun nedeni, bu durumda MQCIH yapısının veri dönüştürmesi gerçekleştirilmemesinden kaynaklanır.
- Applications that connect to other queue managers can provide an MQCIH structure that is in any of the supported character sets and encodings; conversion of the MQCIH is performed by the receiving message channel agent connected to the queue manager that owns the CICS bridge queue.

**Not:** Bu konuda bir istisna var. If the queue manager that owns the CICS bridge queue is using CICS for distributed queuing, the MQCIH must be in the character set and encoding of the queue manager that owns the CICS bridge queue.

- MQCIH yapısından sonra gelen uygulama iletisi verileri, MQCIH yapısıyla aynı karakter kümesinde ve kodlamada olmalıdır. MQCIH yapısındaki CICS ve CIENC alanları, uygulama iletisi verisinin karakter kümesini ve kodlamasını belirtmek için kullanılamaz.

Veri, kuyruk yöneticisi tarafından desteklenen yerleşik biçimlerden biri değilse, uygulama iletisi verilerini dönüştürmek için kullanıcı tarafından bir veri dönüştürme çıkışı sağlanmalıdır.

**Kullanım:** Uygulamanın gerektirdiği değerler [Çizelge 692 sayfa 1027](#)' ta gösterilen ilk değerlerle aynıysa ve köprü AUTH=LOCAL ya da AUTH=TSUP ile çalışıyorsa, MQCIH yapısı iletiden atlanabilir. Diğer tüm durumlarda, yapının mevcut olması gerekir.

Köprü, version-1 ya da version-2 MQCIH yapısını kabul eder, ancak 3270 işlemleri için bir version-2 yapısı kullanılmalıdır.

The application must ensure that fields documented as "istik" fields have appropriate values in the message sent to the bridge; these fields are input to the bridge.

"Yanıt" alanları olarak belgelenmiş alanlar, köprünün uygulamaya gönderdiği yanıt iletisinde CICS bridge tarafından ayarlanır. Error information is returned in the CIRET, CIFNC, CICC, CIREA, and CIAC fields, but not all of them are set in all cases. [Çizelge 691 sayfa 1018](#) , farklı CIRET değerleri için hangi alanların belirlendiğini gösterir.

Çizelge 691. MQCIH yapısındaki hata bilgileri alanlarının içeriği				
CIRET	CIFNC	CICC	CIREA	CIAC
CRC000	-	-	-	-
CRC003	-	-	GB*	-

Çizelge 691. MÇCIH yapısındaki hata bilgileri alanlarının içeriği (devamı var)				
<b>CIRET</b>	<b>CIFNC</b>	<b>CICC</b>	<b>CIREA</b>	<b>CIAC</b>
CRC002 CRC008	IBM MQ arama adı	IBM MQ <i>CMPCOD</i>	IBM MQ <i>REASON</i>	-
CRC001 CRC006 CRC007 CRC009	CICS EIBFN	CICS EIBRESP	CICS EIBRESP2	-
CRC004 CRC005	-	-	-	CICS ABCODE

- “Alanlar” sayfa 1019
- “Başlangıçtaki değerler” sayfa 1027
- “RPG bildirim” sayfa 1028

## Alanlar

MÇCIH yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

### CCIAC (4 baytlık karakter dizilimi)

Olağandışı kod.

The value returned in this field is significant only if the *CIRET* field has the value CRC005 or CRC004. Yapılırsa, *CIAC* , CICS ABCODE değerini içerir.

Bu bir yanıt alanıdır. Bu alanın uzunluğu LNABNC tarafından verilir. Bu alanın ilk değeri 4 boş karakterdir.

Bu gösterge, ADS tanımlayıcılarının SEND ve RECEIVE BMS isteklerinde gönderilip gönderilmeyeceğini belirleyen bir göstergedir. Aşağıdaki değerler tanımlanır:

#### YOK

ADS tanımlayıcısı göndermeyin ya da göndermeyin.

#### ADSEND

ADS tanımlayıcısı gönderin.

#### DREDCV

ADS tanımlayıcısını al

#### ADMSGF

ADS tanımlayıcısı için ileti biçimini kullanın.

Bu, ADS tanımlayıcısının, ADS tanımlayıcısının uzun biçimi kullanılarak gönderilmesine ya da alınmasına neden olur. Uzun biçimli alanlar, 4 baytlık sınırlar üzerinde hizalanmış alanlar içerir.

*CIADS* alanı aşağıdaki gibi ayarlanmalıdır:

- ADS tanımlayıcıları kullanılmıyorsa, alanı ADNONE olarak ayarlayın.
- ADS tanımlayıcıları : kullanılıyorsa ve her ortamda *aynı* CCSID 'si ile, alanı ADSEND ve ADRECV toplamıyla ayarlayın.
- ADS tanımlayıcıları : kullanılıyorsa, ancak her ortamda *farklı* CCSID ' lerle birlikte, alanı ADSEND, ADRECV ve ADMSGF toplamını toplayacak şekilde ayarlayın.

Bu, yalnızca 3270 işlemleri için kullanılan bir istek alanıdır. Bu alanın ilk değeri ADNONE olur.

### CIADS (10 basamaklı işaretli tamsayı)

Gönderme/alma ADS tanımlayıcısı.

Bu gösterge, ADS tanımlayıcılarının SEND ve RECEIVE BMS isteklerinde gönderilip gönderilmeyeceğini belirleyen bir göstergedir. Aşağıdaki değerler tanımlanır:

#### YOK

ADS tanımlayıcısı göndermeyin ya da göndermeyin.

**ADSEND**

ADS tanımlayıcısı gönderin.

**DREDCV**

ADS tanımlayıcısını al

**ADMSGF**

ADS tanımlayıcısı için ileti biçimini kullanın.

Bu, ADS tanımlayıcısının, ADS tanımlayıcısının uzun biçimi kullanılarak gönderilmesine ya da alınmasına neden olur. Uzun biçimli alanlar, 4 baytlık sınırlar üzerinde hizalanmış alanlar içerir.

CIADS alanı aşağıdaki gibi ayarlanmalıdır:

- ADS tanımlayıcıları kullanılmıyorsa, alanı ADNONE olarak ayarlayın.
- ADS tanımlayıcıları : kullanılıyorsa ve her ortamda *aynı* CCSID 'si ile, alanı ADSEND ve ADRECV toplamıyla ayarlayın.
- ADS tanımlayıcıları : kullanılıyorsa, ancak her ortamda *farklı* CCSID ' lerle birlikte, alanı ADSEND, ADRECV ve ADMSGF toplamını toplayacak şekilde ayarlayın.

Bu, yalnızca 3270 işlemleri için kullanılan bir istek alanıdır. Bu alanın ilk değeri ADNONE olur.

**CIAI (4 baytlık karakter dizilimi)**

AID anahtarı.

Bu, işlem başlatıldığında AID tuşunun ilk değeridir. 1 byte 'lık bir değer, sola hizalanmış.

Bu, yalnızca 3270 işlemleri için kullanılan bir istek alanıdır. Bu alanın uzunluğuna LNATID değeri verilir. Bu alanın ilk değeri 4 boşluktur.

**CIAUT (8 baytlık karakter dizilimi)**

Şifre ya da passticket.

Bu bir şifre ya da passticket. If user-identifier authentication is active for the CICS bridge, CIAUT is used with the user identifier in the MQMD identity context to authenticate the sender of the message.

Bu bir istek alanıdır. Bu alanın uzunluğu LNAUTH tarafından verilir. Bu alanın ilk değeri 8 boşluktur.

**CIC (10 basamaklı işaretli tamsayı)**

IBM MQ completion code or CICS EIBRESP.

Bu alanda döndürülen değer CIRET ' a bağlıdır; bkz. [Çizelge 691 sayfa 1018](#).

Bu bir yanıt alanıdır. Bu alanın ilk değeri CCOK 'tır.

**CICSC (4 baytlık karakter dizilimi)**

Olağandışı işlem kodu.

Bu, hareketi sonlandırmak için kullanılacak olağandışı son koddur (genellikle daha fazla veri isteyen bir etkileşimli işlem). Ters durumda, bu alan boşluk karakterine ayarlanır.

Bu, yalnızca 3270 işlemleri için kullanılan bir istek alanıdır. Bu alanın uzunluğu LNCNCL tarafından verilir. Bu alanın ilk değeri 4 boşluktur.

**CICSP (10 basamaklı işaretli tamsayı)**

İmleç konumu.

Bu, hareket başlatıldığında ilk imleç konumdur. Daha sonra, etkileşimli işlemler için, imleç konumu RECEIVE vektöründe yer alıyor.

Bu, yalnızca 3270 işlemleri için kullanılan bir istek alanıdır. Bu alanın ilk değeri 0 'tır. CIVER , CIVER2değerinden küçükse bu alan yoktur.

**CICSI (10 basamaklı işaretli tamsayı)**

Ayrılmış.



Bu ayrılmış bir alandır; değeri anlamlı değildir. Bu alanın ilk değeri 0 'tır.

#### **CICST (10 basamaklı işaretli tamsayı)**

Görevin etkileşimli olup olmayacağını.

Bu, görevin daha fazla bilgi için istek yayınına izin verilip verilmeyeceğini ya da olağandışı sona ermesi gerektiğini belirten bir göstergedir. Değer aşağıdakilerden biri olmalıdır:

##### **CTYES**

Görev birbirine yakındır.

##### **CTNO**

Görev birbirine uygun değil.

Bu, yalnızca 3270 işlemleri için kullanılan bir istek alanıdır. Bu alanın ilk değeri CTNO ' dur.

#### **CIENC (10 basamaklı işaretli tamsayı)**

Ayrılmış.

Bu ayrılmış bir alandır; değeri anlamlı değildir. Bu alanın ilk değeri 0 'tır.

#### **CIEO (10 basamaklı işaretli tamsayı)**

İletide hata görelili konumu.

Bu, köprü çıkışı tarafından saptanan geçersiz veri konumudur. Bu alan, iletinin başlangıcından, geçersiz verilerin yerine görelili konum sağlar.

Bu, yalnızca 3270 işlemleri için kullanılan bir yanıt alanıdır. Bu alanın ilk değeri 0 'tır. *CIVER* , *CIVER2* değerinden küçükse bu alan yoktur.

#### **CIFAC (8 baytlık bit dizesi)**

Köprü tesisi simgesi.

Bu, 8 baytlık bir köprü olanağı simgesiyle. Bir köprü olanağı simgesinin amacı, aynı köprü tesisini (sanal 3270 uçbirimi) kullanmak için bir sözde etkileşimde birden çok işlemin yapılmasına izin vermesidir. İlk ya da yalnızca, sözde bir etkileşimde ileti, FCNONE değeri ayarlanmalıdır; bu, CICS ' a bu ileti için yeni bir köprü olanağı ayırmayı bildirir. Giriş iletisinde sıfır olmayan bir *CIFKT* belirtildiğinde yanıt iletilerinde bir köprü olanağı simgesi döndürülür. Sonraki giriş iletileri daha sonra aynı köprü olanağı simgesini kullanabilir.

Aşağıdaki özel değer tanımlıdır:

##### **FCNONE**

BVT simgesi belirtilmedi.

Bu hem bir istek, hem de yalnızca 3270 işlemleri için kullanılan bir yanıt alanıdır. Bu alanın uzunluğu LNFAC tarafından verilir. Bu alanın ilk değeri FCNONE 'tır.

#### **CIFKT (10 basamaklı işaretli tamsayı)**

Köprü tesisi serbest bırakma zamanı.

Bu, kullanıcı işlemi sona erdikten sonra köprü tesisinin alıkonacağı saniye cinsinden süre uzunluğudur. Etkileşimli olmayan işlemler için değer sıfır olmalıdır.

Bu, yalnızca 3270 işlemleri için kullanılan bir istek alanıdır. Bu alanın ilk değeri 0 'tır.

#### **CIFL (4 baytlık karakter dizilimi)**

Uçbirim öykünülen öznitelikleri.

Bu, köprü tesisine model olarak kullanılmak üzere kurulmuş bir uçbirimin adıdır. A value of blanks means that *CIFL* is taken from the bridge transaction profile definition, or a default value is used.

Bu, yalnızca 3270 işlemleri için kullanılan bir istek alanıdır. Bu alanın uzunluğu LNFACL tarafından verilir. Bu alanın ilk değeri 4 boşluktur.

### **CIFLG (10 basamaklı işaretli tamsayı)**

Bayraklar.

Değer şu olmalıdır:

#### **DOSYA Dışı**

Bayrak yok.

Bu bir istek alanıdır. Bu alanın ilk değeri CIFON değeridir.

### **CFMT (8 baytlık karakter dizilimi)**

IBM MQ format name of data that follows MQCIH.

Bu, MQCIH yapısını izleyen verilerin IBM MQ biçim adını belirtir.

MQPUT ya da MQPUT1 çağrısında, uygulama bu alanı veri için uygun değere ayarlamalıdır. Bu alanı kodlamaya ilişkin kurallar, MQMD 'deki *MDFMT* alanı için yapılan kurallarla aynıdır.

*CIRFM* alanında FMNONE değeri varsa, yanıt iletisi için bu biçim adı da kullanılır.

- DPL istekleri için *CIFMT* , COMMAREA 'nın biçim adı olmalıdır.
- 3270 istekleri için *CIFMT* , CSQCBDCI olmalı ve *CIRFM* CSQCBDCO olmalıdır.

Bu biçimlere ilişkin veri dönüştürme çıkışları, çalıştırılacağı kuyruk yöneticisine kurulmalıdır.

İstek iletisi, bir hata yanıtı iletisinde sonuçlanırsa, hata yanıtı iletisinin biçim adı FMSTR olur.

Bu bir istek alanıdır. Bu alanın uzunluğu LNFMT tarafından verilir. Bu alanın ilk değeri FMNONE 'dir.

### **CIFNC (4 baytlık karakter dizilimi)**

IBM MQ çağrı adı ya da CICS EIBFN işlevi.

Bu alanda döndürülen değer *CIRET* 'a bağlıdır; bkz. [Çizelge 691 sayfa 1018](#). *CIFNC* bir IBM MQ arama adı içerdiğinde aşağıdaki değerler kullanılabilir:

#### **CFCONN**

MQCONN çağrısı.

#### **CFGET**

MQGET çağrısı.

#### **CİNGER**

MQINQ çağrısı.

#### **CFAç**

MQOPEN çağrısı.

#### **CFPUT**

MQPUT çağrısı.

#### **CFPUT1**

MQPUT1 çağrısı.

#### **CFYOK**

Arama yok.

Bu bir yanıt alanıdır. Bu alanın uzunluğu LNFUNC tarafından verilir. Bu alanın ilk değeri CFNONE 'dir.

### **CIIGWI (10 basamaklı işaretli tamsayı)**

Köprü görevi tarafından MQGET çağrısına ilişkin bekleme aralığı.

Bu alan yalnızca *CIUOW* 'in CUFIRST değerine sahip olduğunda geçerlidir. Bu ileti, gönderme uygulamasının, köprü tarafından yayınlanan MQGET çağrılarının, bu ileti tarafından başlatılan iş birimine ilişkin ikinci ve sonraki istek iletilerini beklemesi için gereken yaklaşık süreyi milisaniye cinsinden belirtmesini sağlar. Bu, köprü tarafından kullanılan varsayılan bekleme aralığını geçersiz kılar. Aşağıdaki özel değerler kullanılabilir:

**WIDFLT**

Varsayılan bekleme aralığı.

Bu, CICS bridge ' in köprü başlatıldığında belirtilen dönemi beklemesine neden olur.

**WIULIM**

Sınırsız bekleme aralığı.

Bu bir istek alanıdır. Bu alanın ilk değeri WIDFLT ' dir.

**CIII (10 basamaklı işaretli tamsayı)**

Ayrılmış.

Bu ayrılmış bir alandır. Değer 0 olmalıdır. CIVER , CIVER2değerinden küçükse bu alan yoktur.

**CILEN (10 basamaklı işaretli tamsayı)**

MQCIH yapısının uzunluğu.

Değer aşağıdakilerden biri olmalıdır:

**CILEN1**

version-1 CICS bilgi üstbilgisi yapısının uzunluğu.

**CILEN2**

version-2 CICS bilgi üstbilgisi yapısının uzunluğu.

Aşağıdaki değişmez, yürürlükteki sürümün uzunluğunu belirtir:

**CILENC**

CICS bilgi üstbilgisi yapısının geçerli sürümünün uzunluğu.

Bu bir istek alanıdır. Bu alanın ilk değeri CILEN2' dir.

**CILT (10 basamaklı işaretli tamsayı)**

Link tipi.

Bu, köprünün bağlantı oluşturmak için denediği nesnenin tipini belirtir. Değer aşağıdakilerden biri olmalıdır:

**TLTPROG**

DPL programı.

**LTTRAN**

3270 hareketi.

Bu bir istek alanıdır. Bu alanın ilk değeri LTPROG ' dir.

**CINTI (4 baytlık karakter dizilimi)**

Eklenecek sonraki hareket.

Bu ad, kullanıcı hareketi tarafından döndürülen (genellikle EXEC CICS RETURN TRANSID) tarafından döndürülen sonraki işlemin adıdır. Sonraki hareket yoksa, bu alan boşluk karakterine ayarlanır.

Bu, yalnızca 3270 işlemleri için kullanılan bir yanıt alanıdır. Bu alanın uzunluğuna LNTRID değeri verilir. Bu alanın ilk değeri 4 boşluktur.

**CIODL (10 basamaklı işaretli tamsayı)**

Çıkış COMMAREA veri uzunluğu.

Bu, bir yanıt iletisinde istemciye döndürülebilmek için kullanıcı verilerinin uzunluğudur. Bu uzunluk, 8 byte 'lık program adını içerir. Bağlantılı programa geçirilen COMMAREA uzunluğu, bu alanın üst sınırı ve istek iletisinde kullanıcı verilerinin uzunluğunun (eksi 8) uzunluğudur.

**Not:** Bir iletteki kullanıcı verilerinin uzunluğu, MQCIH yapısını *hariç* iletinin uzunluğudur.

İstek iletisinde kullanıcı verilerinin uzunluğu CIODL'dan küçükse, LINK komutunun DATALENGTH seçeneği kullanılır; bu, LINK ' un başka bir CICS bölgesine verimli bir şekilde işlev göndermesine olanak tanır.

Aşağıdaki özel değer kullanılabilir:

#### **OLINT**

Çıkış uzunluğu giriş uzunluğuna göre aynı.

Bağlantılı programa geçirilen COMMAREA ' nın yeterli boyutta olduğundan emin olmak için, yanıt istenirse de bu değer gerekebilir.

Bu, yalnızca DPL programları için kullanılan bir istek alanıdır. Bu alan OLINPT alanının ilk değeri.

#### **CIREA (10 basamaklı işaretli tamsayı)**

IBM MQ neden ya da geribildirim kodu ya da CICS EIBRESP2.

Bu alanda döndürülen değer *CIRET* ' a bağlıdır; bkz. [Çizelge 691 sayfa 1018](#).

Bu bir yanıt alanıdır. Bu alanın ilk değeri RCNONE olur.

#### **CIRET (10 basamaklı işaretli tamsayı)**

Köprüden dönüş kodu.

Bu, köprü tarafından gerçekleştirilen işlemin sonucunu açıklayan CICS bridge ' den gelen dönüş kodudur. *CIFNC*, *CICC*, *CIREAve* *CIAC* alanları ek bilgi içerebilir (bkz. [Çizelge 691 sayfa 1018](#) ). Değer aşağıdakilerden biridir:

##### **CRC000**

(0, X'000 ') Hata yok.

##### **CRC001**

(1, X'001 ') EXEC CICS deyimi bir hata saptadı.

##### **CRC002**

(2, X'002 ') IBM MQ çağrısı bir hata saptadı.

##### **CRC003**

(3, X'003 ') CICS bridge bir hata algıladı.

##### **CRC004**

(4, X'004 ') CICS bridge olağandışı şekilde sona erdi.

##### **CRC005**

(5, X'005 ') Uygulama olağandışı sona erdi.

##### **CRC006**

(6, X'006 ') Güvenlik hatası oluştu.

##### **CRC007**

(7, X'007 ') Program kullanılmıyor.

##### **CRC008**

(8, X'008 ') Yürürlükteki iş birimi içinde ikinci ya da daha sonraki bir ileti belirlenen süre içinde alınmadı.

##### **CRC009**

(9, X'009 ') İşlem kullanılmıyor.

Bu bir yanıt alanıdır. Bu alanın ilk değeri CRC000' dir.

#### **CIRFM (8 baytlık karakter dizilimi)**

Yanıt iletisinin IBM MQ biçim adı.

Bu, yürürlükteki iletiye yanıt olarak gönderilecek yanıt iletisinin IBM MQ biçim adıdır. Bu kodu kodlamaya ilişkin kurallar, MQMD ' deki *MDFMT* alanı için yapılan kurallarla aynıdır.

Bu, yalnızca DPL programları için kullanılan bir istek alanıdır. Bu alanın uzunluğu LNFMT tarafından verilir. Bu alanın ilk değeri FMNONE ' dir.

#### **CIRSI (4 baytlık karakter dizilimi)**

Ayrılmış.

Bu ayrılmış bir alandır. Değer 4 boşluk olmalıdır. Bu alanın uzunluğuna LNRSID değeri verilir.

**CIRS1 (8 baytlık karakter dizgisi)**

Ayrılmış.

Bu ayrılmış bir alandır. Değer 8 boşluk olmalıdır.

**CIRS2 (8 baytlık karakter dizgisi)**

Ayrılmış.

Bu ayrılmış bir alandır. Değer 8 boşluk olmalıdır.

**CIRS3 (8 baytlık karakter dizgisi)**

Ayrılmış.

Bu ayrılmış bir alandır. Değer 8 boşluk olmalıdır.

**CIRS4 (10 basamaklı işaretli tamsayı)**

Ayrılmış.

Bu ayrılmış bir alandır. Değer 0 olmalıdır. *CIVER* , *CIVER2*değerinden küçükse bu alan yoktur.

**CIRTI (4 baytlık karakter dizilimi)**

Ayrılmış.

Bu ayrılmış bir alandır. Değer 4 boşluk olmalıdır. Bu alanın uzunluğuna LNTRID değeri verilir.

**CISC (4 baytlık karakter dizilimi)**

İşlem başlatma kodu.

Bu, köprünün bir uçbirim işlemi mi, yoksa bir START işlemi mi öyküneceğini belirleyen bir göstergedir. Değer aşağıdakilerden biri olmalıdır:

**SSCSTRT**

Başlatma.

**SCDATA**

Verileri başlatın.

**SCTERM**

Girişi sona erdirin.

**SSCU**

Yok.

Köprüden gelen yanıtta, bu alan, *CINTI* alanında bulunan bir sonraki işlem tanıtıcısına uygun başlangıç koduna ayarlanır. Yanıtta aşağıdaki başlangıç kodları olabilir:

- SSCSTRT
- SCDATA
- SCTERM

CICS Transaction Server 1.2 için bu alan yalnızca bir istek alanıdır; yanıtta değeri tanımsız olur.

CICS Transaction Server 1.3 ve sonraki yayınlar için bu hem bir istek, hem de bir yanıt alanıdır.

Bu alan yalnızca 3270 hareketleri için kullanılır. Bu alanın uzunluğu LNSTCO tarafından verilir. Bu alanın ilk değeri SCNONE değeridir.

**CISID (4 baytlık karakter dizilimi)**

Yapı tanıtıcısı.

Değer şu olmalıdır:

**CISIDV**

CICS bilgi üstbilgisi yapısına ilişkin tanıtıcı.

Bu bir istek alanıdır. Bu alanın ilk değeri CISIDV ' dir.

### **CITES (10 basamaklı işaretli tamsayı)**

Görevin sonunda durum.

Bu alan, görev sonunda kullanıcı hareketinin durumunu gösterir. Aşağıdaki değerlerden biri döndürülür:

#### **TENOSY**

Eşitlenmedi.

Kullanıcı hareketi henüz tamamlanmamış ve eşitlenmiş değil. MQMD 'deki *MDMT* alanı, bu durumda *MTRQST* ' dir.

#### **TECMIT**

İşlerin kesinleştirilmesinin birimi.

Kullanıcı işlemi henüz tamamlanmadı, ancak ilk iş birimini eşitlemiştir. MQMD 'deki *MDMT* alanı, bu vakadaki *MTDGRM* ' dir.

#### **ARKA**

Çalışma birimi geri döndü.

Kullanıcı işlemi henüz tamamlanmadı. Yürürlükteki iş birimi geriletilen. MQMD 'deki *MDMT* alanı, bu vakadaki *MTDGRM* ' dir.

#### **TEKNDT**

Görevi sona erdirin.

Kullanıcı işlemi sona erdi (ya da olağandışı sona erdirildi). MQMD 'deki *MDMT* alanı, bu durumda *MTRPLY* ' dir.

Bu, yalnızca 3270 işlemleri için kullanılan bir yanıt alanıdır. Bu alanın ilk değeri TENOSY 'dir.

### **CITI (4 baytlık karakter dizilimi)**

Eklenecek hareket.

*CILT* değeri *LTTRAN* değerine sahipse, *CITI* çalıştırılacak kullanıcı işleminin işlem tanıtıcısıdır; bu durumda boş olmayan bir değer belirtilmelidir.

If *CILT* has the value *LTPROG*, *CITI* is the transaction code under which all programs within the unit of work are to be run. Belirlenen değer boş bırakılırsa, CICS DPL köprüsü varsayılan hareket kodu (*CKBP*) kullanılır. If the value is nonblank, it must have been defined to CICS as a local TRANSACTION with an initial program of *CSQCBP00*. This field is applicable only when *CIUOW* has the value *CUFRST* or *CUONLY*.

Bu bir istek alanıdır. Bu alanın uzunluğuna *LNTRID* değeri verilir. Bu alanın ilk değeri 4 boşluktur.

### **CIUOW (10 basamaklı işaretli tamsayı)**

İş birimi kontrolü.

Bu, CICS bridgetarafından gerçekleştirilen iş birimi işlemlerini denetler. Köprüyü tek bir işlem çalıştırmasını ya da bir iş birimi içinde bir ya da daha fazla program çalıştırmasını isteyebilirsiniz. Bu alan, CICS bridge ' in bir iş birimi başlatması, yürürlükteki iş birimi içinde istenen işlevi gerçekleştirilmesi ya da kesinleştirerek ya da yedeklemesiyle iş birimini sona erdirmesi gerekip gerekmediğini belirtir. Veri iletim akışlarını eniyilemek için çeşitli birleşimler desteklenir.

Değer aşağıdakilerden biri olmalıdır:

#### **SADECE**

İş birimini başlatın, işlev gerçekleştirin, daha sonra çalışma birimini (DPL ve 3270) kesinleştirin.

#### **CUCONT**

Yürürlükteki iş birimine ilişkin ek veriler (yalnızca 3270).

#### **CUFRST**

İş birimini başlatma ve işlev gerçekleştirme (yalnızca DPL).

**CUMIDL**

Yürürlükteki iş birimi içinde işlev gerçekleştir (yalnızca DPL).

**CULAST**

İşlevi gerçekleştir, daha sonra çalışma birimini (yalnızca DPL) kesinleştirin.

**CUCMIT**

İş birimini kesinleştirin (yalnızca DPL).

**GERI DÖN**

Çalışma birimini yedekle (yalnızca DPL).

Bu bir istek alanıdır. Bu alanın ilk değeri CUONLY değeridir.

**CIVER (10 basamaklı işaretli tamsayı)**

Yapı sürüm numarası.

Değer aşağıdakilerden biri olmalıdır:

**CIVER1**

Version-1 CICS bilgi üstbilgisi yapısı.

**CIVER2**

Version-2 CICS bilgi üstbilgisi yapısı.

Yalnızca yapının daha yeni sürümünde var olan alanlar, alanların açıklamalarında olduğu gibi tanımlanır. Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

**CIVERC**

CICS bilgi üstbilgisi yapısının yürürlükteki sürümü.

Bu bir istek alanıdır. Bu alanın ilk değeri CIVER2' dir.

**Başlangıçtaki değerler**

<i>Çizelge 692. MÖCIH içindeki alanların ilk değerleri</i>		
<b>Alan adı</b>	<b>Değişmezin adı</b>	<b>Değişmezin değeri</b>
<i>CISID</i>	CISIDV	'CIH→'
<i>CIVER</i>	CIVER2	2
<i>CILEN</i>	CILEN2	180
<i>CIENC</i>	Yok	0
<i>CICSI</i>	Yok	0
<i>CIFMT</i>	FMNONE	Boşluklar
<i>CIFLG</i>	DOSYA Dışı	0
<i>CIRET</i>	CRC000	0
<i>CICC</i>	CCOK	0
<i>CIREA</i>	YOK	0
<i>CIUOW</i>	SADECE	273
<i>CIGWI</i>	WIDFLT	-2
<i>CILT</i>	TLTPROG	1
<i>CIODL</i>	OLINT	-1
<i>CIFKT</i>	Yok	0
<i>CIADS</i>	YOK	0

Çizelge 692. MÇCIH içindeki alanların ilk değerleri (devamı var)

Alan adı	Değişmezin adı	Değişmezin değeri
CICT	CTNO	0
CITES	TENOSY	0
CIFAC	FCNONE	Boş Değerler
CIFNC	CFYOK	Boşluklar
CIAC	Yok	Boşluklar
CIAUT	Yok	Boşluklar
CIRS1	Yok	Boşluklar
CIRFM	FMNONE	Boşluklar
CIRSI	Yok	Boşluklar
CIRTI	Yok	Boşluklar
CITI	Yok	Boşluklar
CIFL	Yok	Boşluklar
CIAI	Yok	Boşluklar
CISC	SSCU	Boşluklar
CICNC	Yok	Boşluklar
CINTI	Yok	Boşluklar
CIRS2	Yok	Boşluklar
CIRS3	Yok	Boşluklar
CICP	Yok	0
CIE0	Yok	0
CIII	Yok	0
CIRS4	Yok	0

#### Notlar:

1. → simgesi tek bir boş karakteri temsil eder.

#### RPG bildirimi

```

D*..1.....2.....3.....4.....5.....6.....7..
D* MÇCIH Structure
D*
D* Structure identifier
D CISID          1      4    INZ('CIH ')
D* Structure version number
D CIVER          5      8I 0 INZ(2)
D* Length of MÇCIH structure
D CILEN          9     12I 0 INZ(180)
D* Reserved
D CIENC          13     16I 0 INZ(0)
D* Reserved
D CICSI          17     20I 0 INZ(0)
D* MQ format name of data that followsMÇCIH
D CIFMT          21     28    INZ('      ')
D* Flags
D CIFLG          29     32I 0 INZ(0)

```



```

D* Return code from bridge
D CIRET 33 36I 0 INZ(0)
D* MQ completion code or CICSEIBRESP
D CICC 37 40I 0 INZ(0)
D* MQ reason or feedback code, or CICSEIBRESP2
D CIREA 41 44I 0 INZ(0)
D* Unit-of-work control
D CIUOW 45 48I 0 INZ(273)
D* Wait interval for MQGET call issued by bridge task
D CIGWI 49 52I 0 INZ(-2)
D* Link type
D CILT 53 56I 0 INZ(1)
D* Output COMMAREA data length
D CIODL 57 60I 0 INZ(-1)
D* Bridge facility release time
D CIFKT 61 64I 0 INZ(0)
D* Send/receive ADS descriptor
D CIAADS 65 68I 0 INZ(0)
D* Whether task can be conversational
D CICT 69 72I 0 INZ(0)
D* Status at end of task
D CITES 73 76I 0 INZ(0)
D* Bridge facility token
D CIFAC 77 84 INZ('00000000000000-00')
D
D* MQ call name or CICS EIBFNfunction
D CIFNC 85 88 INZ(' ')
D* Abend code
D CIAC 89 92 INZ
D* Password or passticket
D CIAUT 93 100 INZ
D* Reserved
D CIRS1 101 108 INZ
D* MQ format name of reply message
D CIRFM 109 116 INZ(' ')
D* Remote CICS system ID to use
D CIRSI 117 120 INZ
D* CICS RTRANSID to use
D CIRTI 121 124 INZ
D* Transaction to attach
D CITI 125 128 INZ
D* Terminal emulated attributes
D CIFL 129 132 INZ
D* AID key
D CIAI 133 136 INZ
D* Transaction start code
D CISC 137 140 INZ(' ')
D* Abend transaction code
D CICNC 141 144 INZ
D* Next transaction to attach
D CINTI 145 148 INZ
D* Reserved
D CIRS2 149 156 INZ
D* Reserved
D CIRS3 157 164 INZ
D* Cursor position
D CICP 165 168I 0 INZ(0)
D* Offset of error in message
D CIEO 169 172I 0 INZ(0)
D* Reserved
D CIII 173 176I 0 INZ(0)
D* Reserved
D CIRS4 177 180I 0 INZ(0)
D*

```



## IBM üzerinde MQCMHO (İleti işleme seçenekleri yarat)

**MQCMHO** yapısı, uygulamaların ileti tutamaçlarının nasıl yaratıldığını denetleyen seçenekleri belirlemesine olanak sağlar.

### Genel Bakış

#### Amaç

Yapı, **MQCRTMH** çağrısındaki bir giriş parametresidir.

## Karakter kümesi ve kodlama

**MQCMHO** içindeki veriler, uygulamanın uygulama ve kodlamasının (ENNAT) karakter kümesinde yer almalıdır.

- “Alanlar” sayfa 1030
- “Başlangıçtaki değerler” sayfa 1031
- “RPG bildirimi” sayfa 1031

## Alanlar

MQCMHO yapısı aşağıdaki alanları içerir; alanlar alfabetik sırayla anlatılır:

### CMOPT (10 basamaklı işaretli tamsayı)

Aşağıdaki seçeneklerden biri belirlenebilir:

#### CMVAL

Bu ileti tanıtıcısında bir özellik ayarlamak için **MQSETMP** çağrıldığında, özellik adının doğrulanmasını sağlamak için bu özellik adının geçerliliği denetlenir:

- geçersiz karakter içeriyor.
- aşağıdaki durumlar dışında "JMS" ya da "usr.JMS" başlatmaz:
  - JMSCorrelationID
  - JMSReplyTo
  - JMSType
  - JMSXGroupID
  - JMSXGroupSeq

Bu adlar JMS özellikleri için ayrılmıştır.

- aşağıdaki anahtar sözcüklerden biri, üst veya küçük harflerin herhangi bir karışımında yer almaz:
  - "VE"
  - "ARALARı"
  - "KAÇıŞ"
  - "FALSE"
  - "IN"
  - "IS"
  - "BENZER"
  - "DEĞİL"
  - "BOŞ"
  - "YA DA"
  - "TRUE"
- "Ceset" e başlamaz. ya da "Kök." ("Root.MQMDdışında.").

Özellik MQ-defind ("mq. \*") ise ve ad tanındı, özellik tanımlayıcı alanları, özellik için doğru değerlere ayarlanır. Özellik tanınmadıysa, özellik tanımlayıcısının *Support* alanı **PDSUP** olarak ayarlanır (ek bilgi için [PDSUP](#) başlıklı konuya bakın).

#### CMDEFV

Bu, özellik adlarına ilişkin varsayılan geçerlilik denetimi düzeyinin ortaya çıkmasını belirtir.

Varsayılan doğrulama düzeyi, **CMVAL** ile belirtilen değere eşdeğerdir.

Gelecekteki bir yayında, **CMDEFV** tanımlandığında gerçekleştirilecek geçerlilik denetimi düzeyini değiştirecek bir yönetim seçeneği tanımlanabilirdi.

Bu varsayılan değerdir.

### CMNOVA

Özellik adı üzerinde doğrulama gerçekleştirilmez. **CMVAL'** in açıklamasına bakın.

**Varsayılan seçenek:** Bu bölümde önceden anlatılan seçeneklerden hiçbiri gerekmiyorsa, aşağıdaki seçenek kullanılabilir:

### CMYOK

Tüm seçenekler varsayılan değerlerini varsayar. Başka bir seçenek belirtilmemeyi belirtmek için bu değeri kullanın. **CMNONE** , program belgelerine yardımcı olur; bu seçeneğin diğeriyle birlikte kullanılması amaçlanmaz, ancak değeri sıfır olduğu için, bu tür bir kullanım saptanamaz.

Bu her zaman bir giriş alanıdır. Bu alanın başlangıç değeri **CMDEFV'** dir.

### CMSID (10 basamaklı işaretli tamsayı)

Bu, yapı tanıtıcısıdır; değer şöyle olmalıdır:

#### CMSIDV

İleti tanıtıcısı seçenekleri yapısının yaratılmasına ilişkin tanıtıcı.

Bu her zaman bir giriş alanıdır. Bu alanın başlangıç değeri **CMSIDV'** dir.

### CMVER (10 basamaklı işaretli tamsayı)

Bu, yapı sürüm numarasıdır; değer şu olmalıdır:

#### CMVER1

Version-1 , ileti tanıtıcısı seçenekleri yapısını yaratır.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

#### CMVERC

İleti tanıtıcısı yaratma seçenekleri yapısının yürürlükteki sürümü.

Bu her zaman bir giriş alanıdır. Bu alanın başlangıç değeri **CMVER1'** dir.

## Başlangıçtaki değerler

Çizelge 693. MÖCMHO 'daki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
CMSID	CMSIDV	'CMHO'
CMVER	CMVER1	1
CMOPT	CMDEFV	0

## RPG bildirimi

```
D* MÖCMHO Structure
D*
D*
D* Structure identifier
D CMSID          1      4  INZ('CMHO')
D*
D* Structure version number
D CMVER          5      8I 0 INZ(1)
D*
```

## IBM i IBM i üzerinde MQCNO (Bağlantı seçenekleri)

MQCNO yapısı, uygulamanın yerel kuyruk yöneticisine yönelik bağlantıyla ilgili seçenekleri belirtmesine olanak sağlar.

### Genel Bakış

**Amaç:** Yapı, MQCONNX çağrısındaki bir giriş/çıkış değiştirgesidir.

**Sürüm:** MQCNO ' nun yürürlükteki sürümü CNVER6. Yapının yalnızca daha yeni sürümlerinde bulunan alanlar, izleyen açıklamalarda bu şekilde tanımlanır.

Sağlanan COPY dosyası, ortam tarafından desteklenen en son MQCNO sürümünü içeriyor; ancak, CNVER alanının ilk değeri CNVER1olarak ayarlanmış. version-1 yapısında var olmayan alanları kullanmak için uygulamanın CNVER alanını gerekli sürümün sürüm numarasına ayarlaması gerekir.

**Karakter kümesi ve kodlama:** MQCNO içindeki veriler, **CodedCharSetId** kuyruk yöneticisi özneliği ve ENNAT tarafından verilen yerel kuyruk yöneticisinin kodlaması tarafından belirtilen karakter kümesinde olmalıdır.

- [“Alanlar” sayfa 1032](#)
- [“Başlangıçtaki değerler” sayfa 1037](#)
- [“RPG bildirimi” sayfa 1038](#)

### Alanlar

MQCNO yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** açıklanır:

#### CCDTUL (10 basamaklı işaretli tamsayı)

CCDTUL, bağlantı için kullanılacak istemci bağlantı kanalı çizelgesinin yerini tanımlayan bir URL içeren CCDTUP ya da CCDTUO tarafından tanımlanan dizginin uzunluğudur.

CCDTUL ' yi yalnızca MQCONNX çağrılarını yayınlayan uygulama IBM MQ MQI clientolarak çalıştığında kullanın.

Bu, [MQCHLLIB](#) ve [MQCHLTAB](#) ortam değişkenlerini ayarlamaya yönelik programlı bir alternatiftir.

Uygulama istemci olarak çalışmıyorsa, CCDTUL yoksayılr.

CNVER CNVER6değerinden küçükse bu alan yoksayılr.

#### CCDTUO (10 basamaklı işaretli tamsayı)

CCDTUO, MQCNO yapısının başlangıcından, bağlantı için kullanılacak istemci bağlantı kanalı çizelgesinin yerini tanımlayan bir URL içeren bir dizgiye kadar bayt cinsinden görelî konumdur. Görelî konum pozitif ya da negatif olabilir.

CCDTUL ' yi yalnızca MQCONNX çağrılarını yayınlayan uygulama IBM MQ MQI clientolarak çalıştığında kullanın.

**Önemli:** CCDTUP ve CCDTUO ' dan yalnızca birini kullanabilirsiniz. Her iki alan da sıfır değilse, çağrı RC2600 neden koduyla başarısız olur.

Bu, [MQCHLLIB](#) ve [MQCHLTAB](#) ortam değişkenlerini ayarlamaya yönelik programlı bir alternatiftir.

Uygulama istemci olarak çalışmıyorsa, CCDTUO yoksayılr.

CNVER CNVER6değerinden küçükse bu alan yoksayılr.

#### CCDTUP (gösterge)

CCDTUP, bağlantı için kullanılacak istemci bağlantı kanalı çizelgesinin yerini saptamak için, URL içeren bir dizgiye ilişkin isteğe bağlı bir göstergedir.

CCDTUP özelliğini, MQCONNX çağrısı yayınlayan uygulama IBM MQ MQI clientolarak çalıştığında kullanın.

**Önemli:** CCDTUP ve CCDTUO ' dan yalnızca birini kullanabilirsiniz. Her iki alan da sıfır değilse, çağrı RC2600 neden koduyla başarısız olur.

Bu, MQCHLLIB ve MQCHLTAB ortam değişkenlerini ayarlamaya yönelik programlı bir alternatiftir.

Uygulama istemci olarak çalışmıyorsa, CCDTUP yoksayılr.

CNVER CNVER6değerinden küçükse bu alan yoksayılr.

#### **V 9.2.0 CNAN (28 baytlık karakter dizgisi)**

Kuyruk yöneticisiyle bağlantıyı tanımlamak için uygulama tarafından ayarlanan ad. Alanın ilk değeri boş karakterdir.

CNVER CNVER7değerinden küçükse bu alan yoksayılr.

#### **CNCCO (10 basamaklı işaretli tamsayı)**

MQCNO yapısının başlangıcından itibaren bir MQCD kanal tanımlaması yapısının byte cinsinden görelî konumudur.

#### **CNCCP (gösterge)**

Bu, bir MQCD kanal tanımlaması yapısına ilişkin bir işaretçidir.

#### **CNCONID (24 baytlık karakter dizgisi)**

Benzersiz bağlantı tanıtıcısı. Bu alan, kuyruk yöneticisinin kuyruk yöneticisine ilk bağlandığında benzersiz bir tanıtıcı atayarak bir uygulama işlemini güvenilir bir şekilde tanımasını sağlar.

Uygulamalar PUT ve GET çağrıları yaparken ilişkilendirme amacıyla bağlantı tanıtıcısını kullanır. Bağlantı nasıl kurulduğı fark etmez, tüm bağlantılara kuyruk yöneticisi tarafından bir tanıtıcı atanır.

Uzun süreli bir iş biriminin sona ermesini zorlamak için bağlantı tanıtıcısı kullanılabilir. Bunu yapmak için, 'Bağlantıyı Durdur' PCF komutunu ya da STOP CONN MQSC komutunu kullanarak bağlantı tanıtıcısını belirtin. Bu komutların kullanılmasıyla ilgili ek bilgi için ilgili bağlantılara bakın.

Alanın ilk değeri 24 boş bayttır.

#### **CNCT (128 baytlık bit dizgisi)**

Bu, kuyruk yöneticisinin bu bağlantı sırasında uygulamadan etkilenen kaynaklarla ilişkilendirdiğı bir etikettir.

Kuyruk yöneticisi bağlantı etiketi.

Kuyruk yöneticisinin etkilenen kaynaklara erişimi doğru şekilde diziselleştirebilmesi için, her uygulama ya da uygulama yönetim ortamının etiket için farklı bir değer kullanması gerekir. Daha fazla ayrıntı için CN\* CT\* seçeneklerinin açıklamalarına bakın. Uygulama sona erdiğinde ya da MQDISC çağrısı verildiğinde etiket geçerli olmaktan çıkar.

Etiket gerekmiyorsa aşağıdaki özel değeri kullanın:

#### **CTNONE.**

Bağlantı etiketi belirtilmedi.

Değer, alanın uzunluğu için ikili sıfırdır.

Bu bir giriş alanıdır. Bu alanın uzunluğu LNCTAG tarafından verilir. Bu alanın ilk değeri CTNONE. CNVER CNVER3değerinden küçükse bu alan yoksayılr.

Bir z/OS kuyruk yöneticisine bağlanırken ConnTag alanını kullanın.

#### **V 9.2.0 CNNORES2 (4 baytlık karakter dizgisi)**

Yapıyı 64 bitlik bir sınıra çıkarmak için ayrılmış bir alan. Alanın ilk değeri, alanın uzunluğu için ikili sıfırdır.

CNVER CNVER7değerinden küçükse bu alan yoksayılr.

## CNOPT (10 basamaklı işaretli tamsayı)

MQCONNXeylemine denetleyen seçenekler.

### Bağ tanımlama seçenekleri

Bağlama seçenekleri, kullanılan IBM MQ bağ tanımının tipini denetler; bu seçeneklerden yalnızca birini belirtin:

#### CNSBND

Standart bağlama.

Standart bağlama seçeneği, uygulamanın ve yerel kuyruk yöneticisi aracısının genellikle ayrı işlemlerde ayrı yürütme birimlerinde çalışmasına neden olur. Düzenleme, kuyruk yöneticisinin bütünlüğünü korur; yani, kuyruk yöneticisini hatalı programlardan korur.

Uygulamanın tam olarak sınanmamış olabileceği ya da güvenilir ya da güvenilir olmayabileceği durumlarda CNSBND komutunu kullanın. CNSBND varsayılan değerdir.

CNSBND , program belgelerine yardımcı olmak için tanımlanır. Bu seçeneği, kullanılan bağ tanımının tipini denetleyen başka bir seçenekle kullanmayın; ancak değeri sıfır olduğu için, bu tür bir kullanım algılanamaz.

Bu seçenek tüm ortamlarda desteklenir.

#### CNFBND

Hızlı yol bağlama.

Hızlı yol bağlama seçeneği, uygulamanın ve yerel kuyruk yöneticisi aracısının aynı yürütme biriminin bir parçası olmasına neden olur. Hızlı yol, uygulamanın ve yerel kuyruk yöneticisi aracısının ayrı yürütme birimlerinde çalıştığı standart bağ tanımından farklıdır.

Kuyruk yöneticisi bu tip bir bağ tanımını desteklemiyorsa CNFBND yoksayılr; işlem, seçenek belirtilmemiş gibi devam eder.

CNFBND , birden çok sürecin uygulama tarafından kullanılan genel kaynaktan daha fazla kaynak tükettiği durumlarda avantajlı olabilir.Hızlı yol bağ tanımını kullanan bir uygulama, *güvenilir uygulama* olarak bilinir.

Hızlı yol bağlamanın kullanılıp kullanılmayacağına karar verirken aşağıdaki önemli noktaları göz önünde bulundurun:

- **CNFBND seçeneğinin kullanılması, bir uygulamanın iletileri ve kuyruk yöneticisine ait diğer veri alanlarını değiştirmesini ya da bozmasını önlemez. Bu seçeneği yalnızca bu sorunları tam olarak değerlendirdiğiniz durumlarda kullanın.**
- Uygulama, CNFBNDile zamanuyumsuz sinyaller ya da süreölçer kesintileri ( sigkll gibi) kullanmamalıdır. Paylaşılan bellek parçalarının kullanımına ilişkin kısıtlamalar da vardır.
- Uygulamada, kuyruk yöneticisine aynı anda birden çok iş parçacığı bağlı olmamalıdır.
- Uygulamanın kuyruk yöneticisiyle bağlantısını kesmek için MQDISC çağrısı kullanması gerekir.
- Uygulama, kuyruk yöneticisini endmqm komutuyla sona erdirmeden önce bitmelidir.

Aşağıdaki noktalar, belirtilen ortamlarda CNFBND kullanımı için geçerlidir:

- IBM sistemlerinde iş, QMQMADM grubuna ait QMQM kullanıcı tanıtımı altında çalıştırılmalıdır. Ayrıca, program olağandışı sonlanmamalıdır, aksi takdirde beklenmedik sonuçlar ortaya çıkabilir.

Güvenilir uygulamaların kullanılmasının etkilerine ilişkin ek bilgi için [MQCONNX çağrısı kullanılarak kuyruk yöneticisine bağlanma ve Güvenilen uygulamalara ilişkin kısıtlamalar](#) başlıklı konuya bakın.

#### CNSHBD

Paylaşılan Bağ Tanımları.

Paylaşılan bağ tanımları seçeneği, uygulamanın ve yerel kuyruk yöneticisi aracısının genellikle ayrı işlemlerde ayrı yürütme birimlerinde çalışmasına neden olur. Düzenleme, kuyruk yöneticisinin bütünlüğünü korur; yani, kuyruk yöneticisini hatalı programlardan korur. Ancak, bazı kaynaklar uygulama ile yerel kuyruk yöneticisi aracı arasında paylaşılır. Kuyruk yöneticisi bu tip bir bağ tanımını desteklemiyorsa CNSHBD yoksayılar. İşlem, seçenek belirtilmemiş gibi devam eder.

### **CNIBND**

Yalıtılmış Bağ Tanımları.

Yalıtılmış bağ tanımları seçeneği, uygulamanın ve yerel kuyruk yöneticisi aracısının genellikle ayrı işlemlerde ayrı yürütme birimlerinde çalışmasına neden olur. Düzenleme, kuyruk yöneticisinin bütünlüğünü korur; yani, kuyruk yöneticisini hatalı programlardan korur. Uygulama süreci ve yerel kuyruk yöneticisi aracı, kaynakları paylaşmadıkları için birbirinden yalıtılır. Kuyruk yöneticisi bu tip bir bağ tanımını desteklemiyorsa, CNIBND yoksayılar. İşlem, seçenek belirtilmemiş gibi devam eder.

### **İşle paylaşma seçenekleri**

Aşağıdaki seçenekler, aynı işlem içindeki farklı iş parçacıkları (koşut işleme birimleri) arasındaki çekme noktalarının paylaşılmasını denetler. Bu seçeneklerden yalnızca biri belirtilebilir.

#### **CNHSN**

İş parçacıkları arasında tanıtıcı paylaşımı yok.

İş parçacıkları arasındaki tanıtıcı paylaşımı yok seçeneği, bağlantı ve nesne tanıtıcılarının yalnızca tutamacın ayrılmasına neden olan iş parçacığı tarafından kullanılabilmesini; diğer bir deyişle, MQCONN, MQCONNXya da MQOPEN çağrısını yayınlayan iş parçacığı tarafından kullanılabilmesini belirtir. Tutamaçlar, aynı işleme ait diğer iş parçacıkları tarafından kullanılamaz.

#### **CNHSB**

Arama bloklamasıyla, iş parçacıkları arasında dizisel tutamaç paylaşımı.

İş parçacıkları arasındaki dizisel tutamaç paylaşımı (çağrı engelleme ile) seçeneği, bir işlemin bir iş parçacığı tarafından ayrılan bağlantı ve nesne tanıtıcılarının aynı işleme ait diğer iş parçacıkları tarafından kullanılabilmesini gösterir. Ancak, aynı anda yalnızca bir iş parçacığı belirli bir tutamacı kullanabilir; yani, yalnızca bir tutamacın dizisel kullanımına izin verilir. Bir iş parçacığı, başka bir iş parçacığı tarafından kullanılmakta olan bir tanıtıcıyı kullanmayı denerse, çağrı blokları (tutamaç kullanılabilir oluncaya kadar bekler).

#### **CNHSNB**

Arama engellemesi olmadan, iş parçacıkları arasında dizisel tutamaç paylaşımı.

Arama engellemesi olmadan, iş parçacıkları arasındaki dizisel tutamaç paylaşımı seçeneği, " ile blocking " seçeneği dışında, tutamaç başka bir iş parçacığı tarafından kullanılırsa, çağrı tutamaç kullanılabilir oluncaya kadar engelleme yerine CCFAIL ve RC2219 ile hemen tamamlanır.

Bir iş parçacığının sıfır ya da bir paylaşılmayan tanıtıcısı ve sıfır ya da daha fazla paylaşılan tanıtıcısı olabilir:

- CNHSN belirten her bir MQCONN ya da MQCONNX çağrısı, ilk çağrıda yeni bir paylaşılmayan tanıtıcı ve sonraki çağrılarda aynı paylaşılmayan tanıtıcı döndürür (MQDISC çağrısını müdahalede bulunmadığı varsayılarak). Neden kodu, ikinci ve sonraki çağrılar için RC2002 ' dir.
- CNHSB ya da CNHSNB belirten her bir MQCONNX çağrısı, her çağrıda yeni bir paylaşılan tanıtıcı döndürür.

Nesne tanıtıcıları, nesne tanıtıcısını yaratan MQOPEN çağrısında belirtilen bağlantı tanıtıcısıyla aynı paylaşım özelliklerini devralır. Ayrıca, iş birimleri, iş birimini başlatmak için kullanılan bağlantı tanıtıcısıyla aynı paylaşım özelliklerini devralır; iş birimi, paylaşılan bir tanıtıcı

kullanılarak bir iş parçacığında başlatılırsa, iş birimi aynı tanıtıcı kullanılarak başka bir iş parçacığında güncellenebilir.

Bir tanıtıcı paylaşım seçeneği belirlemezseniz, varsayılan değer ortam tarafından belirlenir:

- Microsoft Transaction Server (MTS) ortamında varsayılan değer CNHSBile aynıdır.
- Diğer ortamlarda, varsayılan değer CNHSNile aynıdır.

### **Yeniden bağlanma seçenekleri**

Yeniden bağlanma seçenekleri, bir bağlantının yeniden bağlanabilir olup olmadığını belirler. Yalnızca istemci bağlantıları yeniden bağlanabilir.

#### **CNRCDF**

Yeniden bağlanma seçeneği varsayılan değerine çözülür. Varsayılan değer ayarlanmazsa, bu seçeneğin değeri DEVRE Dışı olarak çözülür. Seçeneğin değeri sunucuya geçirilir ve **PCF** ve **MQSC** tarafından sorgulanabilir.

#### **CNRC**

Uygulama, MQCONNX **QMNAME** değiştirgesinin değeriyle tutarlı herhangi bir kuyruk yöneticisine yeniden bağlanabilir. CNRC seçeneğini yalnızca istemci uygulaması ile başlangıçta bağlantı kurduğu kuyruk yöneticisi arasında benzerlik yoksa kullanın. Seçeneğin değeri sunucuya geçirilir ve **PCF** ve **MQSC** tarafından sorgulanabilir.

#### **CNRC D**

Uygulama yeniden bağlanamıyor. Seçeneğin değeri sunucuya iletilmedi.

#### **CNRCQM**

Uygulama yalnızca, başlangıçta bağlandığı kuyruk yöneticisine yeniden bağlanabilir. Bir istemci yeniden bağlanabiliyorsa, ancak istemci uygulaması ile başlangıçta bağlantı kurduğu kuyruk yöneticisi arasında bir benzerlik varsa bu değeri kullanın. Bir istemcinin yüksek kullanılabilirlikli bir kuyruk yöneticisinin yedek yönetim ortamına otomatik olarak yeniden bağlanmasını istiyorsanız bu değeri seçin. Seçeneğin değeri sunucuya geçirilir ve **PCF** ve **MQSC** tarafından sorgulanabilir.

Yalnızca istemci bağlantıları için CNRC, CNRC D ve CNRCQM seçeneklerini kullanın. Seçenekler bir bağ tanımlama bağlantısı için kullanılıyorsa, MQCONNX tamamlanma kodu MQCC\_FAILED ve neden kodu MQRC\_OPTIONS\_ERROR ile başarısız olur.

**Varsayılan seçenek:** Açıklanan seçeneklerin hiçbiri gerekmiyorsa, aşağıdaki seçenek kullanılabilir:

#### **CNNONE**

Seçenek belirtilmedi.

CNNONE , program belgelerine yardımcı olmak için tanımlanır. Bu seçeneğin başka bir CN\* seçeneğiyle birlikte kullanılması amaçlanmamıştır, ancak değeri sıfır olduğu için bu tür bir kullanım algılanamaz.

### **CNSCO (10 basamaklı işaretli tamsayı)**

MQCNO yapısının başlangıcından itibaren bir MQSCO yapısının byte cinsinden görelî konumu.

CNVER CNVER4 değerinden küçükse bu alan yoksayılr.

### **CNSCP (gösterge)**

Bu, bir MQSCO yapısının adresidir.

CNVER CNVER4 değerinden küçükse bu alan yoksayılr.

### **CNSECPO (10 basamaklı işaretli tamsayı)**

Güvenlik parametreleri görelî konumu. Bir kullanıcı kimliği ve parola belirtmek için kullanılan MQCSP yapısının görelî konumu.

Değer pozitif ya da negatif olabilir. Bu alanın ilk değeri 0 'dır.

CNVER CNVER5 değerinden küçükse bu alan yoksayılr.



## CNSECPP (gösterge)

Güvenlik deęiřtirgeleri göstergesi. Kullanıcı kimlięi ve parola belirtmek için kullanılan MQCSP yapısının adresi.

Bu alanın ilk deęeri boş deęerli bir gösterge ya da boş deęerli byte.

CNVER CNVER5deęerinden küçükse bu alan yoksayılr.

## CNSID (4 baytlık karakter dizgisi)

MQCNO yapısına iliřkin yapı tanıtıcısı.

Deęer řöyle olmalıdır:

### CNSIDV

Baęlantı seęenekleri yapısına iliřkin tanıtıcı.

Bu her zaman bir giriş alanıdır. Bu alanın ilk deęeri CNSIDV 'dir.

## CNVER (10 basamaklı iřaretli tamsayı)

MQCNO yapısına iliřkin yapı sürümü numarası.

Deęer řöyle olmalıdır:

### CNVER6

Version-6 baęlantı seęenekleri yapısı.

Bu sürüm tüm ortamlarda desteklenir.

### V 9.2.0 CNVER7

Version-7 baęlantı seęenekleri yapısı.

Bu sürüm tüm ortamlarda desteklenir.

Ařaęıdaki deęiřmez, yürürlükteki sürümün sürüm numarasını belirtir:

### CNVERC

Baęlantı seęenekleri yapısının yürürlükteki sürümü.

V 9.2.0 Bu her zaman bir giriş alanıdır. Bu alanın ilk deęeri CNVER7' dir.

## Başlangıçtaki deęerler

Çizelge 694. MQCNO 'daki alanların başlangıç deęerleri		
Alan adı	Deęiřmezin adı	Deęiřmezin deęeri
CNSID	CNSIDV	' CNO-
CNVER	CNVER5	1
CNOPT	CNNONE	0
CNCCO	Yok	0
CNCCP	Yok	Boş deęerli gösterge ya da boş deęerli byte
CNCT	CTNONE.	Boş Deęerler
CNSCP	Yok	Boş deęerli gösterge ya da boş deęerli byte
CNSCO	Yok	0
CNCONID	Yok	Boş Deęerler
CNSECPO	Yok	0

Çizelge 694. MQCNO 'daki alanların başlangıç değerleri (devamı var)

Alan adı	Değişmezin adı	Değişmezin değeri
CNSECPP	Yok	Boş değerli gösterge ya da boş değerli byte
CCDTUL	Yok	0
CCDTUO	Yok	0
CCDTUP	Yok	Boş değerli gösterge ya da boş değerli byte

#### Notlar:

1. → simgesi tek bir boş karakteri temsil eder.

#### RPG bildirimi

```

D*****
D**                                     **
D**          IBM MQ for IBM i          **
D**                                     **
D** FILE NAME:      CMQCNOG           **
D**                                     **
D** DESCRIPTION:    MQCNO Structure -- Connect Options **
D**                                     **
D*****
D** <N_OCO_COPYRIGHT>                  **
D** Licensed Materials - Property of IBM **
D**                                     **
D** 5724-H72                            **
D** (c) Copyright IBM Corp. 1993, 2024. All Rights Reserved. **
D**                                     **
D** US Government Users Restricted Rights - Use, duplication or **
D** disclosure restricted by GSA ADP Schedule Contract with **
D** IBM Corp.                            **
D** <NOC_COPYRIGHT>                    **
D*****
D** FUNCTION:        This file declares the structure MQCNO, **
D**                  which is used by the main MQI.         **
D**                                     **
D** PROCESSOR:      RPG (ILE)           **
D**                                     **
D*****
D*
D*
D*****
D** <BEGIN_BUILDINFO>                  **
D** Generated on:   08/02/16 13:50      **
D** Build Level:    L000000            **
D** Build Type:     Production          **
D** Pointer Size:   128 Bit             **
D** Source File:    **
D** CMQCNOG         **
D** <END_BUILDINFO>                    **
D*****
D*
D*.1....:....2....:....3....:....4....:....5....:....6....:....7..
D*
D*
D* MQCNO Structure
D*
D* Structure identifier
D CNSID          1          4      INZ('CNO ')
D* Structure version number
D CNVER          5          8I 0 INZ(1)
D* Options that control the action of MQCONN
D CNOPT          9          12I 0 INZ(0)
D* Ver:1 **
D* Offset of MQCD structure for client connection
D CNCCO          13         16I 0 INZ(0)
D* Address of MQCD structure for client connection
D CNCCP          17         32*   INZ(*NULL)

```

```

D* Ver:2 **
D* Queue managerconnection tag
D  CNCT          33    160    INZ(X'0000000000000000-
D                                     00000000000000000000-
D                                     00000000000000000000-
D                                     00000000000000000000-
D                                     00000000000000000000-
D                                     00000000000000000000-
D                                     00000000000000000000-
D                                     00000000000000000000-
D                                     00000000000000000000-
D                                     00000000000000000000-
D                                     00000000000000000000-
D                                     00000000000000000000-
D                                     00000000000000000000')
D* Ver:3 **
D* Address of MQSCO structure for client connection
D  CNSCP          161    176*   INZ(*NULL)
D* Offset of MQSCO structure for client connection
D  CNSCO          177    180I 0 INZ(0)
D* Ver:4 **
D* Unique Connection Identifier
D  CNCONID        181    204    INZ(X'0000000000000000-
D                                     00000000000000000000-
D                                     000000')
D* Offset of MQCSP structure
D  CNSECP0        205    208I 0 INZ(0)
D* Address of MQCSP structure
D  CNSECPP        209    224*   INZ(*NULL)
D* Ver:5 **
D* Address of CCDT URL string
D  CNCCDTUP       225    240*   INZ(*NULL)
D* Offset of CCDT URL string
D  CNCCDTU0       241    244I 0 INZ(0)
D* Length of CCDT URL
D  CNCCDTUL       245    248I 0 INZ(0)
D* Ver:6 **
D*
D*****
D** End of CMQCNUG **
D*****

```

## IBM i IBM üzerinde MQCSP (güvenlik deęiřtirgeleri)

IBM için MQCSP yapısına iliřkin özet.

### Genel Bakıř

**Amaç:** MQCSP yapısı, yetki hizmetinin bir kullanıcı kimlięi ve parola doęrulamasına olanak saęlar. MQCONNX çağırısında MQCSP baęlantı güvenlięi deęiřtirgeleri yapısını belirtmenizi saęlar.

**Karakter kümesi ve kodlama:** Data in MQCSP must be in the character set given by the **CodedCharSetId** queue manager attribute and encoding of the local queue manager given by ENNAT.

- [“Alanlar” sayfa 1039](#)
- [“Bařlangıçtaki deęerler” sayfa 1041](#)
- [“RPG bildirimini” sayfa 1041](#)

### Alanlar

MQCSP yapısı ařaęıdaki alanları ierir; alanlar **alfabetik sırayla** anlatılır:

#### CSAUTH (10 basamaklı iřaretili tamsayı)

Bu, gerekleřtirilecek kimlik doęrulamasının tipidir.

Geerli deęerler řunlardır:

#### CSAN

Kullanıcı kimlięi ve parola alanlarını kullanmayın.

#### CSAUIAP

Kullanıcı kimlięi ve parola alanlarını doęrulayın.

Bu bir giriş alanıdır. Bu alanın ilk değeri CSAN ' dir.

#### **CSCPPL (10 basamaklı işaretli tamsayı)**

Bu, kimlik doğrulamasında kullanılacak parola uzunluğudur.

Parola uzunluğu üst sınırı altyapıya bağımlı değildir. Parolanın uzunluğu izin verilenden büyükse, kimlik doğrulama isteği bir RC2035 ile başarısız olur.

Bu bir giriş alanıdır. Bu alanın ilk değeri 0 'tır.

#### **CSCPPO (10 basamaklı işaretli tamsayı)**

Bu, kimlik doğrulamasında kullanılacak parolanın bayt cinsinden görelidir.

Görelidir konum pozitif ya da negatif olabilir.

Bu bir giriş alanıdır. Bu alanın ilk değeri 0 'tır.

#### **CSCPPP (işaretçi)**

Bu, kimlik doğrulamasında kullanılacak parolanın adresidir.

Bu bir giriş alanıdır. Bu alanın başlangıç değeri, boş değerli göstergedir.

#### **CSCSPUIL (10 basamaklı işaretli tamsayı)**

Bu, kimlik doğrulamasında kullanılacak kullanıcı kimliğinin uzunluğudur.

Kullanıcı kimliğinin uzunluk üst sınırı altyapıya bağımlı değildir. Kullanıcı kimliğinin uzunluğu izin verilenden büyükse, kimlik doğrulama isteği bir RC2035 ile başarısız olur.

Bu bir giriş alanıdır. Bu alanın ilk değeri 0 'tır.

#### **CSCSPUIO (10 basamaklı işaretli tamsayı)**

Bu, kimlik doğrulamasında kullanılacak kullanıcı kimliğinin bayt cinsinden görelidir.

Görelidir konum pozitif ya da negatif olabilir.

Bu bir giriş alanıdır. Bu alanın ilk değeri 0 'tır.

#### **CSCSPUIP (işaretçi)**

Bu, kimlik doğrulamasında kullanılacak kullanıcı kimliğinin adresidir.

Bu bir giriş alanıdır. Bu alanın başlangıç değeri, boş değerli göstergedir. CSVER değeri CSVER5 değerinden küçükse bu alan dikkate alınmaz.

#### **CSRE1 (4 baytlık karakter dizilimi)**

IBM üzerinde işaretçi hizalaması için gereken ayrılmış bir alan.

Bu bir giriş alanıdır. Bu alanın ilk değeri boş değerli.

#### **CSRS2 (8 baytlık karakter dizilimi)**

IBM üzerinde işaretçi hizalaması için gereken ayrılmış bir alan.

Bu bir giriş alanıdır. Bu alanın ilk değeri boş değerli.

#### **CSSID (4 baytlık karakter dizilimi)**

Yapı tanıtıcısı.

Değer şu olmalıdır:

#### **CSSIDV**

Güvenlik parametreleri yapısına ilişkin tanıtıcı.

#### **CSVER (10 basamaklı işaretli tamsayı)**

Yapı sürüm numarası.

Değer şu olmalıdır:

## CSVVER1

Version-1 güvenlik parametreleri yapısı.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

## CSVVERC

Güvenlik değiştirgeleri yapısının yürürlükteki sürümü.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri CSVVER1' dir.

## Başlangıçtaki değerler

Çizelge 695. MQCNO içindeki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
CSSID	CSSIDV	' CSP↵ '
CSVVER	CSVVER1	1
CSAUTH	Yok	0
CSRE1	Yok	Boş Değerler
CSCSPUIP	Yok	Boş değerli gösterge
CSCSPUIO	Yok	0
CSCSPUIL	Yok	0
CSRS2	Yok	Boş Değerler
CSCPPP	Yok	Boş değerli gösterge
CSCPP0	Yok	0
CSCPPL	Yok	0

### Not:

1. ↵ simgesi tek bir boş karakteri temsil eder.

## RPG bildirim

```
D*..1.....2.....3.....4.....5.....6.....7..
D*
D* MQCSP Structure
D*
D* Structure identifier
D CSSID 1 4 INZ('CSP ')
D* Structure version number
D CSVVER 5 8I 0 INZ(1)
D* Type of authentication
D CSAUTH 9 12I 0 INZ(0)
D* Reserved
D CSRE1 13 16 INZ(X'00000000')
D* Address of user ID
D CSCSPUIP 17 32* INZ(*NULL)
D* Offset of user ID
D CSCSPUIO 33 36I 0 INZ(0)
D* Length of user ID
D CSCSPUIL 37 40I 0 INZ(0)
D* Reserved
D CSRS2 41 48 INZ(X'0000000000000000')
D* Address of password
D CSCPPP 49 64* INZ(*NULL)
D* Offset of password
D CSCPP0 65 68I 0 INZ(0)
D* Length of password
D CSCPPL 69 72I 0 INZ(0)
```

Denetim geri bildirme işlevini belirten yapı.

## Genel Bakış

### Amaç

MQCTLO yapısı, denetim çağrılarını işleviyle ilgili seçenekleri belirlemek için kullanılır.

Yapı, [MQCTL](#) çağrısındaki bir giriş ve çıkış parametresidir.

### S\u00fcr\u00fcm

MQCTLO 'nun yürürlükteki sürümü CTLV1' dir.

### Karakter kümesi ve kodlama

Data in MQCTLO must be in the character set given by the **CodedCharSetId** queue manager attribute and encoding of the local queue manager given by ENNAT. Ancak, uygulama bir IBM MQ istemcisi olarak çalışıyorsa, yapı, istemcinin karakter kümesinde ve kodlamasında olmalıdır.

- “Alanlar” sayfa 1042
- “Başlangıçtaki değerler” sayfa 1043
- “RPG bildirim” sayfa 1043

## Alanlar

MQCTLO yapısı aşağıdaki alanları içerir; alanlar alfabetik sırayla anlatılır:

### COCONNARA (10 basamaklı işaretli tamsayı)

Denetim seçenekleri yapısı- ConnectionArea alanı.

Bu alan, geri bildirme işlevi için kullanılabilir bir alandır.

Kuyruk yöneticisi, bu alanın içeriğine dayalı olarak hiçbir karar vermez ve bu, MQCB çağrısındaki bir parametre olan MQCBC yapısındaki [CBCCONNAREA](#) alanından değiştirilmeden geçirilir.

Bu alan, CTLSR ve CTLSW dışındaki tüm işlemler için dikkate alınmaz.

Bu, geri bildirme işlevi için bir giriş ve çıkış alanıdır. Bu alanın başlangıç değeri, boş değerli ya da boş değerli bayttır.

### COOPT (10 basamaklı işaretli tamsayı)

MQCTLO ' nun işlemini denetleyen seçenekler.

#### CTLFQ

Kuyruk yöneticisi ya da bağlantı susturma durumdaysa, MQCTLO çağrısını başarısız olacak şekilde zorlayın.

MQCB çağrısında geçirilen MQGMO seçeneklerinde, susturma sırasında ileti tüketicilerine bildirim yol açabilmek için GMFIQ değerini belirtin.

#### CTLTHR

Bu seçenek, uygulamanın, aynı iş parçacığının üzerinde aynı bağlantı için tüm ileti tüketicilerinin çağrılmasını gerektirdiğini bildirir.

**Varsayılan seçenek:** Açıklanan seçeneklerden herhangi birine gereksiniminiz yoksa, aşağıdaki seçeneği kullanın:

#### CCTLNO

Başka bir seçenek belirtilmemeyi belirtmek için bu değeri kullanın; tüm seçenekler varsayılan değerlerini kabul eder. CTLNO, program belgelerine yardımcı olacak şekilde tanımlıdır; bu seçeneğin başka biriyle birlikte kullanılması amaçlanmaz, ancak değeri sıfır olduğu için, bu tür bir kullanım saptanamaz.

Bu bir giriş alanıdır. *COOPT* alanının ilk değeri CTLNO değerine sahip olur.

### **CORSV (10 basamaklı işaretli tamsayı)**

Bu ayrılmış bir alandır. Bu alanın ilk değeri boşluk karakteridir.

### **COSID (10 basamaklı işaretli tamsayı)**

Denetim seçenekleri yapısı- StrucId alanı.

Bu, yapı tanıtıcısıdır; değer şöyle olmalıdır:

#### **CTLSI**

Denetim Seçenekleri yapısına ilişkin tanıtıcı.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri CTLSI 'dir.

### **COVER (10 basamaklı işaretli tamsayı)**

Denetim seçenekleri yapısı-Sürüm alanı.

Bu, yapı sürüm numarasıdır; değer şu olmalıdır:

#### **CTLV1**

Version-1 Denetim seçenekleri yapısı.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

#### **CTLCV**

Denetim seçenekleri yapısının yürürlükteki sürümü.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri CTLV1' dir.

## **Başlangıçtaki değerler**

<i>Çizelge 696. MÖCTLO içindeki alanların ilk değerleri</i>		
<b>Alan adı</b>	<b>Değişmezin adı</b>	<b>Değişmezin değeri</b>
<i>COSID</i>	CTLSI	'CTLO'
<i>COVER</i>	CTLV1	1
<i>COOPT</i>	CCTLNO	Boş Değerler
<i>CORSV</i>	Ayrılmış alan	
<i>COCONNAREA</i>	Yok	Boş değerli gösterge ya da boş değerli byte

## **RPG bildirimi**

```
D* MÖCTLO Structure
D*
D*
D* Structure identifier
D COSID          1      4  INZ('CTLO')
D*
D* Structure version number
D COVER          5      8I 0 INZ(1)
D*
D* Options that control the action of MÖCTL
D COOPT          9      12I 0 INZ(0)
D*
D* Reserved
D CORSV         13     16I 0 INZ(-1)
D*
D* MÖCTL Data area passed to the function
D COCONNAREA    17     32*  INZ(*NULL)
```

MQDH yapısı, ileti bir iletim kuyruğunda saklanan bir dağıtım listesi ileti olduğunda, bir iletime var olan ek verileri açıklar.

## Genel Bakış

**Amaç:** Dağıtım listesi ileti, birden çok hedef kuyruğuna gönderilen bir iletidir. Ek veriler, MQDH yapısından ve ardından bir MQOR kaydı dizisine ve bir MQPMR kayıtlarından oluşan bir diziye sahiptir.

Bu yapı, iletileri doğrudan iletim kuyruklarına koyan ya da iletim kuyruklarından iletilerin kaldırıldığı (örneğin, ileti kanalı araçları) özel uygulamalar tarafından kullanılmak üzere kullanılır.

Bu yapı, yalnızca dağıtım listelerine ileti koymak isteyen normal uygulamalar tarafından kullanılmalıdır. Bu uygulamalar, dağıtım listesindeki hedefleri tanımlamak için MQOD yapısını ve ileti özelliklerini belirtmek için MQPMO yapısını kullanmalı ya da tek tek hedeflere gönderilen iletilerle ilgili bilgileri almak için kullanılmalıdır.

**Karakter kümesi ve kodlama:** MQDH 'deki veriler, C programlama dili için ENNAT tarafından verilen yerel kuyruk yöneticisinin kodlaması ve **CodedCharSetId** kuyruk yöneticisi özniteliği tarafından verilen karakter kümesinde olmalıdır.

The character set and encoding of the MQDH must be set into the *MDCSI* and *MDENC* fields in:

- MQMD (MQDH yapısı ileti verilerinin başlangıcındaki ise) ya da
- MQDH yapısından önce gelen üstbilgi yapısı (diğer tüm vakalar).

**Kullanım:** Bir uygulama bir dağıtım listesine ileti yerleştirdiğinde ve hedeflerin bazıları ya da tümü uzaksa, kuyruk yöneticisi, uygulama ileti verilerini MQXQH ve MQDH yapılarıyla örnekler ve iletiyi ilgili iletim kuyruğuna yerleştirir. Bu nedenle, ileti bir iletim kuyruğunda olduğunda, aşağıdaki sırada yer alan veriler görüntülenir:

- MQXQH yapısı
- MQDH yapısı artı MQOR ve MQPMR kayıtları dizileri
- Uygulama ileti verileri

Variş noktalarına bağlı olarak, kuyruk yöneticisi tarafından birden çok ileti oluşturulabilir ve farklı iletim kuyruklarına yerleştirilecektir. Bu durumda, bu iletilerdeki MQDH yapıları, uygulama tarafından açılan dağıtım listesi tarafından tanımlanan hedeflerin farklı alt kümelerini tanımlar.

Dağıtım listesi iletilerini doğrudan bir iletim kuyruğuna yerleştiren bir uygulama, önceden açıklanan sıraya uymalıdır ve MQDH yapısının doğru olduğundan emin olmalıdır. MQDH yapısı geçerli değilse, kuyruk yöneticisi MQPUT ya da MQPUT1 çağrısının başarısız olması için neden kodu RC2135 ile başarısız olabilir.

İletiler dağıtım listesi formundaki bir kuyrukta saklanabilir; ancak, kuyruk dağıtım listesi iletilerini destekleyebilecek şekilde tanımlandıysa ( "Kuyruklara ilişkin öznitelikler" sayfa 1345 içinde açıklanan **DistLists** kuyruğu özniteliğine bakın). Bir uygulama dağıtım listesi iletilerini dağıtım listelerini desteklemeyen bir kuyruğa doğrudan koyarsa, kuyruk yöneticisi dağıtım listesi iletilerini tek tek iletilere ayırır ve bunun yerine bunları kuyruğun üzerine yerleştirir.

- "Alanlar" sayfa 1044
- "Başlangıçtaki değerler" sayfa 1047
- "RPG bildirim" sayfa 1048

## Alanlar

MQDH yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

### DHCNT (10 basamaklı işaretli tamsayı)

Var olan MQOR kayıt sayısı.



Bu, hedef sayısını tanımlar. Bir dağıtım listesi her zaman en az bir hedef içermeli, bu nedenle *DHCNT* her zaman sıfırdan büyük olmalıdır.

Bu alanın ilk değeri 0 'tır.

### **DHCSI (10 basamaklı işaretli tamsayı)**

MQOR ve MQPMR kayıtlarını izleyen verilerin karakter kümesi tanıtıcısı.

Bu, MQOR ve MQPMR kayıtları dizilerini izleyen verilerin karakter kümesi tanıtıcısını belirtir; bu, MQDH yapısındaki karakter verilerine uygulanmaz.

MQPUT ya da MQPUT1 çağrısında, uygulama bu alanı veri için uygun değere ayarlamalıdır. Aşağıdaki özel değer kullanılabilir:

#### **CİNT**

Bu yapının karakter kümesi tanıtıcısını devralır.

Character data in the data *takip edilen* this structure is in the same character set as this structure.

Kuyruk yöneticisi, bu değeri, iletide belirtilen yapının gerçek karakter kümesi tanıtıcısına göre değiştirir. Hata oluşmaması durumunda, MQGET çağrısıyla CSINHT değeri döndürülmez.

MQMD 'deki *MDPAT* alanının değeri ATBRKR ise, CSINHT kullanılamaz.

Bu alanın ilk değeri CSUNDF 'dir.

### **DHENC (10 basamaklı işaretli tamsayı)**

MQOR ve MQPMR kayıtlarını izleyen verilerin sayısal kodlaması.

Bu, MQOR ve MQPMR kayıtları dizilerini izleyen verilerin sayısal kodlamasını belirtir; MQDH yapısındaki sayısal veriler için geçerli değildir.

MQPUT ya da MQPUT1 çağrısında, uygulama bu alanı veri için uygun değere ayarlamalıdır.

Bu alanın ilk değeri 0 'tır.

### **DHFLG (10 basamaklı işaretli tamsayı)**

Genel bayraklar.

Aşağıdaki işaret belirlenebilir:

#### **YENI**

Yeni ileti tanıtıcıları oluşturur.

Bu işaret, dağıtım listesindeki her hedef için yeni bir ileti tanıtıcısı oluşturulacağını belirtir.

Bu yalnızca, herhangi bir put-message kaydı yoksa ya da kayıtlar varsa, ancak *PRMID* alanını içermediği durumlarda ayarlanabilir.

Bu işaret, ileti tanıtıcılarının, olası son ana kadar, yani dağıtım listesi iletisinin son olarak tek tek iletilere bölünmesine kadar, ileti tanıtıcılarının oluşturulmalarını sağlar. Bu, dağıtım listesi iletiliyle akışı gereken denetim bilgileri miktarını en aza indirir.

Bir uygulama bir dağıtım listesine ileti koyduğunda, kuyruk yöneticisi aşağıdaki deyimlerin her ikisi de doğru olduğunda MQDH 'de DHFNEW' i ayarlar:

- Uygulama tarafından sağlanan herhangi bir put-message kaydı yok ya da sağlanan kayıtlar *PRMID* alanını içermiyor.
- MQMD 'deki *MDMID* alanı IKINCILDIR ya da MQPMO' daki *PMOPT* alanı PMNMID ' yi içerir.

Herhangi bir işaret gerekli değilse, aşağıdaki ifade kullanılabilir:

#### **DHFNON**

Bayrak yok.

Bu değişmez, herhangi bir işaretin belirlenmemesine neden olur. DHFON, program belgelerine yardımcı olmak için tanımlıdır. Bu değişmezin başka biriyle birlikte kullanılması amaçlanmaz, ancak değeri sıfır olduğu için, bu tür bir kullanım saptanamaz.

Bu alanın ilk deęeri DHFON ' dur.

### **DHFMT (8 baytlık karakter dizilimi)**

MQOR ve MQPMR kayıtlarını izleyen verilerin adını biçimle.

Bu, MQOD ve MQPMR kayıtlarının dizilerini izleyen verilerin biçim adını belirtir (hangisi son olursa olsun).

MQPUT ya da MQPUT1 çağrısında, uygulama bu alanı veri için uygun değere ayarlamalıdır. Bu alanı kodlamaya ilişkin kurallar, MQMD ' deki *MDFMT* alanı için yapılan kurallarla aynıdır.

Bu alanın ilk deęeri FMNONE ' dır.

### **DHLEN (10 basamaklı işaretli tamsayı)**

MQDH yapısı ve izleyen MQOR ve MQPMR kayıtlarının uzunluğu.

Bu, MQDH yapısının başlangıcından, MQOR ve MQPMR kayıtlarının dizilerini izleyen ileti verilerinin başlangıcından bayt sayısıdır. Veriler aşağıdaki sırayla gerçekleşir:

- MQDH yapısı
- MQOR kayıtları dizisi
- MQPMR kayıtları dizisi
- İleti Verileri

MQOR ve MQPMR kayıtlarının dizileri, MQDH yapısının içerdiği görelî kümelerle adreslenir. Bu görelî konum, bir ya da daha çok MQDH yapısı, kayıt dizileri ve ileti verileri arasında kullanılmayan baytlarla sonuçlanırsa, kullanılmayan byte 'lar *DHLEN*değerine eklenmelidir; ancak bu baytların içerięi kuyruk yöneticisi tarafından korunmaz. MQPMR kayıtları dizisi için, MQOR kayıtları diziden önce gelen dizi için geçerlidir.

Bu alanın ilk deęeri 0 'tır.

### **DHORO (10 basamaklı işaretli tamsayı)**

MQDH ' nin başlangıcından ilk MQOR kaydının görelî konumu.

Bu alan, hedef kuyrukların adlarını içeren MQOR nesne kayıtları dizideki ilk kaydın bayt cinsinden görelî konumu verir. Bu dizideki *DHCNT* kayıtları var. Bu kayıtlar (artı ilk nesne kaydı ile önceki alan arasında atlanan tüm baytlar), *DHLEN* alanı tarafından verilen uzunluęa eklenir.

Bir dağıtım listesi her zaman en az bir hedef içermeli, bu nedenle *DHORO* her zaman sıfırdan büyük olmalıdır.

Bu alanın ilk deęeri 0 'tır.

### **DHPRF (10 basamaklı işaretli tamsayı)**

Hangi MQPMR alanlarının var olduğunu gösteren işaretler.

Aşağıdaki işaretlerin sıfır ya da daha fazlası belirtilebilir:

#### **PFMID**

İleti tanıtıcısı alanı var.

#### **PFCID**

İlinti tanıtıcısı alanı var.

#### **PFA**

Grup tanıtıcısı alanı var.

#### **PFFB**

Geribildirim alanı mevcut.

#### **PFACC**

Muhasebe belirteci alanı var.

Hiçbir MQPMR alanı yoksa, aşağıdaki deęer belirlenebilir:

## FPNONE

Hiçbir put-message kayıt alanı mevcut değil.

FPNONE, program belgelerine yardımcı olacak şekilde tanımlıdır. Bu değişmezin başka biriyle birlikte kullanılması amaçlanmaz, ancak değeri sıfır olduğu için bu tür bir kullanım saptanamaz.

Bu alanın ilk değeri FPNONE 'dir.

## DHPRO (10 basamaklı işaretli tamsayı)

MQDH ' nin başlangıcından ilk MQPMR kaydının kayması.

Bu alan, MQPMR dizesindeki ilk kaydın byte cinsinden görelî konumu verir, ileti özelliklerini içeren ileti kayıtlarını içerir. Varsa, bu dizideki *DHCNT* kayıtları vardır. Bu kayıtlar (artı, ilk put iletilisi kaydı ile önceki alan arasında atlanan tüm baytlar), *DHLEN* alanı tarafından verilen uzunluğa eklenir.

Put ileti kayıtları isteğe bağlıdır; kayıt sağlanmaz, *DHPRO* sıfır ve *DHPRF* değeri PFYOK değerine sahiptir.

Bu alanın ilk değeri 0 'tır.

## DHSID (4 baytlık karakter dizgisi)

Yapı tanıtıcısı.

Değer şu olmalıdır:

### DHSIDV

Dağıtım üstbilgisi yapısına ilişkin tanıtıcı.

Bu alanın ilk değeri DHSIDV 'dir.

## DHVER (10 basamaklı işaretli tamsayı)

Yapı sürüm numarası.

Değer şu olmalıdır:

### DHVER1

Dağıtım üstbilgisi yapısına ilişkin sürüm numarası.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

### DHVERC

Dağıtım üstbilgisi yapısının yürürlükteki sürümü.

Bu alanın ilk değeri DHVER1' dir.

## Başlangıçtaki değerler

Çizelge 697. MQDH ' deki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
<i>DHSID</i>	DHSIDV	'DH---
<i>DHVER</i>	DHVER1	1
<i>DHLEN</i>	Yok	0
<i>DHENC</i>	Yok	0
<i>DHCSI</i>	CSUNDF	0
<i>DHFMT</i>	FMNONE	Boşluklar
<i>DHFLG</i>	DHFNON	0
<i>DHPRF</i>	FPNONE	0
<i>DHCNT</i>	Yok	0

Çizelge 697. MQDH 'deki alanların ilk değerleri (devamı var)

Alan adı	Değişmezin adı	Değişmezin değeri
DHORO	Yok	0
DHPRO	Yok	0

#### Notlar:

1. - simgesi tek bir boş karakteri temsil eder.

#### RPG bildirim

```
D*..1.....2.....3.....4.....5.....6.....7..
D* MQDH Structure
D*
D* Structure identifier
D  DHSID          1      4    INZ('DH ')
D* Structure version number
D  DHVER          5      8I 0 INZ(1)
D* Length of MQDH structure plus following MQOR and MQPMR records
D  DHLEN          9     12I 0 INZ(0)
D* Numeric encoding of data that followsthe MQOR and MQPMR records
D  DHENC         13     16I 0 INZ(0)
D* Character set identifier of data thatfollows the MQOR and MQPMR
D* records
D  DHCSI         17     20I 0 INZ(0)
D* Format name of data that follows theMQOR and MQPMR records
D  DHFMT         21     28    INZ(' ')
D* General flags
D  DHFLG         29     32I 0 INZ(0)
D* Flags indicating which MQPMR fieldsare present
D  DHPRF         33     36I 0 INZ(0)
D* Number of MQOR records present
D  DHCNT         37     40I 0 INZ(0)
D* Offset of first MQOR record from startof MQDH
D  DHORO         41     44I 0 INZ(0)
D* Offset of first MQPMR record fromstart of MQDH
D  DHPRO         45     48I 0 INZ(0)
```

IBM i

## IBM üzerindeMQDLH (Dead-letter üstbilgisi)

### Genel Bakış

#### Amaç

MQDLH yapısı, ölü-mektup (undelivered-message) kuyruğunda bulunan iletilerin uygulama iletileri verilerini önleyen bilgileri açıklar. Kuyruk yöneticisi ya da ileti kanalı aracısı tarafından kuyruğa yönlendirdiği için, ileti, ölü-harf kuyruğunda gelebilir. Bir uygulama iletiyi doğrudan kuyruğa koyabilir.

#### Biçim Adı

FMDLH

#### Karakter kümesi ve kodlama

MQDLH , uygulama iletileri verilerinin başlangıcında olabilir. Böyle bir durumda, MQDLH yapısındaki alanlar, MDCSI ve MDENC alanları tarafından verilen karakter kümesi ve kodlamadır. Değilse, karakter kümesi ve kodlama, MQDLH' den önce gelen üstbilgi yapısındaki MDCSI ve MDENC alanları tarafından ayarlanır.

Karakter takımı, kuyruk adlarında geçerli olan karakterler için tek byte 'lık karakterler içeren bir karakter takımı olmalıdır.

#### Kullanım

Applications that put messages directly on the dead-letter queue must prefix the message data with an MQDLH structure, and initialize the fields with appropriate values. Ancak kuyruk yöneticisi bir MQDLH yapısının var olmasını gerektirmez ya da alanlar için geçerli değerler belirtilmesini gerektirmez.

İletinin gönderileceği ileti kuyruğunda çok uzun olması durumunda, uygulamanın aşağıdaki şeylerden birini gerçekleştirmesi gerekir:

- İleti verilerinin, ölü-mektup kuyruğuna sığabilmek için kesilmesi.
- İletiyi yardımcı belleğe kaydedin ve ileti, ileti kuyruğunda çok uzun olduğunu belirten bir kural dışı durum raporu iletilmesi yerletir.
- İletiyi atar ve bir hatayı orijinaline geri döndürür. İleti kritik bir iletiyse. İletiyi yalnızca, başlatan iletinin bir kopyası olduğu biliniyorsa atın. Örneğin, bir iletişim kanalından bir ileti kanalı aracısı tarafından alınan bir ileti.

Seçimlerden hangisinin uygun olduğunu, uygulamanın tasarımına bağlıdır.

Kuyruk yöneticisi, bir bölüm olan bir ileti, öndeki bir MQDLH yapısıyla konduğunda özel işleme gerçekleştirir. Ek ayrıntılar için MQMDE yapısına ilişkin açıklamalara bakın.

- [“İletilerin ölü-mektup kuyruğuna konması” sayfa 1049](#)
- [“Ölü-mektup kuyruğundan ileti alınması” sayfa 1050](#)
- [“Alanlar” sayfa 1050](#)
- [“Başlangıçtaki değerler” sayfa 1053](#)
- [“RPG bildirim” sayfa 1053](#)

## İletilerin ölü-mektup kuyruğuna konması

Bir ileti, ölü-mektup kuyruğuna konursa, MQPUT ya da MQPUT1 çağrısı için kullanılan MQMD yapısı, iletiyle ilişkili MQMD ile aynı olmalıdır. MQMD genellikle aşağıdaki durumlar dışında, MQGET çağrısı tarafından döndürülen bir adrestir:

- MDCSI ve MENDENC alanları, MQDLH yapısındaki alanlar için karakter kümesi ve kodlama her ne için kullanılsa ayarlanmalıdır.
- Verilerin MQDLH yapısıyla başladığını belirtmek için MDFMT alanının FMDLH olarak ayarlanması gerekir.
- Koşullara uygun bir bağlam seçeneği kullanılarak bağlam alanları, MDACC, MDAID, MDAOD, MDPAN, MDPAT, MDPD, MDPTve MDUID ayarlanması gerekir:
  - Ölü-mektup kuyruğuna gönderilen bir uygulamanın, önceki herhangi bir iletiyle ilgili olmayan bir iletiyi PMDEFEC seçeneğini kullanmaları gerekir. PMDEFEC seçeneği, kuyruk yöneticisinin ileti tanımlayıcısındaki bağlam alanlarının tümünü varsayılan değerlerine ayarlamasına neden olur.
  - Özgün bağlam bilgilerini korumak için, ölü-mektup kuyruğuna gönderilen bir ileti kuyruğuna gönderilen bir sunucu uygulamasının, PMPASA seçeneğini kullanması gerekir.
  - Ölü-mektup kuyruğuna gönderilen bir yanıt iletilisini içeren bir sunucu uygulaması, aldığı iletiye yanıt olarak PMPASI seçeneğini kullanmalıdır. PMPASI seçeneği, kimlik bilgilerini korur, ancak sunucu uygulamasının başlangıç noktası bilgilerini belirler.
  - Bir ileti kanalı aracısının, iletişim kanalından aldığı ileti kuyruğuna bir ileti yerleştirmesi PMSETA seçeneğini kullanmalıdır. PMSETA seçeneği, özgün bağlam bilgilerini korur.

MQDLH yapısının kendisinde, alanların aşağıdaki gibi ayarlanması gerekir:

- DLCSI, DLENCve DLFMT alanları, MQDLH yapısını izleyen verileri tanımlayan değerlere ayarlanmış olmalıdır. Bu değerler genellikle özgün ileti tanımlayıcısından gelen değerlerdir.
- DLPAT, DLPAN, DLPDve DLPT bağlam alanları, iletiyi ölü-mektup kuyruğuna koyan uygulamaya uygun değerlere ayarlanmalıdır. Bu değerler özgün iletiyle ilişkili değerlerdir.
- Diğer alanlar için uygun bir değer belirlenmeli.

Uygulama, tüm alanların geçerli değerlere sahip olduğundan ve karakter alanlarının, alanın tanımlı uzunluğuna kadar boşluklarla doldurulduğundan emin olmalıdır. Karakter verileri, boş değerli bir karakter kullanılarak zamanından önce sonlandırılmamalı. Kuyruk yöneticisi boş (null) ve sonraki karakterleri MQDLH yapısındaki boşluklara dönüştürmez.

## Ölü-mektup kuyruğundan ileti alınması

Ölü-mektup kuyruğundan ileti alan uygulamaların, iletilerin MQDLH yapısıyla başladığını doğrulamalıdır. Uygulama, MQMDileti tanımlayıcısındaki MDFMT alanını inceleyerek bir MQDLH yapısının var olup olmadığını belirleyebilir. Alanda FMDLHdeğeri varsa, ileti verileri bir MQDLH yapısıyla başlar. Kuyruklanan kuyruklardaki kuyruklardaki iletiler, kuyruğun amaçlandığı kuyruk için çok uzunsa kesilebilir.

### Alanlar

MQDLH yapısı aşağıdaki alanları içerir; alanlar alfabetik sırayla anlatılır:

#### **DLCSI (10 basamaklı işaretli tamsayı)**

Character set identifier of data that follows MQDLH.

DLCSI , MQDLH yapısını izleyen verilerin karakter kümesi tanıttıcısını belirtir. Veriler genellikle özgün iletiden gelir. Bu, MQDLH yapısındaki karakter verileri için geçerli değildir.

On the MQPUT or MQPUT1 call, the application must set this field to the value appropriate to the data. Aşağıdaki özel değer kullanılabilir:

#### **CSINHT**

Bu yapının karakter kümesi tanıttıcısını devralır.

Bu yapıyı izleyen verilerdeki karakter verileri, bu yapıyla aynı karakter kümesinde yer alıyor.

Kuyruk yöneticisi, bu değeri, iletide belirtilen yapının gerçek karakter kümesi tanıttıcısına göre değiştirir. Provided no error occurs, the value CSINHT is not returned by the MQGET call.

MQMD içindeki MDPAT alanının değeri ATBRKRise,CSINHT kullanılamaz.

Bu alanın başlangıç değeri CSUNDF' dir.

#### **DLDM (48 baytlık karakter dizgisi)**

Özgün hedef kuyruk yöneticisinin adı.

Bu ad, ileti için özgün hedef olan kuyruk yöneticisinin adıdır.

Bu alanın uzunluğu LNQMntarafından verilir. Bu alanın ilk değeri 48 boş karakterdir.

#### **DLDQ (48 baytlık karakter dizilimi)**

Özgün hedef kuyruğunun adı.

Bu ad, iletinin özgün hedefi olan ileti kuyruğunun adıdır.

Bu alanın uzunluğu LNQNtarafından verilir. Bu alanın ilk değeri 48 boş karakterdir.

#### **DLENC (10 basamaklı işaretli tamsayı)**

MQDLH' u izleyen verilerin sayısal kodlaması.

DLENC , MQDLH yapısını izleyen verilerin sayısal kodlamasını belirtir. Veriler genellikle özgün iletiden gelir. MQDLH yapısındaki sayısal veriler için geçerli değildir.

On the MQPUT or MQPUT1 call, the application must set this field to the value appropriate to the data.

Bu alanın ilk değeri 0 'tır.

#### **DLFMT (8 baytlık karakter dizilimi)**

MQDLH' i izleyen verilerin biçimi.

Bu, MQDLH yapısını izleyen verilerin biçim adını belirtir (genellikle özgün iletiden gelen veriler).

On the MQPUT or MQPUT1 call, the application must set this field to the value appropriate to the data. Bu alanın kodlanması için kurallar, MQMDiçindeki MDFMT alanının kurallarıyla aynıdır.

Bu alanın uzunluğu LNFMTtarafından verilir. Bu alanın başlangıç değeri FMNONE' dir.

### **DLPAN (28 baytlık karakter dizilimi)**

İletinin ölüme mektup (teslim edilemeyen ileti) kuyruğuna ileti koyan uygulamanın adı.

Adın biçimi DLPAT alanına bağlıdır. [“MQMD \(Message descriptor\) on IBM i” sayfa 1092](#) içindeki MDPAN alanının açıklamalarına bakın.

İletiyi ölü-mektup kuyruğuna yeniden yönlendiren kuyruk yöneticisi, DLPAN kuyruk yöneticisi adının ilk 28 karakterini içerir. Ad, gerekirse boşluklarla doldurulur.

Bu alanın uzunluğu LNPAN tarafından verilir. Bu alanın ilk değeri 28 boşluk karakteridir.

### **DLPAT (10 basamaklı işaretli tamsayı)**

İletinin ölüme mektup (teslim edilemeyen ileti) kuyruğuna konması için kullanılan uygulama tipi.

Bu alan, MQMD ileti tanımlayıcısındaki MDPAT alanıyla aynı anlama sahiptir (ayrıntılar için [“MQMD \(Message descriptor\) on IBM i” sayfa 1092](#) konusuna bakın).

İletiyi ölü-harfli kuyruğuna yeniden yönlendiren kuyruk yöneticisi, DLPAT değeri ATQM olur.

Bu alanın ilk değeri 0 'tır.

### **DLPD (8 baytlık karakter dizilimi)**

İletinin, ölüme mektup (teslim edilemeyen ileti) kuyruğuna konduğu tarih.

Bu alanın kuyruk yöneticisi tarafından oluşturulduğu tarih için kullanılan biçim şöyledir:

• YYYYMMDD

karakterlerin gösterdiği yer:

**YYYY**

yıl (dört sayısal rakam)

**MM**

yıl ay (01-12)

**DD**

ay günü (01-31 arası)

Greenwich Mean Time (GMT) is used for the DLPD and DLPT fields, subject to the system clock being set accurately to GMT.

Bu alanın uzunluğu LNPDAT tarafından verilir. Bu alanın ilk değeri sekiz boş karakterdir.

### **DLPT (8 baytlık karakter dizilimi)**

İletinin, ölü-mektup (teslim edilemeyen ileti) kuyruğuna konduğu saat.

Bu alanın kuyruk yöneticisi tarafından oluşturulduğu saat için kullanılan biçim şöyledir:

• HHMMSSSTH

karakterlerin gösterdiği (sırayla):

**HH**

saat (00-23 arası)

**MM**

dakika (00-59)

**SS**

saniye (00-59; bu konuda daha sonra not et)

**T**

saniyenin onda biri (0-9 arası)

**H**

saniyenin kamarası (0-9 arası)

**Not:** If the system clock is synchronized to an accurate time standard, it is possible for 60 or 61 to be returned for the seconds in DLPT. Ek saniye, genel saat standardına sıçrama saniyeleri eklenince gerçekleşir.

Greenwich Mean Time (GMT) is used for the DLPD and DLPT fields, subject to the system clock being set accurately to GMT.

Bu alanın uzunluğu LNPTIMtarafından verilir. Bu alanın ilk değeri sekiz boş karakterdir.

### **DLREA (10 basamaklı işaretli tamsayı)**

Neden iletisi, ölüme mektup (teslim edilemeyen ileti) kuyruğuna geldi.

Bu durum, iletinin, özgün hedef kuyruğu yerine, ölüme neden olan kuyruğun üzerine yerleştirilmesinin nedenini belirtir. FB\* ya da RC\* değerlerinden biri olmalıdır (örneğin, RC2053). Oluşabilecek ortak FB\* değerlerine ilişkin ayrıntılar için "[MQMD \(Message descriptor\) on IBM i](#)" sayfa 1092 içindeki *MDFB* alanının açıklamasına bakın.

Değer FBIFST ile FBILSTarasında bir değerse, gerçek IMS hata kodu *DLREA* alanının değerinden FBIERR çıkarılarak saptlanabilir.

Bazı FB\* değerleri yalnızca bu alanda oluşur. Bunlar, hedef kuyruk kuyruğuna aktarılan havuz iletilerine, tetikleme iletilerine ya da iletim kuyruğu iletilerine ilişkindir. Bu değerler şunlardır:

#### **FBABEG**

Uygulama başlatılamaz.

Tetikleme iletisini işleyen bir uygulama, tetikleme iletisinin TMAI alanında adı belirtilen uygulamayı başlatamadı; bkz. "[MQTM-Tetikleme iletisi](#)" sayfa 1215.

#### **FBATYP**

Uygulama tipi hatası.

Tetikleme iletisinin TMAI alanı geçerli olmadığından, bir tetikleme iletisini işleyen bir uygulama, uygulamayı başlatamadı; bkz. "[MQTM-Tetikleme iletisi](#)" sayfa 1215.

#### **FBOCD**

Küme alıcı kanalı silindi.

İleti, FBIERR seçeneğiyle açılan bir küme kuyruğuna ilişkin bir küme iletim kuyruğunda yer aldı. İletiyi hedef kuyruğa iletmek için kullanılacak uzak küme alıcı kanalı, iletinin gönderilebilmesi için önce silindi. FBIERR belirtildiği için, yalnızca kuyruk açıldığı sırada seçilen kanal iletiyi iletmek için kullanılabilir. Bu kanal artık kullanılabilir durumda olmadığı için, ileti, ölü-mektup kuyruğuna yerleştirildi.

#### **FBNARM**

İleti bir havuz iletisi değil.

#### **FBSBCX**

İleti, kanal otomatik tanımlama çıkışa göre durduruldu.

#### **FBSBMX**

İleti, kanal iletisi çıkışa göre durduruldu.

#### **FBTM**

MQTM yapısı geçerli değil ya da eksik.

MQMD içindeki MDFMT alanı FMTMbelirtir, ancak ileti geçerli bir MQTM yapısıyla başlamamaktadır. Örneğin, *TMSID* anımsatıcısı göz yakalayıcısı geçerli olmayabilir. *TMVER* tanınmamış olabilir. Tetikleme iletisinin uzunluğu, MQTM yapısını içermek için yetersiz olabilir.

#### **FBXQME**

İletim kuyruğunda ileti doğru biçimde değil.

İleti kanalı aracısı, iletim kuyruğunda bir iletinin doğru biçimde olmadığını saptadı. İleti kanalı aracısı, bu geribildirim kodunu kullanarak iletiyi ölü harf kuyruğuna yerleştiriyor.

Bu alanın başlangıç değeri RCNONE' dir.



## DLSID (4 baytlık karakter dizilimi)

Yapı tanıtıcısı.

Değer şu olmalıdır:

### DLSIDV

Ölü-harf üstbilgisi yapısına ilişkin tanıtıcı.

Bu alanın başlangıç değeri DLSIDV' dir.

## DLVER (10 basamaklı işaretli tamsayı)

Yapı sürüm numarası.

Değer şu olmalıdır:

### DLVER1

Ölü-mektup üstbilgisi yapısına ilişkin sürüm numarası.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

### DLVERC

Yürürlükteki ölü harf üstbilgisi yapısının yürürlükteki sürümü.

Bu alanın başlangıç değeri DLVER1' dir.

## Başlangıçtaki değerler

Çizelge 698. Initial values of fields in MQLH		
Alan adı	Değişmezin adı	Değişmezin değeri
DLSID	DLSIDV	'DLH↵'
DLVER	DLVER1	1
DLREA	RCNONE	0
DLDQ	Yok	Boşluklar
DLDM	Yok	Boşluklar
DLENC	Yok	0
DLCSI	CSUNDF	0
DLFMT	FMNONE	Boşluklar
DLPAT	Yok	0
DLPAN	Yok	Boşluklar
DLPD	Yok	Boşluklar
DLPT	Yok	Boşluklar

### Notlar:

1. ↵ simgesi tek bir boş karakteri temsil eder.

## RPG bildirim

```
D*..1.....2.....3.....4.....5.....6.....7..
D* MQLH Structure
D*
D* Structure identifier
D DLSID 1 4 INZ('DLH ')
D* Structure version number
D DLVER 5 8I 0 INZ(1)
```

```

D* Reason message arrived on dead-letter(undelivered-message) queue
D DLREA          9      12I 0 INZ(0)
D* Name of original destination queue
D DLDQ           13      60   INZ
D* Name of original destination queue manager
D DLDM           61      108  INZ
D* Numeric encoding of data that followsMQDLH
D DLENC          109     112I 0 INZ(0)
D* Character set identifier of data thatfollows MQDLH
D DLCSI          113     116I 0 INZ(0)
D* Format name of data that followsMQDLH
D DLFMT          117     124   INZ(' ')
D* Type of application that put messageon dead-letter
D* (undelivered-message)queue
D DLPAT          125     128I 0 INZ(0)
D* Name of application that put messageon dead-letter
D* (undelivered-message)queue
D DLPAN          129     156   INZ
D* Date when message was put ondead-letter (undelivered-message)queue
D DLPD           157     164   INZ
D* Time when message was put on thedead-letter (undelivered-message)queue
D DLPT           165     172   INZ

```

IBM i

## IBM üzerinde MQDMHO (İleti tanıtıcısı seçeneklerini sil)

**MQDMHO** yapısı, uygulamaların ileti tutamaçlarının nasıl silineceğini denetleyen seçenekleri belirlemesine olanak sağlar.

### Genel Bakış

**Amaç:** Yapı, **MQDLTMH** çağrısında bir giriş parametresidir.

**Karakter kümesi ve kodlama:** **MQDMHO** içindeki veriler, uygulamanın uygulama ve kodlamasının (ENNAT) karakter kümesinde yer almalıdır.

- [“Alanlar” sayfa 1054](#)
- [“Başlangıçtaki değerler” sayfa 1055](#)
- [“RPG bildirimi” sayfa 1055](#)

### Alanlar

MQDMHO yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

#### **DMOPT (10 basamaklı işaretli tamsayı)**

Değer şu olmalıdır:

**DMYOK**

Seçenek belirtilmedi.

Bu her zaman bir giriş alanıdır. Bu alanın başlangıç değeri **DMNONE'** dir.

#### **DMSID (10 basamaklı işaretli tamsayı)**

Bu, yapı tanıtıcısıdır; değer şöyle olmalıdır:

**DMSIDV**

İleti tanıtıcısı seçenekleri yapısının silinmesine ilişkin tanıtıcı.

Bu her zaman bir giriş alanıdır. Bu alanın başlangıç değeri **DMSIDV'** dir.

#### **DMVER (10 basamaklı işaretli tamsayı)**

Bu, yapı sürüm numarasıdır; değer şu olmalıdır:

## DMVER1

Version-1 , ileti tanıtıcısı seçenekleri yapısını siler.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

## DDMVERC

İleti silme işlemi seçenekleri yapısının yürürlükteki sürümü.

Bu her zaman bir giriş alanıdır. Bu alanın başlangıç değeri **DMVER1**' dir.

## Başlangıçtaki değerler

Çizelge 699. MQDMHO 'daki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
DMSID	DMSIDV	' DMHO '
DMVER	DMVER1	1
DMOPT	DMYOK	0

## RPG bildirimi

```
D* MQDMHO Structure
D*
D*
D* Structure identifier
D DMSID          1      4  INZ('DMHO')
D*
D* Structure version number
D DMVER          5      8I 0 INZ(1)
D*
D* Options that control the action of MQDLTMH
D DMOPT          9      12I 0 INZ(0)
```

## IBM i IBM üzerinde MQDMPO (İleti özelliği seçeneklerini sil)

İleti silme özelliği seçeneklerini tanımlayan yapı.

## Genel Bakış

**Amaç:** MQDMPO yapısı, uygulamaların, iletilerin özelliklerini nasıl sileceğini denetleyen seçenekleri belirlemesine olanak sağlar. Yapı, MQDLTMP çağrısına ilişkin bir giriş değiştirmedir.

**Karakter kümesi ve kodlama:** MQDMPO 'daki veriler, uygulamanın (ENNAT) uygulama ve kodlamasının karakter kümesinde yer almalıdır.

- “Alanlar” sayfa 1055
- “Başlangıçtaki değerler” sayfa 1056
- “RPG bildirimi” sayfa 1057

## Alanlar

MQDMPO yapısı aşağıdaki alanları içerir; alanlar alfabetik sırayla anlatılır:

### DPOPT (10 basamaklı işaretli tamsayı)

İleti özelliği seçenekleri yapısını sil-DPOPT alanı.

**Konum seçenekleri:** Aşağıdaki seçenekler, özellik imlecine göre özelliğin görece konularıyla ilgilidir.

**DPDELF**

Belirtilen adla eşleşen ilk özelliğin silinmesine neden olur.

**DPDELC**

Özellik imlecinin işaret ettiği özellik siler; son sorgulayan özellik, IPINQF ya da IPINQN seçeneğini kullanarak son sorguya girilir.

İleti tanıtıcısı yeniden kullanıldığında özellik imleci sıfırlanır. MQGMO ' nun HMSG alanında bir MQGET çağrısında ya da bir MQPUT çağrısında MQPMO yapısıyla ileti tanıtıcısı belirtildiğinde de bu değer sıfırlanır.

İleti tanıtıcısı yeniden kullanıldığında ya da MQGet çağrısında bir MQGET ya da MQPMO yapısındaki MQGMO yapısının HMSG alanında ileti tanıtıcısı belirtildiğinde, özellik imleci ilk durumuna getirilir ya da ileti tanıtıcısı belirtildiğinde, imleç ilk durumuna getirilir.

Özellik geçici çizelgesi henüz oluşturulmadığında bu seçenek kullanılırsa, çağrı tamamlanma kodu CCFAIL ve neden RC2471 ile başarısız olur. Özellik imlecinin işaret ettiği özellik zaten silindiye, bu kodlarda da başarısız olur.

Bu seçeneklerin hiçbiri gerekmiyorsa, aşağıdaki seçenek kullanılabilir:

**DPNONE**

Seçenek belirtilmedi.

Bu giriş alanının başlangıç değeri DPDELF ' dir.

**DPSID (10 basamaklı işaretli tamsayı)**

İleti özelliği seçenekleri yapısını sil-DPSID alanı.

Bu, yapı tanıtıcısıdır. Değer şu olmalıdır:

**DPSİDETİ**

İleti özelliği seçenekleri yapısının silinmesine ilişkin tanıtıcı.

Bu alan her zaman bir giriş alanıdır. Bu alanın ilk değeri DPSIDV 'dir.

**DPVER (10 basamaklı işaretli tamsayı)**

İleti özelliği seçenekleri yapısını sil-DPVER alanı.

Bu, yapı sürüm numarasıdır. Değer şu olmalıdır:

**DPVER1**

İleti özelliği seçenekleri yapısının silinmesine ilişkin sürüm numarası.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

**DPVERC**

İleti özelliği seçenekleri yapısının yürürlükteki sürümü.

Bu alan her zaman bir giriş alanıdır. Bu alanın ilk değeri DPVER1' dir.

**Başlangıçtaki değerler**

Çizelge 700. MQDPMO ' daki alanların ilk değerleri		
Alan adı	Değişmez adı	Değişmez değeri
DPSID	DPSİDETİ	' DMPO '
DPVER	DPVER1	1
DPOPT	MQDLTMP ' nin işlemini denetleyen seçenekler	DPNONE

## RPG bildirimi

```
D* MQDMPO Structure
D*
D*
D* Structure identifier
D  DPSID          1      4    INZ('DMPO')
D*
D* Structure version number
D  DPVER          5      8I 0  INZ(1)
D*
** Options that control the action of
D* MQDLTMP
D  DPOPT          9      12I 0 INZ(0)
```

IBM i

## MQEPH (Embedded PCF header) on IBM i

### Genel Bakış

#### Amaç

MQEPH yapısı, iletide programlanabilir bir komut biçimi (PCF) iletisi olduğunda, bir iletide var olan ek verileri açıklar. *EPPFH* alanı, bu yapıyı izleyen PCF parametrelerini tanımlar ve bu, PCF iletisi verilerini diğer üstbilgilerle izlemenize olanak tanır.

#### Biçim Adı

EPFMT

#### Karakter kümesi ve kodlama

MQEPH içindeki veriler, yerel kuyruk yöneticisinin karakter kümesinde ve kodlamasında olmalıdır; bu değer, **CCSID** kuyruk yöneticisi özneliği tarafından verilir.

MQEPH ' nin karakter kümesini ve kodlamasını *MDCSI* ve *MDENC* alanlarına kodlayın:

- MQMD (MQPH yapısı ileti verilerinin başlangıcındaki ise) ya da
- MQEPH yapısından önce gelen üstbilgi yapısı (diğer tüm vakalar).

#### Kullanım

Komut sunucusuna ya da başka bir kuyruk yöneticisi PCF kabul eden sunucuya komut göndermek için MQEPH yapılarını kullanamazsınız.

Benzer şekilde, komut sunucusu ya da başka bir kuyruk yöneticisi PCF-kabul eden sunucu, MQEPH yapılarını içeren yanıtlar ya da olaylar oluşturamaz.

- [“Alanlar” sayfa 1057](#)
- [“Başlangıçtaki değerler” sayfa 1059](#)
- [“RPG bildirimi” sayfa 1059](#)

### Alanlar

MQEPH yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

#### EPCSI (10 basamaklı işaretli tamsayı)

Bu, MQEPH yapısını ve ilişkili PCF parametrelerini izleyen verilerin karakter takımı tanıtıcısıdır; MQEPH yapısındaki karakter verileri için geçerli değildir.

Bu alanın ilk değeri EPCund 'tur.

#### EPENC (10 basamaklı işaretli tamsayı)

Bu, MQEPH yapısını ve ilişkili PCF parametrelerini izleyen verilerin sayısal kodlamasıdır; MQEPH yapısındaki karakter verileri için geçerli değildir.

Bu alanın ilk değeri 0 'tır.

## EPFLG (10 basamaklı işaretli tamsayı)

Aşağıdaki değerler kullanılabilir:

### EPNONE

Herhangi bir işaret belirtilmedi. *MDCSI* EPNONE, program belgelerine yardımcı olmak için tanımlıdır. Bu değişmezin başka biriyle birlikte kullanılması amaçlanmaz, ancak değeri sıfır olduğu için bu tür bir kullanım saptanamaz.

### EPCSEM

Karakter verilerini içeren parametrelerin karakter kümesi, her yapıdaki *CCSID* alanı içinde tek tek belirtilir. The character set of the *EPSID* and *EPFMT* fields are defined by the *CCSID* in the header structure that precedes the *MQEPH* structure, or by the *MDCSI* field in the *MQMD* if the *MQEPH* is at the start of the message.

Bu alanın ilk değeri EPNONE 'dir.

## EPFMT (8 baytlık karakter dizilimi)

Bu ad, *MQEPH* yapısını ve ilişkili PCF değiştirgelerini izleyen verilerin biçim adıdır.

Bu alanın ilk değeri EPFMNO 'dur.

## EPLEN (10 basamaklı işaretli tamsayı)

Bu, bir sonraki üstbilgi yapısından önceki verilerin miktidir. Bu ürün aşağıdakileri içerir:

- *MQEPH* üstbilgisinin uzunluğu
- Üstbilgiyi izleyen tüm PCF parametrelerinin uzunluğu
- Bu değiştirgeleri izleyen boş doldurma işlemi

EPLEN dörtten çok olmalı.

Yapının değişmez uzunluktaki kısmı EPSTLF tarafından tanımlanır.

Bu alanın ilk değeri 68 'dir.

## EPPCFH (MQCFH)

Bu, PCF (programlanabilir komut biçimi) üstbilgisidir ve *MQEPH* yapısını izleyen PCF parametrelerini tanımlıyor. Bu, PCF ileti verilerini diğer üstbilgilerle izlemenize olanak sağlar.

PCF üstbilgisi başlangıçta aşağıdaki değerlerle tanımlanır:

Çizelge 701. EPPCFH 'deki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
EP3TYP	CFTNON	0
EP3LEN	FHLENV	36
EP3VER	FHVER3	3
EP3CMD	CMYOK	0
EP3SEQ	Yok	1
EP3CTL	CFCLST	1
EEP3CC	CCOK	0
EP3REA	YOK	0
EP3CNT	Yok	0

Uygulamanın, gömülü PCF üstbilgisinde yaptığı kullanıma ilişkin geçerli bir yapı tipi olan CFTON 'dan EP3TYP ' i değiştirmesi gerekir.

## EPSID (4 baytlık karakter dizilimi)

Değer şu olmalıdır:

### EPSTID

Embedded PCF üstbilgi yapısıyla ilgili tanıttıcı.

Bu alanın ilk değeri EPSTID 'dir.

## EPVER (10 basamaklı işaretli tamsayı)

Değer şu şekilde olabilir:

### EPVER1

Gömülü PCF üstbilgi yapısına ilişkin sürüm numarası.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

### EPVER3

Yerleşik PCF üstbilgi yapısının yürürlükteki sürümü.

Bu alanın ilk değeri EPVER3' dir.

## Başlangıçtaki değerler

Çizelge 702. MÇEPH 'deki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
EPSID	EPSTID	'EP↯↯'
EPVER	EPVER1	1
EPLEN	EPSTLF	68
EPENC	Yok	0
EPCSI	EPCUND	0
EPFMT	EPFMNO	Boşluklar
EPFLG	EPNONE	0
EPPCFH	Çizelge 701 sayfa 1058içinde tanımlandığı şekilde adlar ve değerler	0

### Not:

1. ↯ simgesi tek bir boş karakteri temsil eder.

## RPG bildirimi

```
D*..1.....2.....3.....4.....5.....6.....7..
D* MÇEPH Structure
D*
D* Structure identifier
D EPSID          1      4
D* Structure version number
D EPVER          5      8I 0
D* Total length of MÇEPH including MÇCFHand parameter structures
D* that follow
D EPLEN          9      12I 0
D* Numeric encoding of data that follows last PCF parameter structure
D EPENC          13     16I 0
D* Character set identifier of data that follows last PCF parameter
D* structure
D EPCSI          17     20I 0
D* Format name of data that follows last PCF parameter structure
D EPFMT          21     28
D* Flags
D EPFLG          29     32I 0
```

D*	Programmable Command	Format	Header
D	EP3TYP	33	36I 0
D	EP3LEN	37	40I 0
D	EP3VER	41	44I 0
D	EP3CMD	45	48I 0
D	EP3SEQ	49	52I 0
D	EP3CTL	53	56I 0
D	EP3CC	57	60I 0
D	EP3REA	61	64I 0
D	EP3CNT	65	68I 0

## IBM i IBM üzerinde MQGMO (get-message options)

MQGMO yapısı, uygulamanın iletilerin kuyruktan nasıl kaldırıldığını denetleyen seçenekleri belirlemesine olanak sağlar.

### Genel Bakış

#### Amaç

Yapı, MQGET çağrısındaki bir giriş/çıkış parametresidir.

#### S\u00fcr\u00fcm

MQGMO 'nun yürürlükteki sürümü GMVER4' tür. Yalnızca yapının daha yeni sürümlerinde var olan alanlar, izleyen açıklamalarda olduğu gibi tanımlanır.

Sağlanan COPY dosyası, MQGMO ' nun ortam tarafından desteklenen en son sürümünü içerir, ancak *GMVER* alanının ilk değeri GMVER1olarak ayarlanmış olmalıdır. version-1 yapısında mevcut olmayan alanları kullanmak için, uygulamanın *GMVER* alanını, gereken sürümün sürüm numarasına ayarlaması gerekir.

#### Karakter kümesi ve kodlama

Data in MQGMO must be in the character set given by the **CodedCharSetId** queue manager attribute and encoding of the local queue manager given by ENNAT. Ancak, uygulama bir IBM MQ istemcisi olarak çalışıyorsa, yapı, istemcinin karakter kümesinde ve kodlamasında olmalıdır.

- [“Alanlar” sayfa 1060](#)
- [“Başlangıçtaki değerler” sayfa 1079](#)
- [“RPG bildirim” sayfa 1080](#)

### Alanlar

MQGMO yapısı aşağıdaki alanları içerir; alanlar alfabetik sırayla anlatılır:

#### GMGT (1 byte 'lık karakter dizgisi)

Alınan iletinin bir grup içinde olup olmadığını belirten işaret.

Aşağıdaki değerlerden birine sahiptir:

##### GSNIG

İleti bir grupta yer almıyor.

##### GSMIG

İleti bir grup içinde, ancak gruptaki son kişi değil.

##### GSLMIG

Bu, gruptaki son iletidir.

Bu değer, grup yalnızca bir iletiyle oluşuyorsa, döndürülen değer de olur.

Bu alan bir çıkış alanıdır. Bu alanın ilk değeri GSNIG 'dir. *GMVER* , *GMVER2*değerinden küçükse bu alan yok sayılır.

#### GMMH (10 basamaklı işaretli tamsayı)

İleti Tanıtıcısı



GM PRAQ seçeneği belirtilirse ve PRPCTL kuyruk özniteliği PRPRFH olarak ayarlanmadıysa, bu, kuyruktan alınan iletinin özellikleriyle doldurulan bir iletiyle ilgili tanıtıcıdır. Tanıtıcı, bir MQCRTMH çağrısı tarafından yaratılır. Bir ileti alınmadan önce, bu tanıtıcı ile ilişkilendirilmiş olan tüm özellikler temizlenir.

Aşağıdaki değer de belirtilebilir:

MQHM\_NONE

İleti tanıtıcısı sağlanmadı.

Geçerli bir ileti tanıtıcısı sağlandıysa ve ileti özelliklerini içermek üzere çıkışta kullanılırsa, MQGET çağrısında ileti tanımlayıcısı gerekmez; giriş alanları için ileti tanıtıcısı ile ilişkili ileti tanımlayıcısı kullanılır.

MQGET çağrısında bir ileti tanımlayıcısı belirtilirse, ileti tanıtıcısı ile ilişkili ileti tanımlayıcısına her zaman öncelik kazanır.

GM PRRF belirtilirse ya da GM PRAQ belirtilirse ve PRPCTL kuyruk özniteliği PRPRFH ise, hiçbir ileti tanımlayıcı parametresi belirtilmediğinde, çağrı neden kodu RC2026 ile başarısız olur.

MQGET çağrısından dönüşte, bu ileti tanıtıcısı ile ilişkilendirilmiş özellikler ve ileti tanımlayıcısı, alınan iletinin durumunu (MQGET çağrısında belirtildiyse, ileti tanımlayıcısının yanı sıra ileti tanımlayıcısını) yansıtacak şekilde güncellenir. Bundan sonra, iletinin özellikleri MQINQMP çağrısını kullanarak sorgulanabilir.

İleti tanımlayıcı uzantıları dışında, varsa, MQINQMP çağrısıyla sorgulanabilen bir özellik ileti verilerinde bulunmuyorsa; kuyruksa bulunan ileti verilerinde yer alan özellikler, veriler uygulamaya döndürülmeden önce ileti verilerinden kaldırılmışsa.

Herhangi bir ileti tanıtıcısı sağlanmamışsa ya da sürüm GMVER4 değerinden küçükse, MQGET çağrısında geçerli bir ileti tanımlayıcısı belirtmelisiniz. İleti verileri (ileti tanımlayıcısında bulunan özellikler dışında), MQGMO yapısındaki özellik seçeneklerinin değeri ve PRPCTL kuyruk öznitelikteki değerine ilişkin ileti özellikleri (ileti tanımlayıcısında yer alır) döndürülür.

Bu alan her zaman bir giriş alanıdır. Bu alanın ilk değeri HMNONE değeridir. *GMVER*, *GMVER4* değerinden küçükse bu alan yok sayılır.

### **GMMO (10 basamaklı işaretli tamsayı)**

MQGET için kullanılan seçim ölçütlerini denetleyen seçenekler.

Bu seçenekler, uygulamanın, MQGET çağrısının döndürdüğü iletiyi seçmek için **MSGDSC** parametresindeki hangi alanların kullanılacağını seçmesine olanak tanır. Uygulama, bu alanda gerekli seçenekleri belirler ve **MSGDSC** parametresindeki ilgili alanları, bu alanlar için gereken değerlere ayarlar. Yalnızca, ileti için MQMD ' de bu değerleri içeren iletiler, MQGET çağrısındaki **MSGDSC** değiştirgesini kullanarak alma için adaydır. Döndürülemeyecek ileti seçilirken, karşılık gelen eşleşme seçeneğinin belirlenmediği alanlar yoksayılar. MQGET çağrısında seçim ölçütü kullanılmayacaksa (yani, herhangi bir ileti kabul edilebilir), *GMMO* MONONE olarak ayarlanmalıdır.

GMLOGO belirtilirse, yalnızca belirli iletiler bir sonraki MQGET çağrısıyla döndürülebilecek şekilde seçilebilir:

- If there is no current group or logical message, only messages that have *MDSEQ* equal to 1 and *MDOFF* equal to 0 are eligible for return. Bu durumda, aşağıdaki seçeneklerden biri ya da birkaçı, döndürülen uygun iletilerden hangilerinin döndürülmesini seçmek için kullanılabilir:
  - MOMSGI
  - MOCORD
  - MOGRPI
- Geçerli bir grup ya da mantıksal ileti varsa, mantıksal iletteki gruptaki ya da sonraki bölümdeki sonraki ileti dönüş için uygun olur ve MO\* seçenekleri belirlenerek bu ileti değiştirilemez.

Her iki durumda da, geçerli olmayan eşleştirme seçenekleri belirlenebilir; ancak, **MSGDSC** parametresindeki ilgili alanın değeri, döndürülebilecek iletiyle ilgili alanın değeriyle eşleşmelidir; çağrı, RC2247 neden koduyla başarısız olur; bu koşul karşılanmaz.

GMMUC ya da GMBRWC belirtilirse, *GMMO* yoksayılır.

Aşağıdaki seçeneklerden biri ya da daha fazlası belirtilebilir:

#### **MOMSGI**

Belirtilen ileti tanıtıcısını içeren iletiyi al.

Bu seçenek, alınacak iletinin, MQGET çağrısının **MSGDSC** parametresindeki *MDMID* alanının değeriyle eşleşen bir ileti tanıtıcısı olması gerektiğini belirtir. Bu eşleşme, geçerli olabilecek diğer eşleşmelere ek olarak (örneğin, ilinti tanıtıcısıdır).

Bu seçenek belirlenmezse, **MSGDSC** parametresindeki *MDMID* alanı yoksayılır ve herhangi bir ileti tanıtıcısı eşleşir.

**Not:** MINONE ileti tanıtıcısı, ileti için MQMD ' deki herhangi bir ileti tanıtıcısıyla eşleşen özel bir değer. Bu nedenle MINONE ile MOMSGI belirtilirken, MOMSGI belirtmemesiyle aynıdır.

#### **MOCORD**

Belirtilen ilinti tanıtıcısına sahip iletiyi al.

Bu seçenek, alınacak iletinin, MQGET çağrısının **MSGDSC** parametresindeki *MDCID* alanının değeriyle eşleşen bir ilinti tanıtıcısına sahip olması gerektiğini belirtir. Bu eşleşme, geçerli olabilecek diğer eşleşmelere ek olarak (örneğin, ileti tanıtıcısı).

Bu seçenek belirlenmezse, **MSGDSC** parametresindeki *MDCID* alanı yoksayılır ve herhangi bir ilinti tanıtıcısı eşleşir.

**Not:** İlinti tanıtıcısı CINONE, ileti için MQMD ' deki herhangi bir ilinti tanıtıcısıyla eşleşen özel bir değerdir. Bu nedenle, MOCORI ile MOCORI değerinin belirlenmesi, MOCORI belirtmemesiyle aynıdır.

#### **MOGRPI**

Belirtilen grup tanıtıcısına sahip iletiyi al.

Bu seçenek, alınacak iletinin, MQGET çağrısının **MSGDSC** parametresindeki *MDGID* alanının değeriyle eşleşen bir grup tanıtıcısına sahip olması gerektiğini belirtir. Bu eşleşme, geçerli olabilecek diğer eşleşmelere ek olarak (örneğin, ilinti tanıtıcısıdır).

Bu seçenek belirlenmezse, **MSGDSC** parametresindeki *MDGID* alanı yoksayılır ve grup tanıtıcı eşleşmeleri eşleşir.

**Not:** Grup tanıtıcısı GINONE, ileti için MQMD ' deki herhangi bir grup tanıtıcısıyla eşleşen özel bir değerdir. Bu nedenle, GINONE ile MOGRPI belirtilerek MOGRPI belirtmemesiyle aynı olur.

#### **MOSEQN**

İleti sıra numarasını belirtilen iletiyi al.

Bu seçenek, alınacak iletinin, MQGET çağrısının **MSGDSC** parametresindeki *MDSEQ* alanının değeriyle eşleşen bir ileti sıra numarasına sahip olması gerektiğini belirtir. Bu eşleşme, geçerli olabilecek diğer eşleşmelere ek olarak (örneğin, grup tanıtıcısı).

Bu seçenek belirlenmezse, **MSGDSC** parametresindeki *MDSEQ* alanı yoksayılır ve herhangi bir ileti dizisi numarası eşleşir.

#### **MOOFFS**

İletiyi belirtilen görel konum ile al.

Bu seçenek, alınacak iletinin, MQGET çağrısının **MSGDSC** parametresindeki *MDOFF* alanının değeriyle eşleşen bir görel konumuna sahip olması gerektiğini belirtir. Bu eşleşme, geçerli olabilecek diğer eşleşmelere ek olarak (örneğin, ileti sıra numarası).

Bu seçenek belirlenmezse, **MSGDSC** parametresindeki *MDOFF* alanı yoksayılır ve tüm görel konum eşleşmeleri yoksayılır.

Belirtilen seçeneklerden hiçbirini belirtilmediyse, aşağıdaki seçenek kullanılabilir:

#### **MONONE**

Eşleşme yok.

Bu seçenek, döndürülebilecek iletiyi seçerken hiçbir eşleşmenin kullanılmadığını belirtir; bu nedenle, kuyrukta bulunan tüm iletiler alma için uygun olur (ancak, GMAMSA, GMASGA ve GMCMPM seçenekleri tarafından denetime tabidir).

MONONE, program belgelerine yardımcı olmak için tanımlıdır. Bu seçeneğin diğer MO\* seçeneklerinde kullanılması amaçlanmaz, ancak değeri sıfır olduğu için bu tür bir kullanım algılanamaz.

Bu alan bir giriş alanıdır. Bu alanın ilk değeri MOCORI ile MOMSGI ' dir. *GMVER* , *GMVER2*değerinden küçükse bu alan yok sayılır.

**Not:** *GMMO* alanının ilk değeri, önceki sürüm kuyruğu yöneticileriyle uyumluluk için tanımlıdır. Ancak, seçim ölçütlerini kullanmadan bir kuyruktan ileti dizisi okunurken, bu ilk değer uygulamanın *MDMID* ve *MDCID* alanlarını her MQGET çağrısından önce MINONE ve CINONE olarak ilk durumuna getirmesini gerektirir. *MDMID* ve *MDCID* 'yi ilk duruma getirme gereksinimi, *GMVER* ayarını *GMVER2*ve *GMMO* ' dan MONONE ' a ayarlamaktan kaçınılabilir.

### **GMOPT (10 basamaklı işaretli tamsayı)**

MQGET işlemini denetleyen seçenekler.

Aşağıdaki açıklanan seçeneklerin sıfır ya da daha fazlası belirtilebilir. Birden fazla değer gerekliyse, değerler eklenebilir (aynı değışmezi bir kereden fazla eklemeyin). Geçerli olmayan seçenek birleşimleri dikkat edilir; diğer tüm birleşimler geçerlidir.

**Bekleme seçenekleri:** Aşağıdaki seçenekler, kuyruğun varolması için bekleyen iletilerin beklenmesiyle ilgilidir:

#### **GGMWT**

İletinin gelmesini bekleyin.

Uygulama, uygun bir ileti gelene kadar beklemeniz gerekir. Uygulamanın bekleyeceği süre üst sınırı *GMWT* içinde belirtilir.

MQGET istekleri engellenirse ya da bekleme sırasında MQGET istekleri engellenirse, bekleme işlemi iptal edilir ve çağrı CCFAIL ile tamamlanır ve kuyruğun üzerinde uygun ileti olup olmadığına bakılmaksızın RC2016neden kodu ile tamamlanır.

Bu seçenek, GMBRWF ya da GMBRWN seçenekleri ile kullanılabilir.

Aynı paylaşılan kuyrukta, uygulamada ya da uygulamalarda çok sayıda uygulama bekliyorsa, bu bölümde daha sonra uygun bir ileti geldiğinde etkinleştirilen uygulamalar daha sonra açıklanmıştır.

**Not:** Aşağıdaki tanımda, bir MQGET göz atma çağrısına göz atma seçeneklerinden birini belirtir, ancak GMLK değil; GMLK seçeneğini belirten bir MQGET çağrısı, göz atma dışı bir çağrı olarak işlenir.

- Bir ya da daha çok göz atmayan MQGET çağrıları bekliyorsa, ancak MQGET çağrılarına göz atma işlemi beklenmiyorsa, bir kişi etkinleşir.
- Bir ya da daha fazla tarama MQGET çağrıları bekliyorsa, ancak bekleyen MQGET çağrıları beklemiyorsa, tüm bunlar etkinleşir.
- Bir ya da daha çok MQGET çağrısının ve MQGET çağrılarının bir ya da daha fazlasının beklemesi durumunda, bir MQGET çağrısına göz atmayan bir MQGET çağrısı etkinleştirilir ve hiçbiri, bazıları ya da tüm MQGET çağrıları etkinleştirilir. (Etkinleştirilen MQGET çağrılarının sayısı, işletim sisteminin zamanlama konularına ve diğer etkenlere bağlı olduğundan öngörülemez.)

Aynı kuyrukta birden çok MQGET dışı sorgu çağrısı bekleniyorsa, yalnızca biri etkinleştirilir; bu durumda kuyruk yöneticisi, göz atma bekleyen çağrıları beklemek için şu sırada öncelik vermeyi dener:

1. Yalnızca belirli iletiler (örneğin, belirli bir *MDMID* ya da *MDCID* ) ile (ya da her ikisi) karşılanabilen belirli bekleme istekleri.
2. Herhangi bir iletiyle karşılanabilen genel alma isteği istekleri.

Aşağıdaki noktalara dikkat edilmelidir:

- İlk kategori içinde, daha özel alma bekleme isteklerine ek öncelik verilmez; örneğin, hem *MDMID* hem de *MDCID* belirtilenler.
  - Her iki kategori içinde, hangi uygulamanın seçildiği tahmin edilemez. Özellikle, en uzun süre bekleyen uygulama, seçilen en uzun süre için gerekli değildir.
  - İşletim sisteminin yol uzunluğu ve öncelik zamanlaması konuları, düşük işletim sistemi önceliğinin beklenenden beklenen bir uygulamanın iletiyi almasının beklenenden daha düşük bir şekilde uygulanması anlamına gelebilir.
  - Beklemedeki bir uygulamanın, tercihteki iletiyi olan bir iletiyi almadığı da ortaya gelebilir.
- GMBRWC ya da GMMUC ile belirtilirse, GMWT yoksayılr; hata ortaya çıktı.

#### **GDNWT**

Uygun bir ileti yoksa hemen geri dönün.

Uygun bir ileti yoksa, uygulama beklemez. Bu, GMWT seçeneğinin tam tersidir ve program belgelerine yardımcı olmak için tanımlanır. İkisi de belirtilmediyse, bu varsayılan değerdir.

#### **GMFIQ**

Kuyruk yöneticisi susturulmuş durumda olduğunda başarısız olun.

Bu seçenek, kuyruk yöneticisi susturma durumdaysa, MQGET çağrısını başarısız olarak zorlar.

Bu seçenek, GMWT ile birlikte belirtilirse ve kuyruk yöneticisinin susturma durumuna girdiği sırada bekleme olağanüstüse:

- Bekleme iptal edildi ve çağrı, RC2161 neden koduyla CCFAIL tamamlanma kodunu döndürdü.

GMFIQ belirlenmediyse ve kuyruk yöneticisi susturma durumuna girerse bekleme iptal edilmez.

**Syncpoint seçenekleri:** Aşağıdaki seçenekler, bir iş birimi içindeki MQGET çağrısının katılımlarıyla ilgilidir:

#### **GGMSYP**

Syncpoint denetimiyle ileti alın.

Bu istek, olağan çalışma birimi protokolleri içinde işlem yapmak. İleti, diğer uygulamalar tarafından kullanılamaz olarak işaretlenir, ancak iş birimi kesinleştirildiğinde kuyruktan silinir. İş birimi getirilirse, ileti yeniden kullanılabilir kılındır.

Bu seçenek ya da GMNSYP belirtilmediyse, alma isteği bir iş birimi içinde değildir.

Bu seçenek, aşağıdaki seçeneklerin hiçbirisiyle geçerli değildir:

- GMBRWF
- GMBRWC
- GMBRWN
- GMLK
- GMNSYP
- GMPSYP
- GGMUNLK

#### **GMPSYP**

İleti kalıcıysa, eşitleme noktası denetimi ile ileti alın.

Bu istek, yalnızca alınan iletinin kalıcı olması durumunda, olağan çalışma birimi protokollerinde işlem görmektedir. Kalıcı ileti, MQMD 'deki *MDPER* alanında PEPER değerini içerir.

- İleti kalıcıysa, kuyruk yöneticisi çağrısı, uygulamanın GMSYP belirtmiş gibi işlemlerini sağlar.
- İleti kalıcı değilse, kuyruk yöneticisi çağrısı uygulamanın GMNSYP belirtmiş gibi işlemlerini sağlar (ayrıntılar için aşağıdaki bölüme bakın).

Bu seçenek, aşağıdaki seçeneklerin hiçbirisiyle geçerli değildir:

- GMBRWF

- GMBRWC
- GMBRWN
- GMCMM
- GMNSYP
- GGMSYP
- GGMUNLK

#### **GMNSYP**

Eşitleme noktası denetimi olmadan ileti al.

Bu istek, olağan çalışma birimi protokollerinin dışında işlem yapmak. İleti kuyruktan hemen silinir (bu bir göz atma isteği değilse). İleti, iş biriminin yedeklenmesiyle yeniden kullanılabilir kılınamaz.

Bu seçenek, GMBRWF ya da GMBRWN belirtildiğinde kabul edilir.

Bu seçenek ve GMSYP belirtilmediyse, alma isteği bir iş birimi içinde değildir.

Bu seçenek, aşağıdaki seçeneklerin hiçbirleriyle geçerli değildir:

- GGMSYP
- GMPSYP

**Göz at seçenekleri:** Aşağıdaki seçenekler kuyruklardaki iletilere göz atmak için ilişkilendirilir:

#### **GMBRWF**

Kuyruğun başlangıcından göz atın.

OBRWF seçeneğiyle bir kuyruk açıldığında, kuyruktaki ilk iletiden önce mantıksal olarak konumlandırılmış bir göz atma imleci oluşturulur. Daha sonra, kuyruktan yok edilemeyen iletileri almak için, GMBRWF, GMBRWN ya da GMBRWC seçeneğini belirten sonraki MQGET çağrılarını kullanılabilir. Göz atma imleci, kuyruktaki iletiler içinde, GMBRWN ile ilgili sonraki MQGET çağrısının uygun bir ileti için arama yaptığı iletiler içinde yer alan konumdaki konumu işaretler.

GMBRWF ile MQGET çağrısı, göz atma imlecinin önceki konumunun yoksayılmasına neden oluyor. Kuyrukte belirtilen ilk ileti, ileti tanımlayıcısında belirtilen koşulları karşıladığını gösterir. İleti kuyruğun üzerinde kalır ve göz atma imleci bu iletiye konumlanır.

Bu çağrıdan sonra, göz atma imleci döndürülen iletiye konumlanır. İleti, GMBRWN ile bir sonraki MQGET çağrısından önce kuyruktan kaldırılırsa, bu konum artık boş olsa bile, göz atma imleci, iletinin meşgul olduğu kuyrukte konumdaki konumda kalır.

GMMUC seçeneği, iletiyi kuyruktan kaldırmak için, gerekliyse, göz atmayan bir MQGET çağrısıyla birlikte kullanılabilir.

Göz atma imleci, aynı *HOB*J tanıtıcısını kullanan bir MQGET çağrısına göre taşınmaz. Nor is it moved by a browse MQGET call that returns a completion code of CCFAIL, or a reason code of RC2080 .

GGMLK seçeneği, bu seçenekle birlikte, göz atıldığı iletinin kilitlemiş olmasına yol açabilir.

GMBRWF, mantıksal iletilerin gruplarındaki ve bölümlerindeki iletilerin işlenmesini denetleyen GM\* ve MO\* seçeneklerinin geçerli herhangi bir birleşimiyle belirlenebilir.

GMLOGO belirtilirse, iletiler mantıksal sırayla göz atılır. Bu seçenek atılırsa, iletiler fiziksel sırayla göz atılır. GMBRWF belirtildiğinde, mantıksal sıralama ile fiziksel sıralama arasında geçiş yapmak mümkündür; ancak, GMBRWN kullanan sonraki MQGET çağrılarında kuyruk tanıtıcısı için GMBRWF belirlenmiş en son çağrıyla aynı sırayla kuyruğa göz atmanız gerekir.

Kuyruk yöneticisinin kuyruklardaki iletilere göz atarak MQGET çağrılarını için sakladığı grup ve kesim bilgileri, kuyruk yöneticisinin kuyruktan iletileri kaldırtan MQGET çağrılarını için sakladığı grup ve kesim bilgilerinden ayrı olarak kullanılır. GMBRWF belirtildiğinde, kuyruk yöneticisi, göz atma için grup ve bölüm bilgilerini yoksayar ve yürürlükteki grup ve yürürlükteki mantıksal ileti yok gibi kuyrukları tarar. MQGET çağrısı başarılı olursa (tamamlanma kodu CCOK ya da CCWARN), göz atma

için grup ve kesim bilgileri, döndürülen iletinin o değerine ayarlanır; arama başarısız olursa, grup ve bölüm bilgileri aramadan önce olduğu gibi kalır.

Bu seçenek, aşağıdaki seçeneklerin hiçbirisiyle geçerli değildir:

- GMBRWC
- GMBRWN
- GMMUC
- GGMSYP
- GMPSYP
- GGMUNLK

Ayrıca, kuyruk göz atma için açılmamışsa da bu bir hatadır.

## **GMBRWN**

Kuyrukta yürürlükteki konumdan göz atın.

Göz atma imleci, kuyruğun, MQGET çağrısında belirlenen seçim ölçütlerini karşılayan bir sonraki iletiye iletildiğinden emin olun. İleti uygulamaya geri döndürüldü, ancak kuyruğun üzerinde kalır.

Bir kuyruk göz atma için açıldıktan sonra, tutamacı kullanarak ilk göz atma çağrısı, GMBRWF ya da GMBRWN seçeneğini belirleyip belirtmediğine göre aynı etkiye sahiptir.

İleti, GMBRWN ile bir sonraki MQGET çağrısından önce kuyruktan kaldırılırsa, göz atma imleci, o konum artık boş olsa bile, iletinin meşgul olduğu kuyrukta mantıksal olarak konumun üzerinde kalır.

İletiler kuyrukta iki yoldan biriyle saklanır:

- Öncelik içindeki FIFO (MSPRIO) ya da
- Öncelik (MSFIFO) dikkate alınmaksızın FIFO (FIFO)

**MsgDeliverySequence** kuyruk özniteliği, hangi yöntemin geçerli olduğunu belirtir (ayrıntılar için [“Kuyruklara ilişkin öznitelikler” sayfa 1345](#) konusuna bakın).

If the queue has a *MsgDeliverySequence* of MSPRIO, and a message arrives on the queue that is of a higher priority than the one currently pointed to by the browse cursor, that message is not found during the current sweep of the queue using GMBRWN. Yalnızca, imleç GMBRWF (ya da kuyruğun yeniden açılarak) ile ilk durumuna getirildikten sonra bulunabilir.

GMMUC seçeneği daha sonra, iletiyi kuyruktan kaldırmak için, gerekiyorsa, bir MQGET çağrısıyla birlikte kullanılabilir.

Göz atma imleci, aynı *HOB*J tanıtıcısı kullanılarak MQGET çağrılarında göz atılmaz bir şekilde taşınmaz.

GGMLK seçeneği, bu seçenekle birlikte, göz atıldığı iletinin kilitlemiş olmasına yol açabilir.

GMBRWN, mantıksal iletilerin gruplarındaki ve bölümlerindeki iletilerin işlenmesini denetleyen GM\* ve MO\* seçeneklerinin geçerli bir birleşimiyle belirlenebilir.

GMLOGO belirtilirse, iletiler mantıksal sırayla göz atılır. Bu seçenek atılırsa, iletiler fiziksel sırayla göz atılır. GMBRWF belirtildiğinde, mantıksal sıralama ile fiziksel sıralama arasında geçiş yapmak mümkündür; ancak, GMBRWN kullanan sonraki MQGET çağrılarında kuyruk tanıtıcısı için GMBRWF belirlenmiş en son çağrıyla aynı sırayla kuyruğa göz atmanız gerekir. Bu koşul karşılanmazsa, çağrı neden kodu RC2259 ile başarısız olur.

**Not:** Bir MQGET çağrısı, GMLOGO belirtilmediğinde bir ileti grubunun (ya da bir grupta olmayan mantıksal ileti) bitmesinin ötesine göz atmak için kullanılıyorsa, özel bakım gerekir. For example, if the last message in the group happens to precede the first message in the group on the queue, using GMBRWN to browse beyond the end of the group, specifying MOSEQN with MDSEQ set to 1 (to find the first message of the next group) would return again the first message in the group

already browsed. Bu durum hemen olabilir ya da daha sonra (araya giren gruplar varsa) MQGET çağrılarının sayısı hemen olabilir.

Göz atma için kuyruk iki kez açılarak sonsuz döngü olasılığı önlenir:

- Her gruptaki ilk iletiye göz atmak için ilk tanıtıcıyı kullanın.
- Yalnızca belirli bir grup içindeki iletilere göz atmak için ikinci tanıtıcıyı kullanın.
- Gruptaki iletilere göz atmadan önce, ikinci göz atma imlecini ilk göz atma imlecinin konumuna taşımak için MO\* seçeneklerini kullanın.
- Bir grubun sonunun ötesine göz atmak için GMBRWN kullanmayın.

Kuyruk yöneticisinin kuyruklardaki iletilere göz atmasını içeren MQGET çağrılarının için sakladığı grup ve kesim bilgileri, kuyruktan iletilerin kaldırıldığı MQGET çağrılarının için sakladığı grup ve bölüm bilgilerinden ayrıdır.

Bu seçenek, aşağıdaki seçeneklerin hiçbirisiyle geçerli değildir:

- GMBRWF
- GMBRWC
- GMMUC
- GGMSYP
- GMPSYP
- GGMUNLK

Ayrıca, kuyruk göz atma için açılmamışsa da bu bir hatadır.

## **GMBRWC**

Göz atma imleci altındaki iletiye göz atın.

Bu seçenek, MQGMO ' da GMMO alanında belirtilen MO\* seçeneklerinden bağımsız olarak, göz atma imlecinin, yok edici olmayan bir duruma getirilmesine neden olur.

Göz atma imlecinin gösterdiği ileti, son olarak GMBRWF ya da GMBRWN seçeneği kullanılarak alınan bir ileti. Açıldığı için bu kuyruk için bu çağrılarının hiçbirisi verilmezse ya da Göz At imlecinin altındaki ileti geçici olarak geri alındıktan sonra çağrılırsa arama başarısız olur.

Göz atma imlecinin konumu, bu çağrıya göre değiştirilmez.

GMMUC seçeneği, iletiyi kuyruktan kaldırmak için, gerekiyorsa, göz atmayan bir MQGET çağrısıyla birlikte kullanılabilir.

Göz atma imleci, aynı HOBJS tanıtıcısını kullanan bir MQGET çağrısına göre taşınmaz. Ayrıca, CCFAIL ' in tamamlanma kodunu ya da RC2080neden kodunu döndüren bir MQGET çağrısıyla hareket etmiştir.

GMBRWC, GMLK ile belirtilirse:

- Önceden kilitlemiş bir ileti varsa, bu ileti, imlecin altındaki, bu nedenle kilidi açmadan ve yeniden kilitlemeden döndürülmesi gerekir; ileti kilitli kalır.
- Kilitli bir ileti yoksa, göz at imlecinin altındaki ileti (varsa) kilitletir ve uygulamaya geri döndürülür; Göz At imlecinin altında bir ileti yoksa, arama başarısız olur.

GMBRWC, GMLK olmadan belirtilirse:

- Önceden kilitlemiş bir ileti varsa, imlecin altındaki bu ileti olmalıdır. Bu ileti uygulamaya geri döndürüldü ve daha sonra kilidi açıldı. İletinin kilidi açıldığı için, yeniden göz atılabileceğinin ya da (kuyruktan ileti alınması başka bir uygulama tarafından yok edici olarak alınabilir) ileti alınmadığına ilişkin bir garanti yoktur.
- Kilitli bir ileti yoksa, (varsa) göz atma imlecinin altındaki ileti uygulamaya geri döndürülür; Göz At imlecinin altında bir ileti yoksa çağrı başarısız olur.

If GMCMPM is specified with GMBRWC, the browse cursor must identify a message with a MDOFF field in MQMD that is zero. Bu koşul karşılanmazsa, çağrı neden kodu RC2246 ile başarısız olur.

Kuyruk yöneticisinin kuyruklardaki iletilere göz atmasını içeren MQGET çağrılarını için sakladığı grup ve kesim bilgileri, kuyruktan iletilerin kaldırıldığı MQGET çağrılarını için sakladığı grup ve bölüm bilgilerinden ayrıdır.

Bu seçenek, aşağıdaki seçeneklerin hiçbirisiyle geçerli değildir:

- GMBRWF
- GMBRWN
- GMMUC
- GGMSYP
- GMPSYP
- GGMUNLK

Ayrıca, kuyruk göz atma için açılmamışsa da bu bir hatadır.

### **GMMUC**

Göz atma imlecini altına ileti al.

Bu seçenek, MQGMO ' da *GMMO* alanında belirtilen MO\* seçeneklerinden bağımsız olarak, göz atma imlecini işaret ettiği iletinin alınmasına neden olur. İleti kuyruktan kaldırılır.

Göz atma imlecini gösterdiği ileti, son olarak GMBRWF ya da GMBRWN seçeneği kullanılarak alınan bir ileti.

If GMCMPM is specified with GMMUC, the browse cursor must identify a message with a *MDOFF* field in MQMD that is zero. Bu koşul karşılanmazsa, çağrı neden kodu RC2246 ile başarısız olur.

Bu seçenek, aşağıdaki seçeneklerin hiçbirisiyle geçerli değildir:

- GMBRWF
- GMBRWC
- GMBRWN
- GGMUNLK

Ayrıca, hem göz atma hem de giriş için kuyruk açılmamışsa, bu bir hatadır. Göz atma imleci yeniden alınabilir bir ileti göstermiyorsa, MQGET çağrısıyla bir hata döndürülür.

**Kilit seçenekleri:** Aşağıdaki seçenekler kuyruklardaki iletileri kilitlemek için ilişkilendir:

### **GMLK**

İletiyi kilitle.

Bu seçenek, iletinin, kuyruk için açık olan diğer herhangi bir tanıtıcı için görünmez hale gelmesi için, göz atıldığı iletiyi kilitlet. Bu seçenek, yalnızca aşağıdaki seçeneklerden biri de belirtilirse belirlenebilir:

- GMBRWF
- GMBRWN
- GMBRWC

Kuyruk tanıtıcısı başına tek bir ileti kilitletlenir, ancak bu bir mantıksal ileti ya da fiziksel bir ileti olabilir:

- GMCMPM belirtilirse, mantıksal iletiyi oluşturan tüm ileti bölümleri kuyruk tanıtıcılarına kilitletlenir (kuyruğun tümü varsa ve alma için kullanılabilir).
- GMCMPM belirtilmemişse, kuyruk sapına yalnızca tek bir fiziksel ileti kilitletlenir. Bu ileti bir mantıksal iletinin bir bölümü olursa, kilitli kesim mantıksal iletiyi almak ya da göz atmak için GMCMPM kullanan diğer uygulamaları öner.

Kilitletlenen ileti her zaman, göz atma imlecini altındaki ileti ve ileti, GMMUC seçeneğini belirten daha sonraki bir MQGET çağrısıyla kuyruktan kaldırılabilir. Kuyruk tanıtıcısını kullanan diğer MQGET çağrılarını da iletiyi kaldırabilir (örneğin, kilitli iletinin ileti tanıtıcısını belirten bir çağrıdır).



If the call returns completion code CCFAIL, or CCWARN with reason code RC2080, no message is locked.

Uygulama iletiyi kuyruktan kaldırmamaya karar verirse, kilit şu tarafından serbest bırakılır:

- Bu tanıtıcı için, GMBRWF ya da GMBRWN ile (GMLK ile ya da GDLK olmadan) belirtilen başka bir MQGET çağrısı yayınlayarak, çağrı CCOK ya da CCWARN ile tamamlanır, ileti kilitlenir, ancak çağrı CCFAIL ile tamamlanır kilitli kalır. Ancak, aşağıdaki istisnalar geçerlidir:

- RC2080 ile CCWARN döndürülürse, iletinin kilidi açılmaz.
- CCFAIL, RC2033 ile birlikte döndürülürse, iletinin kilidi açılır.

GMLK de belirtilirse, döndürülen ileti kilitlenir. GMLK belirtilmediyse, aramadan sonra kilitli bir ileti yoktur.

GMWT belirtilirse ve hemen kullanılabilir bir ileti yoksa, özgün iletide kilidi açma işlemi beklenenin başlangıcından önce gerçekleşir (çağrıyı sağlamak, hatadan başka bir şekilde serbest değildir).

- GMBRWC (GMLK olmadan) ile bu tanıtıcı için başka bir MQGET çağrısı yayınlayın; çağrı CCOK ya da CCWARN ile tamamlanır, ileti kilitlenir, ancak çağrı CCFAIL ile tamamlanır kilitli kalır. Ancak, aşağıdaki kural dışı durum geçerlidir:
  - RC2080 ile CCWARN döndürülürse, iletinin kilidi açılmaz.
- GMUNLK ile bu tanıtıcı için başka bir MQGET çağrısı yayınlayın.
- Bu tanıtıcı için bir MQCLOSE çağrısının verilmesi (belirtik olarak ya da uygulamanın sona erdirilmesiyle örtük olarak).

Eşlik eden göz atma seçeneğini belirlemek için gereken OOBRW dışında, bu seçeneği belirtmek için özel bir açma seçeneği gerekli değildir.

Bu seçenek, aşağıdaki seçeneklerin hiçbirisiyle geçerli değildir:

- GGMSYP
- GMPSYP
- GGMUNLK

### **GGMUNLK**

İletinin kilidini aç.

Kilidi açılacak iletinin daha önce GMLK seçeneğiyle bir MQGET çağrısı tarafından kilitlenmiş olması gerekir. Bu tanıtıcı için kilitlenmiş bir ileti yoksa, çağrı CCWARN ve RC2209 ile tamamlanır.

**MSGDSC, BUFLN, BUFFER** ve **DATLEN** parametreleri, GGMUNLK belirtilirse denetlenmez ya da değiştirilmez. *BUFFER*' da ileti döndürülmez.

Bu seçeneği belirlemek için özel bir açma seçeneği gerekmez (kilit isteğini ilk sırada OOBRW komutu verebilmek için OOBRW gerekir).

Bu seçenek, aşağıdakiler dışında herhangi bir seçenekle geçerli değildir:

- GDNWT
- GMNSYP

Bu seçeneklerin her ikisi de, belirtilip belirlenmemekte olup olmadığı varsayılır.

**İleti verisi seçenekleri:** Aşağıdaki seçenekler, ileti kuyruktan okunduğunda ileti verilerinin işlenmesiyle ilgilidir:

### **GMATM**

İleti verilerinin kesilmesine izin verir.

İleti arabelleği, iletinin tamamını tutmak için çok küçükse, bu seçenek, MQGET çağrısının arabelleği, arabelleğin tutabileceği kadar ileti doldurabilmesini, bir uyarı tamamlama kodu yayınlamasını ve işlemeyi tamamlayabilmesini sağlar. Bunun anlamı:

- İletilere göz atılırken, göz at imleci döndürülen iletiye kadar ilerletilir.
- İletileri kaldırırken, döndürülen ileti kuyruktan kaldırılır.
- Başka bir hata oluşmazsa neden kodu RC2079 döndürülür.

Bu seçenek olmadan, arabellek hala tutulabildiği kadar ileti doldurur, bir uyarı tamamlama kodu verilir, ancak işleme tamamlanmaz. Bunun anlamı:

- İletilere göz atılırken göz at imleci gelişmiş değildir.
- İletileri kaldırırken, ileti kuyruktan kaldırılmaz.
- Başka bir hata oluşmazsa neden kodu RC2080 döndürülür.

## GMCONV

İleti verilerini dönüştürün.

Bu seçenek, veriler **BUFFER** parametresine kopyalanmadan önce, iletteki uygulama verilerinin, MQGET çağrısındaki **MSGDSC** parametresinde belirtilen *MDCSI* ve *MDENC* değerlerine uygun bir şekilde dönüştürülmesini ister.

İleti konduğunda belirtilen *MDFMT* alanı, iletteki verilerin niteini tanımlamak için dönüştürme işlemi tarafından kabul edilir. İleti verilerinin dönüştürülmesi, yerleşik biçimler için kuyruk yöneticisi tarafından ve diğer biçimler için kullanıcı tarafından yazılan bir çıkışa dönüştürülebilir.

- Dönüştürme başarılı bir şekilde gerçekleştirilirse, **MSGDSC** parametresinde belirtilen *MDCSI* ve *MDENC* alanları, MQGET çağrısından geri dönüşünde değişmeden kalır.
- Dönüştürme işlemi başarılı bir şekilde gerçekleştirilemezse (ancak, MQGET çağrısı hatasız olarak tamamlanırsa), ileti verileri dönüştürülemez olarak döndürülür; *MSGDSC* içindeki *MDCSI* ve *MDENC* alanları, dönüştürülemez ileti için değer ayarlıdır. Bu durumda, tamamlama kodu CCWARN olur.

Her iki durumda da, bu alanlar, **BUFFER** parametresine döndürülen ileti verilerinin karakter kümesi tanıtıcısını ve kodlamasını açıklar.

Kuyruk yöneticisinin dönüştürmeyi gerçekleştirdiği biçim adlarının listesi için "[MQMD \(Message descriptor\) on IBM i](#)" sayfa 1092 içinde açıklanan *MDFMT* alanına bakın.

**Grup ve bölüm seçenekleri:** Aşağıdaki seçenekler, mantıksal iletilerin gruplarındaki ve kısımlarındaki iletilerin işlenmesiyle ilgilidir. Bu tanımlamalar, seçeneklerin anlaşılması konusunda yardımcı olabilir:

### Fiziksel ileti

Bu, bir kuyruğa yerleştirilebilecek ya da kuyruktan kaldırılacak en küçük bilgi birimidir; genellikle, tek bir MQPUT, MQPUT1 ya da MQGET çağrısında belirtilen ya da alınan bilgilere karşılık gelir. Her fiziksel ileti, kendi ileti tanımlayıcısına (MQMD) sahiptir. Genellikle, fiziksel iletiler ileti tanıtıcısı (MQMD ' de *MDMID* alanı) için farklı değerlere göre ayırt edilir; ancak bu, kuyruk yöneticisi tarafından zorunlu kılınmaz.

### Mantıksal ileti

Bu, uygulama bilgilerinin tek bir birimidir. Sistem koşullarının yokluğunda, mantıksal bir ileti fiziksel bir iletiyle aynı olur. Ancak mantıksal iletilerin büyük olduğu durumlarda, sistem kısıtlamaları, mantıksal bir iletiyi bölümler olarak adlandırılan iki ya da daha çok fiziksel iletiye bölmeyi önerebilir ya da zorunlu kılabilir.

Kesimlere ayrılmış bir mantıksal ileti, aynı boş değerli grup tanıtıcısına (MQMD ' de *MDGID* alanı) ve aynı ileti dizisi numarasına (MQMD ' de *MDSEQ* alanı) sahip iki ya da daha çok fiziksel iletten oluşur. Kesimler, mantıksal iletide verilerin başlangıcındaki fiziksel iletteki verilerin görel konumunu veren bölüm görel konumu (MQMD ' de *MDOFF* alanı) için farklı değerlere göre ayırt edilir. Her bir bölüm fiziksel bir ileti olduğu için, mantıksal bir iletteki kesimler genellikle farklı ileti tanıtıcılarına sahiptir.

Ayrılmamış bir mantıksal ileti; ancak, gönderme uygulaması tarafından bölümlenmeye izin verilen, boş değerli olmayan bir grup tanıtıcısı da vardır; ancak, bu durumda, mantıksal ileti bir ileti grubuna ait değilse, o grup tanıtıcısına sahip tek bir fiziksel ileti vardır. Gönderme uygulaması tarafından engellendiği mantıksal iletiler, mantıksal ileti bir ileti grubuna ait değilse, boş değerli grup tanıtıcısına (GINONE) sahip olur.

## İleti grubu

Bu, aynı boş değerli grup tanıtıcısına sahip bir ya da daha fazla mantıksal ileti kümesidir. Gruptaki mantıksal iletiler, 1-n aralığında bir tamsayı olan ileti sıra numarası için farklı değerlere göre ayırılır; burada n, gruptaki mantıksal iletilerin sayısıdır. Mantıksal iletilerden biri ya da daha fazlası bölümlendiyse, grupta n fiziksel ileti sayısı daha fazla olur.

## GMLOGO

Mantıksal iletilerin gruplarındaki ve kısımlarındaki iletiler mantıksal sırayla döndürülür.

Bu seçenek, kuyruk tanıtıcısı için art arda gelen MQGET çağrılarını tarafından iletilerin döndürüldüğü sırayı denetler. Bir etkinin olması için, bu çağrıların her birinde seçeneğin belirlenmesi gerekir.

Kuyruk tanıtıcısı için art arda gelen MQGET çağrılarını için GMLOGO belirtilirse, gruplardaki iletiler, ileti sıra numaraları tarafından verilen sırayla döndürülür ve mantıksal iletilerin bölümleri, bölüm görelili konumları tarafından verilen sırayla döndürülür. Bu sipariş, kuyrukta ileti ve kesimlerin oluştuğu sıradan farklı olabilir.

**Not:** GMLOGO belirtiminin, gruplara ait olmayan ve bölüm olmayan iletilerde herhangi bir olumsuz sonucu yoktur. Sonuç olarak, bu tür iletiler her biri tek bir iletiyle oluşan bir ileti grubuna aitmiş gibi işlem görür. Bu nedenle, gruplar, ileti bölümleri ve kesimlere ayrılmış iletiler gruplarda olmayan iletiler içerebilecek kuyruklardan ileti alınırken GMLOGO ' nun belirtilmesi son derece güvenli olur.

İletileri gereken sırayla geri döndürmek için, kuyruk yöneticisi grup ve bölüm bilgilerini ardışık MQGET çağrılarını arasında tutar. Bu bilgiler, yürürlükteki ileti grubunu ve kuyruk tanıtıcısı için yürürlükteki mantıksal iletiyi, grup ve mantıksal ileti içindeki yürürlükteki konumu ve iletilerin iş birimi içinde alınıp alınmadığını belirtir. Kuyruk yöneticisi bu bilgileri sakladığı için, uygulamanın her MQGET çağrısından önce grup ve bölüm bilgilerini ayarlamaya gerek yoktur. Özellikle, uygulamanın MQMD ' de MDGID, MDSEQve MDOFF alanlarını ayarlamaya gerek olmadığı anlamına gelir. Ancak, uygulamanın her çağrıda GMSYP ya da GMNSYP seçeneğini doğru ayarlamaya gerek vardır.

Kuyruk açıldığında, yürürlükteki ileti grubu yok ve yürürlükteki mantıksal ileti yok. MQGET çağrısıyla MFMIG işaretine sahip bir ileti döndürüldüğünde, ileti grubu yürürlükteki ileti grubu olur. Art arda çağrılarda belirtilen GMLOGO ile, bu grup, aşağıdaki gibi bir ileti döndürülünceye kadar yürürlükteki grup olarak kalır:

- MFSEG olmadan MFLMIG (yani, gruptaki son mantıksal ileti kesimlere ayrılmış değildir) ya da
- MFLSEG ile MFLMIG (yani, döndürülen ileti, gruptaki son mantıksal iletinin son bölümüdür).

Böyle bir ileti döndürüldüğünde, ileti grubu sonlandırılır ve MQGET çağrısının başarıyla tamamlanmasıyla artık yürürlükteki bir grup yoktur. Benzer bir şekilde, MQGET çağrısıyla MFSEG işaretine sahip bir ileti döndürüldüğünde, mantıksal ileti yürürlükteki mantıksal ileti olur ve MFLSEG işaretine sahip ileti döndürüldüğünde mantıksal ileti sonlandırılır.

Seçim ölçütü belirtilmezse, sonraki MQGET çağrılarını (doğru sırayla) kuyrukta ilk ileti grubuna ilişkin iletileri, ikinci ileti grubuna ilişkin iletileri vb. daha fazla ileti kalmayınca kadar çağırır. It is possible to select the particular message groups returned by specifying one or more of the following options in the GMMO field:

- MOMSGI
- MOCORD
- MOGRPI

Ancak, bu seçenekler yalnızca, yürürlükteki ileti grubu ya da mantıksal ileti olmadığında etkindir; bu konuda açıklanan GMMO alanına bakın.

Çizelge 703 sayfa 1072 , kuyruk yöneticisinin MQGET çağrısına geri dönmek için bir ileti bulunmaya çalışırken, kuyruk yöneticisinin aradığı MDMID, MDCID, MDGID, MDSEQve MDOFF alanlarının değerlerini gösterir. Bu, hem iletilerin kuyruktan kaldırılması, hem de kuyruklardaki iletilere göz atılması için geçerlidir. Çizelgedeki kolonlar aşağıdaki anlamlara sahiptir:

## OTURUM KAPAT

Çağrıda GMLOGO seçeneğinin belirtilip belirtilmediğini belirtir.

**Cur grp**

Çağrıdan önce geçerli bir ileti grubunun var olup olmadığını belirtir.

**Cur günlük iletisi**

Çağrıdan önce geçerli bir mantıksal iletinin var olup olmadığını belirtir.

**Diğer kolonlar**

Kuyruk yöneticisinin görünüşü için gereken değerleri gösterir. "Önceki" değeri, kuyruk tanıtıcısı için önceki iletide alan için döndürülen değeri belirtir.

Çizelge 703. Mantıksal iletilerin gruplarındaki ve kısımlarındaki iletilere ilişkin MQGET seçenekleri							
Belirttiğiniz seçenekler	Aramadan önce grup ve günlük-msg durumu		Kuyruk yöneticisinin bakışlarına ilişkin değerler				
	Günlük ORD	Cur Grp	Ccur günlük iletisi	MDMID	MDCID	MDGID	MDSEQ
Evet	Hayır	Hayır	GMMOtarafında n denetlenen	GMMOtarafında n denetlenen	GMMOtarafında n denetlenen	1	0
Evet	Hayır	Evet	Herhangi bir ileti tanıtıcısı	Herhangi bir ilinti tanıtıcısı	Önceki grup tanıtıcısı	1	Önceki görel konum + önceki bölüm uzunluğu
Evet	Evet	Hayır	Herhangi bir ileti tanıtıcısı	Herhangi bir ilinti tanıtıcısı	Önceki grup tanıtıcısı	Önceki sıra numarası + 1	0
Evet	Evet	Evet	Herhangi bir ileti tanıtıcısı	Herhangi bir ilinti tanıtıcısı	Önceki grup tanıtıcısı	Önceki sıra numarası	Önceki görel konum + önceki bölüm uzunluğu
Hayır	Herhangi biri	Herhangi biri	GMMOtarafında n denetlenen	GMMOtarafında n denetlenen	GMMOtarafında n denetlenen	GMMOtarafında n denetlenen	GMMOtarafında n denetlenen

Kuyrukta birden çok ileti grubu varsa ve geri dönüş için uygun olduğunda, gruplar, her gruptaki ilk mantıksal iletinin ilk bölümünün kuyruğunda belirlenen sırayla döndürülür (yani, ileti sıra numaraları 1 olan fiziksel iletiler ve görel gruplar 0 olan, uygun grupların döndürüldüğü sırayı belirleyin).

GMLOGO seçeneği, iş birimlerini aşağıdaki gibi etkiler:

- Bir iş birimi içinde bir gruptaki ilk mantıksal ileti ya da bölüm alınırsa, aynı kuyruk tanıtıcısı kullanılsa, gruptaki diğer tüm mantıksal iletiler ve kesimler bir iş birimi içinde alınmalıdır. Ancak, bunlar aynı iş birimi içinde alınmaları gerekir. Bu, birçok fiziksel iletilerden oluşan bir ileti grubunun kuyruk tanıtıcısı için iki ya da daha fazla sayıda iş birimi bölünmesine olanak sağlar.
- Bir gruptaki ilk mantıksal ileti ya da bölüm bir iş birimi içinde alınmadıysa, aynı kuyruk tanıtıcısı kullanılırsa, gruptaki diğer mantıksal ileti ve kesimler bir iş birimi içinde alınmaz.

If these conditions are not satisfied, the MQGET call fails with reason code RC2245 .

GMLOGO belirtildiğinde, MQGET çağrısında sağlanan MQGMO GMVER2değerinden küçük olmamalıdır ve MQMD MDVER2değerinden küçük olmamalıdır. Bu koşul karşılanmazsa, uygun olduğu şekilde, çağrı neden kodu RC2256 ya da RC2257 ile başarısız olur.

Kuyruk tanıtıcısı için ardışık MQGET çağrıları için GMLOGO belirtilmemişse, ileti gruplarına ait olup olmamaları ya da mantıksal iletilerin parçası olup olmamaları dikkate alınmadan iletiler döndürülür. Bu, belirli bir gruptan ya da mantıksal iletiden gelen iletilerin ya da bölümlerin sıradan geri döndürülebileceği ya da diğer gruplardan ya da mantıksal iletilerden gelen iletilerle ya

da bölümlerle ya da gruplarda olmayan ve kesimler olmayan iletilerle bir araya getirilebileceği anlamına gelir. Bu durumda, ardışık MQGET çağrılarının döndürdüğü iletilerin, bu çağrılarda belirtilen MO\* seçenekleri tarafından denetlenir (bu seçeneklerin ayrıntıları için “IBM üzerinde MQGMO (get-message options)” sayfa 1060 içinde açıklanan GMMO alanına bakın).

Sistem başarısızlığı oluştuktan sonra, ortadaki bir ileti grubunu ya da mantıksal iletiyi yeniden başlatmak için kullanılacak teknik budur. Sistem yeniden başlatıldığında, uygulama MDGID, MDSEQ, MDOFF ve GMMO alanlarını uygun değerlere ayarlayabilir ve daha sonra, GMSYP ya da GMNSYP ayarlarıyla MQGET çağrısını gerektiği şekilde ayarlayabilir, ancak GMLOGO belirtmeden ayarlanır. Bu çağrı başarılı olursa, kuyruk yöneticisi grup ve bölüm bilgilerini saklar ve o kuyruk tanıtıcısını kullanan sonraki MQGET çağrıları, GMLOGO ' yı normal olarak belirtebilir.

Kuyruk yöneticisinin MQGET çağrısı için sakladığı grup ve kesim bilgileri, MQPUT çağrısına ilişkin sakladığı grup ve bölüm bilgilerinden ayrıdır. Ayrıca, kuyruk yöneticisi aşağıdakine ilişkin ayrı bilgileri saklar:

- İletileri kuyruktan kaldırma MQGET çağrıları.
- MQGET, kuyruklardaki iletilere göz atmasını sağlar.

Herhangi bir kuyruk tanıtıcısı için uygulama, MQGET çağrıları ile birlikte GMLOGO belirten MQGET çağrılarını karışık olarak karıştırıyor; ancak, aşağıdaki noktalar dikkate alındığında aşağıda belirtilmeli:

- GMLOGO belirtilmezse, başarılı olan her MQGET çağrısı, kuyruk yöneticisinin kayıtlı grubu ve bölüm bilgilerini döndürülen iletiye karşılık gelen değerlere ayarlamasına neden olur; bu durum, kuyruk tanıtıcısı için kuyruk yöneticisi tarafından korunan varolan grup ve kesim bilgilerinin yerini alır. Yalnızca arama işlemine uygun olan bilgiler (göz atma ya da kaldırma) değiştirilir.
- GMLOGO belirtilmediyse, geçerli bir ileti grubu ya da mantıksal ileti varsa arama başarısız olmaz; ancak, çağrı bir CCWARN tamamlanma koduyla başarılı olabilir. Çizelge 704 sayfa 1073 , oluşabilecek çeşitli vakaları gösterir. Bu durumlarda, tamamlanma kodu CCOK değilse, neden kodu aşağıdakilerden biridir:
  - RC2241
  - RC2242
  - RC2245

**Not:** Kuyruk yöneticisi, bir kuyruğa göz atılırken grup ve bölüm bilgilerini denetmez ya da göz atma için açılan ancak giriş yapmayan bir kuyruğu kapatırken, tamamlama kodu her zaman CCOK (başka hata olmadığı varsayılarak) olur.

Çizelge 704. MQGET ya da MQCLOSE çağrısının grup ve bölüm bilgileriyle tutarlı olmadığı bir sonuç		
Yürürlükteki çağrı	Önceki arama, GMLOGO ile MQGET oldu	Önceki arama, GMLOGO olmadan MQGET oldu
GMLOGO ile MQGET	CCCFAIL	CCCFAIL
GMLOGO olmadan MQGET	CCWARN	CCOK
Sonlandırılmamış bir gruba ya da mantıksal iletiyle MQCLOSE	CCWARN	CCOK

Mantıksal sırayla iletileri ve bölümleri almak isteyen uygulamaların, en basit kullanım seçeneği olduğu için GMLOGO belirtilmesini önerilmektedir. Bu seçenek, kuyruk yöneticisi bu bilgileri yönettiği için, grup ve bölüm bilgilerini yönetme gereksiniminin uygulanını giderir. Ancak, özelleştirilmiş uygulamaların GMLOGO seçeneği tarafından sağlanandan daha fazla denetime gereksinimi olabilir ve bu, bu seçeneği belirtmeyerek elde edilebilir. If this is done, the application must ensure that the MDMID, MDCID, MDGID, MDSEQ, and MDOFF fields in MQMD, and the MO\* options in GMMO in MQGMO, are set correctly, before each MQGET call.

Örneğin, bu iletilerin grup halinde mi, yoksa mantıksal ileti bölümleri mi olduğunu dikkate almadan, aldığı fiziksel iletileri iletmek isteyen bir uygulama GMLOGO belirtmemelidir. Bunun nedeni, kuyruk yöneticisi gönderme ve alma arasında birden çok yol içeren karmaşık bir ağda, fiziksel iletiler sıradan ulaşır olabilir. MQPUT çağrısında GMLOGO ve karşılık gelen PMLOGO belirtmeyerek, iletmeye uygulaması her fiziksel iletiyi alır almaz ve mantıksal sırada gelecek mantıksal sırada beklemeye gerek kalmadan alabilir ve iletebilirler.

GMLOGO, diğer GM\* seçeneklerinin herhangi biriyle ve uygun koşullarda MO\* seçeneklerinin çeşitli biriyle belirtilebilir.

## GMCM

Yalnızca tam mantıksal iletiler alınabilir.

Bu seçenek, MQGET çağrısıyla yalnızca tam mantıksal bir iletinin döndürülebileceğini belirtir. Mantıksal ileti bölümlenmiş bir iletiyse, kuyruk yöneticisi kesimleri yeniden oluşturur ve uygulamaya ilişkin tam mantıksal iletiyi döndürür; mantıksal iletinin bölümlenmiş olması, uygulamanın alınırken uygulamanın belirgin olmamasını sağlar.

**Not:** Kuyruk yöneticisinin ileti bölümlerini yeniden bir araya getirmesine neden olan tek seçenek budur. Belirlenmezse, kuyruğun üzerinde bulunmaları durumunda kesimler uygulamaya tek tek döndürülür (ve MQGET çağrısında belirtilen diğer seçim ölçütlerini karşılar). Bu nedenle, tek tek parça almak istemeyen uygulamaların her zaman GMCM ' yi belirtmesi gerekir.

Bu seçeneği kullanmak için, uygulamanın, iletiyi tamamlamak için yeterli büyüklükte bir arabellek sağlamalı ya da GMATM seçeneğini belirtmesi gerekir.

Kuyruk bazı kesimlere ayrılmış iletiler içeriyorsa (ağda gecikmiş olduğundan ve henüz gelmemiş oldukları için), GMCM belirtilmesi, eksik mantıksal iletilere ait kesimlerin alınmasını önler. Ancak, bu ileti bölümleri yine de **CurrentQDepth** kuyruk özniteliğinin değerine katkıda bulunur; bu, *CurrentQDepth* sıfırdan büyük olmasına rağmen yeniden alınabilecek mantıksal iletiler olabileceği anlamına gelir.

Kalıcı iletiler için, kuyruk yöneticisi kesimleri yalnızca bir iş birimi içinde yeniden birleştirebilir:

- MQGET çağrısı, kullanıcı tanımlı bir iş birimi içinde çalışıyorsa, bu iş birimi kullanılır. Arama, yeniden çevirme işlemi boyunca başarısız olursa, kuyruk yöneticisi yeniden birleştirme sırasında kaldırılan kesimleri yeniden kuyruğa alma işlemini geri yükler. Ancak, hata, iş biriminin başarıyla kesinleştirilmesini engellemektedir.
- Arama, kullanıcı tanımlı bir iş birimi dışında çalışıyorsa ve var olan bir kullanıcı tanımlı iş birimi yoksa, kuyruk yöneticisi yalnızca arama süresi boyunca bir iş birimi yaratır. Arama başarılı olursa, kuyruk yöneticisi iş birimini otomatik olarak kesinleştirir (uygulamanın bu işlemi yapması gerekmez). Arama başarısız olursa, kuyruk yöneticisi iş birimini geri arar.
- Arama, kullanıcı tanımlı bir iş birimi dışında çalışıyorsa, ancak kullanıcı tanımlı bir iş birimi varsa, kuyruk yöneticisi yeniden birleştirme gerçekleştiremez. İleti yeniden yapılanmasını gerektirmiyorsa, arama hala başarılı olabilir. Ancak, ileti yeniden yapılanmasını gerektiriyorsa, çağrı neden kodu RC2255 ile başarısız olur.

Kalıcı olmayan iletiler için, kuyruk yöneticisi yeniden birleştirme işlemini gerçekleştirmek için bir iş biriminin kullanılabilir olmasını gerektirmez.

Bir kesim olan her fiziksel ileti, kendi ileti tanımlayıcısına sahiptir. Tek bir mantıksal ileti oluşturan kesimler için, ileti tanımlayıcısındaki alanların çoğu mantıksal iletteki tüm kesimler için aynıdır; tipik olarak, mantıksal iletide bölümler arasında farklılık gösteren yalnızca *MDMID*, *MDOFFve* *MDMFL* alanları vardır. Ancak, bir bölüm, ara kuyruk yöneticisinde bir kesilen iletiler kuyruğuna yerleştirilirse, DLQ işleyicisi, GMCONV seçeneğini belirten iletiyi alır ve bu, karakterlerin karakter kümesi ya da kodlamasıyla sonuçlanabilir ve bu durum değiştirilmekte. DLQ işleyicisi, kesimi kendi yolunda başarıyla gönderirse, kesim hedef kuyruk yöneticisine en sonunda geldiğinde, mantıksal iletteki diğer kesimlerden farklı olan bir karakter takımı ya da kodlama olabilir.

*MDCSI*, *MDENCya* da her iki alanın farklı olduğu bölümlerden oluşan mantıksal bir ileti, kuyruk yöneticisi tarafından tek bir mantıksal iletiye yeniden birleştirilemez. Bunun yerine, kuyruk yöneticisi, aynı karakter kümesi tanıttıklarına ve kodlamalara sahip mantıksal iletinin

başlangıcındaki ilk birkaç ardışık kesimi yeniden birleştirir ve tamamlanma kodu CCWARN ve neden kodu RC2243 ya da RC2244 ile MQGET çağrısının tamamlanmasını sağlar ve uygun şekilde. Bu durum, GMCONV ' nin belirtilip belirlenmediğine bakılmaksızın gerçekleşir. Geri kalan bölümleri almak için, uygulama MQGET çağrısını GMCMPM seçeneği olmadan yeniden yayınlamalı ve kesimleri tek tek yayınlamalıdır. GMLOGO, siparişteki kalan bölümleri almak için kullanılabilir.

Ayrıca, ileti tanımlayıcısındaki diğer alanları, bölümler arasında farklılık gösteren değerlere ayarlayan bir uygulama için de olanaklıdır. Ancak, alma uygulaması mantıksal iletiyi almak için GMCMPM kullanıyorsa, bunu gerçekleştirmede bir avantaj yoktur. Kuyruk yöneticisi bir mantıksal iletiyi yeniden derlerken, ileti tanımlayıcısında ilk kesime ilişkin ileti tanımlayıcısındaki değerleri döndürür; tek kural dışı durum *MDMFL* alanıdır; kuyruk yöneticisi, yeniden birleştirilen iletinin tek kesim olduğunu göstermek için ayarlar.

Bir rapor iletisi için GMCMPM belirtilirse, kuyruk yöneticisi özel işleme gerçekleştirir. Kuyruk yöneticisi, kuyrukta mantıksal iletiyle ilgili farklı bölümlerle ilgili rapor tipine ilişkin tüm rapor iletilerinin olup olmadığını görmek için kuyruğu denetler. Varsa, GMCMPM belirtilerek tek bir ileti olarak alınabilir. Bunun mümkün olması için, rapor iletileri bir kuyruk yöneticisi tarafından ya da bölümlemenin desteklenmesini destekleyen bir MCA tarafından oluşturulmalıdır; ya da kaynak uygulama en az 100 bayt ileti verisi istemelidir (yani, uygun RO\* D ya da RO\* F seçenekleri belirtilmelidir). Bir kesime ilişkin uygulama verilerinin tam miktarından azsa, eksik byte 'lar, döndürülen rapor iletisinde boş değerler ile değiştirilir.

GMCMPM, GMMUC ya da GMBRWC ile belirlenirse, göz atma imlecinin, MQMD ' de 0 değeri olan bir *MDOFF* alanına sahip bir iletide konumlandırılması gerekir. Bu koşul karşılanmazsa, çağrı neden kodu RC2246 ile başarısız olur.

GMCMPM, bu nedenle belirtilmemesi gereken GMASGA ' yı belirtir.

GMCMPM, GMPSYP dışındaki diğer GM\* seçeneklerinin herhangi biriyle ve MOOFFS dışında MO\* seçeneklerinden herhangi biriyle belirtilebilir.

## GMAMSA

Gruptaki tüm iletilerin kullanılabilir olması gerekir.

Bu seçenek, gruptaki iletilerin yalnızca gruptaki tüm iletiler kullanılabilir durumda olduğunda alınabilmesini sağlar. Kuyrukta bazı iletileri eksik ileti grupları varsa (ağda gecikmiş olduğundan ve henüz gelmedikleri için), GMAMSA belirtilerek eksik gruplara ait iletilerin alınmasını önler. Ancak, bu iletiler yine de **CurrentQDepth** kuyruk özniteliğinin değerine katkıda bulunur; bu, **CurrentQDepth** sıfırdan büyük olmasına rağmen yeniden alınabilecek ileti grupları olabileceğinden anlamına gelir. Yeniden alınabilir başka bir ileti yoksa, belirtilen bekleme aralığı (varsa) sona erdikten sonra neden kodu RC2033 döndürülmez.

GMAMSA 'nın işlenmesi, GMLOGO' nun da belirtilip belirtilmediğine bağlıdır:

- Her iki seçenek de belirlendiyse, GMAMSA yalnızca geçerli grup ya da mantıksal ileti olmadığında etkiler. Geçerli bir grup ya da mantıksal ileti varsa, GMAMSA ' nın iletileri mantıksal sırada işlerken üzerinde kalabileceği anlamına gelir.
- GMAMSA, GMLOGO olmadan belirtilirse, GMAMSA her zaman bir etkiye sahiptir. Bu, gruptaki geri kalan iletileri kaldırabilmek için, gruptaki ilk iletinin kuyruktan kaldırılmasından sonra, seçeneğin kapatılması gerektiği anlamına gelir.

GMAMSA belirtisini belirten bir MQGET çağrısının başarıyla tamamlanması, MQGET çağrısının yayımlandığı sırada gruptaki tüm iletilerin kuyrukta olduğu anlamına gelir. Ancak, diğer uygulamaların gruptaki iletileri kaldırmayı hala başaramadığından haberdar olun (grup, gruptaki ilk iletiyi alan uygulamaya kilitlemez).

Bu seçenek belirlenmezse, grup tamamlanmamış olsa bile gruplara ait iletiler alınabilir.

GMAMSA, bu nedenle belirtilmemesi gereken GMASGA ' yı belirtir.

GMAMSA, diğer GM\* seçeneklerinde ve MO\* seçeneklerinden herhangi biriyle belirtilebilir.

## GMASGA

Bir mantıksal iletteki tüm kesimler kullanılabilir olmalıdır.

Bu seçenek, bir mantıksal iletteki kesimlerin yalnızca mantıksal iletteki tüm kesimler kullanılabilir olduğunda alınabilmeye hazır olacağını belirtir. Kuyrukta kesimlere ayrılmış bazı kesimler varsa (ağda gecikmiş ve henüz gelmedikleri için), GMASGA belirtilerek eksik mantıksal iletelere ait kesimlerin alınmasını engeller. Ancak bu kesimler yine de **CurrentQDepth** kuyruk özneliğinin değerine katkıda bulunur; bu, **CurrentQDepth** sıfırdan büyük olsa bile, yeniden alınabilir mantıksal ileteler olabileceği anlamına gelir. Yeniden alınabilir başka bir ileti yoksa, belirtilen bekleme aralığı (varsa) sona erdikten sonra neden kodu RC2033 döndürülmez.

GMASGA 'nın işlenmesi, GMLOGO' nun da belirtilip belirtilmediğine bağlıdır:

- Her iki seçenek de belirlenirse, GMASGA ' nın geçerli bir mantıksal ileti olmadığına bir etkisi vardır. Yürürlükteki bir mantıksal ileti varsa, GMASGA yoksayılır. Bu, GMASGA ' nın iletileri mantıksal sırada işlerken üzerinde kalabileceği anlamına gelir.
- GMASGA, GMLOGO olmadan belirtilirse, GMASGA her zaman bir etkiye sahiptir. Bu, mantıksal iletiden kalan bölümleri kaldırabilmek için, mantıksal iletiden gelen ilk kesimin kuyruktan kaldırılmasından sonra, seçeneğin kapatılması gerektiği anlamına gelir.

Bu seçenek belirlenmezse, mantıksal ileti tamamlanmamış olsa bile ileti bölümleri alınabilir.

Hem GMCMPM hem de GMASGA herhangi birinin alınabilmesi için tüm kesimlerin kullanılabilir olmasını gerektirirken, eski ileti, iletinin tamamını döndürür, ancak ikincisi kesimlerin birer birer alınmasını sağlar.

Bir rapor iletisi için GMASGA belirtilirse, kuyruk yöneticisi özel işleme gerçekleştirir. Kuyruk yöneticisi, tam mantıksal iletiyi oluşturan her bir bölüm için en az bir rapor iletisi olup olmadığını görmek için kuyruğu denetler. Varsa, GMASGA koşulu karşılanır. Ancak kuyruk yöneticisi, yürürlükteki rapor iletilerinin tipini denetmez ve bu nedenle, rapor iletisinde mantıksal iletinin bölümleriyle ilgili rapor tipleri karışımı olabilir. Sonuç olarak GMASGA 'nın başarısı, GMCMM' nin başarılı olacağı anlamına gelmez. Belirli bir mantıksal iletinin kesimleri için bir rapor tipi karışımı varsa, bu rapor iletileri birer birer alınmalıdır.

GMASGA, diğer GM\* seçeneklerinde ve MO\* seçeneklerinden herhangi biriyle belirlenebilir.

**Varsayılan seçenek:** Tanımlanan seçeneklerden hiçbiri gerekli değilse, aşağıdaki seçenek kullanılabilir:

#### **GGMNONE**

Seçenek belirtilmedi.

Bu değer, başka bir seçeneğin belirlenmemesine işaret etmek için kullanılabilir; tüm seçenekler varsayılan değerlerini varsayar. GGMNONE, program belgelerine yardımcı olmak için tanımlıdır; bu seçeneğin başka biriyle birlikte kullanılması amaçlanmaz, ancak değeri sıfır olduğu için, bu tür bir kullanım algılanamaz.

*GMOPT* alanının ilk değeri *GMNWT* ' dir.

#### **GMRE1 (1 byte 'lık karakter dizisi)**

Ayrılmış.

Bu ayrılmış bir alandır. Bu alanın ilk değeri boşluk karakteridir. *GMVER* , *GMVER2* değerinden küçükse bu alan yok sayılır.

#### **GMRL (10 basamaklı işaretli tamsayı)**

Döndürülen ileti verilerinin uzunluğu (byte).

Bu, kuyruk yöneticisi tarafından, **BUFFER** parametresindeki MQGET çağrısının döndürdüğü ileti verilerinin bayt cinsinden uzunluğuna göre ayarlanan bir çıkış alanıdır. If the queue manager does not support this capability, *GMRL* is set to the value *RLUNDF*.

İletiler kodlamalar ya da karakter kümeleri arasında dönüştürüldüğünde, ileti verileri bazen boyut değiştirebilir. MQGET çağrısından geri dönüş:

- *GMRL* , *RLUNDF* değilse, döndürülen ileti verilerinin bayt sayısı *GMRL* tarafından verilir.



- *GMRL* , *RLUNDF* değerine sahipse, döndürülen ileti verilerinin bayt sayısı genellikle *BUFLLEN* ve *DATLEN*' nin daha küçü tarafından verilir; ancak, *MQGET* çağrısı neden kodu *RC2079* ile tamamlanırsa, bundan küçük olabilir. Bu gerçekleşirse, **BUFFER** parametresindeki önemsiz byte 'lar boş değere ayarlanır.

Aşağıdaki özel değer tanımlıdır:

#### **RUNDF**

Döndürülen verilerin uzunluğu tanımlı değil.

Bu alanın ilk değeri *RLUNDF* ' dir. *GMVER* , *GMVER3* değerinden küçükse bu alan yok sayılır.

#### **GMRQN (48 baytlık karakter dizilimi)**

Hedef kuyruğun çözümlenmiş adı.

Bu, kuyruk yöneticisi tarafından, iletinin alındığı kuyruğun yerel adına, yerel kuyruk yöneticisine tanımlandığı gibi, yerel adına ayarlanacak bir çıkış alanıdır. Bu, aşağıdaki durumlarda kuyruğu açmak için kullanılan addan farklıdır:

- Bir diğer ad kuyruğu açıldı (bu durumda, diğer adın çözüldüğü yerel kuyruğun adı döndürülür) ya da
- Bir model kuyruğu açıldı (bu durumda, dinamik yerel kuyruğun adı döndürülür).

Bu alanın uzunluğu *LNQN* tarafından verilir. Bu alanın ilk değeri 48 boş karakterdir.

#### **GMRS2 (1 byte 'lık karakter dizgisi)**

Ayrılmış.

Bu ayrılmış bir alandır. Bu alanın ilk değeri boşluk karakteridir. *GMVER* , *GMVER4* değerinden küçükse bu alan yok sayılır.

#### **GMSEG (1 byte karakter dizgisi)**

Alınan ileti için ek bölümlenmeye izin verilip verilmediğini belirten işaret.

Aşağıdaki değerlerden birine sahiptir:

#### **EGYHB**

Bölümlenmeye izin verilmiyor.

#### **SİGALW**

Bölümlenmeye izin verilir.

Bu bir çıkış alanıdır. Bu alanın ilk değeri *SEGIHB* 'dir. *GMVER* , *GMVER2* değerinden küçükse bu alan yok sayılır.

#### **MSG1 (10 basamaklı işaretli tamsayı)**

Sinyal.

Bu ayrılmış bir alandır; değeri anlamlı değildir. Bu alanın ilk değeri 0 'tır.

#### **MSG2 (10 basamaklı işaretli tamsayı)**

İşaret tanıtıcısı.

Bu ayrılmış bir alandır; değeri anlamlı değildir.

#### **GMSID (4 baytlık karakter dizgisi)**

Yapı tanıtıcısı.

Değer şu olmalıdır:

#### **GMSIDR**

Get-message options yapısına ilişkin tanıtıcı.

Bu alan her zaman bir giriş alanıdır. Bu alanın ilk değeri *GMSIDV* 'dir.

#### **GMSST (1 byte 'lık karakter dizgisi)**

Alınan iletinin bir mantıksal iletinin bir parçası olup olmadığını belirten işaret.

Aşağıdaki değerlerden birine sahiptir:

**SSNSEG**

İleti bir bölüm değil.

**SSSSEG**

İleti bir bölümdür, ancak mantıksal iletinin son bölümü değildir.

**SSLSEG**

İleti, mantıksal iletinin son bölümüdür.

Mantıksal ileti yalnızca bir kesimden oluşuyorsa, bu değer de döndürülen değerdir.

Bu alan bir çıkış alanıdır. Bu alanın ilk değeri SSNSEG 'dir. *GMVER* , *GMVER2* değerinden küçükse bu alan yok sayılır.

**GMTOK (16 baytlık bit dizgisi)**

İleti simgesi.

Bu ayrılmış bir alandır; değeri anlamlı değildir. Aşağıdaki özel değer tanımlıdır:

**MTKNON**

İleti simgesi yok.

Değer, alanın uzunluğuna ilişkin ikili sıfır değeridir.

Bu alanın uzunluğuna LNMTOK değeri verilir. Bu alanın ilk değeri MTKON değeridir. *GMVER* , *GMVER3* değerinden küçükse bu alan yok sayılır.

**GMVER (10 basamaklı işaretli tamsayı)**

Yapı sürüm numarası.

Değer aşağıdakilerden biri olmalıdır:

**GMVER1**

Version-1 alma iletisi seçenekleri yapısı.

**GMVER2**

Version-2 get-message options yapısı.

**GMVER3**

Version-3 alma iletisi seçenekleri yapısı.

**GMVER4**

Version-4 alma iletisi seçenekleri yapısı.

Yalnızca yapının daha yeni sürümlerinde var olan alanlar, alanların açıklamalarında olduğu gibi tanımlanır. Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

**GDVERC**

İleti alma seçenekleri yapısının yürürlükteki sürümü.

Bu alan her zaman bir giriş alanıdır. Bu alanın ilk değeri GMVER1' dir.

**GMVER (10 basamaklı işaretli tamsayı)**

Yapı sürüm numarası.

Değer aşağıdakilerden biri olmalıdır:

**GMVER1**

Version-1 alma iletisi seçenekleri yapısı.

**GMVER2**

Version-2 get-message options yapısı.

**GMVER3**

Version-3 alma iletisi seçenekleri yapısı.

**GMVER4**

Version-4 alma iletisi seçenekleri yapısı.

Yalnızca yapının daha yeni sürümlerinde var olan alanlar, alanların açıklamalarında olduğu gibi tanımlanır. Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

#### **GDVERC**

İleti alma seçenekleri yapısının yürürlükteki sürümü.

Bu alan her zaman bir giriş alanıdır. Bu alanın ilk değeri GMVER1' dir.

#### **GMWI (10 basamaklı işaretli tamsayı)**

Bekleme aralığı.

Bu, MQGET çağrısının gelmesi için uygun bir ileti beklemesi (yani, MQGET çağrısının **MSGDSC** parametresinde belirtilen seçim ölçütlerini karşılayan bir ileti; daha ayrıntılı bilgi için "MQMD (Message descriptor) on IBM i" sayfa 1092 içinde açıklanan **MDMID** alanına bakın), yaklaşık olarak milisaniye cinsinden ifade edilir. Bu süre geçtikten sonra uygun bir ileti gelmezse, çağrı CCFAIL ile tamamlanır ve neden kodu RC2033 ile tamamlanır.

**GMWI** , **GMWT** seçeneğiyle birlikte kullanılır. Bu seçenek belirlenmezse, bu değer dikkate alınmaz. Belirtilirse, **GMWI** sıfıra eşit ya da sıfırdan büyük olmalı ya da aşağıdaki özel değer olmalıdır:

#### **WIULIM**

Sınırsız bekleme aralığı.

Bu alanın ilk değeri 0 'tır.

### **Başlangıçtaki değerler**

Çizelge 705. MQGMO 'daki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
GMSID	GMSIDR	' GMO- '
GMVER	GMVER1	1
GMOPT	GDNWT	0
GMWI	Yok	0
GMSG1	Yok	0
GMSG2	Yok	0
GMRQN	Yok	Boşluklar
GMMO	MOMSGI + MOCORD	3
GMGST	GSNIG	' - '
GMSST	SSNSEG	' - '
GMSEG	EGYHB	' - '
GMRE1	Yok	' - '
GMTOK	MTKNON	Boş Değerler
GMRL	RUNDF	-1
GMRS2	Yok	' - '
GMMH	HMYOK	0

#### **Notlar:**

1. - simgesi tek bir boş karakteri temsil eder.

## RPG bildirimi

```
D*..1....:....2....:....3....:....4....:....5....:....6....:....7..
D*
D* MQGMO Structure
D*
D* Structure identifier
D  GMSID          1      4    INZ('GMO ')
D* Structure version number
D  GMVER          5      8I 0 INZ(1)
D* Options that control the action ofMQGET
D  GMOPT          9      12I 0 INZ(0)
D* Wait interval
D  GMWI           13     16I 0 INZ(0)
D* Signal
D  GMSG1          17     20I 0 INZ(0)
D* Signal identifier
D  GMSG2          21     24I 0 INZ(0)
D* Resolved name of destination queue
D  GMRQN          25     72    INZ
D* Options controlling selection criteriaused for MQGET
D  GMMO           73     76I 0 INZ(3)
D* Flag indicating whether messageretrieved is in a group
D  GMGST          77     77    INZ(' ')
D* Flag indicating whether messageretrieved is a segment of a
D* logicalmessage
D  GMSST          78     78    INZ(' ')
D* Flag indicating whether furthersegmentation is allowed for themessage
D* retrieved
D  GMSEG          79     79    INZ(' ')
D* Reserved
D  GMRE1          80     80    INZ
D* Message token
D  GMTOK          81     96    INZ('X'0000000000000000-
D  00000000000000000000')
D* Length of message data returned(bytes)
D  GMRL           97     100I 0 INZ(-1)
D* Reserved
D  GMRS2          101    104I 0 INZ(0)
D* Message handle
D  GMMH           105    112I 0 INZ(0)
```

IBM i

## MQIIH (IMS information header) on IBM i

MQIIH yapısı, IBM MQ for z/OSaracılığıyla IMS köprüsine gönderilen bir iletinin başlangıcında var olması gereken bilgileri açıklar.

### Genel Bakış

**Biçim adı:** FMIMS.

**Karakter kümesi ve kodlama:** Özel koşullar, MQIIH yapısı ve uygulama iletisi verileri için kullanılan karakter kümesi ve kodlama için geçerlidir:

- IMS köprü kuyruğunun sahibi olan kuyruk yöneticisine bağlanan uygulamalar, kuyruk yöneticisinin karakter kümesinde ve kodlamasında bulunan bir MQIIH yapısı sağlamalıdır. Bunun nedeni, bu durumda MQIIH yapısının veri dönüştürmesi gerçekleştirilmemesinden kaynaklanır.
- Applications that connect to other queue managers can provide an MQIIH structure that is in any of the supported character sets and encodings; conversion of the MQIIH is performed by the receiving message channel agent connected to the queue manager that owns the IMS bridge queue.

**Not:** Bu konuda bir istisna var. IMS köprü kuyruğunun sahibi olan kuyruk yöneticisi dağıtılmış kuyruklama için CICS kullanıyorsa, MQIIH, IMS köprü kuyruğunun sahibi olan kuyruk yöneticisinin karakter kümesinde ve kodlamasında olmalıdır.

- MQIIH yapısının ardından gelen uygulama iletisi verileri, MQIIH yapısıyla aynı karakter kümesinde ve kodlamada olmalıdır. MQIIH yapısındaki *IICSI* ve *IIENC* alanları, uygulama iletisi verisinin karakter kümesini ve kodlamasını belirtmek için kullanılamaz.

Veri, kuyruk yöneticisi tarafından desteklenen yerleşik biçimlerden biri değilse, uygulama iletisi verilerini dönüştürmek için kullanıcı tarafından bir veri dönüştürme çıkışı sağlanmalıdır.

- [“IMS köprü uygulamaları için passtickets doğrulanıyor” sayfa 1081](#)
- [“Alanlar” sayfa 1081](#)
- [“Başlangıçtaki değerler” sayfa 1084](#)
- [“RPG bildirim” sayfa 1084](#)

## IMS köprü uygulamaları için passtickets doğrulanıyor

It is now possible for IBM MQ administrators to specify the application name to be used for authenticating passtickets, for IMS bridge applications. Bunu yapmak için, uygulama adı STGCLASS nesne tanımlaması için yeni bir PTKTAPPL özneteliği olarak, 1-8 karakterden oluşan alfasayısal bir dizgi olarak belirtilir.

Boş değer, kimlik doğrulamasının önceki IBM MQ yayın düzeyleriyle olduğu gibi, kimlik doğrulama isteğinde uygulama adı akışı olmayan ve bunun yerine kullanılacak MVSxxxx değerinin olduğu anlamına gelir.

A value of 1 - 8 alphanumeric characters must follow the rules for passticket application names as described in the RACF publications.

IBM MQ denetimcileri ve RACF denetimcileri, kullanılacak geçerli uygulama adlarını kabul etmelidir. RACF yöneticisinin, erişim izni verilecek tüm uygulamaların kullanıcı kimliklerine okuma erişimi veren PTKTDATA sınıfında bir tanım yaratması gerekir. The IBM MQ administrator must create or alter the required STGCLASS definitions that specify the application name to be used for passticket authentication.

İlgili bilgiler için *Script (MQSC) Command Reference* adlı elkitabına bakın.

## Alanlar

MQIIH yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

### IIAUT (8 baytlık karakter dizilimi)

RACF parolası ya da passticket.

Bu isteğe bağlıdır; belirtildiyse, güvenlik bağlamı sağlamak üzere IMS 'a gönderilen bir UTOKEN oluşturmak için MQMD güvenlik bağlamındaki kullanıcı kimliğiyle birlikte kullanılır. Belirlenmezse, doğrulama işlemi yapılmaksızın kullanıcı kimliği kullanılır. Bu, RACF anahtarlarının ayarına bağlıdır; bu, kimlik doğrulayıcının var olmasını gerektirebilir.

İlk byte boş ya da boş değerliyse bu değer dikkate alınmaz. Aşağıdaki özel değer kullanılabilir:

#### IAUNON

Kimlik doğrulama yok.

Bu alanın uzunluğu LNAUTH tarafından verilir. Bu alanın ilk değeri IAUNON 'tır.

### IICMT (1 byte 'lık karakter dizisi)

Kesinleştirme kipi.

IMS kesinleştirme kiplerine ilişkin ek bilgi için *OTMA Reference* belgesine bakın. Değer aşağıdakilerden biri olmalıdır:

#### ICMCTS

Gönder ve gönder.

Bu kip, çıkışın iki kez kuyruğa girmesini, ancak bölge doluluk zamanlarının daha kısa olduğunu belirtir. Hızlı yol ve etkileşimli işlemler bu kiple çalışmaz.

#### ICMSTC

O zaman teslim et.

Bu alanın ilk değeri ICMCTS 'dir.

**IICSI (10 basamaklı işaretli tamsayı)**

Ayrılmış.

Bu ayrılmış bir alandır; değeri anlamlı değildir. Bu alanın ilk değeri 0 'tır.

**IIENC (10 basamaklı işaretli tamsayı)**

Ayrılmış.

Bu ayrılmış bir alandır; değeri anlamlı değildir. Bu alanın ilk değeri 0 'tır.

**IIFLG (10 basamaklı işaretli tamsayı)**

Bayraklar.

Değer şu olmalıdır:

**IINONE**

Bayrak yok.

Bu alanın ilk değeri IINONE değeridir.

**IIFMT (8 baytlık karakter dizgisi)**

IBM MQ format name of data that follows MQIIH.

Bu, MQIIH yapısını izleyen verilerin IBM MQ biçim adını belirtir.

MQPUT ya da MQPUT1 çağrısında, uygulama bu alanı veri için uygun değere ayarlamalıdır. Bu alanı kodlamaya ilişkin kurallar, MQMD ' deki *MDFMT* alanı için yapılan kurallarla aynıdır.

Bu alanın uzunluğu LNFMT tarafından verilir. Bu alanın ilk değeri FMNONE ' dir.

**IILEN (10 basamaklı işaretli tamsayı)**

MQIIH yapısının uzunluğu.

Değer şu olmalıdır:

**IILEN1**

IMS bilgi üstbilgisi yapısının uzunluğu.

Bu alanın ilk değeri IILEN1' dir.

**IILTO (8 baytlık karakter dizgisi)**

Mantıksal uçbirim geçersiz kılma.

Bu, GÇ PCB alanına yerleştirilir. Bu, isteğe bağlıdır; belirtilmediyse, TPIPE adı kullanılır. İlk baytın boş olması ya da boş değer olması dikkate alınmaz.

Bu alanın uzunluğu LNLTOV tarafından verilir. Bu alanın ilk değeri 8 boş karakterdir.

**IIMMN (8 baytlık karakter dizilimi)**

İleti biçimi hizmetleri eşlemi adı.

Bu, GÇ PCB alanına yerleştirilir. İsteğe bağlıdır. Giriş sırasında, MID 'yi temsil eder ve MOD' yi temsil eder. İlk byte boş ya da boş değerliyse bu değer dikkate alınmaz.

Bu alanın uzunluğu LNMFMN tarafından verilir. Bu alanın ilk değeri 8 boş karakterdir.

**IIRFM (8 baytlık karakter dizgisi)**

Yanıt iletisinin IBM MQ biçim adı.

Bu, yürürlükteki iletiye yanıt olarak gönderilecek yanıt iletisinin IBM MQ biçim adıdır. Bu kodu kodlamaya ilişkin kurallar, MQMD ' deki *MDFMT* alanı için yapılan kurallarla aynıdır.

Bu alanın uzunluğu LNFMT tarafından verilir. Bu alanın ilk değeri FMNONE ' dir.

**IIRSV (1 byte 'lık karakter dizgisi)**

Ayrılmış.

Bu ayrılmış bir alandır; boş bırakılmalıdır.

### **IISEC (1 baytlık karakter dizilimi)**

Güvenlik kapsamı.

Bu, gerekli IMS güvenlik işlemini gösterir. Aşağıdaki değerler tanımlanır:

#### **ISSCHK**

Güvenlik kapsamını denetleyin.

Denetim bölgesi içinde bir ACEE oluşturulur, ancak bağımlı bölgede oluşturulur.

#### **ÇOK KÖTÜ**

Tam güvenlik kapsamı.

Önbelleğe alınan bir ACEE, denetim bölgesinde oluşturulur ve bağımlı bölgede önbelleğe alınmamış bir ACEE oluşturulur. ISSFUL kullanıyorsanız, ACEE 'nin yerleşik olduğu kullanıcı kimliğinin bağımlı bölgede kullanılan kaynaklara erişimi olduğundan emin olun.

Bu alan için ISSCHK ve ISSFUL belirlenmediyse, ISSCHK kabul edilir.

Bu alanın ilk değeri ISSCHK ' dir.

### **IISID (4 baytlık karakter dizilimi)**

Yapı tanıtıcısı.

Değer şu olmalıdır:

#### **IISIDV**

IMS bilgi üstbilgisi yapısına ilişkin tanıtıcı.

Bu alanın ilk değeri IISIDV ' dir.

### **IITID (16 baytlık bit dizesi)**

Hareket eşgörünümü tanıtıcısı.

This field is used by output messages from IMS so is ignored on first input. If *IITST* is set to *ITSIC*, this must be provided in the next input, and all subsequent inputs, to enable IMS to correlate the messages to the correct conversation. Aşağıdaki özel değer kullanılabilir:

#### **ITINON**

İşlem eşgörünümü tanıtıcısı yok.

Bu alanın uzunluğuna LNTIID değeri verilir. Bu alanın ilk değeri ITINON 'tır.

### **IITST (1 byte 'lık karakter dizgisi)**

Hareket durumu.

Bu, IMS etkileşim durumunu gösterir. Bu, ilk girişte yoksayılr, çünkü etkileşim yok. Sonraki girişlerde, bir sohbetin etkin olup olmadığını belirtir. Çıkış sırasında IMStarafından ayarlanır. Değer aşağıdakilerden biri olmalıdır:

#### **ITSIC**

Sohbete.

#### **ITSNIC**

Konuşmakta değil.

#### **ITSARC**

Hareket durumu verilerini, mimari biçimiyle geri döndürür.

Bu değer yalnızca IMS /DISPLAY TRAN komutu ile kullanılır. Bu, hareket durumu verilerinin, karakter formu yerine IMS archiptited formunun döndürmesine neden olur. Ek ayrıntılar için [Writing IMS transaction programs through IBM MQ](#) ' e bakın.

Bu alanın ilk değeri ITSNIC 'dir.

## IIVER (10 basamaklı işaretli tamsayı)

Yapı sürüm numarası.

Değer şu olmalıdır:

### IIVER1

IMS bilgi üstbilgisi yapısına ilişkin sürüm numarası.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

### IIVERC

IMS bilgi üstbilgisi yapısının yürürlükteki sürümü.

Bu alanın ilk değeri IIVER1' dir.

## Başlangıçtaki değerler

Çizelge 706. MQIIH 'daki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
IISID	IISIDV	' IIH- '
IIVER	IIVER1	1
IILEN	IILEN1	84
IIENC	Yok	0
IICSI	Yok	0
IIFMT	FMNONE	Boşluklar
IIFLG	IINONE	0
IILTO	Yok	Boşluklar
IIMMN	Yok	Boşluklar
IIRFM	FMNONE	Boşluklar
IIAUT	IAUNON	Boşluklar
IITID	ITINON	Boş Değerler
IITST	ITSNIC	' - '
IICMT	ICMCTS	' 0 '
IISEC	ISSCHK	' C '
IIRSV	Yok	' - '

### Notlar:

1. - simgesi tek bir boş karakteri temsil eder.

## RPG bildirim

```
D*..1.....2.....3.....4.....5.....6.....7..
D*
D* MQIIH Structure
D*
D* Structure identifier
D IISID 1 4 INZ(' IIH ')
D* Structure version number
D IIVER 5 8I 0 INZ(1)
D* Length of MQIIH structure
D IILEN 9 12I 0 INZ(84)
```



D*	Reserved			
D	IIENC	13	16I 0	INZ(0)
D*	Reserved			
D	IICSI	17	20I 0	INZ(0)
D*	MQ format name of data that followsMQIIH			
D	IIFMT	21	28	INZ(' ')
D*	Flags			
D	IIFLG	29	32I 0	INZ(0)
D*	Logical terminal override			
D	IILTO	33	40	INZ
D*	Message format services map name			
D	IIMMN	41	48	INZ
D*	MQ format name of reply message			
D	IIRFM	49	56	INZ(' ')
D*	RACF password or passticket			
D	IIAUT	57	64	INZ(' ')
D*	Transaction instance identifier			
D	IITID	65	80	INZ('X'0000000000000000-0000000000000000')
D				
D*	Transaction state			
D	IITST	81	81	INZ(' ')
D*	Commit mode			
D	IICMT	82	82	INZ('0')
D*	Security scope			
D	IISEC	83	83	INZ('C')
D*	Reserved			
D	IIRSV	84	84	INZ

IBM i

## IBM üzerinde MQIMPO (Sorgula ileti özelliği seçenekleri)

MQIMPO yapısı, uygulamaların, iletilerin özelliklerini nasıl sorguladığına ilişkin seçenekleri belirlemesine olanak sağlar.

### Genel Bakış

**Amaç:** Yapı, MQINQMP çağrısında bir giriş parametresidir.

**Karakter kümesi ve kodlama:** MQIMPO 'daki veriler, uygulamanın (ENNAT) uygulama ve kodlamasının karakter kümesinde yer almalıdır.

- [“Alanlar” sayfa 1085](#)
- [“Başlangıçtaki değerler” sayfa 1091](#)
- [“RPG bildirimi” sayfa 1091](#)

### Alanlar

MQIMPO yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

#### IPOPT (10 basamaklı işaretli tamsayı)

İzleyen seçenekler MQINQMP 'nin işlemini denetler. Bu seçeneklerden birini ya da birkaçını belirleyebilirsiniz. birden çok seçenek belirtmek için, değerleri bir araya ekleyin (aynı değişmezi bir kereden fazla eklemeyin) ya da bit ya da işlemi kullanarak değerleri birleştirin (programlama dili, bit işlemleri destekliyorsa). Geçerli olmayan seçenek birleşimleri not edilir; diğer tüm birleşimler geçerlidir.

**Değer verisi seçenekleri:** Aşağıdaki seçenekler, özellik iletiden alınınca değer verilerinin işlenmesiyle ilgilidir.

#### IPCVAL

Bu seçenek, özellik değerinin, MQINQMP çağrısından önce belirtilen *IPREQCSI* ve *IPREQENC* değerlerine, *Value* alanındaki özellik değerini döndürmesi için dönüştürülmesini ister.

- Dönüştürme başarılı olursa, *IPRETCSI* ve *IPRETENC* alanları, MQINQMP çağrısından dönüşte *IPREQCSI* ve *IPREQENC* ile aynı şekilde ayarlanır.

- Dönüştürme başarısız olursa, ancak MQINQMP çağrısı hatasız tamamlanırsa, özellik değeri dönüştürülmez olarak döndürülür.

Özellik bir dizelse, *IPRETCSI* ve *IPRETENC* alanları, dönüştürülenmemiş dizginin karakter takımı ve kodlamasına ayarlanır.

Bu durumda, tamamlanma kodu CCWARN olur; neden kodu RC2466 ile. Özellik imleci döndürülen özelliğe iletilir.

Özellik değeri dönüştürme sırasında genişlerse ve **Value** parametresinin boyutunu aşarsa, değer dönüştürülmez olarak döndürülür; tamamlanma kodu CCFAIL; neden kodu RC2469 olarak ayarlıdır.

MQINQMP çağrısının **DataLength** değiştirgesi, uygulamanın dönüştürülen özellik değerini sığdırmak için gereken arabelleğin büyüklüğünü belirlemesine olanak vermek için, özellik değerinin dönüştürüleceğini döndürür. Özellik imleci değişmeden kalır.

Bu seçenek aşağıdaki bilgileri de ister:

- Özellik adı bir genel arama karakteri içeriyorsa ve
- *IPRETNAMECHRP* alanı, döndürülen ad için bir adres ya da görel konum ile başlatılır.

Döndürülen ad, *IPREQCSI* ve *IPREQENC* değerlerine uygun olarak dönüştürülür.

- Dönüştürme başarılı olursa, *IPRETNAMECHRP* ' ın *VSCCSID* alanı ve döndürülen adın kodlaması, *IPREQCSI* ve *IPREQENC* giriş değerine ayarlanır.
- Dönüştürme başarısız olursa, ancak MQINQMP çağrısı hata ya da uyarı olmadan tamamlanırsa, döndürülen ad dönüştürülmez. Bu durumda, tamamlanma kodu CCWARN olur; neden kodu RC2492 ile.

Özellik imleci döndürülen özelliğe iletilir. Hem değer, hem de ad dönüştürülmezse, RC2466 döndürülür.

Döndürülen ad dönüştürme sırasında genişler ve *RequestedName* alanının *VSBufsize* alanının büyüklüğünü aşarsa, döndürülen dizgi dönüştürülemez olarak bırakılır; tamamlanma kodu CCFAIL ve neden kodu RC2465 olarak ayarlanır.

MQCHARV yapısının *VSLength* alanı, uygulamanın dönüştürülen özellik değerini barındırmak için gereken arabelleğin büyüklüğünü belirlemesine olanak vermek için, özellik değerinin dönüştürüleceğini döndürür. Özellik imleci değişmeden kalır.

## IPCTYP

Bu seçenek, özelliğin değerinin yürürlükteki veri tipinden, MQINQMP çağrısının **Type** parametresinde belirtilen veri tipine dönüştürülmesini ister.

- Dönüştürme başarılı olursa, MQINQMP çağrısının geri döndürülmesi sırasında **Type** parametresi değişmez.
- Dönüştürme başarısız olursa, ancak MQINQMP çağrısı hatasız olarak tamamlanırsa, arama başarısız olur ve neden RC2470 ile başarısız olur. Özellik imleci değişmeden kalır.

Veri tipinin dönüştürülmesi dönüştürme sırasında değer genişletilmesine neden olursa ve dönüştürülen değer, **Value** parametresinin boyutunu aşarsa, değer dönüştürülmez olarak döndürülür; tamamlanma kodu CCFAIL ve neden kodu RC2469 olarak ayarlıdır.

MQINQMP çağrısının **DataLength** değiştirgesi, uygulamanın dönüştürülen özellik değerini sığdırmak için gereken arabelleğin büyüklüğünü belirlemesine olanak vermek için, özellik değerinin dönüştürüleceğini döndürür. Özellik imleci değişmeden kalır.

MQINQMP çağrısının **Type** değiştirgesinin değeri geçerli değilse, çağrı neden RC2473 ile başarısız olur.

İstenen veri tipi dönüşümü desteklenmiyorsa, çağrı neden RC2470 ile başarısız olur. Aşağıdaki veri tipi dönüşümleri desteklenmektedir:

Çizelge 707. Desteklenen veri tipi dönüştürmeleri

Özellik veri tipi	Desteklenen hedef veri tipleri
TYPEYÜKLEME	TYPESTR, TYPI8, TYPI16, TYPI32, TYPI64
TYPBST	TYPSTR
TYPI8	TYPESTR, TYPI16, TYPI32, TYPI64
TYPI16	TYPESTR, TYPI32, TYPI64
TYPI32	TYPESTR, TYPI64
TYPI64	TYPSTR
TYPF32	TYPESTR, TYPF64
TYPF64	TYPSTR
TYPSTR	TYPEYÜKLEME, TYPI8, TYPI16, TYPI32, TYPI64, TYPF32, TYPF64
TYPNUL	Yok

Desteklenen dönüştürmeleri yöneten genel kurallar aşağıdaki gibidir:

- Sayısal özellik değerleri, dönüştürme sırasında hiçbir veri kaybolmaması koşuluyla, bir veri tipinden diğerine dönüştürülebilir.

For example, the value of a property with data type TYPI32 can be converted into a value with data type TYPI64, but cannot be converted into a value with data type TYPI16.

- Herhangi bir veri tipindeki bir özellik değeri bir dizgiye dönüştürülebilir.
- Bir dizgi özelliği değeri, dizginin dönüştürme için doğru şekilde biçimlendirilebilmesi koşuluyla, diğer herhangi bir veri tipine dönüştürülebilir. Bir uygulama, biçimlendirilmemiş bir dizgi özelliği değerini doğru şekilde dönüştürmeye çalışırsa, IBM MQ neden kodu RC2472değerini döndürür.
- Bir uygulama, desteklenmeyen bir dönüştürme girişiminde bulunursa, IBM MQ neden kodu RC2470döndürür.

Bir özellik değerini bir veri tipinden diğerine dönüştürmeye ilişkin belirli kurallar şunlardır:

- Bir TYPBOL özellik değerini bir dizgiye dönüştürürken, TRUE değeri "TRUE" dizgisine dönüştürülür ve false değeri "FALSE" dizgisine dönüştürülür.
- Bir TYPBOL özellik değerini sayısal bir veri tipine dönüştürürken, TRUE değeri bir değere dönüştürülür ve FALSE değeri sifıra dönüştürülür.
- Bir dizgi özelliği değerini TYPEBOL değerine dönüştürürken, "TRUE" ya da "1" dizgisi TRUE değerine dönüştürülür ve "FALSE" ya da "0" dizgisi FALSE olarak dönüştürülür.

"TRUE" ve "FALSE" terimlerinin büyük ve küçük harfe duyarlı olmadığını unutmayın.

Any other string cannot be converted; IBM MQ returns reason code RC2472.

- Bir dizgi özelliği değerini TYPI8, TYPI16, TYPI32 ya da TYPI64veri tipli bir değere dönüştürürken, dizginin şu biçimde olması gerekir:

```
[blanks][sign]digits
```

Dizginin bileşenlerinin anlamları aşağıdaki gibidir:

**blanks**

İsteğe bağlı olarak kullanılan boş karakterler

**sign**

İsteğe bağlı artı işareti (+) ya da eksi işareti (-) karakteri.

## digits

Ardışık rakam karakterleri dizisi (0-9). En az bir sayı karakteri var olmalıdır.

Sayı karakterleri sırasından sonra dizgi, rakam karakteri olmayan diğer karakterleri içerebilir, ancak bu karakterlerin ilk kez ulaşıldığı anda dönüştürme durakları durur. Dizilimin ondalık bir tamsayıyı temsil ettiği varsayılır.

IBM MQ returns reason code RC2472 if the string is not formatted correctly.

- Bir dizgi özelliği değerini TYPF32 ya da TYPF64 veri tipli bir değere dönüştürürken, dizginin şu biçimde olması gerekir:

```
[blanks][sign]digits[.digits][e_char[e_sign]e_digits]
```

Dizginin bileşenlerinin anlamları aşağıdaki gibidir:

### blanks

İsteğe bağlı olarak kullanılan boş karakterler

### sign

İsteğe bağlı artı işareti (+) ya da eksi işareti (-) karakteri.

### digits

Ardışık rakam karakterleri dizisi (0-9). En az bir sayı karakteri var olmalıdır.

### e\_char

"E" ya da "e" olan bir üstel karakter.

### e\_sign

Üstel için isteğe bağlı artı işareti (+) ya da eksi işareti (-) karakteri.

### e\_digits

Üs için bitişik sayı karakterleri dizisi (0-9). Dizgi bir üstel karakter içeriyorsa en az bir sayı karakteri var olmalıdır.

Sayı karakterleri sırasından sonra ya da bir üssü temsil eden isteğe bağlı karakterler sonra, dizgi sayı karakteri olmayan diğer karakterleri içerebilir, ancak bu karakterlere ilk ulaşıldığında dönüştürme durakları durur. Dizginin, 10 'un gücü olan bir üstel bir ondalık kayan noktalı sayıyı temsil ettiği varsayılır.

IBM MQ returns reason code RC2472 if the string is not formatted correctly.

- Bir sayısal özellik değerini bir dizgiye dönüştürürken, değer, bu değere ilişkin ASCII karakterini içeren dizgi değil, değer dizgi gösterimine ondalık sayı olarak dönüştürülür. Örneğin, 65 tamsayı, "A" dizgisine değil, "65" dizgisine dönüştürülür.
- Bir bayt dizilimi özellik değerini bir dizgiye dönüştürürken, her bayt bayt 'ı temsil eden iki onaltılı karaktere dönüştürülür. Örneğin, bayt dizisi {0xF1, 0x12, 0x00, 0xFF} , "F11200FF" dizgisine dönüştürülür.

## IPQLEN

Özellik değerinin tipini ve uzunluğunu sorgulayın. MQINQMP çağrısının **DataLength** parametresine uzunluk değeri döndürülür. Özellik değeri döndürülmez.

Bir *ReturnedName* arabelleği belirtilirse, MQCHARV yapısının *VSLength* alanı, özellik adının uzunluğuna uygun olarak doldurulur. Özellik adı döndürülmez.

**Yineleme seçenekleri:** Aşağıdaki seçenekler, genel arama karakteri içeren bir ad kullanarak, özellikler üzerinde yineleme ile ilgilidir.

## IPINQF

Belirtilen adla eşleşen ilk özelliğe ilişkin bilgi edinmek için. Bu çağrıdan sonra, döndürülen özellik üzerinde bir geçici çizelge oluşturulur.

Bu varsayılan değerdir.

Daha sonra, aynı özelliği yeniden sorgulamak için gerekirse, bir MQINQMP çağrısıyla IPINQC seçeneği kullanılabilir.

Yalnızca bir özellik imlecinin olduğunu unutmayın; bu nedenle, MQINQMP çağrısında belirlenen özellik adı belirlenirse, imleç ilk durumuna getirilir.

Bu seçenek, aşağıdaki seçeneklerin herhangi biri ile geçerli değildir:

İPINQN  
IPINQC

### İPINQN

Belirtilen adla eşleşen sonraki özelliği sorgular, aramayı özellik imlecinden devam ettirir. İmleç, döndürülen özelliğe ilerletilir.

Bu, belirtilen ad için ilk MQINQMP çağrısı, belirtilen ad ile eşleşen ilk özellik döndürülür.

Daha sonra, aynı özelliği yeniden sorgulamak için gerekirse, bir MQINQMP çağrısıyla IPINQC seçeneği kullanılabilir.

İmlecın altındaki özellik silindiyse, MQINQMP, silinen bir sonraki özellik değerini döndürür.

Genel arama karakteriyle eşleşen bir özellik eklenirse, bir yineleme devam ederken, bu özellik yinelemenin tamamlanması sırasında döndürülmeyebilir ya da iade edilmeyebilir. IPINQF kullanılarak yineleme yeniden başlatıldıktan sonra özellik döndürülür.

Silme işlemi devam ederken silinen genel arama karakteriyle eşleşen bir özellik, silme işleminin ardından döndürülmez.

Bu seçenek, aşağıdaki seçeneklerin herhangi biri ile geçerli değildir:

IPINQF  
IPINQC

### IPINQC

Özellik imlecine işaret edilen özelliğin değerini alır. Özellik imlecinin işaret ettiği özellik, IPINQF ya da İPINQN seçeneğini kullanarak son olarak sorgulanana işaret eder.

İleti tanıtıcısı yeniden kullanıldığında, bir MQGET çağrısındaki MQGMO ' nun *MsgHandle* alanında ileti tanıtıcısı belirtildiğinde ya da bir MQPUT çağrısındaki MQPMO yapısının *OriginalMsgHandle* ya da *NewMsgHandle* alanlarında ileti tanıtıcısı belirtildiğinde, özellik imleci sıfırlanır.

Bu seçenek, özellik imleci henüz oluşturulmadığında ya da özellik imlecinin işaret ettiği özellik silindiğinde kullanılıyorsa, çağrı tamamlanma kodu CCFAIL ve neden RC2471 ile başarısız olur.

Bu seçenek, aşağıdaki seçeneklerin herhangi biri ile geçerli değildir:

IPINQF  
İPINQN

Önceden tanımlanan seçeneklerden hiçbiri gerekli değilse, aşağıdaki seçenek kullanılabilir:

### IPYT

Başka bir seçenek belirtilmemeyi belirtmek için bu değeri kullanın; tüm seçenekler varsayılan değerlerini kabul eder.

IPNONE yardımcı programı belgeleri; bu seçeneğin diğeriyle birlikte kullanılması amaçlanmaz, ancak değeri sıfır olduğu için, bu tür bir kullanım algılanamaz.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri IPINQF 'dir.

### İPREQCSI (10 basamaklı işaretli tamsayı)

Değer, bir karakter dizgisiyse, sorgulayan özellik değerinin dönüştürülemesini sağlar. Bu, IPCVAL ya da IPCTYP belirtildiğinde, *ReturnedName* ' in dönüştürüleceği karakter kümesidir.

Bu alanın ilk değeri CSAPL ' dir.

### İPREQENC (10 basamaklı işaretli tamsayı)

Bu, IPCVAL ya da IPCTYP belirtildiğinde, sorgulayan özellik değerinin dönüştürülebileceği kodlamadır.

Bu alanın ilk değeri ENNAT ' tır.

#### **IPRE1 (10 basamaklı işaretli tamsayı)**

Bu ayrılmış bir alandır. Bu alanın ilk değeri boşluk karakteridir.

#### **IPRETCSI (10 basamaklı işaretli tamsayı)**

Çıkışta, bu, MQINQMP çağrısının **Type** değiştirgesi TYPSTR ise, döndürülen değer karakter takısıdır.

IPCVAL seçeneği belirtilirse ve dönüştürme başarılı olursa, dönüş sırasında *ReturnedCCSID* alanı, geçirilen değerle aynı değerde olur.

Bu alanın ilk değeri sıfır.

#### **IBORAC (10 basamaklı işaretli tamsayı)**

Çıkışta, döndürülen değer kodlaması bu.

IPCVAL seçeneği belirtilirse ve dönüştürme başarılı olursa, dönüş sırasında *ReturnedEncoding* alanı, geçirilen değerle aynı değerde olur.

Bu alanın ilk değeri ENNAT ' tır.

#### **IPRETNAMCHRP (10 basamaklı işaretli tamsayı)**

Sorgulamanın gerçek adı.

Girişte, MQCHARV yapısının *VSPtr* ya da *VSOffset* alanı kullanılarak bir dizgi arabelleği geçirilebilir. Dizgi arabelleğindeki uzunluk, MQCHARV yapısının *VSBuFSIZE* alanı kullanılarak belirtilir.

MQINQMP çağrısından geri dönerek, dizgi arabelleğinin adı tam olarak içermeye yetecek kadar uzun olması koşuluyla, dizgi arabelleği sorgulayan özelliğin adıyla tamamlanır. MQCHARV yapısına ilişkin *VSLength* alanı, özellik adının uzunluğuna sahip olarak doldurulur. Döndürülen adın karakter kümesini belirtmek için, MQCHARV yapısına ilişkin *VSCCSID* alanı doldurulur. Bu ad, adın dönüştürülmesinin başarısız olup olmadığını gösterir.

Bu bir giriş/çıkış alanıdır. Bu alanın ilk değeri MQCHARV\_DEFAULT 'tır.

#### **IPSID (10 basamaklı işaretli tamsayı)**

Bu, yapı tanıtıcısıdır. Değer şu olmalıdır:

##### **IPSIDV**

Sorgu iletisi özellik seçenekleri yapısının tanıtıcısı.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri IPSIDV 'dir.

#### **IPTYP (10 basamaklı işaretli tamsayı)**

Özelliğin veri tipinin dizgi gösterimi.

Özellik bir MQRFH2 üstbilgisinde belirtildiyse ve MQRFH2 dt özniteliği tanınmadıysa, özelliğin veri tipini saptamak için bu alan kullanılabilir. *TypeString* , 1208 (UTF-8) kodlu karakter kümesinde döndürülür ve tanınması başarısız olan özelliğin dt özniteliğinin ilk sekiz bayttır.

Bu her zaman bir çıkış alanıdır. Bu alanın başlangıç değeri, C programlama dilinde boş değer, diğer programlama dillerindeki 8 boş karakter dizilimidir.

#### **IPVER (10 basamaklı işaretli tamsayı)**

Bu, yapı sürüm numarasıdır. Değer şu olmalıdır:

## IPVER1

Sorgu iletisi özellik seçenekleri yapısı için sürüm numarası.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

## IPVERC

Sorgu iletisi özellik seçenekleri yapısının yürürlükteki sürümü.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri IPVER1' dir.

## Başlangıçtaki değerler

Çizelge 708. MQIPMO 'daki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
IPSID	IPSIDV	' IMPO '
IPVER	IPVER1	1
IPOPT	IPINQF	
IPREQENC	ENNAT	
IPREQCSI	CSAPL	
IPRETENC	ENNAT	
IPRETCSI	0	
IPRE1	0	
IPRETAMCHRP		
IPTYP		boşluk

## RPG bildirim

```
D* MQIMPO Structure
D*
D*
D* Structure identifier
D IPSID      1   4 INZ('IMPO')
D*
D* Structure version number
D IPVER      5   8I 0 INZ(1)
D*
** Options that control the action of
D* MQINQMP
D IPOPT      9  12I 0 INZ(0)
D*
D* Requested encoding of Value
D IPREQENC   13  16I 0 INZ(273)
D*
** Requested character set identifier
D* of Value
D IPREQCSI   17  20I 0 INZ(-3)
D*
D* Returned encoding of Value
D IPRETENC   21  24I 0 INZ(273)
D*
** Returned character set identifier of
D* Value
D IPRETCSI   25  28I 0 INZ(0)
D*
D* Reserved
D IPRE1      29  32I 0 INZ(0)
D*
D* Returned property name
D* Address of variable length string
D IPRETAMCHRP 33  48* INZ(*NULL)
D* Offset of variable length string
```

```
D IPRETNAMCHRO      49   52I 0 INZ(0)
D* Size of buffer
D IPRETNAMVSBS      53   56I 0 INZ(-1)
D* Length of variable length string
D IPRETNAMCHRL      57   60I 0 INZ(0)
D* CCSID of variable length string
D IPRETNAMCHRC      61   64I 0 INZ(-3)
D*
D* Property data type as a string
D IPTYP              65   72  INZ
```

## IBM i MQMD (Message descriptor) on IBM i

### Genel Bakış

**Amaç:** MQMD yapısı, gönderme ve alma uygulamaları arasında bir ileti dolaşırken, uygulama verileriyle birlikte gönderilen denetim bilgilerini içerir. Bu yapı, MQGET, MQPUT ve MQPUT1 çağrılarında bir giriş/çıkış deęiştirir.

**Sürüm:** MQMD 'nin yürürlükteki sürümü MDVER2' dir. Yalnızca yapının daha yeni sürümlerinde var olan alanlar, izleyen açıklamalarda olduęu gibi tanımlanır.

Saęlanan COPY dosyası, ortam tarafından desteklenen MQMD ' nin en son sürümünü içerir, ancak MDVER alanının ilk deęeri MDVER1olarak ayarlıdır. version-1 yapısında mevcut olmayan alanları kullanmak için, uygulamanın MDVER alanını, gereken sürümün sürüm numarasına ayarlaması gerekir.

version-1 yapısına ilişkin bir bildirge MQMD1adıyla kullanılabilir.

**Karakter kümesi ve kodlama:** MQMD ' deki veriler, ENNAT tarafından verilen yerel kuyruk yöneticisinin **CodedCharSetId** kuyruk yöneticisi öznitelięi ve kodlaması tarafından verilen karakter kümesinde olmalıdır. Ancak, uygulama bir IBM MQ MQI clientolarak çalışıyorsa, yapı istemcinin karakter kümesinde ve kodlamasında olmalıdır.

Gönderme ve alma kuyruęu yöneticileri farklı karakter kümeleri ya da kodlamalar kullanırsa, MQMD ' deki veriler otomatik olarak dönüştürülür. Uygulamanın MQMD ' yi dönüştürmesi gerekli deęildir.

- [“MQMD ' nin farklı sürümlerinin kullanılması” sayfa 1092](#)
- [“İleti baęlamı” sayfa 1093](#)
- [“İleti süre bitimi” sayfa 1093](#)
- [“Alanlar” sayfa 1093](#)
- [“Başlangıçtaki deęerler” sayfa 1132](#)
- [“RPG bildirimi” sayfa 1133](#)

### MQMD ' nin farklı sürümlerinin kullanılması

version-2 MQMD, genellikle bir version-1 MQMD kullanılmasına ve ileti verilerinin MQMDE yapısıyla önceden düzeltilmesine eşdeęerdir. Ancak, MQMDE yapısındaki tüm alanların varsayılan deęerleri varsa, MQMDE atlanabilir. Bir version-1 MQMD artı MQMDE, daha sonra bu bölümde anlatıldıęı gibi kullanılır.

- MQPUT ve MQPUT1 çağrılarında, uygulama bir version-1 MQMD saęlıyorsa, uygulama isteęe baęlı olarak bir MQMDE ile ileti verilerini önleyebilir, MQMD ' de MDFMT alanını FMMDE olarak ayarlayarak bir MQMDE olduęunu göstermek için bu alanı FMDE olarak ayarlayabilir. Uygulama bir MQMDE saęlamazsa, kuyruk yöneticisi MQMDE içindeki alanlar için varsayılan deęerleri devralır.

**Not:** version-2 MQMD 'de var olan, ancak version-1 MQMD' de olmayan bazı alanlar, MQPUT ve MQPUT1 çağrılarında giriş/çıkış alanlarında yer alıyor. Ancak, kuyruk yöneticisi, MQPUT ve MQPUT1 çağrılarının çıktısındaki MQMD ' deki eşdeęer alanlarda herhangi bir deęer döndürmez; uygulama bu çıkış deęerlerini gerektiriyorsa, bu deęer bir version-2 MQMD kullanmalıdır.

- MQGET çağrısında, uygulama bir version-1 MQMD saęlıyorsa, kuyruk yöneticisi önekleri bir MQMDE ile döndürülür; ancak, MQMDE içindeki alanlardan biri ya da daha fazlası varsayılan olmayan bir deęere



sahip olur. MQMD 'deki MDFMT alanı, bir MQMDE' nin var olduğunu göstermek için FMMDE değerine sahip olur.

MQMDE 'deki alanlar için kuyruk yöneticisinin kullandığı varsayılan değerler, [Çizelge 710 sayfa 1132](#) ' ta gösterilen bu alanların ilk değerleriyle aynıdır.

Bir ileti iletim kuyruğunda olduğunda, MQMD ' deki bazı alanlar belirli değerlere ayarlanır; ayrıntılar için [“IBM üzerinde MQXQH \(Transmission-queue header\)” sayfa 1225](#) konusuna bakın.

## İleti bağlamı

MQMD ' deki bazı alanlar ileti bağlamını içerir. Tipik olarak:

- *Tanıttıcı bağlamı* , iletiyi ilk olarak koyan uygulamayla ilişkilidir.
- *Köken bağlamı* , iletiyi en son yerleştiren uygulama ile ilgilidir
- *Kullanıcı bağlamı* , iletiyi ilk olarak koyan uygulama ile ilgilidir.

Bu iki uygulama aynı uygulama olabilir, ancak bunlar aynı zamanda farklı uygulamalar da olabilir (örneğin, bir ileti bir uygulamadan başka bir uygulamaya iletildiğinde).

Tanıttıcı ve kaynak bağlamı genellikle daha önce açıklanan anlamlara sahip olsa da, MQMD ' deki bağlam alanlarının her iki tipinin içeriği aslında, ileti bulunduğu belirlenen PM\* seçeneklerine bağlıdır. Sonuç olarak, kimlik bağlamı, ilk başta iletiyi koyan uygulamayla ilişkili değildir ve kaynak bağlamı, iletiyi en son yerleştiren uygulamayla ilişkili değildir; bu, uygulama takımının tasarımına bağlıdır.

İleti kanalı aracı (MCA), ileti bağlamını hiçbir zaman değiştirmeyen bir uygulama sınıfı vardır. Uzak kuyruk yöneticilerinden gelen iletileri alan MCA ' lar, MQPUT ya da MQPUT1 çağrısında PMSETA bağlam seçeneğini kullanır. Bu işlem, MCA 'nın gönderilen MCA' dan gelen iletiyle birlikte gönderilen ileti bağlamını tam olarak korumasını sağlar. Ancak sonuç, kaynak köken bağlamının en son iletiyi (alma MCA ' yı) koyduğu uygulamayla ilişkili olmadığından, bunun yerine, iletiyi (büyük olasılıkla kaynak uygulamanın kendisi) koyan daha önceki bir uygulamayla ilişkilendirilir.

Daha fazla bilgi için bkz. [İleti bağlamı](#).

## İleti süre bitimi

Yüklü bir kuyruğun süresi dolmuş iletiler (açılmış olan bir kuyruk), süre bitiminden sonra, makul bir süre içinde otomatik olarak kuyruktan kaldırılır. Bu IBM MQ yayınının diğer bazı yeni özellikleri, yüklenen kuyrukların önceki ürün sürümünden daha sık taranmasını sağlar; ancak, yüklenen kuyruklarda süresi dolan iletiler süre bitiminin makul bir dönemi içinde her zaman kaldırılabilir.

## Alanlar

MQMD yapısı aşağıdaki alanları içerir; alanlar alfabetik sırayla anlatılır:

### MDMACC (32 baytlık bit dizgisi)


Muhasebe simgesi.

Bu, iletinin *kimlik bağlamının* bir parçasıdır. İleti bağlamına ilişkin daha fazla bilgi için bkz. [İleti bağlamı](#) ve [Bağlam bilgilerini denetleme](#).

MDACC , uygulamanın uygun şekilde ücretlendirilmesine neden olarak bir uygulamanın çalışmasına neden olur. Kuyruk yöneticisi bu bilgileri bir bit dizgisi olarak değerlendirir ve içeriğini denetmez.

Kuyruk yöneticisi bu bilgileri oluşturduğunda, aşağıdaki gibi ayarlanır:

- Alanın ilk baytı, izleyen baytlardaki muhasebe bilgilerinin uzunluğuna ayarlanır; bu uzunluk, sıfır ile 30 aralığındadır ve ikili tamsayı olarak ilk bayta saklanır.
- İkinci ve sonraki baytlar (uzunluk alanı tarafından belirtildiği gibi), ortama uygun muhasebe bilgilerine ayarlanır.

–  On z/OS the accounting information is set to:

- z/OS toplu işi için, JES JOB kartındaki muhasebe bilgileri ya da EXEC kartındaki JES ACCT deyiminden (virgül ayırıcılar X'FF ' olarak değiştirilir). Bu bilgiler gerekirse, 31 bayta kısaltılır.
  - TSO için, kullanıcının hesap numarası.
  - CICS için, LU 6.2 iş tanıtıcısı birimi (UEPUOWDS) (26 bayt).
  - IMS için, 16 karakterlik IMS kırtarma simgesiyle bitıştırilen 8 karakterlik PSB adı.
  - **IBM i** IBM i' ta, muhasebe bilgileri işe ilişkin muhasebe koduna ayarlanır.
  - **Linux** **AIX** AIX and Linux üzerinde, hesap bilgileri ASCII karakterlerinde sayısal kullanıcı tanımlayıcıyla ayarlanır.
  - **Windows** Windows üzerinde, muhasebe bilgileri sıkıştırılmış biçimde bir Windows NT güvenlik tanıtıcısı (SID) ayarlıdır. SID, MDUID alanında saklanan kullanıcı kimliğini benzersiz olarak tanımlar. SID, MDACC alanında saklandığında, 6 baytlık Tanıtıcı Yetkisi (SID ' nin üçüncü ve sonraki baytlarında bulunur) atlanır. For example, if the Windows NT SID is 28 bytes long, 22 bytes of SID information are stored in the MDACC field.
- Son bayt, hesap belirteci tipine ayarlanır; aşağıdaki değerlerden biri:

**KATILMA**

CICS LUOW tanıtıcısı.

**KATLAN**

PC DOS varsayılan muhasebe simgesi.

**ATTWNT**

Windows güvenlik tanıtıcısı.

**ATT400**

IBM i hesap simgesi.

**AYIRMA**

AIX and Linux sayısal tanıtıcısı.

**AYIKTIR**

Kullanıcı tanımlı hesap belirteci.

**KATIL**

Bilinmeyen muhasebe belirteci tipi.

Muhasebe belirteci tipi, yalnızca aşağıdaki ortamlarda açık bir değere ayarlanır:

- **AIX** AIX
- **IBM i** IBM i
- **Windows** Windows

ve bu sistemlere bağlı IBM MQ MQI clients için.

Diğer ortamlarda, muhasebe belirteci tipi ATTUNK değerine ayarlanır. Bu ortamlarda, alınan hesap simgesiyle ortaya çıkarmak için MDPAT alanı kullanılabilir.

- Diğer tüm baytlar ikili sifıra ayarlanır.

For the MQPUT and MQPUT1 calls, this is an input/output field if PMSETI or PMSETA is specified in the **PMO** parameter. PMSETI ya da PMSETA belirlenmezse, bu alan giriş sırasında yoksayılır ve yalnızca çıkış alanına sahip olur. İleti bağlamına ilişkin daha fazla bilgi için bkz. [İleti bağlamı](#) ve [Bağlam bilgilerini denetleme](#).

Bir MQPUT ya da MQPUT1 çağrısının başarıyla tamamlanmasından sonra, bu alan, bir kuyruğa konduysa, iletiyle birlikte gönderilen MDACC ögesini içerir. Bu, alıkonursa iletiyle birlikte tutulan MDACC değeri olacaktır (alıkonan yayınlara ilişkin ek ayrıntılar için "[MQPMO \(Put-message options\) on IBM i](#)" sayfa 1154 içindeki PMRET tanımına bakın), ancak ileti, gönderilen tüm yayınlarda MDACC ' i geçersiz kılmak için bir değer sağladığından, bu ileti abonelere bir yayın olarak gönderildiğinde, MDACC olarak kullanılmaz. İletinin bağlamı yoksa, alan tamamen ikili olarak sıfırdır.

Bu, MQGET çağrısına ilişkin bir çıkış alanıdır.

Bu alan, kuyruk yöneticisinin karakter kümesine dayalı herhangi bir çeviriye tabi değildir; alan, bir karakter dizisi olarak değil, bitler dizisi olarak işlenir.

Kuyruk yöneticisi, bu alandaki bilgilerle hiçbir şey yapmaz. Hesap, muhasebe amacıyla bilgileri kullanmak istiyorsa, bilgileri yorumlamalıdır.

MDACC alanı için aşağıdaki özel değer kullanılabilir:

#### **YOK**

Hesap belirteci belirtilmedi.

Değer, alanın uzunluğuna ilişkin ikili sıfır değeridir.

Bu alanın uzunluğu LNAACC tarafından verilir. Bu alanın ilk değeri ACNONE 'dir.

### **MDAID (32 baytlık karakter dizilimi)**

Kimlikle ilgili uygulama verileri.

Bu, iletinin *kimlik bağlamının* bir parçasıdır. İleti bağlamına ilişkin daha fazla bilgi için bkz. [İleti bağlamı](#) ve [Bağlam bilgilerini denetleme](#).

MDAID , uygulama takımı tarafından tanımlanan bilgilerdir ve iletiyle ya da başlatıcıya ilişkin ek bilgi sağlamak için kullanılabilir. Kuyruk yöneticisi bu bilgileri karakter verisi olarak değerlendirir, ancak biçiminin biçimini tanımlamaz. Kuyruk yöneticisi bu bilgileri oluşturduğunda, bu bilgi tamamen boştur.

For the MQPUT and MQPUT1 calls, this is an input/output field if PMSETI or PMSETA is specified in the **PMO** parameter. Boş değerli bir karakter varsa, boş değer ve izleyen karakterler kuyruk yöneticisi tarafından boşluklara dönüştürülür. PMSETI ya da PMSETA belirlenmezse, bu alan giriş sırasında yoksayılr ve yalnızca çıkış alanına sahip olur. İleti bağlamına ilişkin daha fazla bilgi için bkz. [İleti bağlamı](#) ve [Bağlam bilgilerini denetleme](#).

Bir MQPUT ya da MQPUT1 çağrısının başarıyla tamamlanmasından sonra, bu alan, bir kuyruğa konduysa, iletiyle birlikte gönderilen MDAID ögesini içerir. Bu, alıkonursa iletiyle birlikte tutulan MDAID değerinin (alıkonan yayınlarla ilgili daha fazla ayrıntı için PMRET tanımına bakın) değeri olur, ancak bu ileti kendilerine gönderilen tüm yayınlarda MDAID ' i geçersiz kılmak için bir değer sağladığından, bu ileti abonelere bir yayın olarak gönderildiğinde, MDAID olarak kullanılmaz. İletinin bağlamı yoksa, alan tümüyle boştur.

Bu, MQGET çağrısına ilişkin bir çıkış alanıdır. Bu alanın uzunluğu LNAIDD tarafından verilir. Bu alanın ilk değeri 32 boş karakterdir.

### **MDAOD (4 baytlık karakter dizilimi)**

Köken ile ilgili uygulama verileri.

Bu, iletinin *başlangıç bağlamının* bir parçasıdır. İleti bağlamına ilişkin daha fazla bilgi için bkz. [İleti bağlamı](#) ve [Bağlam bilgilerini denetleme](#).

MDAOD , iletinin kökeniyle ilgili ek bilgi sağlamak için kullanılabilir. Örneğin, kimlik verilerinin güvenilir olup olmadığını göstermek için uygun kullanıcı yetkisiyle çalışan uygulamalar tarafından ayarlanabilir.

Kuyruk yöneticisi bu bilgileri karakter verisi olarak değerlendirir, ancak biçiminin biçimini tanımlamaz. Kuyruk yöneticisi bu bilgileri oluşturduğunda, bu bilgi tamamen boştur.

For the MQPUT and MQPUT1 calls, this is an input/output field if PMSETA is specified in the **PMO** parameter. Alan içindeki boş değerli bir karakteri izleyen herhangi bir bilgi atılır. Boş değerli karakter ve izleyen karakterler kuyruk yöneticisi tarafından boşluklara dönüştürülür. PMSETA belirtilmediyse, bu alan girişte yoksayılr ve yalnızca çıkış alanıdır.

Bir MQPUT ya da MQPUT1 çağrısının başarıyla tamamlanmasından sonra, bu alan, bir kuyruğa konduysa, iletiyle birlikte gönderilen MDAOD ögesini içerir. Bu, alıkonursa iletiyle birlikte tutulan MDAOD değerinin (alıkonan yayınlarla ilgili daha fazla ayrıntı için PMRET tanımına bakın) değeri olur, ancak bu ileti kendilerine gönderilen tüm yayınlarda MDAOD ' i geçersiz kılmak için bir değer sağladığından, bu

ileti abonelere bir yayın olarak gönderildiğinde, MDAOD olarak kullanılmaz. İletinin bağlamı yoksa, alan tümüyle boştur.

Bu, MQGET çağrısına ilişkin bir çıkış alanıdır. Bu alanın uzunluğu LNAORD tarafından verilir. Bu alanın ilk değeri 4 boş karakterdir.

### **MDBOC (10 basamaklı işaretli tamsayı)**

Arka çıkış sayacı.

Bu, iletinin daha önce MQGET çağrısının bir iş biriminin bir parçası olarak geri döndürüleceği ve daha sonra geriletildiği sayıların sayısıdır. Bu, ileti içeriğine dayalı işleme hatalarının saptanmasında uygulamaya bir yardımcı olarak sağlanır. Sayım, GMBRW\* seçeneklerinden herhangi birini belirten MQGET çağrılarını dışlar.

Bu sayımın doğruluğu, **HardenGetBackout** kuyruk özniteliden etkilenir; bkz. [“Kuyruklara ilişkin öznitelikler” sayfa 1345](#).

Bu, MQGET çağrısına ilişkin bir çıkış alanıdır. Bu, MQPUT ve MQPUT1 çağrıları için yoksayılr. Bu alanın ilk değeri 0 'tır.

### **Nesne Tanıtıcısı (24 baytlık bit dizesi)**

İlinti tanıtıcısı.

Bu, uygulamanın bir iletiyi başka bir iletiyi ilişkilendirmek ya da iletiyi, uygulamanın gerçekleştirdiği diğer çalışmalarla ilişkilendirmek için kullanabileceği bir bayt dizilimidir. İlinti tanıtıcısı, iletinin kalıcı bir özeldir ve kuyruk yöneticisinin yeniden başlatmalarında devam eder. İlinti tanıtıcısı bir karakter dizgisi değil, bir byte dizgisi olduğu için, ileti bir kuyruk yöneticisinden başka bir kuyruk yöneticisinden diğerine geçtiğinde ilinti tanıtıcısı karakter kümeleri arasında dönüştürülmez.

MQPUT ve MQPUT1 çağrıları için, uygulama herhangi bir değer belirtebilir. Kuyruk yöneticisi bu değeri iletiyle iletir ve iletiye ilişkin alma isteğini içeren uygulamaya iletir.

Uygulama, PMNCID değerini belirtiyorsa, kuyruk yöneticisi iletiyle birlikte gönderilen benzersiz bir ilinti tanıtıcısı oluşturur ve MQPUT ya da MQPUT1 çağrısından çıkış için gönderme uygulamasına döndür.

Bu üretilen ilinti tanıtıcısı, saklanırsa iletiyle tutulur ve MQSUB çağrısına geçirilen MQSD ' deki SDCID alanında CINONE belirten abonelere bir yayın olarak gönderilirken ilinti tanıtıcısı olarak kullanılır.

Alıkonan yayınlarla ilgili daha fazla ayrıntı için bkz. [“MQPMO \(Put-message options\) on IBM i” sayfa 1154](#)

Kuyruk yöneticisi ya da bir ileti kanalı aracısı bir rapor iletisi oluşturduğunda, MDCID alanını, özgün iletinin MDREP alanı, ROCMTC ya da ROPCI iletisinin belirlediği şekilde ayarlar. Bu işlemi, rapor iletileri oluşturan uygulamalar da yapmalıdır.

MQGET çağrısına ilişkin MDCID , kuyruktan alınacak belirli bir iletiyi seçmek için kullanılacak beş alandan biridir. Bu alana ilişkin değerlerin nasıl belirtileceğini görmek için, MDMID alanının açıklamasına bakın.

Korelasyon tanıtıcısı olarak CINONE belirtilmesi, MOCORI belirtmeleriyle aynı etkiye sahiptir; yani, herhangi bir ilinti tanımlayıcısı eşleşir.

MQGET çağrısındaki **GMO** parametresinde GMMUC seçeneği belirtilirse, bu alan yoksayılr.

MQGET çağrısından dönüşte, MDCID alanı, döndürülen iletinin ilinti tanıtıcısına (varsa) ayarlanır.

Aşağıdaki özel değerler kullanılabilir:

#### **CINONE**

İlinti tanıtıcısı belirtilmedi.

Değer, alanın uzunluğuna ilişkin ikili sıfır değeridir.

#### **CINEWS**

İleti, yeni bir oturumun başlangıcıdır.

Bu deęer, CICS bridge tarafından yeni bir oturumun bařlangıcının, yani yeni bir ileti dizisinin bařlangıcının belirtildięi gibi tanınır.

MQGET çağrısına iliřkin bu bir giriř/çıkıř alanıdır. MQPUT ve MQPUT1 çağrıları için, PMNCID belirtilmediyse, bu bir giriř alanıdır ve PMNCID belirtildiyse bir çıkıř alanı olur. Bu alanın uzunluęuna LNCID deęeri verilir. Bu alanın ilk deęeri CINONE 'tır.

### **MDCSI (10 basamaklı iřaretli tamsayı)**

Bu, iletteki karakter verilerinin karakter kümesi tanıtıcısını belirtir.

**Not:** MQMD 'deki karakter verileri ve çağrılar üzerinde deęiřtirge olan dięer IBM MQ veri yapıları, kuyruk yöneticisinin karakter kümesinde bulunmalıdır. Bu, kuyruk yöneticisinin **CodedCharSetId** öznitelięi tarafından tanımlanır; bu öznitelięe iliřkin ayrıntılar için "IBM i üzerinde kuyruk yöneticisine iliřkin öznitelikler" sayfa 1376 'e bakın.

Ařaęıdaki özel deęerler kullanılabilir:

#### **CSQM**

Kuyruk yöneticisinin karakter kümesi tanıtıcısı.

İletideki karakter verileri, kuyruk yöneticisinin karakter takımında yer alıyor.

MQPUT ve MQPUT1 çağrılarında, kuyruk yöneticisi bu deęeri, ileti ile gönderilen MQMD içinde, kuyruk yöneticisinin gerçek karakter kümesi tanıtıcısına (true character-set) deęiřtirir. Sonuç olarak, MQGET çağrısıyla CSQM deęeri hiçbir zaman döndürülmez.

#### **CİNT**

Bu yapının karakter kümesi tanıtıcısını devralır.

İletideki karakter verileri, bu yapıyla aynı karakter takımında yer alıyor; bu, kuyruk yöneticisinin karakter takısıdır. (Yalnızca MQMD için, CSINHT, CSQM ile aynı anlamlara sahiptir).

Kuyruk yöneticisi, MQMD 'nin gerçek karakter kümesi tanıtıcısına iletiyle gönderilen MQMD' de bu deęeri deęiřtirir. Hata oluřmaması durumunda, MQGET çağrısıyla CSINHT deęeri döndürülmez.

MQMD 'deki MDPAT alanının deęeri ATBRKR ise, CSINHT kullanılamaz.

#### **CSEMO**

Gömülü karakter takımı tanıtıcısı.

İletideki karakter verileri, ileti verilerinin kendisinde yer alan tanıtıcıyı içeren bir karakter kümesinde bulunur. İletin farklı bölümlerine uygulanırken, ileti verileri içine yerleřtirilmiř herhangi bir sayıda karakter kümesi tanıtıcısı olabilir. Bu deęer, karakter kümelerinin karıřımında veri içeren PCF iletileri için kullanılmalıdır. PCF iletileri, FMPCF biçim adına sahiptir.

Bu deęeri yalnızca MQPUT ve MQPUT1 çağrılarında belirtin. MQGET çağrısında belirtilirse, iletinin dönüřtürülmesini önler.

MQPUT ve MQPUT1 çağrılarında, kuyruk yöneticisi daha önce açıklandığı gibi iletiyle birlikte gönderilen MQMD 'deki CSQM ve CSINHT deęerlerini deęiřtirir, ancak MQPUT ya da MQPUT1 çağrısında belirtilen MQMD' yi deęiřtirmes. Belirtilen deęerde bařka bir denetim gerçekleřtirilmez.

İletileri alan uygulamaların bu alanı uygulamanın bekledięi deęerle karıřlařtırmaları gerekir; deęerler deęiřirse, uygulamanın karakter verilerini iletide dönüřtürmesi gerekebilir.

MQGET çağrısında GMCONV seęeneęi belirtilirse, bu alan bir giriř/çıkıř alanıdır. Uygulama tarafından belirlenen deęer, gerekli olduęunda ileti verilerinin dönüřtürülmesi gereken kodlanmış karakter takımı tanıtıcısıdır. Dönüřtürme iřlemi bařarılı ya da gereksizse, deęer deęiřmeden kalır (CSQM ya da CSINHT deęeri gerçek deęere dönüřtürülürse). Dönüřtürme bařarısız olursa, MQGET çağrısından sonraki deęer, uygulamaya döndürülen dönüřtürülmemiř iletinin kodlanmış karakter kümesi tanıtıcısını temsil eder.

Tersi durumda, bu, MQGET çağrısına iliřkin bir çıkıř alanıdır ve MQPUT ve MQPUT1 çağrılarına iliřkin bir giriř alanı olur. Bu alanın ilk deęeri CSQM 'dir.

## MDENC (10 basamaklı işaretli tamsayı)

İleti verilerinin sayısal kodlaması.

Bu, iletteki sayısal verilerin sayısal kodlamasını belirtir; MQMD yapısındaki sayısal veriler için geçerli değildir. Sayısal kodlama, ikili tamsayılar, paketlenmiş ondalık tamsayılar ve kayan noktalı sayılar için kullanılan gösterimi tanımlar.

MQPUT ya da MQPUT1 çağrısında, uygulama bu alanı veri için uygun değere ayarlamalıdır. Kuyruk yöneticisi, alanın geçerli olup olmadığını denetmez. Aşağıdaki özel değer tanımlıdır:

### ENNAT

Yerel makine kodlaması.

Kodlama, uygulamanın çalışmakta olduğu programlama dili ve makinesi için varsayılan değerdir.

**Not:** Bu değişmezin değeri, programlama diline ve ortama bağlıdır. Bu nedenle, uygulamaların, uygulamanın çalışacağı ortama uygun olan üstbilgi, makro, COPY ya da INCLUDE dosyaları kullanılarak derlenmeleri gerekir.

İletilerin yerleştiği uygulamaların olağan durumda ENNAT belirtmesi gerekir. İletileri alan uygulamaların bu alanı ENNAT değeriyle karşılaştırmaları gerekir; değerler değişirse, uygulamanın sayısal verileri iletide dönüştürmesi gerekebilir. GMCONV seçeneği, kuyruk yöneticisinin iletiyi MQGET çağrısının işlenmesinin bir parçası olarak dönüştürmesini istemek için kullanılabilir.

MQGET çağrısında GMCONV seçeneği belirtilirse, bu alan bir giriş/çıkış alanıdır. Uygulama tarafından belirtilen değer, gerekiyorsa, ileti verilerinin dönüştürülmesi gereken kodlamadır. Dönüştürme başarılı ya da gereksizse, değer değişmeden kalır. Dönüştürme başarısız olursa, MQGET çağrısından sonraki değer, uygulamaya döndürülen dönüştürülmemiş iletinin kodlamasını temsil eder.

Diğer durumlarda, bu, MQGET çağrısına ilişkin bir çıkış alanıdır ve MQPUT ve MQPUT1 çağrılarında ilişkin bir giriş alanı. Bu alanın ilk değeri ENNAT ' tır.

## MDEXP (10 basamaklı işaretli tamsayı)

İleti geçerlik süresi.

Bu, iletiyi yerleştiren uygulama tarafından belirlenen saniyenin onda biri olarak ifade edilen bir dönemdir. İleti, bu süre geçtikten sonra hedef kuyruktan kaldırılmamışsa, atılabilir duruma gelir.

Bu değer, iletinin hedef kuyruktaki harcandığı süreyi ve uzak bir kuyruğa konursa ara iletim kuyruklarında da yer alan süreyi yansıtacak şekilde belirlenir. İleti kanalı araçlarının iletim sürelerini yansıtması da azaltılabilir; bunlar kayda değer bir durum ise. Benzer şekilde, bu iletiyi başka bir kuyruğa ileten bir uygulama, iletiyi önemli bir süre alırsa, gerekirse değeri azaltılabilir. Ancak, süre bitim süresi yaklaşık olarak kabul edilir ve değer, küçük zaman aralıklarını yansıtacak şekilde azaltılmamalı.

İleti, MQGET çağrısını kullanan bir uygulama tarafından alındığında, MDEXP alanı, halen devam eden özgün süre bitimi süresini gösterir.

Bir iletinin süre bitimi süresi dolduktan sonra, kuyruk yöneticisi tarafından atılır. Yürürlükteki somutlamalarda, iletiye göz atma ya da göz atma olmayan bir MQGET çağrısı gerçekleştiğinde bu ileti atılır; bu durumda iletinin süresi önceden dolmamış olabilir. Örneğin, MQGMO ' daki GMMO alanı ile MONONE okuma değerine ayarlanmış bir MQGET çağrısı, FIFO sıralı bir kuyruktan MONONE okuma değerine ayarlanmış olarak, süresi dolan tüm iletilerin süresi dolmamış ilk iletiye kadar atılmasına neden olur. Aynı çağrı, öncelikli olarak sipariş edilen bir kuyruk ile, süresi dolan daha yüksek önceliğe sahip iletileri ve ilk süresi dolmadan önce kuyruğa gelen eşit önceliğe sahip iletileri atar.

Süresi dolmuş bir ileti, bir uygulamaya hiçbir zaman döndürülmez (göz atma yoluyla ya da bir MQGET dışı çağrıya göre), bu nedenle, başarılı bir MQGET çağrısının ardından ileti tanımlayıcısının MDEXP alanındaki değer sıfırdan büyük ya da EIULIM özel değeri olur.

Bir ileti uzak bir kuyruğa konursa, ileti bir ara iletim kuyruğunda olduğunda, ileti hedef kuyruğa ulaşmadan önce sona erebilir (ve atılabilir).

Bir rapor, süresi dolan bir ileti atıldığında oluşturulur (ileti, ROEXP\* rapor seçeneklerinden birini belirtiyorsa). Bu seçeneklerden hiçbiri belirtilmediyse, böyle bir rapor oluşturulmaz; iletinin bu zaman döneminden sonra ilgili olmadığı kabul edilir (daha sonraki bir iletinin yerine geçtiğinden olabilir).

Son kullanma zamanına dayalı iletileri atan başka bir program da istendiyse, uygun bir rapor ileti göndermelidir.

**Not:**

1. If a message is put with an MDEXP time of zero, the MQPUT or MQPUT1 call fails with reason code RC2013; no report message is generated in this case.
2. Süresi sona ermiş olan bir ileti, daha sonraya kadar atılamayabileceğinden, bir kuyruğun süre bitim sürelerini geçmiş ve bu nedenle alma için uygun olmayan iletiler olabilir. Bu iletiler yine de, derinlik tetikleme de içinde olmak üzere, kuyruğun tüm amaçlar için kuyruklardaki ileti sayısına kadar sayılabiliyor.
3. Bir süre bitim raporu oluşturulur (istenirse), ileti atıldığında değil, atılmanın uygun duruma getirildiği zaman atılır.
4. Süresi dolan bir iletinin atılması ve istenirse bir süre bitimi raporunun oluşturulması, bir iş birimi içinde çalışan bir MQGET çağrısının sonucu olarak atma için zamanlanan ileti olsa bile, uygulamanın iş biriminin hiçbir zaman parçası değildir.
5. Bir iş birimi içinde MQGET çağrısıyla yaklaşık süresi dolan bir ileti alınır ve iş birimi daha sonra gerilettiyse, ileti yeniden alınmadan önce atılabilir duruma gelebilir.
6. Süresi dolan bir ileti, GMLK ile bir MQGET çağrısıyla kilitlenirse, ileti GMMUC ile MQGET çağrısıyla alınmadan önce atılabilir duruma gelebilir; bu durumda, sonraki MQGET çağrısında neden kodu RC2034 döndürülmesi durumunda atılabilir.
7. Sıfırdan büyük bir süre bitimi içeren bir istek ileti alındığında, uygulama yanıt iletilerini gönderdiğinde aşağıdaki işlemlerden birini alabilir:
  - Kalan süre bitimini istek iletilerinden yanıt iletilerine kopyalayın.
  - Yanıt iletilerinde süre bitimi süresini sıfırdan büyük bir açık değere ayarlayın.
  - Yanıt iletilerinde süre bitimi süresini EIULIM olarak ayarlayın.

Yapılacak işlem, uygulama takımının tasarımına bağlıdır. Ancak, iletileri bir ölü harfe (teslim edilemeyen ileti) koymak için varsayılan işlem, iletinin kalan süre bitimini korumak ve azaltmaya devam etmek olmalıdır.

8. Tetikleme iletileri her zaman EIULIM ile oluşturulur.
9. A message (normally on a transmission queue) which has a MDFMT name of FMXQH has a second message descriptor within the MQXQH. Bu nedenle, ilişkili iki MDEXP alanı vardır. Bu durumda aşağıdaki ek noktalar belirtilmeli:
  - Bir uygulama uzak bir kuyruğa ileti yerleştirdiğinde, kuyruk yöneticisi iletiyi başlangıçta yerel bir iletim kuyruğuna yerleştirir ve uygulama iletilerinin örneklerini MQXQH yapısıyla ekler. Kuyruk yöneticisi, iki MDEXP alanının değerlerini, uygulama tarafından belirtilen değerle aynı olacak şekilde ayarlar.

Bir uygulama doğrudan yerel iletim kuyruğuna bir ileti yerleştirirse, ileti verileri önceden MQXQH yapısıyla başlamalıdır ve biçim adı FMXQH (ancak kuyruk yöneticisi bunu zorunlu kılmaz) olmalıdır. Bu durumda uygulama, bu iki MDEXP alanının değerlerini aynı olacak şekilde ayarlamamış olmalıdır. (Kuyruk yöneticisi, MQXQH içindeki MDEXP alanının geçerli bir değer içerdiğini ya da ileti verilerinin onu içermek için yeterince uzun olduğunu denetmiyor.)
  - When a message with a MDFMT name of FMXQH is retrieved from a queue (whether this is a normal or a transmission queue), the queue manager decrements both these MDEXP fields with the time spent waiting on the queue. İleti verileri MQXQH ' deki MDEXP alanını içerecek kadar uzun değilse, hata ortaya çıktı.
  - Kuyruk yöneticisi, iletinin atılmasına uygun olup olmadığını test etmek için ayrı bir ileti tanımlayıcısında (MQXQH yapısı içinde gömülü olan ileti tanımlayıcısında yer alan değil) MDEXP alanını kullanır.
  - İki MDEXP alanının başlangıç değerleri farklıysa, bu nedenle, ileti sıfırdan büyük olacak şekilde alındığında ayrı ileti tanımlayıcısında MDEXP için mümkündür (bu nedenle, MQXQH içindeki

MDEXP alanına göre ileti atılmasına uygun değildir). Bu durumda, MQXQH ' deki MDEXP alanı sıfır olarak ayarlanır.

Şu özel değer tanındı:

**EIULIM**

Sınırsız ömür.

İletinin süre bitimi sınırsız.

Bu, MQGET çağrısına ilişkin bir çıkış alanıdır ve MQPUT ve MQPUT1 çağrılarında ilişkin bir giriş alanı. Bu alanın ilk değeri EIULIM 'dir.

**MDFB (10 basamaklı işaretli tamsayı)**

Geri bildirim ya da neden kodu.

Bu, raporun niteliyi belirtmek için MTRPRT tipinde bir iletiyle kullanılır ve yalnızca bu tip iletiyle anlamlıdır. Alan, FB\* değerlerinden birini ya da RC\* değerlerinden birini içerebilir. Geribildirim kodları aşağıdaki gibi gruplanır:

**YOK**

Geri bildirim sağlanmadı.

**FRSFST**

Sistem tarafından oluşturulan geribildirim için en düşük değer.

**FBSLST**

Sistem tarafından oluşturulan geribildirimlere ilişkin en yüksek değer.

FBSLST aracılığıyla sistem tarafından oluşturulan geribildirim kodlarının aralığı, daha sonra bu bölümde (FB\*) listelenen genel geri bildirim kodlarını ve ileti hedef kuyruğa konmadığında oluşabilecek neden kodlarının (RCA) da yer alır.

**FBAFST**

Uygulama tarafından oluşturulan geribildirim için en düşük değer.

**FBALST**

Uygulama tarafından oluşturulan geribildirimlere ilişkin en yüksek değer.

Rapor iletileri oluşturan uygulamalar, kuyruk yöneticisi ya da ileti kanalı aracısı tarafından oluşturulan rapor iletilerinin benzetimini yapmak istemedikleri sürece, sistem aralığında (FBQUIT dışında) geribildirim kodlarını kullanmamalıdır.

MQPUT ya da MQPUT1 çağrılarında, belirtilen değer FBNONE olması ya da sistem aralığı ya da uygulama aralığı içinde olması gerekir. Bu, MDMTdeğeri ne olursa olsun işaretlendi.

**Genel geri bildirim kodları:**

**FBCOA**

Hedef kuyruğa gelmenin onayı (ROCOA ' ya bakın).

**FBCOD**

Teslim alma uygulamasına teslim edilme onayı (ROCOD ' ya bakın).

**FBEXP**

İletinin süresi doldu.

İleti, süre bitimi süresinin dolmasından önce hedef kuyruktan kaldırılmadığından atıldı.

**FBPAN**

Pozitif işlem bildirim (ROPAN ' a bakın).

**FBNAN**

Negatif işlem bildirim (RONAN 'a bakın).

**FBQUIT**

Uygulama sona ermelidir.

Bu, çalışmakta olan bir uygulama programının eşgörünümlerinin denetlenmesini denetlemek için bir iş yükü zamanlama programı tarafından kullanılabilir. Bu geribildirim koduyla bir MTRPRT



iletisi, uygulama programının bir örneğine gönderilmesi, o yönetim ortamına işlemeyi durdurması gerektiğini belirtir. Ancak, bu sözleşmeye uyulması uygulama için önemli bir maddedir; kuyruk yöneticisi tarafından uygulanmaz.

**IMS-bridge geri bildirim kodları:** IMS köprüsü sıfır olmayan bir IMS-OTMA algılama kodu aldığında, IMS köprüsü durum kodunu onaltıdan ondalık değere dönüştürür, FBIERR değerini (300) ekler ve sonucu yanıt iletisinin MDFB alanına yerleştirir. Bu, bir IMS-OTMA hatası oluştuğunda FBILST (399) aralığında FBIFST (301) aralığında bir değere sahip olan geri bildirim kodunda sonuçlanır.

IMS köprüsü aşağıdaki geri bildirim kodlarını üretebilir:

**FBDLZ**

Veri uzunluğu sıfır.

İletinin uygulama verilerinde bir kesim uzunluğu sıfırdı.

**FBDLN**

Veri uzunluğu negatif.

İletinin uygulama verilerinde bir bölüm uzunluğu negatif oldu.

**FBDLTB**

Veri uzunluğu çok büyük.

İletinin uygulama verilerinde bir bölüm uzunluğu çok büyük.

**FBBUFO**

Arabellek taşması.

Uzunluk alanlarından birinin değeri, verilerin ileti arabelleğindeki taşmasına neden olur.

**FBLOB1**

Hata, hata olarak bir.

Uzunluk alanlarından birinin değeri bir byte çok kısa.

**FBIIH**

MQIIH yapısı geçerli değil ya da eksik.

MQMD 'deki MDFMT alanı FMIMSbelirtir, ancak ileti geçerli bir MQIIH yapısıyla başlamamaktadır.

**FBNAFI**

Kullanıcı kimliği, IMSiçinde kullanım için yetkili değil.

MQMD ileti tanımlayıcısında bulunan kullanıcı kimliği ya da MQIIH yapısındaki IIAUT alanında bulunan parola, IMS köprüsü tarafından gerçekleştirilen geçerlilik denetiminde başarısız oldu. Sonuç olarak ileti IMS'e iletilmedi.

**FBIERR**

IMStarafından beklenmeyen bir hata döndürüldü.

IMStarafından beklenmeyen bir hata döndürüldü. Hatayla ilgili daha fazla bilgi için IMS köprüsünün bulunduğu sistemde IBM MQ hata günlüğüne bakın.

**FBIFST**

IMStarafından oluşturulan geribildirime ilişkin en düşük değer.

IMStarafından oluşturulan geribildirim kodları, FBIFST (300) aralığını FBILST (399) ile doldurur. IMS-OTMA algılama kodunun kendisi MDFB eksi FBIERR 'dir.

**FBILST**

IMStarafından oluşturulan geribildirime ilişkin en yüksek değer.

**CICS-bridge geribildirim kodları:** Aşağıdaki bildirim kodları CICS bridgetarafından oluşturulabilir:

**FBCAAB**

Uygulama sona erdirildi.

İletide belirtilen uygulama programı sona erdirildi. Bu geribildirim kodu, yalnızca MQDLH yapısının DLREA alanında gerçekleşir.

**FCANS**

Uygulama başlatılamaz.

İletide belirtilen uygulama programına ilişkin EXEC CICS LINK başarısız oldu. Bu geribildirim kodu, yalnızca MQDLH yapısının DLREA alanında gerçekleşir.

**BCBBRF**

CICS bridge olağan dışı hata işlemeyi tamamlamadan olağan dışı bir şekilde sonlandırıldı.

**FBST**

Karakter takımı tanıtıcısı geçerli değil.

**FBCIHE**

CICS bilgi üstbilgisi yapısı eksik ya da geçerli değil.

**FBCCAE**

CICS commarea uzunluğu geçerli değil.

**FBCCIE**

İlinti tanıtıcısı geçerli değil.

**FBDDLQ**

Ölü-mektup kuyruğu kullanılmıyor.

CICS bridge görevi, bu istekle gönderilen bir yanıtı ölü-mektup kuyruğuna kopyalayamadı. İstek geriletildi.

**GİCELİ**

Kodlama geçerli değil.

**FCBE**

CICS bridge beklenmeyen bir hata saptadı.

Bu geribildirim kodu, yalnızca MQDLH yapısının DLREA alanında gerçekleşir.

**FCNTA**

Kullanıcı kimliği yetkili değil ya da parola geçerli değil.

Bu geribildirim kodu, yalnızca MQDLH yapısının DLREA alanında gerçekleşir.

**FCBUBO**

İş birimi gerilettili.

İş birimi geriletildi; bunun nedeni aşağıdakilerden biri olabilir:

- Aynı iş birimi içinde başka bir istek işlenirken bir hata saptandı.
- Bir CICS olağandışı sonu, iş birimi devam ederken oluştu.

**FCUMUZ**

İş birimi denetim alanı CIUOW geçerli değil.

**MQ neden kodları:** Kural dışı durum raporu iletileri için MDFB , bir MQ neden kodu içerir. Olası neden kodları arasında şunlar yer alıyor:

**RC2051**

(2051, X'803 ') Kuyruk araması engellenmiş olarak çağrılıyor.

**RC2053**

(2053, X'805 ') Kuyruk, ileti sayısı üst sınırını zaten içeriyor.

**RC2035**

(2035, X'7F3') Erişim için yetkili değil.

**RC2056**

(2056, X'808 ') Kuyruğun diskte kullanılabilir alan yok.

**RC2048**

(2048, X'800 ') Kuyruk kalıcı iletileri desteklemiyor.

**RC2031**

(2031, X'7EF') İleti uzunluğu kuyruk yöneticisi için üst sınırdan büyük.

## RC2030

(2030, X'7EE') İleti uzunluğu, kuyruk için üst sınırdan büyük.

Bu, MQGET çağrısına ilişkin bir çıkış alanıdır ve MQPUT ve MQPUT1 çağrıları için bir giriş alanı. Bu alanın ilk değeri FBNONE değeridir.

### MDFMT (8 baytlık karakter dizilimi)

İleti verilerinin adı biçiminin adı.

Bu ad, iletiyi gönderenin, iletteki verilerin doğasını alıcıya göstermek için kullanabileceği bir addir. Ad için, kuyruk yöneticisinin karakter kümesinde yer alan karakterler belirtilebilir; ancak, adın aşağıda belirtilenlerle kısıtlanmaması önerilir:

- A-Z arasındaki büyük harf
- 0-9 arası sayısal rakamlar

Diğer karakterler kullanılırsa, gönderme ve alma kuyruğu yöneticilerinin karakter kümeleri arasındaki adı çevirmek olanaklı olmayabilir.

Ad, alanın uzunluğuna boşluklarla doldurulmalı ya da alanın sonundan önce adı sonlandırmak için boş bir karakter olmalıdır; boş değer ve sonraki karakterler boşluk olarak işlenir. Baştaki ya da gömülü boşluklar içeren bir ad belirlemeyin. MQGET çağrısına ilişkin kuyruk yöneticisi, boşlukla doldurulan adı alanın uzunluğuna göre döndürür.

Kuyruk yöneticisi, adın önceden açıklanan önerilere uygun olup olmadığını denetleyemez.

"MQ", üst, alt ve karışık büyüklerde başlayan adlar, kuyruk yöneticisi tarafından tanımlanan anlamlara sahiptir; kendi biçimleriniz için bu harflerle başlayan adları kullanmamanız gerekir. Kuyruk yöneticisi yerleşik biçimleri şunlardır:

### FMNONE

Biçim adı yok.

Verilerin türü tanımlı değil. Bu, ileti, GMCONV seçeneği kullanılarak bir kuyruktan alındığında verilerin dönüştürülemeyeceğini belirtir.

MQGET çağrısında GMCONV belirtilirse ve iletteki verilerin karakter kümesi ya da kodlaması **MSGDSC** değiştirilmesinde belirtilenizden farklıysa, ileti aşağıdaki tamamlanma ve neden kodlarıyla döndürülür (başka hata varsayılarak):

- FMNONE verileri iletinin başında yer aldıysa, tamamlanma kodu CCWARN ve neden kodu RC2110 .
- FMNONE iletisi iletinin sonunda (yani bir ya da daha çok MQ üstbilgi yapısından önce gelen), tamamlanma kodu CCOK ve neden kodu RCNONE. MQ üstbilgi yapıları, bu durumda istenen karakter kümesine ve kodlamaya dönüştürüldü.

### FMADMN

Komut sunucusu isteği/yanıt iletisi.

İleti, programlanabilir komut biçiminde (PCF) komut sunucusu isteği ya da yanıt iletidir.

MQGET çağrısında GMCONV seçeneği belirtilirse, bu biçimin iletileri dönüştürülebilmektedir.

Programlanabilir komut biçimi iletilerinin kullanılmasına ilişkin ek bilgi için [Programlanabilir Komut Biçimlerinin Kullanılması](#) konusuna bakın.

### FMCICS

CICS bilgi üstbilgisi.

İleti verileri, uygulama verileri tarafından izlenen CICS bilgi üstbilgisi MQCIH ile başlar. Uygulama verilerine ilişkin biçim adı, MQCIH yapısındaki CIFMT alanı tarafından verilir.

### FMCMD1

1 komut yanıtı iletisi yazın.

İleti, nesne sayısını, tamamlanma kodunu ve neden kodunu içeren bir MQSC komut sunucusu yanıt iletidir. MQGET çağrısında GMCONV seçeneği belirtilirse, bu biçimin iletileri dönüştürülebilmektedir.

#### **FMCM2**

Tip 2 komut yanıt iletisi.

İleti, istenen nesne (ler) e ilişkin bilgileri içeren bir MQSC komut sunucusu yanıt iletidir. MQGET çağrısında GMCONV seçeneği belirtilirse, bu biçimin iletileri dönüştürülebilmektedir.

#### **FBIDLH**

-Ölü harf üstbilgisi.

İleti verileri, MQDLH adlı ölü-harf üstbilgisi ile başlar. Özgün iletiden gelen veriler, MQDLH yapısından hemen sonra gelir. Özgün ileti verilerinin biçim adı, MQDLH yapısındaki DLFMT alanı tarafından verilir; bu yapının ayrıntıları için [“IBM üzerindeMQDLH \(Dead-letter üstbilgisi\)” sayfa 1048 ' e bakın](#). MQGET çağrısında GMCONV seçeneği belirtilirse, bu biçimin iletileri dönüştürülebilmektedir.

Bir MDFMT FMDLH değeri olan iletiler için COA ve COD raporları oluşturulmaz.

#### **FMDH**

Dağıtım listesi üstbilgisi.

İleti verileri, dağıtım listesi üstbilgisiyle (MQDH) başlar; bu, MQOR ve MQPMR kayıtlarının dizilerini içerir. Dağıtım listesi üstbilgisinde ek veriler de izlenebilir. Ek verilerin biçimi (varsa), MQDH yapısındaki DHFMT alanı tarafından verilir; bu yapının ayrıntıları için [“IBM üzerinde MQDH \(Dağıtım üstbilgisi\)” sayfa 1044 ' e bakın](#). MQGET çağrısında GMCONV seçeneği belirtilirse, FMDH biçimi olan iletiler dönüştürülebilmektedir.

#### **FMEVNT**

Olay iletisi.

İleti, oluşan bir olayı bildiren bir MQ olay iletidir. Olay iletileri, programlanabilir komutlarla aynı yapıya sahiptir; bu yapıyla ilgili ek bilgi için [Komutlar ve yanıtlar için yapılarbaşlıklı konuya bakın](#). Olaylarla ilgili bilgi için [Olay izlemebaşlıklı konuya bakın](#).

MQGET çağrısında GMCONV seçeneği belirtilirse,Version-1 olay iletileri dönüştürülebiliyor.

#### **FMIMS**

IMS bilgi üstbilgisi.

İleti verileri, uygulama verileri tarafından izlenen IMS bilgi üstbilgisi MQIIH (MQIIH) ile başlar. Uygulama verilerine ilişkin biçim adı, MQIIH yapısındaki *IIFMT* alanı tarafından verilir. MQGET çağrısında GMCONV seçeneği belirtilirse, bu biçimin iletileri dönüştürülebilmektedir.

#### **FMIMVS**

IMS değişken dizgisi.

İleti, 11zzcccbiçiminin bir dizgisi olan IMS değişkeni dizgisidir; burada:

##### **11**

IMS değişken dizgi ögesinin toplam uzunluğunu belirten 2 baytlık bir uzunluk alanıdır. Bu uzunluk, 11 (2 byte) uzunluğuna eşittir ve zz (2 byte) uzunluğunun yanı sıra, karakter dizgisinin uzunluğuna eşittir. 11 , MDENC alanı tarafından belirtilen kodlamadaki 2 baytlık ikili tamsayıdır.

##### **zz**

IMS için önemli olan işaretleri içeren 2 baytlık bir alandır. zz , iki adet 1 byte bit dizgi alanlarından oluşan bir bayt dizesidir ve gönderenden alıcıya değişiklik yapılmadan iletilir (yani, zz herhangi bir dönüştürmeye tabi değildir).

##### **ccc**

11-4 karakterlerini içeren değişken uzunluklu bir karakter dizilimidir. ccc , MDCSI alanı tarafından belirtilen karakter kümesinde yer alıyor.

MQGET çağrısında GMCONV seçeneği belirtilirse, bu biçimin iletileri dönüştürülebilmektedir.

**FMMDE**

İleti tanımlayıcısı uzantısı.

İleti verileri, MQMDE ileti tanımlayıcısı uzantısıyla başlar ve isteğe bağlı olarak diğer veriler de (genellikle uygulama ileti verileri) izler. MQMDE 'yi izleyen verilerin biçim adı, karakter kümesi ve kodlaması, MQMDE' deki MEFMT, MECSive MEENC alanları tarafından verilir. Bu yapıyla ilgili ayrıntılar için bkz. [“MQMDE \(Message descriptor extension\) on IBM i” sayfa 1134](#) . MQGET çağrısında GMCONV seçeneği belirtilirse, bu biçimin iletileri dönüştürülebilmektedir.

**FMPCF**

Programlanabilir komut biçiminde (PCF) kullanıcı tanımlı ileti.

İleti, programlanabilir komut biçimi (PCF) iletinin yapısına uygun, kullanıcı tanımlı bir iletidir. MQGET çağrısında GMCONV seçeneği belirtilirse, bu biçimin iletileri dönüştürülebilmektedir. Programlanabilir komut biçimi iletileri kullanmaya ilişkin ek bilgi için [Programlanabilir Komut Biçimlerinin Kullanılması](#) konusuna bakın.

**FMRMH**

Başvuru ileti üstbilgisi.

İleti verileri, MQRMH başvuru ileti üstbilgisiyle başlar ve isteğe bağlı olarak diğer veriler tarafından takip edilir. Verilerin biçim adı, karakter kümesi ve kodlaması, MQRMH ' deki RMFMT, RMCSive RMENC alanları tarafından verilir. Bu yapıyla ilgili ayrıntılar için bkz. [“IBM üzerinde MQRMH \(Başvuru ileti üstbilgisi\)” sayfa 1180](#) . MQGET çağrısında GMCONV seçeneği belirtilirse, bu biçimin iletileri dönüştürülebilmektedir.

**FNRFH**

Kurallar ve biçimlendirme üstbilgisi.

İleti verileri, kural ve biçimlendirme üstbilgisi MQRFH ile başlar ve isteğe bağlı olarak diğer veriler tarafından takip edilir. Verilerin biçim adı, karakter kümesi ve kodlanması (varsa), MQRFH içindeki RFFMT, RFCSive RFENC alanları tarafından verilir. MQGET çağrısında GMCONV seçeneği belirtilirse, bu biçimin iletileri dönüştürülebilmektedir.

**FMRFH2**

Kurallar ve biçimlendirme üstbilgisi sürüm 2.

İleti verileri, version-2 kuralları ve biçimlendirme üstbilgisi MQRFH2 ile başlar ve isteğe bağlı olarak diğer veriler tarafından takip edilir. İsteğe bağlı verilerin biçim adı, karakter kümesi ve kodlaması (varsa), MQRFH2 içinde RF2FMT, RF2CSive RF2ENC alanları tarafından verilir. MQGET çağrısında GMCONV seçeneği belirtilirse, bu biçimin iletileri dönüştürülebilmektedir.

**FMSTR**

Tamamen karakterlerden oluşan ileti.

Uygulama ileti verileri bir SBCS dizgisi (tek baytlık karakter takımı) ya da bir DBCS dizilimi (çift baytlık karakter takımı) olabilir. MQGET çağrısında GMCONV seçeneği belirtilirse, bu biçimin iletileri dönüştürülebilmektedir.

**FMTM**

Tetikleyici ileti.

İleti, MQTM yapısı tarafından açıklanan bir tetikleme iletidir; bu yapının ayrıntıları için [“MQTM- Tetikleme ileti” sayfa 1215](#) konusuna bakın. MQGET çağrısında GMCONV seçeneği belirtilirse, bu biçimin iletileri dönüştürülebilmektedir.

**FNWIH**

İş bilgileri üstbilgisi.

İleti verileri, uygulama verileri tarafından izlenen iş bilgisi üstbilgisi MQWIH ile başlar. Uygulama verilerine ilişkin biçim adı, MQWIH yapısındaki WIFMT alanı tarafından verilir.

**FMXQH**

İletim kuyruğu üstbilgisi.

İleti verileri, MQXQH iletim kuyruğu üstbilgisiyle başlar. Özgün iletiden gelen veriler, MQXQH yapısından hemen sonra gelir. Özgün ileti verilerinin biçim adı, iletim kuyruğu üstbilgisi MQXQH ' nin bir parçası olan MQMD yapısındaki MDFMT alanı tarafından verilir. Bu yapıyla ilgili ayrıntılar için bkz. [“IBM üzerinde MQXQH \(Transmission-queue header\)” sayfa 1225](#) .

Bir MDFMT FMXQH değeri olan iletiler için COA ve COD raporları oluşturulmaz.

Bu, MQGET çağrısına ilişkin bir çıkış alanıdır ve MQPUT ve MQPUT1 çağrılarında ilişkin bir giriş alanı. Bu alanın uzunluğu LNFMT tarafından verilir. Bu alanın ilk değeri FMNONE ' dır.

### **MDGID (24 baytlık bit dizesi)**

Grup tanıtıcısı.

Bu, fiziksel iletinin ait olduğu belirli ileti grubunu ya da mantıksal iletiyi tanımlamak için kullanılan bayt dizilimidir. MDGID , ileti için bölümlenmeye izin veriliyorsa da kullanılır. Tüm bu durumlarda, MDGID boş değer olmayan bir değere sahiptir ve MDMFL alanında aşağıdaki işaretlerden biri ya da birkaçı ayarlanır:

- MFMIQ
- MFLMIQ
- MFSEG
- MFLSEG
- MFSEGA

If none of these flags is set, MDGID has the special null value GINONE.

Bu alan, aşağıdaki durumlarda, MQPUT ya da MQGET çağrısında uygulama tarafından belirlenemez:

- MQPUT çağrısında, PMLOGO belirtildi.
- MQGET çağrısında, MOGRPI belirtilmez.

Bu çağrılar, rapor iletisi olmayan iletiler için kullanmayı düşünün. Ancak, uygulama daha fazla denetim gerektiriyorsa ya da çağrı MQPUT1ise, uygulama MDGID ' in uygun bir değere ayarlandığını doğrulamalıdır.

İleti grupları ve bölümleri, yalnızca grup tanıtıcısı benzersizse doğru olarak işlenebilir. Bu nedenle, uygulamalar kendi grup tanıtıcılarını oluşturmamalıdır; yerine, uygulamalar aşağıdakilerden birini yapmalıdır:

- PMLOGO belirtilirse, kuyruk yöneticisi otomatik olarak mantıksal iletinin grubundaki ya da kesiminde ilk ileti için benzersiz bir grup tanıtıcısı oluşturur ve bu grup tanıtıcısını mantıksal iletinin grup ya da kesimlerinde kalan iletiler için kullanır; dolayısıyla, uygulamanın özel bir işlem yapması gerekmez. Bu yordamı kullanmayı düşünün.
- If PMLOGO is not specified, the application should request the queue manager to generate the group identifier, by setting MDGID to GINONE on the first MQPUT or MQPUT1 call for a message in the group or segment of the logical message. Bu çağrıdan çıkışta kuyruk yöneticisi tarafından döndürülen grup tanıtıcısı, daha sonra, mantıksal iletinin grup ya da kesimlerindeki geri kalan iletiler için kullanılmalıdır. Bir ileti grubunda kesimlenmiş iletiler varsa, gruptaki tüm kesimler ve iletiler için aynı grup tanıtıcısı kullanılmalıdır.

PMLOGO belirtilmediğinde, mantıksal iletilerin gruplarındaki ve bölümlerindeki iletiler herhangi bir sırayla (örneğin, ters sırada) yerleştirilebilir, ancak grup tanıtıcısı bu iletilerin herhangi biri için yayınlanan ilk MQPUT ya da MQPUT1 çağrısıyla ayrılmalıdır.

MQPUT ve MQPUT1 çağrılarında giriş sırasında, kuyruk yöneticisi PMOPT' de ayrıntılı olarak belirtilen değeri kullanır. MQPUT ve MQPUT1 çağrılarının çıktısında, kuyruk yöneticisi bu alanı, açılan nesne bir dağıtım listesi değil, tek bir kuyruksa, ancak bir dağıtım listesi değilse, ileti ile birlikte gönderilen değere ayarlar, ancak bu açılan nesne bir dağıtım listesi ise, bu değeri değiştirmez. İkinci durumda, uygulamanın oluşturulan grup tanıtıcılarını bilmesi gerekiyorsa, uygulamanın PRGID alanını içeren MQPMR kayıtlarını sağlamalıdır.

MQGET çağrısına giriş sırasında, kuyruk yöneticisi Tablo 1' de ayrıntılı olarak belirtilen değeri kullanır. MQGET çağrısından çıktıda, kuyruk yöneticisi bu alanı, alınan ileti için değere ayarlar.

Aşağıdaki özel değer tanımlıdır:

### **GINONE**

Grup tanıtıcısı belirtilmedi.

Değer, alanın uzunluğuna ilişkin ikili sıfır değeridir. Bu değer, gruplarda olmayan iletiler için, mantıksal iletilerin kesimleri değil ve kesimlere izin verilmediği iletiler için kullanılır.

Bu alanın uzunluğu LNGID tarafından verilir. Bu alanın ilk değeri GINONE 'tır. MDVER , MDVER2değerinden küçükse bu alan yok sayılır.

### **MDMFL (10 basamaklı işaretli tamsayı)**

Mesaj bayrakları.

Bunlar, iletinin özniteliklerini belirleyen ya da işlemlerini denetleyen işaretlerdir. İşaretler aşağıdaki kategorilere ayrılır:

- Bölümlenme işareti
- Durum işaretleri

Bunlar sırayla anlatılır.

**Bölümlenme işareti:** Bir ileti kuyruk için çok büyük olduğunda, kuyruğa ileti koyma girişimi genellikle başarısız olur. Bölümlenme, kuyruk yöneticisinin ya da uygulamanın iletiyi bölümler adı verilen daha küçük parçalara ayırdığı ve kuyruğun her kesimini ayrı bir fiziksel ileti olarak kuyruğa alan bir tekniktir. İletiyi alan uygulama, kesimleri tek tek alabilir ya da kuyruk yöneticisini, kesimleri MQGET çağrısının döndürdüğü tek bir iletiyle yeniden birleştirmesini isteyebilir. İkinci olarak, MQGET çağrısında GMCMPM seçeneği belirlenerek ve iletiyi tamamlamak için yeterli büyüklükte bir arabellek sağlanarak elde edilir. (GMCMM seçeneğine ilişkin ayrıntılar için bkz. [“IBM üzerinde MQGMO \(get-message options\)” sayfa 1060 .](#)) Bir iletinin bölümlenmesi, gönderen kuyruk yöneticisinde, ara kuyruk yöneticisinde ya da hedef kuyruk yöneticisinde oluşabilir.

Bir iletinin bölümlenmesini denetlemek için aşağıdakilerden birini belirtebilirsiniz:

### **MFSEGI**

Segmentasyon engellenmiş.

Bu seçenek, kuyruk yöneticisi tarafından iletinin kesimlere ayrılmasını önler. Önceden bir bölüm olan bir ileti için belirtilirse, bu seçenek kesimin daha küçük kesimlere ayrılmasını önler.

Bu işaretin değeri ikili sıfırdır. Bu varsayılandır.

### **MFSEGA**

Bölümlenmeye izin verilir.

Bu seçenek, ileti, kuyruk yöneticisi tarafından kesimlere ayrılmasına olanak sağlar. Önceden bir bölüm olan bir ileti için belirtilirse, bu seçenek kesimin daha küçük kesimlere ayrılmasını sağlar. MFSEGA, MFSEG ya da MFLSEG belirlenmeden ayarlanabiliyor.

Kuyruk yöneticisi bir iletiyi kestiğinde, kuyruk yöneticisi her bir bölümle birlikte gönderilen MQMD 'nin kopyasındaki MFSEG işaretini açar; ancak, MQPUT ya da MQPUT1 çağrısında uygulama tarafından sağlanan MQMD' de bu işaretlerin ayarlarını değiştirmez. Mantıksal iletiyle son bölüm için, kuyruk yöneticisi, kesimle gönderilen MQMD ' de MFLSEG işaretini de açar.

**Not:** MFSEGA ile iletiler konulduğunda ancak PMLOGO olmadan bakım gerekir. İleti aşağıdaki gibi olursa:

- Bir kesim değil, ve
- Bir grupta değil,
- İletilmiyor.

Uygulama, her bir MQPUT ya da MQPUT1 çağrısından önce MDGID alanını GINONE olarak sıfırlamayı unutmamalı ve her ileti için kuyruk yöneticisi tarafından benzersiz bir grup tanıtıcısı üretilmesine neden olur. Bu işlem yapılmazsa, ilgisiz iletiler yanlışlıkla aynı grup tanıtıcısına sahip olabilir ve bu da daha sonra yanlış işleme yol açabilir. MDGID alanının ilk durumuna

getirilmesi gerektiği zaman hakkında daha fazla bilgi için, MDGID alanının ve PMLOGO seçeneğinin açıklamalarına bakın.

Kuyruk yöneticisi, iletileri, kuyruğa sığabilecek bölümlerin (artı herhangi bir üstbilgi verisi) kuyrukta olmasını sağlamak için, iletileri gerektiği şekilde böler. Ancak, kuyruk yöneticisi tarafından oluşturulan bir kesimin büyüklüğünün alt sınırı vardır ve yalnızca bir iletiden yaratılan son bölüm bu sınırdan küçük olabilir. (Uygulama tarafından oluşturulan bir kesimin büyüklüğünün alt sınırı bir bayttır.) Kuyruk yöneticisi tarafından oluşturulan kesimler eşit olmayan uzunluğa sahip olabilir. Kuyruk yöneticisi iletiyi şu şekilde işler:

- Kullanıcı tanımlı biçimler, 16 bayttan oluşan sınırlara bölünmektedir. Başka bir deyişle, kuyruk yöneticisi 16 byte 'tan (son kesimden başka) küçük bölümler oluşturmaz.
- FMSTR dışındaki yerleşik biçimler, veri varisinin doğasına uygun olan noktalarda bölünür. Ancak, kuyruk yöneticisi bir iletiyi hiçbir zaman MQ üstbilgi yapısının ortasında bölemez. Başka bir deyişle, tek bir MQ üstbilgi yapısı içeren bir bölüm kuyruk yöneticisi tarafından daha fazla bölünemez ve sonuç olarak, o ileti için olası en düşük kesim büyüklüğü 16 byte 'tan fazla olur.

Kuyruk yöneticisi tarafından oluşturulan ikinci ya da sonraki bölüm aşağıdakilerden biriyle başlar:

- Bir MQ üstbilgi yapısı
- Uygulama iletileri verilerinin başlangıcı
- Uygulama iletileri verileri aracılığıyla bir kısım yolu
- FMSTR, var olan veri (SBCS, DBCS ya da karma SBCS/DBCS) niteliği dikkate alınmadan bölünmeye devam eder. Dizgi DBCS ya da SBCS/DBCS karışık olduğunda, bu durum bir karakter takımından diğerine dönüştürülemeyen kesimlere neden olabilir. Kuyruk yöneticisi, FMSTR iletilerini 16 byte 'tan küçük (son bölümden başka) bölümlere ayırmaz.
- Her kesimin MQMD ' deki MDFMT, MDCSive MDENC alanları, kesimin başlangıcındaki verileri doğru şekilde tanımlamak için kuyruk yöneticisi tarafından ayarlanır; biçim adı, yerleşik bir biçimin adı ya da kullanıcı tanımlı biçimin adı olur.
- Sıfırdan büyük MD0FF olan bölümlerin MQMD ' deki MDREP alanı aşağıdaki gibi değiştirilir:
  - Her bir rapor tipi için, rapor seçeneği RO\* D ise, ancak bölüm, kullanıcı verilerinin ilk 100 bayttan herhangi birini içeremez (yani, var olabilecek herhangi bir MQ üstbilgi yapısından sonraki veriler), rapor seçeneği RO\* olarak değiştirilir.

Kuyruk yöneticisi daha önce kuralları izler, ancak tersi durumda iletileri tahmin edilemez olarak böler; bir iletinin bölündüğü yere ilişkin varsayımlar yapma

Kalıcı iletiler için, kuyruk yöneticisi bölümlendirme işlemini yalnızca bir iş birimi içinde gerçekleştirebilir:

- MQPUT ya da MQPUT1 çağrısı kullanıcı tanımlı bir iş birimi içinde çalışıyorsa, bu iş birimi kullanılır. Arama, bölümlendirme işlemi aracılığıyla partway başarısız olursa, kuyruk yöneticisi, başarısız olan aramaların sonucu olarak kuyruğa yerleştirilen tüm bölümleri kaldırır. Ancak, hata, iş biriminin başarıyla kesinleştirilmesini engellemektedir.
- Arama, kullanıcı tanımlı bir iş birimi dışında çalışıyorsa ve var olan bir kullanıcı tanımlı iş birimi yoksa, kuyruk yöneticisi yalnızca arama süresi boyunca bir iş birimi yaratır. Arama başarılı olursa, kuyruk yöneticisi iş birimini otomatik olarak kesinleştirir (uygulamanın bu işlemi yapması gerekmez). Arama başarısız olursa, kuyruk yöneticisi iş birimini geri arar.
- Arama, kullanıcı tanımlı bir iş birimi dışında çalışıyorsa, ancak kullanıcı tanımlı bir iş birimi varsa, kuyruk yöneticisi bölümlendirme gerçekleştiremez. İleti bölümlenmeye gerek duymuyorsa, arama hala başarılı olabilir. Ancak, ileti kesimlere ayırma gerektiriyorsa, arama başarısız olur ve neden kodu RC2255 ile başarısız olur.

Kalıcı olmayan iletiler için, kuyruk yöneticisi bölümlenmenin gerçekleştirilmesini sağlamak için bir iş birimi kullanılabilir olmasını gerektirmez.

Kesimlere ayrılmış olabilecek iletilerin veri dönüşümlerine özel önem verilmelidir:



- Veri dönüştürme yalnızca, MQGET çağrısında bulunan uygulama tarafından gerçekleştirilir ve uygulama GMCMPM seçeneğini belirtiyorsa, veri dönüştürme çıkışı, çıkışa dönüştürülmek üzere çıkışa ilişkin iletinin tamamını geçirilir ve iletinin bölümlenmiş olması, çıkışa belirgin bir şekilde görünmeyecektir.
- Alma uygulaması bir defada bir parça alıyorsa, bir defada bir kesimi dönüştürmek için veri-dönüştürme çıkışı çağrılır. Bu nedenle, çıkışta, bir kesimde bulunan verileri diğer bölümlerin herhangi birindeki verilerden bağımsız olarak dönüştürme yeteneğine sahip olması gerekir.  
İletideki verilerin doğası, 16 baytlık sınırlardaki verilerin rasgele kesimlerinin çıkışa dönüştürülemez şekilde sonuçlanabileceği gibi, ya da FMSTR ve karakter kümesi DBCS ya da SBCS/DBCS ise, gönderme uygulaması kendi kendine yaratılmayı ve segmentasyonu engellemek için MFSEGI belirterek, kesimlere yol açabilir ve kesimler yerleştirebilir. Bu şekilde, gönderme uygulaması, her kesimin veri dönüştürme çıkışının kesimini başarıyla dönüştürmesine izin vermek için yeterli bilgi içermesini güvenebilir.
- Gönderen bir ileti kanalı aracı (MCA) için gönderici dönüştürme belirtildiyse, MCA yalnızca mantıksal iletilerin kesimleri olmayan iletileri dönüştürür; MCA hiçbir zaman bölüm olan iletileri dönüştürme girişiminde bulunmaz.

Bu işaret, MQPUT ve MQPUT1 çağrılarında bir giriş işaretidir ve MQGET çağrısına ilişkin bir çıkış işareti. İkinci arama sırasında, kuyruk yöneticisi, işaretin değerini MQGMO 'daki GMSEG alanına da yansıtıyor.

Bu işaretin başlangıç değeri MFSEGI 'dir.

**Durum işaretleri:** Bunlar, fiziksel iletinin bir ileti grubuna ait olup olmadığını gösteren işaretlerdir, her ikisi de ya da ikisi de mantıksal iletinin bir bölümüdür. MQPUT ya da MQPUT1 çağrısında, aşağıdakilerden biri ya da daha fazlası belirtilebilir ya da MQGET çağrısının döndürmesini sağlar:

#### **MFMIIG**

İleti, bir grubun üyesidir.

#### **MFLMIIG**

İleti, bir gruptaki son mantıksal iletidir.

Bu işaret ayarlanırsa, kuyruk yöneticisi iletiyle birlikte gönderilen MQMD 'nin kopyasında MFMIIG' yi açar; ancak, MQPUT ya da MQPUT1 çağrısında uygulama tarafından sağlanan MQMD 'de bu işaretlerin ayarlarını değiştirmez.

Bir grup için geçerli, tek bir mantıksal iletilerden oluşur. Böyle bir durumda, MFLMIIG ayarlanır, ancak MDSEQ alanında değer bir değer olur.

#### **MFSEG**

İleti, mantıksal bir iletinin bir bölümüdür.

MFSEG, MFLSEG olmadan belirtildiğinde, kesimde uygulama iletisi verilerinin uzunluğu (var olabilecek MQ üstbilgi yapılarının uzunlukları dışında) en az bir değer olmalıdır. Uzunluk sıfır, MQPUT ya da MQPUT1 çağrısı başarısız olur ve neden kodu RC2253 ile başarısız olur.

#### **MFLSEG**

İleti, bir mantıksal iletinin son bölümdür.

Bu işaret belirlendiyse, kuyruk yöneticisi, iletiyle birlikte gönderilen MQMD 'nin kopyasında MFSEG' i açar, ancak MQPUT ya da MQPUT1 çağrısında uygulama tarafından sağlanan MQMD 'de bu işaretlerin ayarlarını değiştirmez.

Mantıksal bir iletinin tek bir kesimden oluşması için geçerli olur. Bu durumda, MFLSEG ayarlıdır, ancak MDOFF alanının değeri sıfır olur.

MFLSEG belirtildiğinde, (var olabilecek üstbilgi yapılarının uzunlukları hariç) olarak, kesimde uygulama iletisi verilerinin uzunluğunun sıfır olması için izin verilebilir.

Uygulama, ileti yerleştirilirken bu işaretlerin doğru ayarlandığından emin olmalıdır. PMLOGO belirtilirse ya da kuyruk tanıtıcısı için önceki MQPUT çağrısında belirtilmişse, işaretlerin ayarları kuyruk tanıtıcısı için kuyruk yöneticisi tarafından saklanan grup ve bölüm bilgileriyle tutarlı olmalıdır. Aşağıdaki

koşullar, PMLOGO belirlendiğinde, kuyruk tanıtıcısı için arka arkaya gelen MQPUT çağrılarını için geçerlidir:

- Yürürlükte bir grup ya da mantıksal ileti yoksa, bu işaretlerin tümü (ve bunların birleşimleri) geçerli olur.
- MFMIIG belirtildikten sonra, MFLMIIG belirtilinceye kadar bu durumda kalmalıdır. Bu koşul karşılanmazsa, arama neden kodu RC2241 ile başarısız olur.
- MFSEGI belirtildikten sonra, MFLSEGI belirtilinceye kadar bu durumda kalmalıdır. Bu koşul karşılanmazsa, arama neden kodu RC2242 ile başarısız olur.
- MFMIIG olmadan MFSEGI belirlendikten sonra, MFMIIG, MFLSEGI belirtildikten sonra da kapalı kalmalıdır. Bu koşul karşılanmazsa, arama neden kodu RC2242 ile başarısız olur.

**Tablo 1** , işaretlerin geçerli birleşimlerini ve çeşitli alanlar için kullanılan değerleri gösterir.

Bu işaretler, MQPUT ve MQPUT1 çağrılarında ve MQGET çağrısına ilişkin çıkış işaretlerine ilişkin giriş işaretleridir. İkinci çağrı sırasında, kuyruk yöneticisi, işaretlerin değerlerini MQGMO 'daki GMGST ve GMSST alanlarına da yankıyor.

**Varsayılan işaretler:** İletinin varsayılan özniteliklere sahip olduğunu belirtmek için aşağıdaki değer belirlenebilir:

#### **MFYOK**

İleti işaretleri (varsayılan ileti öznitelikleri) yok.

Bu, kesimlere bölünmeyi engeller ve iletinin bir grupta yer almadığını ve mantıksal bir iletinin bir kesimi olmadığını belirtir. MFNONE, program belgelerine yardımcı olmak için tanımlıdır. Bu işaretin başka biriyle birlikte kullanılması amaçlanmaz, ancak değeri sıfır olduğu için bu tür bir kullanım saptanamaz.

MDMFL alanı alt alanlara bölünmektedir; ayrıntılar için ["IBM i" ta rapor seçenekleri ve ileti işaretleri](#) sayfa 1410' e bakın.

Bu alanın ilk değeri MFNONE 'dir. MDVER , MDVER2değerinden küçükse bu alan yok sayılır.

#### **MDMID (24 baytlık bit dizesi)**

İleti tanıtıcısı.

Bu, bir iletiyi diğerinden ayırmak için kullanılan bayt dizilimidir. Genellikle, kuyruk yöneticisi tarafından izin verilmediği halde, iki ileti aynı ileti tanıtıcısına sahip olmalıdır. İleti tanıtıcısı, iletinin kalıcı bir özeldir ve kuyruk yöneticisinin yeniden başlatmalarında devam eder. İleti tanıtıcısı bir karakter dizisi değil, bir byte dizisi olduğu için, ileti tanıtıcısı, ileti bir kuyruk yöneticisinden diğerine akış sırasında karakter kümeleri arasında dönüştürülmez.

MQPUT ve MQPUT1 çağrılarını için, uygulama tarafından MINONE ya da PMNMID belirtilirse, kuyruk yöneticisi ileti konduğunda benzersiz bir ileti tanıtıcısı oluşturur ve iletiyi, iletiyle birlikte gönderilen ileti tanımlayıcısına yerleştirir. Kuyruk yöneticisi, gönderme uygulamasına ait olan ileti tanımlayıcısında bu ileti tanıtıcısını da döndürür. Uygulama, bu değeri belirli iletilere ilişkin bilgileri kaydetmek ve uygulamanın diğer kısımlarındaki sorgulara yanıt vermek için kullanabilir.

An MDMID generated by the queue manager consists of a 4-byte product identifier ( AMQ- or CSQ- in either ASCII or EBCDIC, where - represents a single blank character), followed by a product-specific implementation of a unique string. IBM MQ 'da bu, kuyruk yöneticisi adının ilk 12 karakterini ve sistem saatinden türetilen bir değeri içerir. İletişim kurabilen tüm kuyruk yöneticilerinin, ileti tanıtıcılarının benzersiz olduğundan emin olmak için ilk 12 karakterde farklı adlar olması gerekir. Benzersiz bir dizgi oluşturma yeteneği, sistem saatinin geriye doğru değiştirilmemesine de bağlıdır. Uygulama tarafından oluşturulan bir kuyruk yöneticisi tarafından oluşturulan bir ileti tanıtıcısının olasılığını ortadan kaldırmak için, uygulama ASCII ya da EBCDIC (X'41 '-X'49' ve X'C1'-X'C9') arasında A ile I arasındaki ilk karakterlere sahip tanıtıcıların oluşturulmaması gerekir. Ancak, uygulamanın bu aralıklarda başlangıç karakterleri olan tanıtıcılar oluşturmasını önlemez.

İleti bir konuya konursa, kuyruk yöneticisi yayınlanan her ileti için gerekli olduğu şekilde benzersiz ileti tanıtıcıları oluşturur. Uygulama tarafından PMNMID belirtilirse, kuyruk yöneticisi çıkışta

döndürülebilmek için benzersiz bir ileti tanıtıcısı oluşturur. Uygulama tarafından MINONE belirtilirse, çağrıdan dönüşte MQMD ' deki MDMID alanının değeri değişmeden kalır.

Alıkonan yayınlarla ilgili daha fazla ayrıntı için [PMOPT](#) içindeki PMRET tanımına bakın.

İleti bir dağıtım listesine konursa, kuyruk yöneticisi gereken şekilde benzersiz ileti tanıtıcıları oluşturur, ancak MQMD ' deki MDMID alanının değeri, MINONE ya da PMNMID belirlense bile, çağrıdan geri dönmeye değişmeden kalır. Uygulamanın kuyruk yöneticisi tarafından oluşturulan ileti tanıtıcılarını bilmesi gerekiyorsa, uygulamanın PRMID alanını içeren MQPMR kayıtlarını sağlamalıdır.

Gönderme uygulaması, MINONE dışında ileti tanıtıcısı için belirli bir değeri de belirtebilir; bu değer, kuyruk yöneticisinin benzersiz bir ileti tanıtıcısı oluşturmasını durdurur. Bir iletiyi ileten bir uygulama, bu olanağı özgün iletinin ileti tanıtıcısını yamak için kullanabilir.

Kuyruk yöneticisi şu hariç olmak üzere, bu alanın herhangi bir kullanımını gerçekleştiriyor:

- Daha önce açıklandığı gibi, istenirse benzersiz bir değer oluştur
- İletiyeye ilişkin alma isteğini içeren uygulamaya değer sağlayın
- Değeri, bu ileti hakkında oluşturduğu herhangi bir rapor iletinin MDCID alanına kopyalayın ( MDREP seçeneklerine bağlı olarak).

Kuyruk yöneticisi ya da bir ileti kanalı aracısı bir rapor ileti oluşturduğunda, MDMID alanını, özgün iletinin MDREP alanı, RONMI ya da ROPMI iletinin belirlediği şekilde ayarlar. Rapor iletileri oluşturan uygulamalar da bunu yapmalıdır.

MQGET çağrısına ilişkin MDMID , kuyruktan alınacak belirli bir iletiyi seçmek için kullanılacak beş alandan biridir. Olağan durumda MQGET çağrısı kuyruktan bir sonraki iletiyi döndürür, ancak belirli bir ileti gerekliyse, beş seçim ölçütünün biri ya da daha fazlası herhangi bir birleşimde belirtilirse elde edilebilir; bu alanlar şunlardır:

- MDMID
- MDCID
- MDGID
- MDSEQ
- MDOFF

Uygulama bu alanlardan birini ya da birkaçını gerekli değerlere ayarlar ve sonra bu alanların seçim ölçütü olarak kullanılması gerektiğini belirtmek için MQGMO ' daki GMMO alanında karşılık gelen MO\* eşleşme seçeneklerini ayarlar. Yalnızca, bu alanlarda belirtilen değerlere sahip iletiler alma için adaydır. GMMO alanı için varsayılan değer (uygulama tarafından değiştirilmediyse) hem ileti tanıtıcısı, hem de ilinti tanıtıcısı ile eşleşir.

Olağan durumda, döndürülen ileti, kuyruğun seçim ölçütlerine uygun ilk iletidir. Ancak, GMBRWN belirtilirse, döndürülen ileti seçim ölçütlerini karşılayan bir sonraki iletidir; bu ileti için tarama, imlecin bulunduğu konumdan sonra gelen iletiyle başlar.

**Not:** Kuyruk, seçim ölçütlerini karşılayan bir ileti için sırayla taranır; bu nedenle alma süreleri, seçim ölçütü belirtilmediğinde daha yavaş olur; özellikle, uygun bir öge bulunmadan önce birçok iletinin taranması gerekir.

Seçim ölçütlerinin çeşitli durumlarda nasıl kullanılırla ilgili daha fazla bilgi için [Tablo 1](#) ' e bakın.

MINONE değerinin, ileti tanıtıcısı olarak MOMSGI belirtmeleriyle aynı etkiye sahip olması da, herhangi bir ileti tanıtıcısının eşleşmesine neden olur.

MQGET çağrısının **GMO** parametresinde GMMUC seçeneği belirtilirse bu alan yoksayılır.

MQGET çağrısından dönüşte, MDMID alanı, döndürülen iletinin ileti tanıtıcısına ayarlanır (varsa).

Aşağıdaki özel değer kullanılabilir:

#### **BİRİNCİ**

İleti tanıtıcısı belirtilmedi.

Değer, alanın uzunluğuna ilişkin ikili sıfır değeridir.

Bu, MQGET, MQPUT ve MQPUT1 çağrılarına ilişkin bir giriş/çıkış alanıdır. Bu alanın uzunluğu LNMID tarafından verilir. Bu alanın ilk değeri MINONE değeridir.

### **MDMT (10 basamaklı işaretli tamsayı)**

İleti tipi.

Bu, iletinin tipini gösterir. İleti tipleri şu şekilde gruplanır:

#### **MTSFST**

Sistem tarafından tanımlanan ileti tipleri için en düşük değer.

#### **MTSLST**

Sistem tarafından tanımlanan ileti tipleri için en yüksek değer.

Aşağıdaki değerler sistem aralığı içinde tanımlanmıştır:

#### **MTDGRM**

İleti yanıt gerektirmiyor.

İleti, yanıt gerektirmeyen bir iletidir.

#### **MTRQST**

Yanıt gerektiren ileti.

İleti, yanıt gerektiren bir iletidir.

Yanıtın gönderilmesi gereken kuyruğun adı, MDRQ alanında belirtilmelidir. MDREP alanı, yanıtın MDMID ve MDCID ' in nasıl ayarlanacağını belirtir.

#### **MTRPLY**

Önceki bir istek iletisine yanıt verin.

İleti, daha önceki bir istek iletisine (MTRQST) yanıt olarak gönderilmektedir. İleti, istek iletisinin MDRQ alanı tarafından belirtilen kuyruğa gönderilmelidir. Yanıtın MDMID ve MDCID ' in nasıl ayarlanacağını denetlemek için, isteğin MDREP alanı kullanılmalıdır.

**Not:** Kuyruk yöneticisi, istek-yanıt ilişkisini uygulamaz; bu bir uygulama sorumluluğunda.

#### **MTRPRT**

Rapor iletisi.

İleti, genellikle başka bir iletiyle ilgili (örneğin, geçerli olmayan verileri içeren bir istek iletisi alındı), beklenen ya da beklenmeyen bazı oluşumlarla ilgili raporlama içerir. İleti, özgün iletinin ileti tanımlayıcısının MDRQ alanı tarafından belirtilen kuyruğa gönderilmelidir. Raporun niteliyi belirtmek için MDFB alanının ayarlanması gerekir. Özgün iletinin MDREP alanı, rapor iletisinin MDMID ve MDCID ' in nasıl ayarlanması gerektiğini denetlemek için kullanılabilir.

Kuyruk yöneticisi ya da ileti kanalı aracısı tarafından oluşturulan rapor iletileri, MDFB ve MDCID alanları önceden açıklandığı şekilde ayarlanmış şekilde, her zaman MDRQ kuyruğuna gönderilir.

Sistem aralığı içindeki diğer değerler, MQI ' in sonraki sürümlerinde tanımlanabilir ve MQPUT ve MQPUT1 çağrıları hatasız olarak kabul edilir.

Uygulama tanımlı değerler de kullanılabilir. Bunlar aşağıdaki aralık içinde olmalıdır:

#### **MTAFST**

Uygulama tanımlı ileti tipleri için en düşük değer.

#### **MALST**

Uygulama tanımlı ileti tipleri için en yüksek değer.

MQPUT ve MQPUT1 çağrıları için, MDMT değeri sistem tanımlı aralıkta ya da uygulama tanımlı aralık içinde olmalıdır; değilse, arama başarısız olur ve neden kodu RC2029 ile başarısız olur.

Bu, MQGET çağrısına ilişkin bir çıkış alanıdır ve MQPUT ve MQPUT1 çağrıları için bir giriş alanı. Bu alanın ilk değeri MTDGRM ' dir.

### **MDOFF (10 basamaklı işaretli tamsayı)**

Mantıksal iletinin başlangıcındaki fiziksel iletelerde verilerin göreceli konumu.

Bu, veri formlarının parçası olan mantıksal iletinin başlangıcındaki fiziksel iletteki verilerin bayt cinsinden görelî konudur. Bu verilere *segment* adı verilir. Görelî konum 0 ile 999 999 999 aralığında. Mantıksal iletinin bir bölümü olmayan fiziksel bir ileti, sıfır görelî konutuna sahiptir.

Bu alan, aşağıdaki durumlarda, MQPUT ya da MQGET çağrısında uygulama tarafından belirlenemez:

- MQPUT çağrısında, PMLOGO belirtildi.
- MQGET çağrısında MOOFFS belirtilmedi.

Bunlar, rapor iletisi olmayan ileteler için bu çağrılarını kullanmanın önerilen yollarıdır. Ancak, uygulama bu koşullara uymuyorsa ya da çağrı MQPUT1ise, uygulama MD0FF ' in uygun bir değere ayarlandığından emin olmalıdır.

MQPUT ve MQPUT1 çağrılarında giriş sırasında, kuyruk yöneticisi Tablo 1' de ayrıntılı olarak belirtilen değeri kullanır. MQPUT ve MQPUT1 çağrılarının çıktısında, kuyruk yöneticisi bu alanı, iletiyle birlikte gönderilen değere ayarlar.

Bir mantıksal iletinin bir kesimine ilişkin rapor iletisi raporlaması için, kuyruk yöneticisi tarafından korunan bölüm bilgilerindeki görelî konumu güncellemek için MDOLN alanı (OLUNDF değil, OLUNDF) kullanılır.

MQGET çağrısına giriş sırasında, kuyruk yöneticisi Tablo 1' de ayrıntılı olarak belirtilen değeri kullanır. MQGET çağrısından çıktıda, kuyruk yöneticisi bu alanı, alınan ileti için değere ayarlar.

Bu alanın ilk değeri sıfır. MDVER , MDVER2değerinden küçükse bu alan yok sayılır.

### **MDOLN (10 basamaklı işaretli tamsayı)**

Özgün iletinin uzunluğu.

Bu alan yalnızca bölüm olan rapor iletileri için ilgi alanıdır. Rapor iletisinin ilişkilendirdiği ileti kesiminin uzunluğunu belirtir; bu, bölüm formlarının parçası olan mantıksal iletinin uzunluğunu ya da rapor iletisinde verilerin uzunluğunu belirtmez.

**Not:** When generating a report message for a message that is a segment, the queue manager and message channel agent copy into the MQMD for the report message the MDGID, MDSEQ, MD0FF, and MDMFL, fields from the original message. Sonuç olarak, rapor iletisi de bir bölümdür. Rapor iletileri oluşturan uygulamaların aynı yapılması ve MDOLN alanının doğru olarak ayarlandığından emin olmak için önerilir.

Aşağıdaki özel değer tanımlıdır:

#### **OUNDF**

İletinin özgün uzunluğu tanımlı değil.

MDOLN , MQPUT ve MQPUT1 çağrılarında bir giriş alanıdır; ancak, uygulama tarafından sağlanan değer yalnızca belirli durumlarda kabul edilir:

- Gönderilmekte olan ileti bir bölümse ve aynı zamanda bir rapor iletisiyse, kuyruk yöneticisi belirtilen değeri kabul eder. Değer şu olmalıdır:
  - Kesimin son bölümü değilse sıfırdan büyük
  - Kesim son bölümse sıfırdan küçük değil
  - İletide var olan veri uzunluğundan az değil

Bu koşullar karşılanmazsa, çağrı neden kodu RC2252ile başarısız olur.

- Bir rapor iletisi gönderiliyorsa, ancak bir rapor iletisi değilse, kuyruk yöneticisi alanı yoksayar ve bunun yerine uygulama iletisi verilerinin uzunluğunu kullanır.
- Diğer tüm durumlarda, kuyruk yöneticisi alanı yoksayar ve OLUNUNDF değerini kullanır.

Bu, MQGET çağrısındaki bir çıkış alanıdır.

Bu alanın ilk değeri OLUNDF ' dir. MDVER , MDVER2değerinden küçükse bu alan yok sayılır.

## MDPAN (28 baytlık karakter dizilimi)

İletiyi koyan uygulamanın adı.

Bu, iletinin *başlangıç bağlamının* bir parçasıdır. İleti bağlamına ilişkin daha fazla bilgi için bkz. [İleti bağlamı](#) ve [Bağlam bilgilerini denetleme](#).

MDPAN biçimi, MDPAT değerine bağlıdır.

Bu alan kuyruk yöneticisi (yani, PMSETA dışındaki tüm seçenekler için) tarafından ayarlandığında, ortam tarafından belirlenen değere ayarlanır:

- ▶ **z/OS** On z/OS, the queue manager uses:
  - z/OS toplu işi için, JES JOB kartından 8 karakterlik iş adı
  - TSO için, 7 karakterlik TSO kullanıcı kimliği
  - CICS için, 8 karakterlik uygulama tanıtıcısı ve ardından 4 karakterlik trand
  - IMS için, 8 karakterlik IMS sistem tanıtıcısı ve ardından 8 karakterlik PSB adı.
  - XCF için, 8 karakterden oluşan XCF grup adı ve onu izleyen 16 karakterlik XCF üye adı
  - Kuyruk yöneticisi tarafından oluşturulan bir ileti için, kuyruk yöneticisi adının ilk 28 karakteri
  - CICS olmadan dağıtılmış kuyruklama için, kanal başlatıcısının 8 karakterden oluşan iş adı ve ardından, 8 karakterlik bir görev tanıtıcısı izleyen modülün 8 karakterden oluşan bir adı ve ardından da ölü-mektup kuyruğuna yerleştirilecek bir iş adı.
  - For MQSeries Java language bindings processing with IBM MQ for z/OS the 8-character jobname of the address space created for the z/OS UNIX System Services environment. Tipik olarak, bu, sonuna tek sayısal karakteri eklenmiş bir TSO kullanıcı kimliği olur.

Ad ya da adlar, alanın geri kalanındaki herhangi bir alan gibi, her biri boşlukla boşluğa doldurulur. Birden çok ad varsa, bunlar arasında ayırıcı yoktur.

- ▶ **Windows** PC DOS ve Windows sistemlerinde kuyruk yöneticisi aşağıdakileri kullanır:
  - Bir CICS uygulaması için, CICS işlem adı
  - CICS olmayan bir uygulama için, yürütülebilir dosyanın tam olarak nitelenmiş adı olan 28 karakter
- ▶ **IBM i** IBM üzerinde, kuyruk yöneticisi tam olarak nitelenmiş iş adını kullanır.
- ▶ **Linux** ▶ **AIX** On AIX and Linux, the queue manager uses:
  - Bir CICS uygulaması için, CICS işlem adı
  - CICS uygulaması olmayan bir uygulama için, kuyruk yöneticisi tarafından kullanılabiliriyorsa, yürütülür dosyanın tam olarak nitelenmiş adının en sağdaki 14 karakteri ve tersi durumda (örneğin, AIX üzerinde).
- VSE/ESA üzerinde, kuyruk yöneticisi 8 karakterlik uygulama tanıtıcısını (applid) ve ardından 4 karakterlik bir trand 'i kullanır.

For the MQPUT and MQPUT1 calls, this is an input/output field if PMSETA is specified in the **PMO** parameter. Alan içindeki boş değerli bir karakteri izleyen herhangi bir bilgi atılır. Boş değerli karakter ve izleyen karakterler kuyruk yöneticisi tarafından boşluklara dönüştürülür. PMSETA belirtilmediyse, bu alan girişte yoksayırlı ve yalnızca çıkış alanıdır.

Bu, MQGET çağrısına ilişkin bir çıkış alanıdır. Bu alanın uzunluğu LNPAN tarafından verilir. Bu alanın ilk değeri 28 boşluk karakteridir.

## MDPAT (10 basamaklı işaretli tamsayı)

İletiyi koyan uygulamanın tipi.

Bu, iletinin *başlangıç bağlamının* bir parçasıdır. İleti bağlamına ilişkin daha fazla bilgi için bkz. [İleti bağlamı](#) ve [Bağlam bilgilerini denetleme](#).

*MDPAT* , ařađıdaki standart tiplerden birine sahip olabilir. Kullanıcı tanımlı tipler de kullanılabilir, ancak *ATULST* ile *ATUFST* aralıđındaki deđerlerle sınırlandırılmalıdır.

**ATP**

AIX uygulaması (*ATUNIX* ile aynı deđer).

**ATBRKR**

Aracı.

**ATCICS**

CICS iřlemi.

**ATCIB**

CICS bridge.

**ATVSE**

CICS/VSE iřlemi.

**ATDOS**

PC DOS üzerindeIBM MQ MQI client uygulaması.

**ATDQM**

Dađıtılmıř kuyruk yöneticisi aracı.

**ATGUAR**

Tandem Guardian uygulaması (*ATNSK* ile aynı deđer).

**ATIMS**

IMS uygulaması.

**ATIMSB**

IMS köprüsü.

**ATJAVA**

Java.

**ATMVS**

MVS ya da TSO uygulaması (*ATZOS* ile aynı deđer).

**ATNOT**

Lotus Notes Agent uygulaması.

**ATNKK**

Tandem NonStop Kernel uygulaması.

**AT390**

OS/390 uygulaması (*ATZOS* ile aynı deđer).

**AT400**

IBM i uygulaması.

**ATQM**

Kuyruk yöneticisi.

**ATIUNIX**

UNIX uygulaması.

**ATVOS**

Stratus VOS uygulaması.

**ATWIN**

16 bit Windows uygulaması.

**AKiM**

32 bit Windows uygulaması.

**ATXCF**

XCF.

**ATZOS**

z/OS uygulaması.

**ATDEF**

Varsayılan uygulama tipi.

Bu, uygulamanın çalışmakta olduğu altyapıya ilişkin varsayılan uygulama tipidir.

**Not:** Bu değışmezin değeri ortamdaki özğüdür.

**ATUNK**

Bilinmeyen uygulama tipi.

Bu değeri, diğeri bağlam bilgileri mevcut olsa da, uygulama tipinin bilinmediğini göstermek için kullanılabilir.

**ATUFST**

Kullanıcı tanımlı uygulama tipi için en düşük değeri.

**ATULST**

Kullanıcı tanımlı uygulama tipi için en yüksek değeri.

Aşğıdaki özel değeri de oluşabilir:

**ATNCON**

İletide bağlam bilgisi yok.

Bu değeri, bağlam olmadan bir ileti konduğunda (yani, PMNOC bağlam seçeneği belirtildiğinde) kuyruk yöneticisi tarafından ayarlanır.

Bir ileti alındığında, iletinin bağlamına sahip olup olmadığına karar vermek için MDPAT bu değeri için test edilebilir (örneğin, diğeri bağlam alanlarından herhangi biri boş değilse, MDPAT 'in ATNCON' ya hiçbir zaman ayarlanmaması önerilir).

**ATSIB**

Başka bir IBM MQ ileti sistemi ürününde ortaya çıkan ve SIB (Service Integration Bus) köprüsüyle gelen bir iletiyi belirtir.

Kuyruk yöneticisi bir uygulamanın sonucu olarak bu bilgileri oluşturduğunda, alan, ortam tarafından belirlenen bir değeri ayarlanır.

**IBM i**

IBM i'da bu alan AT400; olarak ayarlansa, kuyruk yöneticisi hiçbir zaman IBM üzerinde ATCICS' i kullanmaz.

For the MQPUT and MQPUT1 calls, this is an input/output field if PMSETA is specified in the **PMO** parameter. PMSETA belirtilmediyse, bu alan girişte yoksayıdır ve yalnızca çıkış alanıdır.

Bir MQPUT ya da MQPUT1 çağrısının başarıyla tamamlanmasından sonra, bu alan, bir kuyruğa konduysa, iletiyle birlikte gönderilen MDPAT ögesini içerir. Bu, alıkonursa iletiyle birlikte tutulan MDPAT değeri (alıkonan yayınlarla ilgili daha fazla ayrıntı için PMRET tanımına bakın) değeri olur, ancak bu ileti kendilerine gönderilen tüm yayınlarda MDPAT ' i geçersiz kılmak için bir değeri sağladığından, bu ileti abonelere bir yayın olarak gönderildiğinde, MDPAT olarak kullanılmaz. İletinin bağlamı yoksa, alan ATNCON olarak ayarlanır.

Bu, MQGET çağrısına ilişkin bir çıkış alanıdır. Bu alanın ilk değeri ATNCON ' dur.

**MDPD (8 baytlık karakter dizilimi)**

İletinin konduğu tarih.

Bu, iletinin *başlangıç bağlamının* bir parçasıdır. İleti bağlamına ilişkin daha fazla bilgi için bkz. [İleti bağlamı](#) ve [Bağlam bilgilerini denetleme](#).

Bu alanın kuyruk yöneticisi tarafından oluşturulduğu tarih için kullanılan biçim şöyledir:

- YYYAAGG

karakterlerin gösterdiği yer:

**YYYY**

yıl (dört sayısal rakam)



**mm**

yıl ay (01-12)

**DD**

ay günü (01-31 arası)

Greenwich Mean Time (GMT) is used for the MDPD and MDPT fields, subject to the system clock being set accurately to GMT.

İleti bir iş biriminin bir parçası olarak konulduysa, tarih, iş biriminin kesinleştirildiği tarih değil, ileti konduğunda olduğu tarih olur.

For the MQPUT and MQPUT1 calls, this is an input/output field if PMSETA is specified in the **PMO** parameter. Alan içeriği kuyruk yöneticisi tarafından denetlenmez; ancak, alan içindeki boş değerli bir karakteri izleyen herhangi bir bilgi atılır. Boş değerli karakter ve izleyen karakterler kuyruk yöneticisi tarafından boşluklara dönüştürülür. PMSETA belirtilmediyse, bu alan girişte yoksayılr ve yalnızca çıkış alanıdır.

Bir MQPUT ya da MQPUT1 çağrısının başarıyla tamamlanmasından sonra, bu alan, bir kuyruğa konduysa, iletiyle birlikte gönderilen MDPD ögesini içerir. Bu, alıkonursa iletiyle birlikte tutulan MDPD değerinin (alıkonan yayınlarla ilgili daha fazla ayrıntı için PMRET tanımına bakın) değeri olur, ancak bu ileti kendilerine gönderilen tüm yayınlarda MDPD ' i geçersiz kılmak için bir değer sağladığından, bu ileti abonelere bir yayın olarak gönderildiğinde, MDPD olarak kullanılmaz. İletinin bağlamı yoksa, alan tümüyle boştur.

Bu, MQGET çağrısına ilişkin bir çıkış alanıdır. Bu alanın uzunluğu LNPDAT tarafından verilir. Bu alanın ilk değeri 8 boş karakterdir.

#### **MDPER (10 basamaklı işaretli tamsayı)**

Mesaj devamlılığı.

Bu, iletinin kuyruk yöneticisinin sistem hatalarından ve yeniden başlatılıp yeniden başlatılıp başlatılmayacağını belirtir. MQPUT ve MQPUT1 çağrıları için, değer aşağıdakilerden biri olması gerekir:

#### **PEPER**

İleti kalıcı.

Bu, ileti, kuyruk yöneticisinin sistem hatalarından ve yeniden başlatılabildiğinden emin olmak anlamına gelir. İleti ortaya konduğunda ve putter çalışma birimi kesinleştirildikten sonra (ileti bir iş biriminin bir parçası olarak gönderildiyse), yardımcı saklama alanı için ileti korunur. İleti kuyruktan kaldırılıncaya ve alıcı iş biriminin (ileti bir iş biriminin bir parçası olarak alındıysa) işleninceye kadar orada kalır.

Uzak bir kuyruğa kalıcı bir ileti gönderildiğinde, ileti bir sonraki kuyruk yöneticisine vardığı bilininceye kadar, her kuyruk yöneticisinde hedefe giden yolda iletiyi tutmak için bir saklama ve iletme mekanizması kullanılır.

Kalıcı iletiler üzerine yerleştirilemiyor:

- Geçici dinamik kuyruklar
- Bağlaşım olanağı yapısı düzeyinin üçten az olduğu ya da bağlaşım olanağı yapısının kurtarılamaz olduğu paylaşılan kuyruklar.

Kalıcı iletiler kalıcı dinamik kuyruklara, önceden tanımlanmış kuyruklara ve bağlaşım olanağı yapısı düzeyinin 3 olduğu ve bağlaşım tesisinin kurtarılabılır olduğu paylaşılan kuyruklara yerleştirilebilir.

#### **PENPER**

İleti kalıcı değil.

Diğer bir deyişle, ileti, kuyruk yöneticisinin sistem hatalarından ya da yeniden başlatılmalarından sağ olarak kurtulmaz. Bu durum, kuyruk yöneticisinin yeniden başlatılması sırasında yardımcı bellekte iletinin el değmemiş bir kopyası bulunsa bile geçerlidir.

Paylaşılan kuyrukların özel durumunda, kalıcı olmayan iletiler *do* kuyruk paylaşım grubundaki kuyruk yöneticilerinin yeniden başlatılmasını sağlar, ancak paylaşılan kuyruklarda iletileri depolamak için kullanılan bağlaşım tesisinin başarısızlıklarını atlatmaz.

### **PPEQDEF**

İleti varsayılan kalıcılığı içeriyor.

- Kuyruk bir küme kuyruğuysa, iletinin kalıcılığı, iletinin yerleştirdiği kuyruğun belirli bir örneğinin sahibi olan hedef kuyruk yöneticisinde tanımlanan **DefPersistence** özniteliğinden alınır. Genellikle, bir küme kuyruğunun tüm eşgörünümlerinin **DefPersistence** özniteliği için aynı değere sahip olması gerekir, ancak bu zorunlu değildir.

İleti hedef kuyruğa yerleştirildiğinde, **DefPersistence** değeri *MDPER* alanına kopyalanır. **DefPersistence** daha sonra değiştirilirse, kuyruğa önceden yerleştirmiş olan iletiler etkilenmez.

- Kuyruk bir küme kuyruğu değilse, hedef kuyruk yöneticisi uzak olsa bile, iletinin kalıcılığı yerel kuyruk yöneticisinde tanımlanan **DefPersistence** özniteliğinden alınır.

Kuyruk adı çözümleme yolunda birden çok tanımlama varsa, yoldaki ilk tanımlamadaki bu özniteliğin değerinden varsayılan kalıcılık varsayılan kalıcılık alınır. Bu durum şöyle olabilir:

- Diğer ad kuyruğu
- Yerel kuyruk
- Uzak kuyruğun yerel tanımlaması
- Kuyruk yöneticisi diğer adı
- Bir iletim kuyruğu (örneğin, *DefXmitQName* kuyruğu)

İleti konduğunda, **DefPersistence** değeri *MDPER* alanına kopyalanır. **DefPersistence** daha sonra değiştirilirse, önceden yerleştirmiş olan iletiler bundan etkilenmez.

Hem kalıcı hem de kalıcı olmayan iletiler aynı kuyruğun üzerinde var olabilir.

Bir iletiyi yanıtlarken, normalde yanıt iletileri için uygulamaların, istek iletilerinin kalıcılırlık için kullanması gerekir.

*MQGET* çağrısı için döndürülen değer, *PEPER* ya da *PENPER* değeri döndürülür.

Bu, *MQGET* çağrısına ilişkin bir çıkış alanıdır ve *MQPUT* ve *MQPUT1* çağrılarında ilişkin bir giriş alanı. Bu alanın ilk değeri *PEQDEF* olur.

### **MDPRI (10 basamaklı işaretli tamsayı)**

İleti önceliği.

*MQPUT* ve *MQPUT1* çağrıları için, değerın sıfırdan büyük ya da sıfıra eşit olması gerekir; sıfır, en düşük önceliğe sahip olmalıdır. Aşağıdaki özel değer de kullanılabilir:

### **PRQDEF**

Kuyruk için varsayılan öncelik.

- Kuyruk bir küme kuyruğıysa, iletiye ilişkin öncelik, iletinin yerleştirildiği kuyruğun belirli bir örneğinin sahibi olan hedef kuyruk yöneticisinde tanımlandığı şekilde **DefPriority** özniteliğinden alınır. Genellikle, bir küme kuyruğunun tüm eşgörünümlerinin **DefPriority** özniteliği için aynı değere sahip olması gerekir, ancak bu zorunlu değildir.

İleti hedef kuyruğa yerleştirildiğinde, **DefPriority** değeri *MDPRI* alanına kopyalanır. **DefPriority** daha sonra değiştirilirse, kuyruğa önceden yerleştirmiş olan iletiler etkilenmez.

- Kuyruk bir küme kuyruğu değilse, hedef kuyruk yöneticisi uzak olsa da, ileti için öncelik yerel kuyruk yöneticisinde tanımlandığı şekilde **DefPriority** özniteliğinden alınır.

Kuyruk adı çözümleme yolunda birden çok tanımlama varsa, yoldaki ilk tanımlamadaki bu özniteliğin değeri varsayılan önceliğe alınır. Bu durum şöyle olabilir:

- Diğer ad kuyruğu

- Yerel kuyruk
- Uzak kuyruğun yerel tanımlaması
- Kuyruk yöneticisi diğer adı
- Bir iletim kuyruğu (örneğin, DefXmitQName kuyruğu)

İleti konduğunda, **DefPriority** değeri MDPRI alanına kopyalanır. **DefPriority** daha sonra değiştirilirse, önceden yerleştirmiş olan iletiler bundan etkilenmez.

MQGET çağrısının döndürdüğü değer her zaman sıfırdan büyük ya da bu değere eşit; PRQDEF değeri hiçbir zaman döndürülmez.

Bir ileti, yerel kuyruk yöneticisi tarafından desteklenen üst sınırdan daha büyük bir önceliğe sahip bir iletiyse (bu üst sınır **MaxPriority** kuyruk yöneticisi özneliği tarafından verilir), ileti kuyruk yöneticisi tarafından kabul edilir, ancak kuyruk yöneticisinin en yüksek önceliğindeki kuyruğa yerleştirilir; MQPUT ya da MQPUT1 çağrısı, CCWARN ve neden kodu RC2049 ile tamamlanır. Ancak, MDPRI alanı, iletiyi koyan uygulama tarafından belirlenen değeri korur.

Bir iletiyi yanıtlarken, normalde yanıt ileti için uygulamaların istek iletinin önceliğini kullanması gerekir. Diğer durumlarda, PRQDEF belirlenmesi, uygulamanın değiştirilmeden öncelikli olarak gerçekleştirilmesini sağlar.

Bu, MQGET çağrısına ilişkin bir çıkış alanıdır ve MQPUT ve MQPUT1 çağrılarında ilişkin bir giriş alanı. Bu alanın ilk değeri PRQDEF olur.

### MDPT (8 baytlık karakter dizilimi)

İletin konulduğu saat.

Bu, iletinin **başlangıç bağlamının** bir parçasıdır. İleti bağlamına ilişkin daha fazla bilgi için bkz. [İleti bağlamı](#) ve [Bağlam bilgilerini denetleme](#).

Bu alanın kuyruk yöneticisi tarafından oluşturulduğu saat için kullanılan biçim şöyledir:

- HHMMSSSTH

karakterlerin gösterdiği (sırayla):

#### HH

saat (00-23 arası)

#### mm

dakika (00-59)

#### SS

saniye (00-59; bkz. [not](#))

#### T

saniyenin onda biri (0-9 arası)

#### H

saniyenin kamarası (0-9 arası)

**Not:** If the system clock is synchronized to a very accurate time standard, it is possible on rare occasions for 60 or 61 to be returned for the seconds in MDPT. Bu, küresel zaman standardına sığrama saniyeleri eklenince oluşur.

Greenwich Mean Time (GMT) is used for the MDPD and MDPT fields, subject to the system clock being set accurately to GMT.

İleti, bir iş biriminin bir parçası olarak konulduysa, iş biriminin kesinleştirildiği zaman değil, ileti konduğunda saat o zaman olur.

For the MQPUT and MQPUT1 calls, this is an input/output field if PMSETA is specified in the **PMO** parameter. Alan içeriği kuyruk yöneticisi tarafından denetlenmez; ancak, alan içindeki boş değerli bir karakteri izleyen herhangi bir bilgi atılır. Boş değerli karakter ve izleyen karakterler kuyruk yöneticisi tarafından boşluklara dönüştürülür. PMSETA belirtilmediyse, bu alan girişte yoksayılr ve yalnızca çıkış alanıdır.

Bir MQPUT ya da MQPUT1 çağrısının başarıyla tamamlanmasından sonra, bu alan, bir kuyruğa konduysa, iletiyle birlikte gönderilen MDPT değerini içerir. Bu, alıkonursa iletiyle birlikte tutulan MDPT değerinin (alıkonan yayınlarla ilgili daha fazla ayrıntı için PMRET tanımına bakın) değeri olur, ancak bu ileti kendilerine gönderilen tüm yayınlarda MDPT ' i geçersiz kılmak için bir değer sağladığından, bu ileti abonelere bir yayın olarak gönderildiğinde, MDPT olarak kullanılmaz. İletinin bağlamı yoksa, alan tümüyle boştur.

Bu, MQGET çağrısına ilişkin bir çıkış alanıdır. Bu alanın uzunluğu LNPTIM tarafından verilir. Bu alanın ilk değeri 8 boş karakterdir.

### **MDREP (10 basamaklı işaretli tamsayı)**

Rapor iletileri için seçenekler.

Rapor ileti, bir uygulamayı, özgün iletiyle ilgili beklenen ya da beklenmeyen olaylarla ilgili bilgilendirmek için kullanılan başka bir iletiyle ilgili bir iletidir. MDREP alanı, uygulamanın, hangi rapor iletilerinin gerekli olduğunu, uygulama ileti verilerinin bunlara dahil edilip edilmeyeceğini ve ayrıca (hem raporlar hem de yanıtlar için) ileti ya da yanıt iletilerinde ileti ve ilinti tanıtıcılarının nasıl ayarlanacağını belirlemek için özgün iletiyi göndermesini sağlar. Aşağıdaki tip rapor iletilerinin herhangi biri ya da tümü (ya da hiçbiri) istenebilir:

- Özel durum
- Süre Bitim Tarihi
- Varışta onayla (COA)
- Teslim edilmeyi onayla (COD)
- Pozitif işlem bildirim (PAN)
- Negatif işlem bildirim (NAN)

Birden fazla rapor ileti tipi gerekliyse ya da diğer rapor seçenekleri gerekiyorsa, değerler birlikte eklenebilir (aynı sabiti bir kereden fazla eklemeyin).

Rapor iletilerini alan uygulama, MQMD ' deki MDFB alanı incelenerek raporun neden oluşturulduğunu belirleyebilir; ek ayrıntılar için MDFB alanına bakın.

Bir konuya ileti yerleştirilirken rapor seçeneklerinin kullanılması sıfır, bir ya da daha çok rapor iletilerinin oluşturulmasını ve uygulamaya gönderilmesine neden olabilir. Bunun nedeni, yayın iletilerinin sıfır, bir ya da daha çok abone olan uygulamaya gönderilebilir olması olabilir.

**Kural dışı durum seçenekleri:** Bir kural dışı durum raporu ileti istemek için aşağıdaki seçeneklerden birini belirleyebilirsiniz.

### **ROACTIVITY**

Etkinlik raporları gerekli

Bu rapor seçeneği, bu rapor seçeneği ayarına sahip bir ileti, destekleyici uygulamalar tarafından işlendiğinde bir etkinlik raporunun oluşturulmasını sağlar.

Bu rapor seçeneği ayarına sahip iletiler, herhangi bir kuyruk yöneticisi tarafından kabul edilmelidir, ancak bu seçenek bu seçeneği 'anlamıyorlar' '. Bu, rapor seçeneğinin, önceki kuyruk yöneticileri tarafından işlenip işlenmemiş olsa bile, herhangi bir kullanıcı iletilerine ayarlanmasını sağlar. Bunu başarmak için rapor seçeneği ROAUM alt alanına yerleştirilir.

Bir süreç (kuyruk yöneticisi ya da kullanıcı süreci), ROACT ayarına sahip bir iletide bir Etkinlik gerçekleştirirse, bir etkinlik raporu oluşturmayı ve yerleştirmeyi seçebilir.

Etkinlik raporu seçeneği, herhangi bir iletinin bir kuyruk yöneticisi ağı boyunca izlenmesine olanak tanır. Rapor seçeneği herhangi bir geçerli kullanıcı iletilerine belirlenebilir ve anında ileti, ağ üzerinden iletinin rotasını hesaplamaya başlayabilirler. İletiyi oluşturan uygulama, etkinlik raporu oluşturmasını etkinleştiremezse, kuyruk yöneticisi yöneticileri tarafından sağlanan bir API geçiş çıkışı kullanılarak etkinleştirilebilir.

Etkinlik raporları için çok sayıda koşul geçerli olur:

1. Ağ içinde etkinlik raporları oluşturabilen daha az sayıda kuyruk yöneticisi varsa, bu rota daha az ayrıntılandırılır.
2. Etkinlik raporları, alınan rotayı belirlemek için kolayca 'sıralanabilir' olmayabilir.
3. Etkinlik raporları, istenen hedefe giden bir rota bulamayabilir.

## **ROEXC**

Kural dışı durum raporları gerekli.

Başka bir kuyruk yöneticisine bir ileti gönderildiğinde ve ileti belirtilen hedef kuyruğa teslim edilemediğinde, bu tip bir rapor ileti kanalı aracısı tarafından oluşturulabilir. Örneğin, hedef kuyruk ya da ara iletim kuyruğu dolu olabilir ya da ileti kuyruk için çok büyük olabilir.

Kural dışı durum raporu iletilsinin oluşturulması, özgün iletinin kalıcılarına ve özgün iletinin seyahat ettiği ileti kanalının (olağan ya da hızlı) hıza bağlıdır.

- Tüm kalıcı iletiler ve normal ileti kanalları üzerinden seyahat eden kalıcı olmayan iletiler için, kural dışı durum raporu yalnızca, hata koşuluna ilişkin gönderme uygulaması tarafından belirtilen işlemin başarıyla tamamlanabilmesi için üretilir. Gönderme uygulaması, hata koşulu ortaya çıkan özgün iletinin durumunu denetlemek için aşağıdaki işlemlerden birini belirtebilir:
  - RODLQ (bu, özgün iletinin, ölü-mektup kuyruğuna yerleştirilmesine neden olur).
  - RODISC (bu, özgün iletinin atılmasına neden olur).

Gönderme uygulaması tarafından belirtilen işlem başarıyla tamamlanamazsa, özgün ileti iletim kuyruğunda bırakılır ve kural dışı durum rapor iletilisi oluşturulamaz.

- Hızlı ileti kanalları üzerinden seyahat eden kalıcı olmayan iletiler için, özgün ileti iletim kuyruğundan kaldırılır ve hata koşulu için belirtilen işlem başarıyla tamamlanamamış olsa bile, kural dışı durum raporu oluşturulur. Örneğin, RODLQ belirtilirse, ancak ilk ileti kuyruğun dolu olduğu için (say), kural dışı durum raporu iletilisi oluşturulduğundan ve özgün ileti atıldığından, özgün ileti bir ölü-mektup kuyruğuna yerleştirilemez.

Normal ve hızlı ileti kanallarına ilişkin ek bilgi için [İleti kalıcılığı](#) başlıklı konuya bakın.

Özgün iletiyi koyan uygulama, sorunun zamanuyumlu olarak MQPUT ya da MQPUT1 çağrısı tarafından döndürülmesi yoluyla sorunun bildirilebileceğini bildiren bir kural dışı durum raporu oluşturulmaz.

Uygulamalar, aldığı bir iletinin işlenemeyeceğini belirtmek için (örneğin, hesabın kredi limitini aşmasına neden olacak bir borç işlemi olduğu için) kural dışı durum raporları gönderebilir.

Özgün iletiden ileti verileri, rapor iletilisine dahil edilmiyor.

ROEXC, ROEXCD ve ROEXCF ' den birden fazlasını belirtmeyin.

## **ROEXCD**

Veri girilmesi zorunlu olan kural dışı durum raporları.

Bu, özgün iletiden gelen uygulama iletilisi verilerinin ilk 100 baytı rapor iletilisine dahil etmek dışında, ROEXC ile aynıdır. Özgün ileti bir ya da daha çok MQ üstbilgisi yapısı içeriyorsa, bunlar rapor iletilisine dahil edilir ve 100 baytlık uygulama verileri eklenir.

ROEXC, ROEXCD ve ROEXCF ' den birden fazlasını belirtmeyin.

## **ROEXCF**

Tam veri gerekli olan kural dışı durum raporları.

Bu, özgün iletiden gelen tüm uygulama iletilisi verilerinin rapor iletilisine dahil olması dışında, ROEXC ile aynıdır.

ROEXC, ROEXCD ve ROEXCF ' den birden fazlasını belirtmeyin.

**Süre bitimi seçenekleri:** Bir süre bitimi rapor iletilisi istemek için aşağıdaki seçeneklerden birini belirleyebilirsiniz.

## **ROEXP**

Son kullanma tarihi raporları gerekli.

This type of report is generated by the queue manager if the message is discarded before delivery to an application because its expiry time has passed (see the MDEXP field). Bu seçenek belirlenmezse, bu nedenle (ROEXC\* seçeneklerinden biri belirlense bile) bir ileti atılırsa, rapor iletisi oluşturulmaz.

Özgün iletiden ileti verileri, rapor iletisine dahil edilmiyor.

ROEXP, ROEXPD ve ROEXPF öğelerinden birden fazlasını belirtmeyin.

### **GENİŞLET**

Veriler gerekli olan süre bitim raporları.

Bu, özgün iletiden gelen uygulama iletisi verilerinin ilk 100 baytı rapor iletisine dahil etmek dışında, ROEXP ile aynıdır. Özgün ileti bir ya da daha çok MQ üstbilgi yapıları içeriyorsa, bunlar rapor iletisine dahil edilir ve 100 baytlık uygulama verileri eklenir.

ROEXP, ROEXPD ve ROEXPF öğelerinden birden fazlasını belirtmeyin.

### **GENİŞLET**

Eksiksiz veri gerektiren süre sonu raporları.

Bu, özgün iletiden gelen tüm uygulama iletisi verilerinin rapor iletisinde yer aldığından, ROEXP ile aynıdır.

ROEXP, ROEXPD ve ROEXPF öğelerinden birden fazlasını belirtmeyin.

**Confirm-on-gelish options**(Gelir-geliş seçenekleri): Bir varışta doğrulama isteği iletisi istemek için aşağıdaki seçeneklerden birini belirleyebilirsiniz.

### **ROCOA**

Varılan geliş raporları gerekli.

Bu tip bir rapor, hedef kuyruğa ilişkin ileti yerleştirildiğinde, hedef kuyruğa sahip olan kuyruk yöneticisi tarafından oluşturulur. Özgün iletiden ileti verileri, rapor iletisine dahil edilmiyor.

İleti bir iş biriminin bir parçası olarak konursa ve hedef kuyruk yerel bir kuyruksa, kuyruk yöneticisi tarafından oluşturulan COA rapor iletisi, iş birimi kesinleştirildiğinde ve iş birimi kesinleştirildiğinde alma için kullanılabilir duruma gelir.

İleti tanımlayıcısındaki MDFMT alanı FMXQH ya da FMDLH ise, COA raporu oluşturulmaz. Bu işlem, ileti bir iletim kuyruğuna konursa ya da teslim edilemediyse ve bir ölü-mektup kuyruğuna konursa COA raporunun oluşturulmasını önler.

ROCOA, ROCOAD ve ROCOAF dışında birden çok ROCOA belirtmeyin.

### **ROCOAD**

Veri girilmesi zorunlu olan raparlarda geliş raporları.

Bu ROCOA ile aynıdır, ancak özgün iletiden gelen uygulama iletisi verilerinin ilk 100 baytı rapor iletisine dahil edilir. Özgün ileti bir ya da daha çok MQ üstbilgi yapıları içeriyorsa, bunlar rapor iletisine dahil edilir ve 100 baytlık uygulama verileri eklenir.

ROCOA, ROCOAD ve ROCOAF dışında birden çok ROCOA belirtmeyin.

### **ROCOAF**

Tam veri gerekli olan geliş raporlarını onaylar.

Bu ROCOA ile aynıdır, ancak özgün iletiden gelen tüm uygulama iletisi verileri rapor iletisine dahil edilir.

ROCOA, ROCOAD ve ROCOAF dışında birden çok ROCOA belirtmeyin.

**At ve süre bitimi seçenekleri:** Rapor iletileri için süre bitimi ve atma işaretini ayarlamak için aşağıdaki seçeneği belirleyebilirsiniz.

### **ROPDAE**

Rapor ileti süre bitimini ayarlayın ve işareti atın.

Bu seçenek, rapor iletilerinin ve yanıt iletilerinin son kullanma tarihini devralmasını ve özgün iletilerinden vazgeçip atmayacağını (atılıp atılmamasını) sağlar. Bu seçenek kümesi, rapor ve yanıt iletileri ile birlikte:

1. RODISC işaretini (ayarlandıysa) devral.
2. İleti süre bitimi raporu değilse, iletinin kalan süre bitimi süresini devralır. İleti süre bitimi raposa, süre bitimi 60 saniyeye ayarlanır.

Bu seçenek ayarında aşağıdakiler geçerlidir:

**Not:**

1. Rapor ve yanıt iletileri, bir atma işareti ve süre bitimi değeriyle oluşturulur ve sistem içinde kalmayamaz.
2. İzleme rotası iletilerinin, izleme dışı rota üzerinde etkinleştirilen kuyruk yöneticilerindeki hedef kuyruklara ulaşması engellenir.
3. İletişim bağlantıları kesilirse, kuyrukların teslim edilemeyen raporlarla doldurulması engellenir.
4. Komut sunucusu yanıtları, isteğin kalan süre bitimini devralır.

**Confirm-on-delivery options**(Teslim alma seçeneklerini onaylayın): Bir teslim alma raporu iletisini istemek için aşağıdaki seçeneklerden birini belirleyebilirsiniz.

**ROCOD**

Teslim edilme raporları gerekli.

Bu tip bir rapor, bir uygulama hedef kuyruktan iletiyi kuyruktan silinmesine neden olan bir şekilde hedef kuyruktan aldığı anda, kuyruk yöneticisi tarafından üretilir. Özgün iletiden ileti verileri, rapor iletisine dahil edilmiyor.

İleti bir iş biriminin bir parçası olarak alınır, rapor iletisi aynı iş birimi içinde oluşturulur; böylece, iş birimi kesinleştirilinceye kadar rapor kullanılabilir değildir. İş birimi yedeklendiyse, rapor gönderilmez.

İleti tanımlayıcısındaki MDFMT alanı FMDLH ise, COD raporu oluşturulmaz. Bu işlem, iletinin teslim edilememesi ve kuyruksuz bir kuyruğa konması durumunda COD raporunun oluşturulmasını önler.

Hedef kuyruk bir XCF kuyruğsa ROCOD geçerli değildir.

ROCOD, ROCODD ve ROCODF ' den birden fazlasını belirtmeyin.

**ROCDD**

Veri girilmesi zorunlu olan teslim alma raporları.

Bu, ROCOD ile aynıdır, ancak özgün iletiden gelen uygulama iletisi verilerinin ilk 100 baytı rapor iletisine dahil edilir. Özgün ileti bir ya da daha çok MQ üstbilgisi yapısı içeriyorsa, bunlar rapor iletisine dahil edilir ve 100 baytlık uygulama verileri eklenir.

Özgün ileti için MQGET çağrısında GMATM belirtilirse ve alınan ileti kısaltılırsa, rapor iletisine yerleştirilen uygulama iletisi verisi miktarı alt sınır olan ileti verileri miktarı alt sınır olur:

- Özgün iletinin uzunluğu
- 100 bayt.

Hedef kuyruk bir XCF kuyruğsa ROCDD geçerli değildir.

ROCOD, ROCODD ve ROCODF ' den birden fazlasını belirtmeyin.

**ROCODF**

Tam veri gerekli olan teslim raporları için onay bilgileri.

Bu, özgün iletiden gelen tüm uygulama iletisi verilerinin rapor iletisinde yer aldığından, ROCOD ile aynıdır.

Hedef kuyruk bir XCF kuyruğsa ROCODF geçerli değildir.

ROCOD, ROCODD ve ROCODF ' den birden fazlasını belirtmeyin.

**Eylem-bildirim seçenekleri:** Alma uygulamasının olumlu ya da olumsuz işlem rapor iletisi göndermesini istemek için aşağıdaki seçeneklerden birini ya da her ikisini belirleyebilirsiniz.

#### **ROPAN**

Olumlu eylem bildirim raporları gerekli.

Bu rapor tipi, iletiyi alan ve üzerinde işlem yapan uygulama tarafından oluşturulur. İletide istenen işlemin başarılı bir şekilde gerçekleştirildiğini gösterir. Raporu oluşturan uygulama, raporla birlikte herhangi bir verinin dahil edilip edilmeyeceğini belirler.

İletiyi alan uygulamaya bu isteği iletmekten başka bir seçenek dışında, kuyruk yöneticisi bu seçeneğe dayalı olarak herhangi bir işlem görmez. Uygunsa, raporu oluşturmak için uygulamanın alma sorumluluğundadır.

#### **RONAN**

Negatif işlem bildirim raporları gerekli.

Bu rapor tipi, iletiyi alan ve üzerinde işlem yapan uygulama tarafından oluşturulur. İletide istenen işlemin başarılı bir şekilde gerçekleştirilmediğini gösterir. Raporu oluşturan uygulama, raporla birlikte herhangi bir verinin dahil edilip edilmeyeceğini belirler. Örneğin, isteğin neden gerçekleştirilemediğini gösteren bazı veriler içermek istenebilir.

İletiyi alan uygulamaya bu isteği iletmekten başka bir seçenek dışında, kuyruk yöneticisi bu seçeneğe dayalı olarak herhangi bir işlem görmez. Uygunsa, raporu oluşturmak için uygulamanın alma sorumluluğundadır.

Olumlu bir işleme karşılık gelen ve olumsuz bir eyleme karşılık gelen koşulların belirlenmesi, uygulamanın sorumluluğunda. Ancak, istek yalnızca kısmen gerçekleştirilmişse, istenirse PAN raporu yerine bir NAN raporu oluşturulmalıdır. Olası her koşulun, olumlu bir işleme ya da olumsuz bir işleme karşılık gelmesi de önerilir, ancak her ikisi de önerilmemelidir.

**İleti tanıtıcısı seçenekleri:** Rapor iletisi (ya da yanıt iletisinin) MDMID ' in nasıl ayarlanacağını denetlemek için aşağıdaki seçeneklerden birini belirleyebilirsiniz.

#### **RONMI**

Yeni ileti tanıtıcısı.

Bu varsayılan işlemdir ve bir rapor ya da yanıt bu iletinin bir sonucu olarak oluşturulursa, rapor ya da yanıt iletisi için yeni bir MDMID oluşturulacağını belirtir.

#### **ROPMI**

İleti tanıtıcısını iletin.

Bir rapor ya da yanıt bu iletinin bir sonucu olarak oluşturulduysa, bu iletinin MDMID 'i rapor ya da yanıt iletisinin MDMID ' e kopyalanır.

Bir yayın iletisinin MsgId , yayının bir kopyasını alan her abone için farklı olur; bu nedenle, rapor ya da yanıt iletisine kopyalanan MsgId her bir abone için farklı olur.

Bu seçenek belirlenmezse, RONMI varsayılr.

**İlinti tanıtıcısı seçenekleri:** Rapor iletisi (ya da yanıt iletisinin) MDCID ' in nasıl ayarlanacağını denetlemek için aşağıdaki seçeneklerden birini belirleyebilirsiniz.

#### **ROCMTC**

İleti tanıtıcısını ilinti tanıtıcısına kopyalayın.

Bu varsayılan işlemdir ve bir rapor ya da yanıt bu iletinin bir sonucu olarak oluşturulursa, bu iletinin MDMID 'un rapor ya da yanıt iletisinin MDCID ' e kopyalanacağına işaret eder.

The MsgId of a publication message will be different for each subscriber that receives a copy of the publication and therefore the MsgId copied into the CorreIID of the report or reply message will be different for each one.

#### **ROPCI**

Geçiş ilintilendirme tanıtıcısı.



Bir rapor ya da yanıt bu iletinin bir sonucu olarak oluşturulduysa, bu iletinin MDCID 'i rapor ya da yanıt iletisinin MDCID ' e kopyalanır.

Bir yayın iletisinin MDCID , SOSCİD seçeneğini kullanmadığı ve MQSD ' deki SCDIC alanını CINONE olarak ayarlamadığı sürece bir aboneye özgü olur. Bu nedenle, rapor ya da yanıt iletisinin MDCID 'ine kopyalanan MDCID ' nin her biri için farklı olması mümkündür.

Bu seçenek belirlenmezse, ROCMT C kabul edilir.

Özgün iletide ROPMI ya da ROPCI seçeneklerinin ayarlanıp ayarlanmadığını denetlemek için istekleri yanıtlama ya da rapor oluşturma iletileri oluşturma önerilir. Bu işlemler gerçekleşirse, sunucular bu seçenekler için açıklanan işlemi kabul etmelidir. İkisi de ayarlanmazsa, sunucuların karşılık gelen varsayılan eylemi alması gerekir.

: Özgün iletinin hedef kuyruğa teslim edilemediğinde, yok etmeyi denetlemek için aşağıdaki seçeneklerden birini belirleyebilirsiniz. Bu seçenekler yalnızca, gönderme yapan uygulama tarafından istendiye, bir kural dışı durum raporu iletisinin oluşturulup oluşturulmamasına neden olacak durumlar için geçerlidir. Uygulama, yok etme seçeneklerini, kural dışı durum raporlarından bağımsız olarak ayarlayabilir.

### **RODLQ**

İletinin gönderilip gönderilmemesine ilişkin ileti.

Bu, varsayılan işlemidir ve ileti hedef kuyruğa teslim edilemezse, iletinin ölü harf kuyruğunda yerleştirileceğini belirtir. Bu, aşağıdaki durumlarda gerçekleşir:

- Özgün iletiyi koyan uygulama, MQPUT ya da MQPUT1 çağrısının döndürdüğü neden kodu yoluyla sorunun zamanuyumlu olarak bildirilememesini sağlar. Gönderen tarafından istendiye, bir kural dışı durum raporu iletisi üretilir.
- Özgün iletiyi koyan uygulama bir konuya yerleştirildi

Gönderen tarafından istendiye, bir kural dışı durum rapor iletisi oluşturulacak.

### **RODISC**

İletiyi atın.

Bu, hedef kuyruğa teslim edilemezse iletinin atılması gerektiğini belirtir. Bu, aşağıdaki durumlarda gerçekleşir:

- Özgün iletiyi koyan uygulama, MQPUT ya da MQPUT1 çağrısının döndürdüğü neden kodu yoluyla sorunun zamanuyumlu olarak bildirilememesini sağlar. Gönderen tarafından istendiye, bir kural dışı durum raporu iletisi üretilir.
- Özgün iletiyi koyan uygulama bir konuya yerleştirildi

Gönderen tarafından istendiye, bir kural dışı durum rapor iletisi oluşturulacak.

Özgün iletiyi gönderene geri döndürmeniz gerekiyorsa, özgün ileti, ölü-mektup kuyruğuna yerleştirilmeden, gönderenin ROEXCF ile RODISC belirtmesi gerekir.

**Varsayılan seçenek:** Rapor seçeneği gerekmiyorsa, aşağıdakileri belirtebilirsiniz:

### **ROYOK**

Rapor gerekli değil.

Bu değer, başka bir seçeneğin belirlenmemesine dikkat etmek için kullanılabilir. RONONE, program belgelerine yardımcı olmak için tanımlıdır. Bu seçeneğin başka biriyle birlikte kullanılması amaçlanmaz, ancak değeri sıfır olduğu için bu tür bir kullanım saptanamaz.

### **Genel bilgiler:**

1. Gerekli tüm rapor tiplerinin, özgün iletiyi gönderen uygulama tarafından özel olarak istenmesi gerekir. Örneğin, bir COA raporu istenirse, ancak bir kural dışı durum raporu yoksa, ileti hedef kuyruğa yerleştirildiğinde COA raporu oluşturulur, ancak ileti oraya ulaştığında hedef kuyruk dolduysa herhangi bir kural dışı durum raporu oluşturulmaz. Hiçbir MDREP seçeneği ayarlanmazsa, kuyruk yöneticisi ya da ileti kanalı aracısı (MCA) tarafından hiçbir rapor iletisi oluşturulmadı.

Bazı rapor seçenekleri, yerel kuyruk yöneticisi bunları tanımamış olsa da belirtilebilir; bu seçenek, hedef kuyruk yöneticisi tarafından işlenmek üzere olduğunda kullanışlıdır. Daha ayrıntılı bilgi için bkz. "IBM i' ta rapor seçenekleri ve ileti işaretleri" sayfa 1410 .

Bir rapor iletilisi istenirse, raporun gönderilmesi gereken kuyruğun adı, MDRQ alanında belirtilmelidir. Bir rapor iletilisi alındığında, raporun niteliği, ileti tanımlayıcısındaki MDFB alanı incelenerek saptanabilir.

2. Bir rapor iletilisi oluşturan kuyruk yöneticisi ya da MCA, rapor iletilisini yanıt kuyruğuna (örneğin, yanıt kuyruğu ya da iletim kuyruğu dolu olduğu için) koyamıyorsa, rapor iletilisi ölü harf kuyruğunda yerine yerleştirilir. Bu hata da başarısız olursa ya da herhangi bir ölü harf kuyruğu yoksa, işlem, rapor iletilisinin tipine bağlıdır:

- Rapor iletilisi bir kural dışı durum bildirimiyse, kural dışı durum raporunun oluşturulmasına neden olan ileti, iletim kuyruğunda bırakılır; bu, iletilinin kaybolmamasını sağlar.
- Diğer tüm rapor tipleri için, rapor iletilisi atılır ve işleme olağan şekilde devam eder. Bu, özgün ileti güvenli bir şekilde (COA ya da COD rapor iletileri için) güvenli bir şekilde teslim edildiğinden ya da artık herhangi bir ilgi göstermediği için yapılır (bir süre sonu rapor iletilisi için).

Bir ileti kuyruğunda (hedef kuyruk ya da ara iletim kuyruğu) başarıyla yerleştirildikten sonra, ileti artık özel işleme tabi değildir; diğer iletiler gibi işlem görür.

3. Rapor oluşturulduğunda, MDRQ kuyruğu açılır ve aşağıdaki durumlar dışında, rapora neden olan iletilinin MQMD ' de MDUID yetki yetkisini kullanarak rapor iletilisi açılır:

- Alıcı bir MCA tarafından oluşturulan kural dışı durum raporları, rapora neden olan iletiyi yerleştirmeye çalışıldığında, MCA ' nın kullandığı yetki ile birlikte ortaya çıkarılır. The CDPA channel attribute determines the user identifier used.
- Kuyruk yöneticisi tarafından oluşturulan COA raporları, rapora neden olan ileti, raporu oluşturan kuyruk yöneticisi üzerine konulduğunda kullanılan herhangi bir yetki ile konabiliyor. Örneğin, ileti MCA 'nın kullanıcı tanımlayıcısı kullanılarak bir alıcı MCA tarafından konulduysa, kuyruk yöneticisi, MCA' nın kullanıcı kimliğini kullanarak COA raporunu koyar.

Rapor oluşturan uygulamalar, normalde bir yanıt oluşturmak için kullanılanlarla aynı yetkiyi kullanmalıdır; bu durumda özgün iletide kullanıcı kimliğinin yetkisi olmalıdır.

Eğer rapor uzak bir hedefe gitmek zorunda ise, gönderenler ve alıcılar diğer mesajlarla aynı şekilde bunu kabul edip etmeyeceğine karar verebilirler.

4. Veri içeren bir rapor iletilisi istenirse:

- Rapor iletilisi her zaman, özgün iletilinin göndereni tarafından istenen veri miktarlarıyla oluşturulur. Rapor iletilisi yanıt kuyruğu için çok büyükse, daha önce açıklanan işleme gerçekleşir; rapor iletilisi, yanıt kuyruğuna sığması için hiçbir zaman kısaltılır.
- Özgün iletilinin MDFMT değeri FMXQH ise, rapordaki veriler MQXQH ' yi içermez. Rapor verileri, özgün iletide MQXQH ' nin ötesindeki ilk veri byte 'ını kullanarak başlar. Bu durum, kuyruğun bir iletim kuyruğu olup olmadığını ortaya çıkar.

5. Yanıt kuyruğunda bir COA, COD ya da süre bitimi raporu alındıysa, özgün iletilinin teslim edildiği, teslim edildiği ya da gerektiği gibi süresi dolduğu garanti edilir. Ancak, bu rapor iletililerinden biri ya da daha fazlası istenirse ve alınmazsa, aşağıdakilerden biri gerçekleşmiş olabileceği için ters olarak ters çevrilemez:

- a. Bir bağlantı kapalı olduğu için rapor iletilisi tutuluyor.
- b. Rapor iletilisi, bir ara iletim kuyruğunda ya da yanıt kuyruğunda bir engelleme koşulu var olduğundan (örneğin, kuyruğun tam ya da engellenmiş olduğu bir kuyruk) olduğu için, bu ileti iletilisi tutuluyor.
- c. Rapor iletilisi, bir ölü-mektup kuyruğunda.
- d. Kuyruk yöneticisi, rapor iletilisini oluşturma girişiminde bulunurken, iletiyi uygun kuyruğa koyamadı ve bu iletiyi ölü harf kuyruğuna yerleştiremedi; böylece, rapor iletilisi oluşturulamadı.

- e. Raporlanan işlem (geliş, teslim ya da süre bitimi) ve karşılık gelen rapor iletilisinin oluşturulması arasında kuyruk yöneticisinin hatası oluştu. (Uygulama bir iş birimi içinde özgün iletiyi aldıysa, COD rapor iletilisi aynı iş birimi içinde oluşturulduğunda COD rapor iletileri için bu gerçekleşmez.)

Kural dışı durum rapor iletileri, önceki 1, 2 ve 3 gibi nedenlerle aynı şekilde tutulabilir. Ancak, bir MCA bir kural dışı durum raporu iletilisi oluşturamazsa (rapor iletilisi yanıt kuyruğuna ya da ölü-mektup kuyruğuna konulamaz), özgün ileti göndericindeki iletim kuyruğunda kalır ve kanal kapatılır. Bu durum, rapor iletilisinin, kanalın gönderme ya da alma sırasında oluşturulup oluşturulmayacağı dikkate alınmalıdır.

6. Özgün ileti geçici olarak engellendiyse (oluşturulmakta olan bir kural dışı durum raporu iletilisi ve özgün ileti bir ölü-mektup kuyruğuna konursa), ancak tıkanıklık temizlenir ve bir uygulama, özgün iletiyi ölü harf kuyruğundan okur ve bu iletiyi yeniden hedefine yerleştirir, aşağıdaki durumlar oluşabilir:
- Bir kural dışı durum raporu iletilisi oluşturulsa da, özgün ileti sonunda hedefine başarıyla ulaşır.
  - Özgün ileti daha sonra başka bir engelle karşılaşılabileceğinden, tek bir özgün iletiye göre birden çok kural dışı durum raporu iletilisi üretilir.

#### **Konu bir konuya yerleştirilirken iletileri raporla:**

1. Bir konuya ileti yerleştirilirken raporlar oluşturulabilir. Bu ileti, tüm abonelere, sıfır, bir ya da daha fazla olabilecek konuya gönderilecek. Sonuç olarak birçok rapor iletilisi oluşturulabilirken, rapor seçeneklerini kullanma seçilirken bu işlem dikkate alınmalıdır.
2. Bir konuya ileti yerleştirilirken, iletilinin bir kopyasını verilecek birçok hedef kuyrukları olabilir. Bu hedef kuyruklardan bazılarının kuyruk dolu gibi bir sorunu varsa, MQPUT ' un başarıyla tamamlanması, NPMGDLV ya da PMSGDLV (iletilinin kalıcısına bağlı olarak) ayarına bağlıdır. Ayar, hedef kuyruğa ileti tesliminin başarılı olması gerekiyorsa (örneğin, kalıcı bir aboneye kalıcı bir iletilidir ve PMSGDLV ALL ya da ALLDIR olarak ayarlanır), başarı aşağıdaki ölçütlerden biri olarak tanımlıdır:
  - Abone kuyruğuna başarılı bir şekilde yerleştirilecek
  - Abone kuyruğu iletiliyi alamazsa, RODLQ ve Dead-letter kuyruğuna başarılı bir şekilde konun.
  - Abone kuyruğu iletiliyi alamazsa RODISC ' yi kullanın.

#### **İleti bölümleri için rapor iletileri:**

1. Kesimlemeye izin verilen iletiler için rapor iletileri istenebilir (MFSEGA işaretinin açıklamasına bakın). Kuyruk yöneticisi iletiliyi bölümlenmek için gerekli bulursa, ilgili koşulla daha sonra karşılaştıran her bir kesim için bir rapor iletilisi oluşturulabilir. Bu nedenle, istekte bulunulan her bir rapor iletilisi tipi için birden çok rapor iletilisi almak üzere uygulamalar hazırlanmalıdır. Rapor iletilisinde yer alan MDGID alanı, özgün iletilinin grup tanıtıcısı ile birden çok raporu ilişkilendirmek için ve her bir rapor iletilisinin tipini tanımlamak için kullanılan MDFB alanı için kullanılabilir.
2. Kesimlere ilişkin rapor iletilerini almak için GMLGO kullanılıyorsa, sonraki MQGET çağrılarını farklı tiplere ait raporların döndürülebileceğini unutmayın. Örneğin, kuyruk yöneticisi tarafından bölümlenmiş bir ileti için hem COA hem de COD raporları istenirse, rapor iletileri için MQGET çağrılarını, COA ve COD rapor iletilerinin öngörülemez bir şekilde serpiştirilmiş olarak dönelmesini sağlar. Bu, GMCMM seçeneği kullanılarak (isteğe bağlı olarak GMATM ' de) önlenemez. GMCMM, kuyruk yöneticisinin aynı rapor tipine sahip rapor iletilerini yeniden bir araya getirmesine neden olur. Örneğin, ilk MQGET çağrısı özgün iletiyle ilgili tüm COA iletilerini yeniden birleştirebilir ve ikinci MQGET çağrısı tüm COD iletilerine yeniden bir araya gelebilir. İlk olarak yeniden birleştirilen, kuyruğun ilk sırasında hangi rapor iletilisi tipinin gerçekleşeceğini bağlıdır.
3. Kesimler yerleştiren uygulamalar, her bir bölüm için farklı rapor seçenekleri belirtebilir. Ancak, aşağıdaki noktalara dikkat edilmelidir:
  - Kesimler GMCMPM seçeneği kullanılarak alınırsa, kuyruk yöneticisi yalnızca ilk segmente rapor seçenekleri onurlandırılır.
  - Kesimler birer birer alınırsa ve bunların çoğu ROCOD\* seçeneklerinden birine sahip olur, ancak en az bir kesim söz etmiyorsa, tek bir MQGET çağrısıyla rapor iletilerini almak için GMCMPM

seçeneğini kullanmak ya da tüm rapor iletilerinin ne zaman geldiğini saptamak için GMASGA seçeneğini kullanmanız mümkün olmaz.

4. Bir MQ ağında, kuyruk yöneticilerinin farklı yeteneklere sahip olması mümkündür. Segmentasyonu desteklemeyen bir kuyruk yöneticisi ya da MCA tarafından bir kesime ilişkin bir rapor ileti oluşturulursa, kuyruk yöneticisi ya da MCA varsayılan olarak rapor iletilerinde gerekli bölüm bilgilerini içermeyecek ve bu durum, raporun oluşturulmasına neden olan özgün iletinin tanınmasını zorlaştırabilir. Bu zorluk, uygun RO\* D ya da RO\* F seçeneklerini belirleyerek, rapor iletileriyle veri istenerek önlenir. Ancak, RO\* D belirtilirse, rapor iletileri, bir kuyruk yöneticisi ya da bölümlendirmeyi desteklemeyen bir MCA tarafından oluşturulduysa, rapor iletilerini alan uygulamaya 100 byte 'tan az uygulama iletileri verisi döndürülebileceğini unutmayın.

**Rapor iletileri için ileti tanımlayıcısının içeriği:** Kuyruk yöneticisi ya da ileti kanalı aracısı (MCA) bir rapor iletileri oluşturduğunda, ileti tanımlayıcısındaki alanları aşağıdaki değerlere ayarlar ve sonra iletiyi olağan şekilde yerleştirir.

*Çizelge 709. Bir rapor iletileri sistem tarafından üretildiğinde, MQMD alanları için kullanılan değerler*

MQMD ' de alan	Kullanılan değer
MDSID	MDSIDV
MDVER	MDVER2
MDREP	ROYOK
MDMT	MTRPRT
MDEXP	EIULIM
MDFB	Raporun niteine uygun olarak (FBCOA, FBCOD, FBEXP ya da RC* değeri)
MDENC	Özgün ileti tanımlayıcısından kopyalandı
MDCSI	Özgün ileti tanımlayıcısından kopyalandı
MDFMT	Özgün ileti tanımlayıcısından kopyalandı
MDPRI	Özgün ileti tanımlayıcısından kopyalandı
MDPER	Özgün ileti tanımlayıcısından kopyalandı
MDMID	Özgün ileti tanımlayıcısındaki rapor seçenekleri tarafından belirlendiği gibi
MDCID	Özgün ileti tanımlayıcısındaki rapor seçenekleri tarafından belirlendiği gibi
MDBOC	0
MDRQ	Boşluklar
MDRM	Kuyruk yöneticisinin adı
MDUID	PMPASI seçeneği tarafından belirlenen seçenek
MDACC	PMPASI seçeneği tarafından belirlenen seçenek
MDAID	PMPASI seçeneği tarafından belirlenen seçenek
MDPAT	ATQM ya da ileti kanalı aracısına uygun olan
MDPAN	Kuyruk yöneticisi adının ya da ileti kanalı aracısı adının ilk 28 byte 'ı. For report messages generated by the IMS bridge, this field contains the XCF group name and XCF member name of the IMS system to which the message relates.
MDPD	Rapor iletilerinin gönderildiği tarih
MDPT	Rapor iletilerinin gönderildiği saat

MQMD ' de alan	Kullanılan değer
MDAOD	Boşluklar
MDGID	Özgün ileti tanımlayıcısından kopyalandı
MDSEQ	Özgün ileti tanımlayıcısından kopyalandı
MDOFF	Özgün ileti tanımlayıcısından kopyalandı
MDMFL	Özgün ileti tanımlayıcısından kopyalandı
MDOLN	OLUNDF değilse, özgün ileti tanımlayıcısından kopyalanır ve özgün ileti verilerinin uzunluğuna göre ayarlanır

Aşağıdakilerin dışında, benzer değerleri ayarlamak için rapor oluşturan bir uygulama önerilir:

- MDRM alanı boşluklara ayarlanabilir (kuyruk yöneticisi, ileti bulunduğu bu değeri yerel kuyruk yöneticisinin adıyla değiştirecektir).
- Bağlam alanları, genellikle PMPASI gibi bir yanıt için kullanılmış olan seçenek kullanılarak ayarlanmalıdır.

**Rapor alanının çözülmesi:** MDREP alanı alt alanlar içerir; bunun nedeniyle, ileti gönderenin belirli bir rapor istemesinin "[Analyzing the report field on IBM i](#)" sayfa 1411' de açıklanan tekniklerden birini kullanması gerekip gerekmediğini denetlemesi gereken uygulamalar.

Bu, MQGET çağrısına ilişkin bir çıkış alanıdır ve MQPUT ve MQPUT1 çağrılarında ilişkin bir giriş alanı. Bu alanın ilk değeri RONONE olur.

#### MDRM (48 baytlık karakter dizisi)

Yanıt kuyruğu yöneticisinin adı.

Bu ad, yanıt iletinin ya da rapor iletinin gönderilmesi gereken kuyruk yöneticisinin adıdır. MDRQ , bu kuyruk yöneticisinde tanımlı olan bir kuyruğun yerel adıdır.

MDRM alanı boşsa, yerel kuyruk yöneticisi, kuyruk tanımlamalarında **MDRQ** adını arar. Bu adı taşıyan bir uzak kuyruğun yerel tanımlaması varsa, iletilen iletteki **MDRM** değeri, uzak kuyruğun tanımlamasındaki **RemoteQMgrName** özneliğinin değeriyle değiştirilir ve alıcı uygulama ileti için bir MQGET çağrısı yayınlarken bu değer ileti tanımlayıcısında döndürülecektir. Uzak kuyruğun yerel tanımlaması yoksa, iletiyle aktarılan MDRM değeri yerel kuyruk yöneticisinin adıdır.

Ad belirlendiyse, sondaki boşluklar olabilir; ilk boş karakter ve izleyen karakterler boşluk olarak işlenir. Ters durumda, adın kuyruk yöneticilerine ilişkin adlandırma kurallarına uygun olması ya da bu adın gönderme kuyruğu yöneticisi tarafından bilinmesi gerekmez; bu değer, iletilen iletide **MDRM** ' in yerine geçiyorsa, iletilen ad için de geçerlidir.

Bir yanıtlama kuyruğu gerekli değilse, MDRM alanının boşluklara ayarlanması gerektiği için (bu işaretlenmemiş olmasına rağmen) önerilir; alan kullanıma hazırlanmamış olarak bırakılmamalı.

MQGET çağrısı için, kuyruk yöneticisi her zaman boşluklarla doldurulan adı, alanın uzunluğuna kadar döndürür.

Bu, MQGET çağrısına ilişkin bir çıkış alanıdır ve MQPUT ve MQPUT1 çağrılarında ilişkin bir giriş alanı. Bu alanın uzunluğu LNQMN tarafından verilir. Bu alanın ilk değeri 48 boş karakterdir.

#### MDRQ (48 baytlık karakter dizilimi)

Yanıt kuyruğunun adı.

Bu, iletiye ilişkin alma isteğini yayınlayan uygulamanın MTRPLY ve MTRPRT iletilerini göndermesi gereken ileti kuyruğunun adıdır. Ad, MDRM ile tanımlanan kuyruk yöneticisinde tanımlı olan bir kuyruğun yerel adıdır. Kuyruk yöneticisi, ileti bulunduğu bunu doğrulaması da, bu kuyruk bir model kuyruğu olmamalıdır.

MQPUT ve MQPUT1 çağrıları için, MDMT alanı MTRQST değerine sahipse ya da MDREP alanı tarafından herhangi bir rapor iletisi istenirse bu alan boş olmamalıdır. Ancak, belirtilen değer (ya da yerine koyulan), ileti tipi ne olursa olsun, iletiye ilişkin alma isteğini içeren uygulamaya geçirilir.

MDRM alanı boşsa, yerel kuyruk yöneticisi kendi kuyruk tanımlamalarında MDRQ adını arar. Bu adı taşıyan bir uzak kuyruğun yerel tanımlaması varsa, iletilen iletteki MDRQ değeri, uzak kuyruğun tanımlamasındaki **RemoteQName** özniteliğinin değeriyle değiştirilir ve alıcı uygulama ileti için bir MQGET çağrısı yayınlarken bu değer ileti tanımlayıcısında döndürülecektir. Uzak kuyruğun yerel tanımlaması yoksa, MDRQ değişmez.

Ad belirlendiyse, sondaki boşluklar olabilir; ilk boş karakter ve izleyen karakterler boşluk olarak işlenir. Ters durumda, adın kuyruklar için adlandırma kurallarına uygun olmasına dikkat edin; MDRQ , iletilen iletide değiştirilirse, iletilen ad için de bu değer geçerlidir. Yapılan tek denetim, koşul gerektiriyorsa, bir ad belirtilmesine neden olur.

Bir yanıtlama kuyruğu gerekli değilse, MDRQ alanının boşluklara ayarlanması gerektiği için (bu işaretlenmemiş olmasına rağmen) önerilir; alan kullanıma hazırlanmamış olarak bırakılmamalı.

MQGET çağrısı için, kuyruk yöneticisi her zaman boşluklarla doldurulan adı, alanın uzunluğuna kadar döndürür.

Bir rapor iletisi gerektiren bir ileti teslim edilemezse ve rapor iletisi de belirtilen kuyruğa teslim edilemezse, hem özgün ileti, hem de rapor iletisi ölü-mektup (teslim edilemeyen ileti) kuyruğuna gider. [“IBM üzerinde kuyruk yöneticisine ilişkin öznitelikler” sayfa 1376](#) içinde açıklanan **DeadLetterQName** özniteliğine bakın.

Bu, MQGET çağrısına ilişkin bir çıkış alanıdır ve MQPUT ve MQPUT1 çağrılarına ilişkin bir giriş alanı. Bu alanın uzunluğu LNQN tarafından verilir. Bu alanın ilk değeri 48 boş karakterdir.

#### **MDSEQ (10 basamaklı işaretli tamsayı)**

Grup içindeki mantıksal iletinin sıra numarası.

Sıra numaraları 1 'den başlar ve gruptaki her yeni mantıksal ileti için en çok 999 999 999 'a kadar artar. Grupta yer almayan fiziksel bir iletinin sıra numarası 1 'dir.

Bu alan, aşağıdaki durumlarda, MQPUT ya da MQGET çağrısında uygulama tarafından belirlenemez:

- MQPUT çağrısında, PMLOGO belirtildi.
- MQGET çağrısında, MOSEQN belirtilmedi.

Bunlar, rapor iletisi olmayan ileteler için bu çağrıları kullanmanın önerilen yollarıdır. Ancak, uygulama daha fazla denetim gerektiriyorsa ya da çağrı MQPUT1 ise, uygulama MDSEQ ' in uygun bir değere ayarlandığını doğrulamalıdır.

MQPUT ve MQPUT1 çağrılarına giriş sırasında, kuyruk yöneticisi [Tablo 1](#) ' de ayrıntılı olarak belirtilen değeri kullanır. MQPUT ve MQPUT1 çağrılarının çıktısında, kuyruk yöneticisi bu alanı, iletiyle birlikte gönderilen değere ayarlar.

MQGET çağrısına giriş sırasında, kuyruk yöneticisi [Tablo 1](#) ' de ayrıntılı olarak belirtilen değeri kullanır. MQGET çağrısından çıktıda, kuyruk yöneticisi bu alanı, alınan ileti için değere ayarlar.

Bu alanın ilk değeri bir alandır. MDVER , MDVER2 değerinden küçükse bu alan yok sayılır.

#### **MDSID (4 baytlık karakter dizilimi)**

Yapı tanıtıcısı.

Değer şu olmalıdır:

##### **MDSIDV**

İleti tanımlayıcı yapısına ilişkin tanıtıcı.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MDSIDV ' dir.

#### **MDUID (12 baytlık karakter dizilimi)**

Kullanıcı kimliği.

Bu, iletinin *kimlik bağlamının* bir parçasıdır. İleti bağlamına ilişkin daha fazla bilgi için bkz. İleti bağlamı ve Bağlam bilgilerini denetleme.

MDUID , iletiyi oluşturan uygulamanın kullanıcı tanımlayıcısını belirtir. Kuyruk yöneticisi bu bilgileri karakter verisi olarak değerlendirir, ancak biçiminin biçimini tanımlamaz.

After a message has been received, MDUID can be used in the ODAU field of the **OBJDSC** parameter of a subsequent MQOPEN or MQPUT1 call, so that the authorization check is performed for the MDUID user instead of the application performing the open.

Kuyruk yöneticisi bir MQPUT ya da MQPUT1 çağrısı için bu bilgileri oluşturduğunda, kuyruk yöneticisi ortamdaki saptanan bir kullanıcı kimliğini kullanır.

Kullanıcı kimliği ortamdaki saptanırken:

- **z/OS** On z/OS, the queue manager uses:
  - Toplu iş için, JES JOB kartından kullanıcı kimliği ya da başlatma görevi
  - TSO için, oturum açma kullanıcı kimliği
  - CICS için, görevle ilişkili kullanıcı kimliği
  - IMS için kullanıcı kimliği, uygulamanın tipine bağlıdır:
    - Bunun için:
      - İleti olmayan BMP bölgeleri
      - İletilmeyen IFP bölgeleri
      - Başarılı bir GU çağrısı yayınlamamış olan ileti BMP ve ileti IFP bölgeleri

kuyruk yöneticisi, bölge JES JOB kartından ya da TSO kullanıcı tanımlayıcısından kullanıcı kimliğini kullanır. Bu değer boş ya da boş değeriye, program belirtimi öbeğinin (PSB) adını kullanır.

- Bunun için:
  - Başarılı bir GU çağrısı yayınlayan ileti BMP ve ileti IFP bölgeleri
  - MPP bölgeleri

Kuyruk yöneticisi aşağıdakilerden birini kullanır:

- İletiyi ilişkilendirilen oturum açmış kullanıcı kimliği
- Mantıksal uçbirim (LTERM) adı
- Bölge JES JOB kartından kullanıcı kimliği
- TSO kullanıcı kimliği
- PSB adı

- **IBM i** On IBM i, the queue manager uses the name of the user profile associated with the application job.

- **Linux** **AIX** On AIX and Linux, the queue manager uses:

- Uygulamanın oturum açma adı
- Oturum açma kullanılmıyorsa, işlemin etkin kullanıcı kimliği
- Uygulama bir CICS hareketiyse, işlemle ilişkili kullanıcı kimliği

- VSE/ESA' da bu ayrılmış bir alandır.

- **Windows** Windows üzerinde, kuyruk yöneticisi, oturum açmış kullanıcı adının ilk 12 karakterini kullanır.

For the MQPUT and MQPUT1 calls, this is an input/output field if PMSETI or PMSETA is specified in the **PMO** parameter. Alan içindeki boş değerli bir karakteri izleyen herhangi bir bilgi atılır. Boş değerli

karakter ve izleyen karakterler kuyruk yöneticisi tarafından boşluklara dönüştürülür. PMSETI ya da PMSETA belirtilmediyse, bu alan girişte yoksayılır ve yalnızca çıkış alanına sahip olur.

Bir MQPUT ya da MQPUT1 çağrısının başarıyla tamamlanmasından sonra, bu alan, bir kuyruğa konduysa, iletiyle birlikte gönderilen MDUID ögesini içerir. Bu, alıkonursa iletiyle birlikte tutulan MDUID değerinin (alıkonan yayınlarla ilgili daha fazla ayrıntı için PMRET tanımına bakın) değeri olur, ancak bu ileti kendilerine gönderilen tüm yayınlarda MDUID ' i geçersiz kılmak için bir değer sağladığından, bu ileti abonelere bir yayın olarak gönderildiğinde, MDUID olarak kullanılmaz. İletinin bağlamı yoksa, alan tümüyle boştur.

Bu, MQGET çağrısına ilişkin bir çıkış alanıdır. Bu alanın uzunluğu LNUID tarafından verilir. Bu alanın ilk değeri 12 boş karakterdir.

### MDVER (10 basamaklı işaretli tamsayı)

Yapı sürüm numarası.

Değer aşağıdakilerden biri olmalıdır:

#### MDVER1

Version-1 ileti tanımlayıcı yapısı.

#### MDVER2

Version-2 ileti tanımlayıcı yapısı.

**Not:** Bir version-2 MQMD (MQMD) kullanıldığında, kuyruk yöneticisi, uygulama iletisi verilerinin başında var olabilecek herhangi bir MQ üstbilgi yapısıyla ilgili ek denetimler gerçekleştirir; ek ayrıntılar için, MQPUT çağrısına ilişkin kullanım notlarına bakın.

Yalnızca yapının daha yeni sürümünde var olan alanlar, alanların açıklamalarında olduğu gibi tanımlanır. Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

#### MDVERC

İleti tanımlayıcı yapısının yürürlükteki sürümü.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MDVER1' dir.

### Başlangıçtaki değerler

Çizelge 710. MQMD ' deki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
MDSID	MDSIDV	' MD---
MDVER	MDVER1	1
MDREP	ROYOK	0
MDMT	MTDGRM	8
MDEXP	EIULIM	-1
MDFB	YOK	0
MDENC	ENNAT	Ortama göre değişir
MDCSI	CSQM	0
MDFMT	FMNONE	Boşluklar
MDPRI	PRQDEF	-1
MDPER	PPEQDEF	2
MDMID	BIRINCI	Boş Değerler
MDCID	CINONE	Boş Değerler



Çizelge 710. MQMD 'deki alanların ilk değerleri (devamı var)

Alan adı	Değişmezin adı	Değişmezin değeri
MDBOC	Yok	0
MDRQ	Yok	Boşluklar
MDRM	Yok	Boşluklar
MDUID	Yok	Boşluklar
MDACC	YOK	Boş Değerler
MDAID	Yok	Boşluklar
MDPAT	ATNCON	0
MDPAN	Yok	Boşluklar
MDPD	Yok	Boşluklar
MDPT	Yok	Boşluklar
MDAOD	Yok	Boşluklar
MDGID	GINONE	Boş Değerler
MDSEQ	Yok	1
MDOFF	Yok	0
MDMFL	MFYOK	0
MDOLN	OUNDF	-1

**Notlar:**

1. - simgesi tek bir boş karakteri temsil eder.

## RPG bildirim

```

D*..1.....2.....3.....4.....5.....6.....7..
D*
D* MQMD Structure
D*
D* Structure identifier
D MDSID          1          4  INZ('MD ')
D* Structure version number
D MDVER          5          8I 0 INZ(1)
D* Options for report messages
D MDREP          9          12I 0 INZ(0)
D* Message type
D MDMT          13         16I 0 INZ(8)
D* Message lifetime
D MDEXP         17         20I 0 INZ(-1)
D* Feedback or reason code
D MDFB          21         24I 0 INZ(0)
D* Numeric encoding of message data
D MDENC         25         28I 0 INZ(273)
D* Character set identifier of messagedata
D MDCSI         29         32I 0 INZ(0)
D* Format name of message data
D MDFMT         33         40  INZ(' ')
D* Message priority
D MDPRI         41         44I 0 INZ(-1)
D* Message persistence
D MDPER         45         48I 0 INZ(2)
D* Message identifier
D MDMID         49         72  INZ(X'00000000000000-
D                                     0000000000000000000000-
D                                     000000000000')

```

```

D* Correlation identifier
D MDCID 73 96 INZ('00000000000000-
D 0000000000000000000000-
D 000000000000')
D* Backout counter
D MBOC 97 100I 0 INZ(0)
D* Name of reply queue
D MDRQ 101 148 INZ
D* Name of reply queue manager
D MDRM 149 196 INZ
D* User identifier
D MDUID 197 208 INZ
D* Accounting token
D MDACC 209 240 INZ('00000000000000-
D 0000000000000000000000-
D 0000000000000000000000-
D 000000')
D* Application data relating to identity
D MDAID 241 272 INZ
D* Type of application that put the message
D MDPAT 273 276I 0 INZ(0)
D* Name of application that put the message
D MDPAN 277 304 INZ
D* Date when message was put
D MDPD 305 312 INZ
D* Time when message was put
D MDPT 313 320 INZ
D* Application data relating to origin
D MDAOD 321 324 INZ
D* Group identifier
D MDGID 325 348 INZ('00000000000000-
D 0000000000000000000000-
D 000000000000')
D* Sequence number of logical message within group
D MDSEQ 349 352I 0 INZ(1)
D* Offset of data in physical message from start of logical message
D MDOFF 353 356I 0 INZ(0)
D* Message flags
D MDMFL 357 360I 0 INZ(0)
D* Length of original message
D MDOLN 361 364I 0 INZ(-1)

```

IBM i

## MQMD (Message descriptor extension) on IBM i

### Genel Bakış

**Amaç:** MQMDE yapısı, bazen uygulama iletileri verilerinden önce ortaya çıkan verileri açıklar. Yapı, version-2 MQMD 'de var olan ancak version-1 MQMD' de bulunmayan MQMD alanlarını içerir.

**Biçim adı:** FMMDE.

**Karakter kümesi ve kodlama:** MQMDE içindeki veriler, C programlama dili için ENNAT tarafından verilen yerel kuyruk yöneticisinin kodlaması ve **CodedCharSetId** kuyruk yöneticisi özneliği tarafından belirtilen karakter kümesinde olmalıdır.

The character set and encoding of the MQMDE must be set into the *MDCSI* and *MDENC* fields in:

- MQMD (MQMDE yapısı iletilerinin başlangıcındaki ise) ya da
- MQMDE yapısından önce gelen üstbilgi yapısı (diğer tüm durumlar).

MQMDE, kuyruk yöneticisinin karakter kümesi ve kodlamasında değilse, MQMDE kabul edilir, ancak onur kabul edilmez; bu, MQMDE 'nin iletileri olarak kabul edilir.

**Kullanım:** Normal uygulamalar bir version-2 MQMD kullanmalı, bu durumda bir MQMDE yapısıyla karşılaşmazlar. Ancak, özelleştirilmiş uygulamalar ve version-1 MQMD kullanmaya devam eden uygulamalar bazı durumlarda bir MQMDE ile karşılaşabilirler. MQMDE yapısı aşağıdaki durumlarda oluşabilir:

- MQPUT ve MQPUT1 çağrılarında belirtilen
- MQGET çağrısının döndürdüğü
- İletim kuyruklarına ilişkin iletilerde

- “MQPUT ve MQPUT1 çağrılarında MQMDE belirtildi” sayfa 1135
- “MQGET çağrısıyla döndürülen MQMDE” sayfa 1135
- “İletim kuyruklarına ilişkin iletilerde MQMDE” sayfa 1136
- “Alanlar” sayfa 1136
- “Başlangıçtaki değerler” sayfa 1138
- “RPG bildirim” sayfa 1138

## MQPUT ve MQPUT1 çağrılarında MQMDE belirtildi

MQPUT ve MQPUT1 çağrılarında, uygulama bir version-1 MQMD sağlıyorsa, uygulama isteğe bağlı olarak bir MQMDE ile ileti verilerini önleyebilir, MQMD ' de *MDFMT* alanını FMMDE olarak ayarlayarak bir MQMDE olduğunu göstermek için bu alanı FMDE olarak ayarlayabilir. Uygulama bir MQMDE sağlamazsa, kuyruk yöneticisi MQMDE içindeki alanlar için varsayılan değerleri devralır. Kuyruk yöneticisinin kullandığı varsayılan değerler, yapı için başlangıç değerleriyle aynıdır; bkz. Çizelge 712 sayfa 1138.

Uygulama bir version-2 MQMD ve örnekleri sağlıyorsa, uygulama ileti verilerini bir MQMDE ile ekler; yapılar, Çizelge 711 sayfa 1135 içinde gösterildiği gibi işlenir.

Çizelge 711. MQPUT ya da MQPUT1 üzerinde MQMDE belirtildiğinde kuyruk yöneticisi işlemi			
MQMD sürümü	version-2 alanlarının değerleri	MQMDE içindeki ilgili alanların değerleri	Kuyruk yöneticisi tarafından alınan işlem
1	-	Geçerli	MQMDE onur
2	Varsayılan	Geçerli	MQMDE onur
2	Varsayılan değil	Geçerli	MQMDE ileti verileri olarak işlem görür
1 ya da 2	Herhangi	Geçerli değil	Arama, uygun bir neden koduyla başarısız olur
1 ya da 2	Herhangi	MQMDE, yanlış karakter kümesi ya da kodlamadır ya da desteklenmeyen bir sürüm	MQMDE ileti verileri olarak işlem görür

Özel bir vaka var. Uygulama bir kesime (yani, MFSEG ya da MFLSEG işareti ayarlıdır) bir ileti yerleştirmek için bir version-2 MQMD kullanıyorsa ve MQMD ' deki biçim adı FMDLH ise, kuyruk yöneticisi bir MQMDE yapısı oluşturur ve bu yapıyı *arasında* MQDLH yapısını ve bunu izleyen verileri ekler. Kuyruk yöneticisinin iletiyle birlikte sakladığı MQMD ' de, version-2 alanları varsayılan değerlerine ayarlanır.

version-2 MQMD 'de var olan, ancak version-1 MQMD' de olmayan bazı alanlar, MQPUT ve MQPUT1 üzerindeki giriş/çıkış alanlarıdır. Ancak, kuyruk yöneticisi, MQPUT ve MQPUT1 çağrılarının çıktısındaki MQMD ' deki eşdeğer alanlarda herhangi bir değer döndürmez; uygulama bu çıkış değerlerini gerektiriyorsa, bu değer bir version-2 MQMD kullanmalıdır.

## MQGET çağrısıyla döndürülen MQMDE

MQGET çağrısında, uygulama bir version-1 MQMD sağlıyorsa, kuyruk yöneticisi örnekleri bir MQMDE ile döndürülür; ancak, MQMDE içindeki alanlardan birinin ya da daha fazlasının varsayılan değeri olmayan bir değer varsa. Kuyruk yöneticisi, MQMD 'nin bir MQMDE olduğunu göstermek için MQMD' deki *MDFMT* alanını FMDE değerine ayarlar.

Uygulama, **BUFFER** parametresinin başlangıcındaki bir MQMDE sağlıyorsa, MQMDE yoksayılır. MQGET çağrısından dönüşte, ileti için MQMDE ile değiştirilir (gerekliyorsa) ya da üzerine uygulama ileti verileri (MQMDE gerekmiyorsa) üzerine yazılır.

MQGET çağrısı tarafından bir MQMDE döndürülürse, MQMDE ' deki veriler tipik olarak kuyruk yöneticisinin karakter kümesi ve kodlamasıdır. Ancak, MQMDE, aşağıdaki durumlarda başka bir karakter takımı ve kodlamada da olabilir:

- MQMDE, MQPUT ya da MQPUT1 çağrısında veri olarak değerlendirildi (bu duruma neden olacak durumlar için [Çizelge 711 sayfa 1135](#) konusuna bakın).
- İleti, bir TCP bağlantısıyla bağlantılı uzak bir kuyruk yöneticisinden alındı ve alıcı ileti kanalı aracısı (MCA) doğru olarak ayarlanmadı (ek bilgi için [IBM MQ for IBM i nesnelere güvenliği](#) konusuna bakın).

## İletim kuyruklarına ilişkin iletilerde MQMDE

İletim kuyruklarındaki iletiler, bir version-1 MQMD içinde bulunan MQXQH yapısıyla önlenir. MQXQH yapısı ve uygulama iletileri arasında konumlandırılmış bir MQMDE de var olabilir; ancak, yalnızca MQMDE içindeki bir ya da daha çok alanın varsayılan değeri olmayan bir değer varsa geçerli olur.

MQXQH yapısı ile uygulama iletileri arasında diğer IBM MQ üstbilgi yapıları da olabilir. Örneğin, MQDLH adlı ölü-mektup üstbilgisi varsa ve ileti bir bölüm değilse, sıralama düzeni şöyledir:

- MQXQH (bir version-1 MQMD içeren)
- MQMDE
- MQDLH
- Uygulama iletileri

## Alanlar

MQMDE yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

### MECSI (10 basamaklı işaretli tamsayı)

MQMDE ' yi izleyen verilerin karakter kümesi tanıtıcısı.

Bu, MQMDE yapısını izleyen verilerin karakter kümesi tanıtıcısını belirtir; bu, MQMDE yapısındaki karakter verilerine uygulanmaz.

MQPUT ya da MQPUT1 çağrısında, uygulama bu alanı veri için uygun değere ayarlamalıdır. Kuyruk yöneticisi bu alanın geçerli olup olmadığını denetmiyor. Aşağıdaki özel değer kullanılabilir:

### CİNT

Bu yapının karakter kümesi tanıtıcısını devralır.

Character data in the data *takip edilen* this structure is in the same character set as this structure.

Kuyruk yöneticisi, bu değeri, iletilde belirtilen yapının gerçek karakter kümesi tanıtıcısına göre değiştirir. Hata oluşmaması durumunda, MQGET çağrısıyla CSINHT değeri döndürülmez.

MQMD ' deki MDPAT alanının değeri ATBRKR ise, CSINHT kullanılamaz.

Bu alanın ilk değeri CSUNDF 'dir.

### MEENC (10 basamaklı işaretli tamsayı)

MEENC (10 basamaklı işaretli tamsayı)

Bu, MQMDE yapısını izleyen verilerin sayısal kodlamasını belirtir; bu, MQMDE yapısındaki sayısal veriler için geçerli değildir.

MQPUT ya da MQPUT1 çağrısında, uygulama bu alanı veri için uygun değere ayarlamalıdır. Kuyruk yöneticisi, alanın geçerli olup olmadığını denetmez. Veri kodlamalarıyla ilgili daha fazla bilgi için [“MQMD \(Message descriptor\) on IBM i” sayfa 1092](#) içinde açıklanan MDENC alanına bakın.

Bu alanın ilk değeri ENNAT ' tır.

### MEFLG (10 basamaklı işaretli tamsayı)

Genel bayraklar.

Aşağıdaki işaret belirlenebilir:

**MEFNON**

Bayrak yok.

Bu alanın ilk deęeri MEFON deęeridir.

**MEFMT (8 baytlık karakter dizilimi)**

MQMDE ' yi izleyen verilerin adını biçimlendirin.

Bu, MQMDE yapısını izleyen verilerin biçim adını belirtir.

MQPUT ya da MQPUT1 çağrısında, uygulama bu alanı veri için uygun deęere ayarlamalıdır. Kuyruk yöneticisi bu alanın geçerli olup olmadığını denetmiyor. Biçim adlarıyla ilgili daha fazla bilgi için [“MQMD \(Message descriptor\) on IBM i” sayfa 1092](#) içinde açıklanan *MDFMT* alanına bakın.

Bu alanın ilk deęeri FMNONE ' dir.

**MEGID (24 baytlık bit dizesi)**

Grup tanıtıcısı.

[“MQMD \(Message descriptor\) on IBM i” sayfa 1092](#) içinde açıklanan *MDGID* alanına bakın. Bu alanın ilk deęeri GINONE 'tır.

**MELEN (10 basamaklı işaretli tamsayı)**

MQMDE yapısının uzunluğu.

Aşağıdaki deęer tanımlıdır:

**MELEN2**

version-2 ileti tanımlayıcı uzantısı yapısının uzunluğu.

Bu alanın ilk deęeri MELEN2' dir.

**MEFL (10 basamaklı işaretli tamsayı)**

Mesaj bayrakları.

[“MQMD \(Message descriptor\) on IBM i” sayfa 1092](#) içinde açıklanan *MDMFL* alanına bakın. Bu alanın ilk deęeri MFNONE 'dir.

**MEOFF (10 basamaklı işaretli tamsayı)**

Mantıksal iletinin başlangıcındaki fiziksel iletelerde verilerin görelî konumu.

[“MQMD \(Message descriptor\) on IBM i” sayfa 1092](#) içinde açıklanan *MDOFF* alanına bakın. Bu alanın ilk deęeri 0 'tır.

**MEOLN (10 basamaklı işaretli tamsayı)**

Özgün iletinin uzunluğu.

[“MQMD \(Message descriptor\) on IBM i” sayfa 1092](#) içinde açıklanan *MDOLN* alanına bakın. Bu alanın ilk deęeri OLUNDF ' dir.

**MESEQ (10 basamaklı işaretli tamsayı)**

Grup içindeki mantıksal iletinin sıra numarası.

[“MQMD \(Message descriptor\) on IBM i” sayfa 1092](#) içinde açıklanan *MDSEQ* alanına bakın. Bu alanın ilk deęeri 1 'dir.

**MESID (4 baytlık karakter dizilimi)**

Yapı tanıtıcısı.

Deęer şu olmalıdır:

**MESIDV**

İleti tanımlayıcı uzantısı yapısına ilişkin tanıtıcı.

Bu alanın ilk deęeri MEMIDV 'dir.

## Mever (10 basamaklı işaretli tamsayı)

Yapı sürüm numarası.

Değer şu olmalıdır:

### MEVER2

Version-2 ileti tanımlayıcı uzantı yapısı.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

### EVERC

İleti tanımlayıcı uzantısı yapısının yürürlükteki sürümü.

Bu alanın ilk değeri MEVER2' dir.

## Başlangıçtaki değerler

Çizelge 712. MQMDE içindeki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
MESID	MESIDV	'MDE↵'
MEVER	MEVER2	2
MELEN	MELEN2	72
MEENC	ENNAT	Ortama göre değişir
MECSI	CSUNDF	0
MEFMT	FMNONE	Boşluklar
MEFLG	MEFNON	0
MEGID	GINONE	Boş Değerler
MESEQ	Yok	1
MEOFF	Yok	0
MEMFL	MFYOK	0
MEOLN	OUNDF	-1

**Notlar:**

- ↵ simgesi tek bir boş karakteri temsil eder.

## RPG bildirimi

```
D*.1.....2.....3.....4.....5.....6.....7..
D*
D* MQMDE Structure
D*
D* Structure identifier
D MESID          1      4   INZ('MDE ')
D* Structure version number
D MEVER          5      8I 0 INZ(2)
D* Length of MQMDE structure
D MELEN          9     12I 0 INZ(72)
D* Numeric encoding of data that followsMQMDE
D MEENC         13     16I 0 INZ(273)
D* Character-set identifier of data thatfollows MQMDE
D MECSI         17     20I 0 INZ(0)
D* Format name of data that followsMQMDE
D MEFMT         21     28   INZ('      ')
D* General flags
D MEFLG         29     32I 0 INZ(0)
D* Group identifier
```

```

D MEGID          33      56      INZ(X'00000000000000-
D              000000000000000000-
D              000000000000')
D* Sequence number of logical message within group
D MESEQ          57      60I 0 INZ(1)
D* Offset of data in physical message from start of logical message
D MEOFF          61      64I 0 INZ(0)
D* Message flags
D MEMFL          65      68I 0 INZ(0)
D* Length of original message
D MEOLN          69      72I 0 INZ(-1)

```

IBM i

## MQMHBO (Message handle to buffer options) on IBM i

İleti tanıtıcılarını arabellek seçeneklerine tanımlayan yapı

### Genel Bakış

**Amaç:** MQMHBO yapısı, uygulamaların, ileti tanıtıcılarından arabelleklerin nasıl üretildiğini denetleyen seçenekleri belirlemesine olanak sağlar. Yapı, MQMHBUF çağrısına ilişkin bir giriş değişikidir.

**Karakter kümesi ve kodlama:** MQMHBO içindeki veriler, uygulamanın uygulama ve kodlamasının (ENNAT) karakter kümesinde yer almalıdır.

- [“Alanlar” sayfa 1139](#)
- [“Başlangıçtaki değerler” sayfa 1140](#)
- [“RPG bildirim” sayfa 1140](#)

### Alanlar

MQMHBO yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

#### MBOPT (10 basamaklı işaretli tamsayı)

İleti tanıtıcısı-arabellek seçenekleri yapısı-MBOPT alanı.

Bu seçenekler MQMHBUF ' un işlemini denetler.

Aşağıdaki seçeneği belirlemelisiniz:

##### MPRRF

Özellikleri bir ileti tanıtıcısından arabelleğe dönüştürürken, bunları MQRFH2 biçimine dönüştürün.

İsteğe bağlı olarak, aşağıdaki seçeneği de belirleyebilirsiniz. birden çok seçenek belirtmek için, değerleri bir araya ekleyin (aynı değışımezi bir kereden fazla eklemeyin) ya da bit ya da işlemi kullanarak değerleri birleştirin (programlama dili, bit işlemleri destekliyse).

##### MBDLPR

Arabelleğe eklenen özellikler ileti tanıtıcısından silinir. Arama başarısız olursa özellikler silinmez.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MBPRRF 'dir.

#### MBSID (10 basamaklı işaretli tamsayı)

İleti tanıtıcısı-arabellek seçenekleri yapısı-MBSID alanı.

Bu, yapı tanıtıcısıdır. Değer şu olmalıdır:

##### MBSİS

Arabellek seçenekleri yapısıyla ileti tanıtıcısı için tanıtıcı.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri isMBSIDV.

#### MBVER (10 basamaklı işaretli tamsayı)

Bu, yapı sürüm numarasıdır. Değer şu olmalıdır:

##### MBVER1

Arabellek seçenekleri yapısıyla ileti tanıtıcısı için sürüm numarası.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

### MBVERC

İleti tutamacının arabellek seçenekleri yapısına ilişkin yürürlükteki sürümü.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MBVER1' dir.

## Başlangıçtaki değerler

Çizelge 713. MQMHBO 'daki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
MVSID	MBSİS	'MHBO'
MBVER	MBVER1	1
MBOPT	MPPRRF	

### Notlar:

1. Boş değerli dizgi ya da boşluk, boş bir karakteri belirtir.

## RPG bildirimi

```
D* MQMHBO Structure
D*
D*
D* Structure identifier
D MBSID 1 4 INZ('MHBO')
D*
D* Structure version number
D MBVER 5 8I 0 INZ(1)
D*
D* Options that control the action of MQMHBUF
D MBOPT 9 12I 0 INZ(1)
```

IBM i

## IBM i üzerinde MQOD (Nesne tanımlayıcı)

MQOD yapısı, ada göre bir nesne belirtmek için kullanılır.

## Genel Bakış

**Amaç:** Aşağıdaki nesne tipleri geçerlidir:

- Kuyruk ya da dağıtım listesi
- Ad listesi
- Süreç tanımlaması
- Kuyruk yöneticisi
- Konu

Yapı, MQOPEN ve MQPUT1 çağrılarında bir giriş/çıkış değiştirgesidir.

**Sürüm:** MQOD ' un yürürlükteki sürümü ODVER4. Yapının yalnızca daha yeni sürümlerinde bulunan alanlar, izleyen açıklamalarda bu şekilde tanımlanır.

Sağlanan COPY dosyası, ortam tarafından desteklenen en son MQOD sürümünü içeriyor; ancak, ODVER alanının ilk değeri ODVER1olarak ayarlanmış. version-1 yapısında var olmayan alanları kullanmak için uygulamanın ODVER alanını gerekli sürümün sürüm numarasına ayarlaması gerekir.

Bir dağıtım listesini açmak için ODVER , ODVER2 ya da daha büyük olmalıdır.

**Karakter kümesi ve kodlama:** MQOD içindeki veriler, CodedCharSetId kuyruk yöneticisi özniteliği ve ENNAT tarafından verilen yerel kuyruk yöneticisinin kodlaması tarafından belirtilen karakter kümesinde



olmalıdır. Ancak, uygulama bir IBM MQ istemcisi olarak çalışıyorsa, yapı istemcinin karakter kümesinde ve kodlamasında olmalıdır.

- “Alanlar” sayfa 1141
- “Başlangıçtaki değerler” sayfa 1147
- “RPG bildirim” sayfa 1148

## Alanlar

MQOD yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** açıklanır:

### ODASI (40 baytlık bit dizgisi)

Diğer güvenlik tanıtıcısı.

Bu, uygun yetki denetimlerinin gerçekleştirilmesine izin vermek için *ODAU* ile yetkilendirme hizmetine iletilen bir güvenlik tanıtıcısıdır. *ODASI* yalnızca aşağıdaki durumda kullanılır:

- MQOPEN çağrısında OOALTU belirtildi ya da
  - MQPUT1 çağrısında PMALTU belirtildi,
- ve *ODAU* alanı, alanın ilk boş karakterine ya da sonuna kadar tamamen boş değildir.

*ODASI* alanı aşağıdaki yapıya sahiptir:

- İlk bayt, izleyen önemli verilerin uzunluğunu içeren ikili bir tamsayıdır; değer, uzunluk byte 'ının kendisini dışlar. Güvenlik tanıtıcısı yoksa, uzunluk sıfır olur.
- İkinci bayt, var olan güvenlik tanıtıcısının tipini gösterir; aşağıdaki değerler kullanılabilir:

#### SITWNT

Windows güvenlik tanıtıcısı.

#### DURUM

Güvenlik tanıtıcısı yok.

- Birinci baytın tanımladığı uzunluğa kadar olan üçüncü ve sonraki baytlar güvenlik tanıtıcısının kendisini içerir.
- Alandaki kalan bayt sayısı ikili sıfır olarak ayarlanır.

Aşağıdaki özel değer kullanılabilir:

#### SINONE.

Güvenlik tanıtıcısı belirtilmedi.

Değer, alanın uzunluğu için ikili sıfırdır.

Bu bir giriş alanıdır. Bu alanın uzunluğu LNSCID tarafından verilir. Bu alanın ilk değeri SINONE. *ODVER* *ODVER3* değerinden küçükse bu alan yoksayılr.

### ODAU (12 baytlık karakter dizgisi)

Diğer kullanıcı kimliği.

MQOPEN çağrısı için OOALTU ya da MQPUT1 çağrısı için PMALTU belirtilirse, bu alan, uygulamanın çalışmakta olduğu kullanıcı kimliği yerine, açma yetkisini denetlemek için kullanılacak diğer bir kullanıcı kimliğini içerir. Ancak, bazı denetimler yürürlükteki kullanıcı kimliğiyle (örneğin, bağlam denetimleri) gerçekleştirilmeye devam eder.

OOALTU ve PMALTU belirtilmezse ve bu alan, ilk boş karakter ya da alanın sonuna kadar tamamen boşsa, açma işlemi yalnızca belirlenen seçeneklerle bu nesneyi açmak için kullanıcı yetkisi gerekmediğinde başarılı olabilir.

OOALTU ya da PMALTU belirtilmezse, bu alan yoksayılr.

Bu bir giriş alanıdır. Bu alanın uzunluğu LNUID tarafından verilir. Bu alanın ilk değeri 12 boş karakterdir.

### **ODDN (48 baytlık karakter dizgisi)**

Dinamik kuyruk adı.

MQOPEN çağrısıyla yaratılacak dinamik kuyruğun adıdır. Bu yalnızca *ODON* bir model kuyruğunun adını belirttiğinde geçerlidir; diğer tüm durumlarda *ODDN* yoksayılr.

Adda geçerli olan karakterler, yıldız işaretinin de geçerli olması dışında, *ODON* için geçerli olan karakterlerle aynıdır. *ODON* bir model kuyruğunun adıysa, boşluk (ya da ilk boş karakterden önce yalnızca boşlukların gösterildiği bir ad) geçerli değildir.

Addaki son boş olmayan karakter bir yıldız işaretiyse (\*), kuyruk yöneticisi yıldız işaretini, kuyruk için oluşturulan adın yerel kuyruk yöneticisinde benzersiz olmasını garanti eden bir karakter dizgisiyle değiştirir. Bunun için yeterli sayıda karaktere izin vermek için, yıldız işareti yalnızca 1-33 arasındaki konumlarda geçerlidir. Yıldız işaretinden sonra boşluk ya da boş karakter dışında bir karakter olmamalıdır.

Yıldız işaretinin ilk karakter konumunda oluşması için geçerlidir; bu durumda ad yalnızca kuyruk yöneticisi tarafından oluşturulan karakterlerden oluşur.

Bu bir giriş alanıdır. Bu alanın uzunluğu LNQN tarafından verilir. Bu alanın ilk değeri 'AMQ.\*', boşluklarla doldurulur.

### **ODIDC (10 basamaklı işaretli tamsayı)**

Açılmayan kuyrukların sayısı.

Bu, dağıtım listesindeki, başarıyla açılmayan kuyrukların sayısıdır. Varsa, dağıtım listesinde olmayan tek bir kuyruk açılırken de bu alan ayarlanır.

**Not:** Varsa, bu alan yalnızca MQOPEN ya da MQPUT1 çağrısındaki **CMPCOD** değiştirgesi CCOK ya da CCWARN ise ayarlanır; **CMPCOD** değiştirgesi CCFail ise ayarlanmaz.

Bu bir çıkış alanıdır. Bu alanın ilk değeri 0'dır. *ODVER* *ODVER2* değerinden küçükse bu alan yoksayılr.

### **ODKDC (10 basamaklı işaretli tamsayı)**

Başarıyla açılan yerel kuyruk sayısı.

Bu, dağıtım listesindeki yerel kuyruklara çözülecek ve başarıyla açılan kuyrukların sayısıdır. Bu sayı, uzak kuyruklara çözülecek kuyrukları içermez (iletiyi saklamak için başlangıçta yerel bir iletim kuyruğu kullanılsa bile). Varsa, dağıtım listesinde olmayan tek bir kuyruk açılırken de bu alan ayarlanır.

Bu bir çıkış alanıdır. Bu alanın ilk değeri 0'dır. *ODVER* *ODVER2* değerinden küçükse bu alan yoksayılr.

### **ODMN (48 baytlık karakter dizgisi)**

Nesne kuyruğu yöneticisi adı.

Bu, *ODON* nesnesinin tanımlandığı kuyruk yöneticisinin adıdır. Adda geçerli olan karakterler, *ODON* ile aynıdır (daha önce bakınız). İlk boş değere kadar ya da alanın sonuna kadar tümüyle boş bırakılan bir ad, uygulamanın bağlı olduğu kuyruk yöneticisini (yerel kuyruk yöneticisi) gösterir.

Aşağıdaki noktalar, belirtilen nesne tipleri için geçerlidir:

- *ODOT* *OTTOP*, *OTNLST*, *OTPRO* ya da *OTQM* ise, *ODMN* boş olmalı ya da yerel kuyruk yöneticisinin adı olmalıdır.
- *ODON* bir model kuyruğunun adıysa, kuyruk yöneticisi model kuyruğunun özniteliklerini içeren dinamik bir kuyruk yaratır ve *ODMN* alanında kuyruğun yaratıldığı kuyruk yöneticisinin adını döndürür; bu, yerel kuyruk yöneticisinin adıdır. Model kuyruğu yalnızca MQOPEN çağrısında belirtilebilir; MQPUT1 çağrısında model kuyruğu geçerli değil.
- *ODON* bir küme kuyruğunun adıysa ve *ODMN* boşsa, MQOPEN çağrısının döndürdüğü kuyruk tanıtıcısı kullanılarak gönderilen iletilerin gerçek hedefi, kuyruk yöneticisi (ya da kuruluysa, küme iş yükü çıkışı) tarafından aşağıdaki gibi seçilir:

- OOBUDO belirtilirse, kuyruk yöneticisi MÇOPEN çağrısının işlenmesi sırasında küme kuyruğunun bir eşgörünümünü seçer ve bu kuyruk tanıtıcısı kullanılarak konan tüm iletiler o yönetim ortamına gönderilir.
- OOBNDN belirtilirse, kuyruk yöneticisi bu kuyruk tanıtıcısını kullanan her ardışık MÇPUT çağrısı için hedef kuyruğun farklı bir eşgörünümünü seçebilir (kümede farklı bir kuyruk yöneticisinde bulunur).

Uygulamanın bir küme kuyruğunun *belirli* bir eşgörünümüne (yani, kümedeki belirli bir kuyruk yöneticisinde bulunan bir kuyruk eşgörünümüne) ileti göndermesi gerekiyorsa, uygulama *ODMN* alanında o kuyruk yöneticisinin adını belirtmelidir. Bu, yerel kuyruk yöneticisini iletiyi belirtilen hedef kuyruk yöneticisine göndermeye zorlar.

- Açılmakta olan nesne bir dağıtım listesiye (yani, *ODREC* sıfırdan büyükse), *ODMN* boş değerli ya da boş değerli bir dizgi olmalıdır. Bu koşul karşılanmazsa, çağrı RC2153neden koduyla başarısız olur.

Bu, *ODON* bir model kuyruğunun adı ve diğer tüm durumlarda yalnızca giriş alanı olduğunda MÇOPEN çağrısına ilişkin bir giriş/çıkış alanıdır. Bu alanın uzunluğu LNQMN tarafından verilir. Bu alanın ilk değeri 48 boş karakterdir.

### **ODON (48 baytlık karakter dizgisi)**

Nesne adı.

Bu, *ODMN* ile tanımlanan kuyruk yöneticisinde tanımlandığı şekilde, nesnenin yerel adıdır. Ad aşağıdaki karakterleri içerebilir:

- Büyük harf alfabetik karakterler (A-Z)
- Küçük harf alfabetik karakterler (a-z)
- Sayısal basamaklar (0-9)
- Nokta (.), eğik çizgi (/), altçizgi (\_), yüzde (%)

Ad başta ya da gömülü boşluklar içermemelidir, ancak sonda boşluklar içerebilir. Addaki önemli verilerin sonunu göstermek için boş karakter kullanılabilir; boş değer ve ardından gelen karakterler boşluk olarak değerlendirilir. Belirtilen ortamlarda aşağıdaki kısıtlamalar geçerlidir:

- EBCDIC Katakana kullanan sistemlerde küçük harfli karakterler kullanılamaz.
- IBM üzerinde, komutlarda belirtildiğinde küçük harf, eğik çizgi ya da yüzde içeren adlar tırnak içine alınmalıdır. Bu tırnak işaretleri, yapılarda alan olarak ya da çağrılarda parametre olarak oluşan adlar için belirlenmemelidir.

Aşağıdaki noktalar, belirtilen nesne tipleri için geçerlidir:

- *ODON* bir model kuyruğunun adıysa, kuyruk yöneticisi model kuyruğunun özniteliklerini içeren dinamik bir kuyruk yaratır ve *ODON* alanında yaratılan kuyruğun adını döndürür. Model kuyruğu yalnızca MÇOPEN çağrısında belirtilebilir; MÇPUT1 çağrısında model kuyruğu geçerli değil.
- Açılmakta olan nesne bir dağıtım listesiye (yani, *ODREC* var ve sıfırdan büyük), *ODON* boş değerli ya da boş değerli olmalıdır. Bu koşul karşılanmazsa, çağrı RC2152neden koduyla başarısız olur.
- *ODOT* OTQM ise, özel kurallar geçerlidir; bu durumda ad, ilk boş karakter ya da alanın sonuna kadar tamamen boş olmalıdır.
- *ODON*, TARGTYPE (TOPIC) ile bir diğer ad kuyruğunun adıysa, diğer ad kuyruklarının kullanımı için olağan olduğu gibi, önce adı belirtilen diğer ad kuyruğunda bir güvenlik denetimi yapılır. Bu güvenlik denetimi başarılı olursa, bu MÇOPEN çağrısı devam eder ve denetim konusu nesnesiyle ilgili bir güvenlik denetimi de içinde olmak üzere, bir OTTOP MÇOPEN çağrısı gibi davranır.

Bu, *ODON* bir model kuyruğunun adı ve diğer tüm durumlarda yalnızca giriş alanı olduğunda MÇOPEN çağrısına ilişkin bir giriş/çıkış alanıdır. Bu alanın uzunluğu LNQN tarafından verilir. Bu alanın ilk değeri 48 boş karakterdir.

Tam konu adı iki farklı alandan oluşturulabilir: *ODON* ve *ODOS*. Bu iki alanın nasıl kullanıldığına ilişkin ayrıntılar için [Konu dizgilerini birleştirme](#) başlıklı konuya bakın.

## **ODORO (10 basamaklı işaretli tamsayı)**

MQOD ' un başlangıcından ilk nesne kaydının görelî konumu.

Bu, MQOD yapısının başlangıcından itibaren ilk MQOR nesne kaydının bayt cinsinden görelî konumudur. Görelî konum pozitif ya da negatif olabilir. *ODORO* yalnızca bir dağıtım listesi açıldığında kullanılır. *ODREC* sıfırda alan yoksayılır.

Dağıtım listesi açılırken, dağıtım listesindeki hedef kuyrukların adlarını belirtmek için bir ya da daha fazla MQOR nesne kaydı dizisi sağlanmalıdır. Bu iki yoldan biriyle yapılabilir:

- *ODORO* görelî konum alanını kullanarak

Bu durumda, uygulama MQOD içeren kendi yapısını ve ardından MQOR kayıtları dizisini (gerektiği kadar dizi ögesi olan) bildirmeli ve *ODORO* ögesini MQOD ' un başlangıcından itibaren dizideki ilk ögenin görelî konumuna ayarlamalıdır. Bu kaymanın doğru olduğundan emin olmak için dikkatli olunmalıdır.

- *ODORP* gösterge alanını kullanarak

Bu durumda, uygulama MQOR yapılarının dizisini MQOD yapısından ayrı olarak bildirebilir ve *ODORP* değerini dizinin adresine ayarlayabilir.

Hangi teknik seçilirse seçilsin, *ODORO* ve *ODORP* yöntemlerinden biri kullanılmalıdır; her ikisi de sıfırda ya da her ikisi de sıfır değilsen, çağrı RC2155 neden koduyla başarısız olur.

Bu bir giriş alanıdır. Bu alanın ilk değeri 0 'dır. *ODVER* *ODVER2* değerinden küçükse bu alan yoksayılır.

## **ODORP (gösterge)**

İlk nesne kaydının adresi.

Bu, ilk MQOR nesne kaydının adresidir. *ODORP* yalnızca bir dağıtım listesi açıldığında kullanılır. *ODREC* sıfırda alan yoksayılır.

Bu bir giriş alanıdır. Bu alanın ilk değeri boş işaretçidir. Nesne kayıtlarını belirlemek için *ODORP* ya da *ODORO* kullanılabilir, ancak her ikisini birden belirleyemez; ayrıntılar için daha önce *ODORO* alanının tanımına bakın. *ODORP* kullanılmazsa, boş değerli göstergeye ya da boş değerli byte 'lara ayarlanmalıdır. *ODVER* *ODVER2* değerinden küçükse bu alan yoksayılır.

## **ODOS (MQCHARV)**

ODOS, kullanılacak uzun nesne adını belirtir.

Bu alana yalnızca belirli *ODOT* değerleri için başvurulur. Bu alanın kullanıldığını gösteren değerlerin ayrıntıları için [ODOT](#) açıklamasına bakın.

*ODOS* yanlış belirtilirse, *MQCHARV* yapısının nasıl kullanılacağına ilişkin açıklamaya göre ya da uzunluk üst sınırını aşarsa, çağrı RC2441 neden koduyla başarısız olur.

Bu bir giriş alanıdır. Bu yapıdaki alanların başlangıç değerleri, *MQCHARV* yapısındaki değerlerle aynıdır.

Tam konu adı iki farklı alandan oluşturulabilir: *ODON* ve *ODOS*. Bu iki alanın nasıl kullanıldığına ilişkin ayrıntılar için [Konu dizgilerini birleştirme](#) başlıklı konuya bakın. *ODVER* *ODVER4* değerinden küçükse bu alan yoksayılır.

## **ODOT (10 basamaklı işaretli tamsayı)**

Nesne tipi.

*ODON* içinde adlandırılmakta olan nesnenin tipi. Olası değerler şunlardır:

### **OTQ**

Sıraya girin. Nesnenin adı *ODON* içinde bulunur.

### **OTNLST**

-Namelist. Nesnenin adı *ODON* içinde bulunur.

### **OTPRO**

Süreç tanımlaması. Nesnenin adı *ODON* içinde bulunur.

## **OTQM (Kalite Yönetimi)**

Kuyruk yöneticisi. Nesnenin adı *ODON* içinde bulunur.

## **OTTOP**

Konu. Tam konu adı iki farklı alandan oluşturulabilir: *ODON* ve *ODOS*.

Bu iki alanın nasıl kullanıldığına ilişkin ayrıntılar için [Konu dizgilerini birleştirme](#) başlıklı konuya bakın.

*ODON* alanıyla tanıtılan nesne bulunamazsa, *ODOS* içinde bir dizgi belirtilmiş olsa bile, çağrı RC2425 neden koduyla başarısız olur.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri OTQ 'dur.

## **ODREC (10 basamaklı işaretli tamsayı)**

Var olan nesne kayıtlarının sayısı.

Bu, uygulama tarafından sağlanan MQOR nesne kayıtlarının sayısıdır. Bu sayı sıfırdan büyükse, bir dağıtım listesinin açılmakta olduğunu ve *ODREC* ' un listedeki hedef kuyrukların sayısı olduğunu gösterir. Bir dağıtım listesinin tek bir hedef içermesi geçerlidir.

*ODREC* değeri sıfırdan küçük olmamalıdır ve sıfırdan büyükse *ODOT* OTQ olmalıdır; bu koşullar karşılanmazsa çağrı RC2154 neden koduyla başarısız olur.

Bu bir giriş alanıdır. Bu alanın ilk değeri 0 'dır. *ODVER* ODVER2 değerinden küçükse bu alan yoksayılr.

## **ODRMN (48 baytlık karakter dizgisi)**

Çözülmüş kuyruk yöneticisi adı.

Bu, ad çözme işlemi yerel kuyruk yöneticisi tarafından gerçekleştirildikten sonra hedef kuyruk yöneticisinin adıdır. Döndürülen ad, *ODRQN* ile tanıtılan kuyruğun iyesi olan kuyruk yöneticisinin adıdır. *ODRMN* , yerel kuyruk yöneticisinin adı olabilir.

*ODRQN* , yerel kuyruk yöneticisinin ait olduğu kuyruk paylaşım grubunun sahip olduğu paylaşılan bir kuyruksa, *ODRMN* , kuyruk paylaşım grubunun adıdır. Kuyruk başka bir kuyruk paylaşım grubuna aitse, *ODRQN* kuyruk paylaşım grubunun adı ya da kuyruk paylaşım grubunun üyesi olan bir kuyruk yöneticisinin adı olabilir (döndürülen değer türü, yerel kuyruk yöneticisinde var olan kuyruk tanımlamalarına göre belirlenir).

Boş olmayan bir değer, nesne göz atma, giriş ya da çıkış (ya da herhangi bir birleşim) için açılmış tek bir kuyruksa döndürülür. Açılan nesne aşağıdakilerden biriye, *ODRMN* boş olarak ayarlanır:

- Kuyruk değil
- Bir kuyruk, ancak göz atma, giriş ya da çıkış için açılmadı
- OOBNDN değeri belirlenmiş bir küme kuyruğu (ya da **DefBind** kuyruk özneliği BNDNOT değerine sahip olduğunda OOBNDQ ile)
- Dağıtım listesi

Bu bir çıkış alanıdır. Bu alanın uzunluğu LNQN tarafından verilir. Bu alanın ilk değeri, C içindeki boş dizgi ve diğer programlama dillerindeki 48 boş karakterdir. *ODVER* ODVER3 değerinden küçükse bu alan yoksayılr.

## **ODRO (MQCHARV)**

ODRO, kuyruk yöneticisi *ODON* içinde sağlanan adı çözdükten sonra uzun nesne adıdır.

Bu alan yalnızca, bir konu nesnesine gönderme yapan belirli nesne, konu ve kuyruk diğer adları tipleri için döndürülür.

Uzun nesne adı *ODOS* içinde sağlanırsa ve *ODON* içinde hiçbir şey sağlanmazsa, bu alanda döndürülen değer, *ODOS* içinde sağlananla aynıdır.

Bu alan atlanırsa ( *ODRO.VSBufSize* sıfır), *ODRO* döndürülmez, ancak uzunluk *ODRO.VSLength*. Uzunluk tam *ODRO* değerinden kısaysa, kesilir ve sağlanan uzunluğa sığabilecek en sağdaki karakterlerin sayısını döndürür.

*ODRO* yanlış belirtilirse, *MQCHARV* yapısının nasıl kullanılacağına ilişkin açıklamaya göre ya da uzunluk üst sınırını aşarsa, çağrı RC2520neden koduyla başarısız olur. *ODVER* *ODVER4* değerinden küçükse bu alan yoksayıdır.

### **ODRQN (48 baytlık karakter dizilimi)**

Çözülmüş kuyruk adı.

Ad çözme işlemi yerel kuyruk yöneticisi tarafından gerçekleştirildikten sonra hedef kuyruğun adıdır. Döndürülen ad, *ODRMN* ile tanımlanan kuyruk yöneticisinde bulunan bir kuyruğun adıdır.

Boş olmayan bir değer, nesne göz atma, giriş ya da çıkış (ya da herhangi bir birleşim) için açılmış tek bir kuyruksa döndürülür. Açılan nesne aşağıdakilerden biriye, *ODRQN* boş olarak ayarlanır:

- Kuyruk değil
- Bir kuyruk, ancak göz atma, giriş ya da çıkış için açılmadı
- Dağıtım listesi
- Bir konu nesnesine gönderme yapan bir diğer ad kuyruğu (bunun yerine "ODRO (MQCHARV)" sayfa 1145 konusuna bakın)

Bu bir çıkış alanıdır. Bu alanın uzunluğu LNQN tarafından verilir. Bu alanın ilk değeri, C içindeki boş dizgi ve diğer programlama dillerindeki 48 boş karakterdir. *ODVER* *ODVER3* değerinden küçükse bu alan yoksayıdır.

### **ODRRO (10 basamaklı işaretli tamsayı)**

MQOD 'nin başlangıcından ilk yanıt kaydının görelî konumu.

Bu, MQOD yapısının başlangıcından itibaren ilk MQRR yanıt kaydının bayt cinsinden görelî konumudur. Görelî konum pozitif ya da negatif olabilir. *ODRRO* yalnızca bir dağıtım listesi açıldığında kullanılır. *ODREC* sıfırda alan yoksayıdır.

Bir dağıtım listesi açıldığında, açılmayan kuyrukları (MQRR 'deki *RRCC* alanı) ve her başarısızlığın nedenini (MQRR 'deki *RRREA* alanı) saptamak için bir ya da daha çok MQRR yanıt kaydı dizisi sağlanabilir. Veriler, yanıt kayıtları dizisinde, nesne kayıtları dizisindeki kuyruk adlarıyla aynı sırayla döndürülür. Kuyruk yöneticisi yanıt kayıtlarını yalnızca çağrı sonucu karışık olduğunda (yani, bazı kuyruklar başarılı bir şekilde açıldığında diğerleri başarısız olduğunda ya da tümü farklı nedenlerle başarısız olduğunda) ayarlar; çağrıdaki RC2136 neden kodu bu durumu gösterir. Aynı neden kodu tüm kuyruklar için de geçerliyse, MQOPEN ya da MQPUT1 çağrısının **REASON** değiştirgesinde bu neden döndürülür ve yanıt kayıtları ayarlanmaz. Yanıt kayıtları isteğe bağlıdır, ancak bunlar sağlarsa *ODREC* olmalıdır.

Yanıt kayıtları, *ODRRO* içinde bir görelî konum belirtilerek ya da *ODRRP* içinde bir adres belirtilerek nesne kayıtlarıyla aynı şekilde sağlanabilir; Bunun nasıl yapılacağına ilişkin ayrıntılar için önceki *ODORO* açıklamasına bakın. Ancak, en çok *ODRRO* ve *ODRRP* kullanılabilir; her ikisi de sıfır değilse, çağrı RC2156 neden koduyla başarısız olur.

MQPUT1 çağrısına ilişkin bu yanıt kayıtları, ileti dağıtım listesindeki kuyruklara gönderildiğinde ortaya çıkan hatalarla ilgili bilgileri ve kuyruklar açıldığında oluşan hataları döndürmek için kullanılır. Bir kuyruk için koyma işlemindeki tamamlanma kodu ve neden kodu, yalnızca kuyruktaki tamamlanma kodu CCOK ya da CCWARN ise, o kuyruk için açık işlemdekileri değiştirir.

Bu bir giriş alanıdır. Bu alanın ilk değeri 0 'dır. *ODVER* *ODVER2* değerinden küçükse bu alan yoksayıdır.

### **ODRRP (işaretçi)**

İlk yanıt kaydının adresi.

Bu, ilk MQRR yanıt kaydının adresidir. *ODRRP* yalnızca bir dağıtım listesi açıldığında kullanılır. *ODREC* sıfırda alan yoksayıdır.

*ODRRP* ya da *ODRRO* yanıt kayıtlarını belirtmek için kullanılabilir, ancak her ikisini birden belirtmez; ayrıntılar için *ODRRO* alanının önceki açıklamasına bakın. *ODRRP* kullanılmazsa, boş değerli göstergeye ya da boş değerli byte 'lara ayarlanmalıdır.

Bu bir giriş alanıdır. Bu alanın ilk değeri boş işaretçidir. *ODVER* *ODVER2*değerinden küçükse bu alan yoksayılır.

#### **ODSID (4 baytlık karakter dizgisi)**

Yapı tanıtıcısı.

Değer şöyle olmalıdır:

##### **ODSIDV**

Nesne tanımlayıcı yapısına ilişkin tanıtıcı.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri *ODSIDV* 'dir.

#### **ODSS (MQCHARV)**

*ODSS*, kuyruktan ileti alınırken kullanılan seçim ölçütlerini sağlamak için kullanılan dizgiyi içerir.

*ODSS* aşağıdaki durumlarda sağlanmamalıdır:

- *ODOT* *OTQ* değilse
- Açılmakta olan kuyruk giriş seçeneklerinden biri kullanılarak açılmıyorsa, *OOINP*\*

Bu durumlarda *ODSS* sağlanırsa, arama *RC2516*neden koduyla başarısız olur.

*ODSS* yanlış belirtilirse, *MQCHARV* yapısının nasıl kullanılacağına ilişkin açıklamaya göre ya da uzunluk üst sınırını aşarsa, çağrı *RC2519*neden koduyla başarısız olur. *ODVER* *ODVER4*değerinden küçükse bu alan yoksayılır.

#### **ODUDC (10 basamaklı işaretli tamsayı)**

Başarıyla açılan uzak kuyruk sayısı

Bu, dağıtım listesindeki uzak kuyruklara çözülecek ve başarıyla açılacak kuyrukların sayısıdır. Varsa, dağıtım listesinde olmayan tek bir kuyruk açılırken de bu alan ayarlanır.

Bu bir çıkış alanıdır. Bu alanın ilk değeri 0 'dir. *ODVER* *ODVER2*değerinden küçükse bu alan yoksayılır.

#### **ODVER (10 basamaklı işaretli tamsayı)**

Yapı sürümü numarası.

Değer aşağıdakilerden biri olmalıdır:

##### **ODVER1**

Version-1 nesne tanımlayıcı yapısı.

##### **ODVER2**

Version-2 nesne tanımlayıcı yapısı.

##### **ODVER3**

Version-3 nesne tanımlayıcı yapısı.

##### **ODVER4**

Version-4 nesne tanımlayıcı yapısı.

Yalnızca yapının daha yeni sürümlerinde bulunan alanlar, alanların tanımlarında bu şekilde tanımlanır. Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

##### **ODVERC**

Nesne tanımlayıcı yapısının yürürlükteki sürümü.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri *ODVER1* 'dir.

#### **Başlangıçtaki değerler**

Çizelge 714. MQOD 'daki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
<i>ODSID</i>	<i>ODSIDV</i>	'ODSID'

Çizelge 714. MÇOD 'daki alanların ilk değerleri (devamı var)

Alan adı	Değişmezin adı	Değişmezin değeri
ODVER	ODVER1	1
ODOT	OTÇ	1
ODON	Yok	Boşluklar
ODMN	Yok	Boşluklar
ODDN	Yok	'AMÇ.*'
ODAU	Yok	Boşluklar
ODREC	Yok	0
ODKDC	Yok	0
ODUDC	Yok	0
ODIDC	Yok	0
ODORO	Yok	0
ODRRO	Yok	0
ODORP	Yok	Boş değerli gösterge ya da boş değerli byte
ODRRP	Yok	Boş değerli gösterge ya da boş değerli byte
ODASI	SINONE.	Boş Değerler
ODRQN	Yok	Boşluklar
ODRMN	Yok	Boşluklar
ODOS	MÇCHARV için tanımlandığı şekilde	MÇCHARV için tanımlandığı şekilde
ODRO	ODOS içinde sağlandığı gibi	ODOS içinde sağlandığı gibi
ODSS	Yok	Boşluklar

**Notlar:**

1. - simgesi, tek bir boş karakteri temsil eder.

## RPG bildirim

```

D*.1.....2.....3.....4.....5.....6.....7..
D*
D* MÇOD Structure
D*
D*
D* Structure identifier
D ODSID          1      4    INZ('OD ')
D*
D* Structure version number
D ODVER          5      8I 0 INZ(1)
D*
D* Object type
D ODOT           9     12I 0 INZ(1)
D*
D* Object name
D ODON          13     60    INZ
D*

```



```

D* Object queue manager name
D ODMN          61    108    INZ
D*
D* Dynamic queue name
D ODDN          109   156    INZ('AMQ.*')
D*
D* Alternate user identifier
D ODAU          157   168    INZ
D*
** Number of object records
D* present
D ODREC         169   172I 0 INZ(0)
D*
** Number of local queues opened
D* successfully
D ODKDC         173   176I 0 INZ(0)
D*
** Number of remote queues opened
D* successfully
D ODUDC         177   180I 0 INZ(0)
D*
** Number of queues that failed to
D* open
D ODIDC         181   184I 0 INZ(0)
D*
** Offset of first object record
D* from start of MQOD
D ODORO         185   188I 0 INZ(0)
D*
** Offset of first response record
D* from start of MQOD
D ODRRO         189   192I 0 INZ(0)
D*
D* Address of first object record
D ODORP         193   208*   INZ(*NULL)
D*
** Address of first response
D* record
D ODRRP         209   224*   INZ(*NULL)
D*
D* Alternate security identifier
D ODASI         225   264    INZ(X'0000000000000000-
D                                     000000000000000000000000-
D                                     000000000000000000000000-
D                                     000000000000')
D*
D* Resolved queue name
D ODRQN         265   312    INZ
D*
D* Resolved queue manager name
D ODRMN         313   360    INZ
D*
D* reserved field
D ODRE1         361   364I 0 INZ(0)
D*
D* reserved field
D ODRS2         365   368I 0 INZ(0)
D*
D* Object long name
D* Address of variable length string
D ODOSCHRP     369   384*   INZ(*NULL)
D* Offset of variable length string
D ODOSCHRO     385   388I 0 INZ(0)
D* Size of buffer
D ODOSVSBS     389   392I 0 INZ(-1)
D* Length of variable length string
D ODOSCHRL     393   396I 0 INZ(0)
D* CCSID of variable length string
D ODOSCHRC     397   400I 0 INZ(-3)
D*
D* Message Selector
D* Address of variable length string
D ODSSCHRP     401   416*   INZ(*NULL)
D* Offset of variable length string
D ODSSCHRO     417   420I 0 INZ(0)
D* Size of buffer
D ODSSVSBS     421   424I 0 INZ(-1)
D* Length of variable length string
D ODSSCHRL     425   428I 0 INZ(0)
D* CCSID of variable length string
D ODSSCHRC     429   432I 0 INZ(-3)
D*

```

```

D* Resolved long object name
D* Address of variable length string
D  ODRSOCHRP          433      448*   INZ(*NULL)
D* Offset of variable length string
D  ODRSOCHRO          449      452I  0 INZ(0)
D* Size of buffer
D  ODRSOVSBS          453      456I  0 INZ(-1)
D* Length of variable length string
D  ODRSOCHRL          457      460I  0 INZ(0)
D* CCSID of variable length string
D  ODRSOCHRC          461      464I  0 INZ(-3)
D*
D* Alias queue resolved object type
D  ODRT                465      468I  0 INZ(0)

```

## IBM i IBM üzerinde MQOR (Nesne kaydı)

MQOR yapısı, tek bir hedef kuyruğun kuyruk adını ve kuyruk yöneticisi adını belirtmek için kullanılır.

### Genel Bakış

**Amaç:** MQOR, MQOPER ve MQPUT1 çağrıları için bir giriş yapısıdır.

**Karakter kümesi ve kodlama:** MQOR ' taki veriler, ENNAT tarafından verilen yerel kuyruk yöneticisinin **CodedCharSetId** kuyruk yöneticisi özneliği ve kodlaması tarafından verilen karakter kümesinde olmalıdır. Ancak, uygulama bir IBM MQ istemcisi olarak çalışıyorsa, yapı, istemcinin karakter kümesinde ve kodlamasında olmalıdır.

**Kullanım:** MQOL çağrısında bu yapılardan oluşan bir dizi sağlayarak, kuyrukların listesini açmak mümkündür; bu listeye *dağıtım listesi* adı verilir. Kuyruk başarıyla açıldıysa, listedeki her bir kuyruğa ilişkin MQOPER çağrısı tarafından döndürülen kuyruk tanıtıcısı kullanılarak verilen her ileti yerleştirilir.

- “Alanlar” sayfa 1150
- “Başlangıçtaki değerler” sayfa 1150
- “RPG bildirimi” sayfa 1151

### Alanlar

MQOR yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

#### ORMN (48 baytlık karakter dizilimi)

Nesne kuyruğu yöneticisi adı.

Bu, MQOD yapısındaki *ODMN* alanı ile aynıdır (ayrıntılar için MQOD ' ye bakın).

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri 48 boş karakterdir.

#### ORON (48 baytlık karakter dizilimi)

Nesne adı.

Bu, MQOD yapısındaki *ODON* alanıyla aynıdır (ayrıntılar için MQOD ' ye bakın). Bunun dışında:

- Bir kuyruğun adı olmalıdır.
- Bu, bir model kuyruğunun adı olmamalıdır.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri 48 boş karakterdir.

### Başlangıçtaki değerler

Çizelge 715. MQYA ' da alanların ilk değerleri		
Alan adı	Değişmez adı	Değişmez değeri
ORON	Yok	Boşluklar

Çizelge 715. MQYA ' da alanların ilk değerleri (devamı var)

Alan adı	Değişimin adı	Değişimin değeri
ORMN	Yok	Boşluklar

## RPG bildirimi

```
D*..1.....2.....3.....4.....5.....6.....7..
D*
D* MQOR Structure
D*
D* Object name
D  ORON                1      48  INZ
D* Object queue manager name
D  ORMN                49     96  INZ
```

## MQPD-Özellik tanımlayıcısı

MQPD özelliği, bir özelliğin özniteliklerini tanımlamak için kullanılır.

## Genel Bakış

**Amaç:** Yapı, MQSETMP çağrısında bir giriş/çıkış parametresi ve MQINQMP çağrısında bir çıkış parametresi.

**Karakter kümesi ve kodlama:** MQPD ' deki veriler, uygulamanın (ENNAT) uygulama ve uygulama kodlamasının karakter kümesinde yer almalıdır.

- “Alanlar” sayfa 1151
- “Başlangıçtaki değerler” sayfa 1154
- “RPG bildirimi” sayfa 1154

## Alanlar

MQPD yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

### PDCT (10 basamaklı işaretli tamsayı)

Bu, özelliğin ait olduğu ileti bağlamını açıklar.

Kuyruk yöneticisi, kuyruk yöneticisinin yanlış olarak tanıdığı IBM MQ-tanımlı bir özelliği içeren bir ileti aldığı anda. Kuyruk yöneticisi, PDCT alanının değerini düzeltir.

Aşağıdaki seçenek belirlenebilir:

### PDUSC

Özellik, kullanıcı bağlamıyla ilişkilendirilir.

MQSETMP çağrısını kullanarak, kullanıcı bağlamıyla ilişkili bir özelliği ayarlayabilmek için özel bir yetki gerekmez.

Bir IBM WebSphere MQ 7.0 kuyruk yöneticisiyle, kullanıcı bağlamıyla ilişkili bir özellik OOSAVA için açıklandığı şekilde saklanır. PMPASA ile belirtilen bir MQPUT çağrısı, özelliğin saklanan bağlamdan yeni iletiye kopyalanmasına neden oluyor.

Önceden açıklanan seçenek gerekli değilse, aşağıdaki seçenek kullanılabilir:

### PNYOC

Özellik bir ileti bağlamıyla ilişkilendirilmemiştir.

Tanınmayan bir değer, RC2482 PDREA kodu ile reddedilir.

Bu, MQSETMP çağrısına ve MQINQMP çağrısından bir çıkış alanına giriş/çıkış alanıdır. Bu alanın ilk değeri PDNOC ' dir.

## **PDCPYOPT (10 basamaklı işaretli tamsayı)**

Bu, özelliğin hangi tip iletilere kopyalanması gerektiğini açıklar.

Bu, tanınan IBM MQ-tanımlı özellikler için yalnızca bir çıkış alanıdır; IBM MQ uygun değeri ayarlar.

Kuyruk yöneticisi, kuyruk yöneticisinin yanlış olarak tanıdığı IBM MQ-tanımlı bir özelliği içeren bir ileti aldığı anda. Kuyruk yöneticisi, *CopyOptions* alanının değerini düzeltir.

Bu seçeneklerden birini ya da birkaçını belirleyebilirsiniz. birden çok seçenek belirtmek için, değerleri bir araya ekleyin (aynı değişmezi bir kereden fazla eklemeyin) ya da bit ya da işlemi kullanarak değerleri birleştirin (programlama dili, bit işlemleri destekliyse).

### **KOPYA**

Bu özellik iletilmekte olan bir iletiye kopyalanır.

### **KOPYA PUANI**

Bu özellik, bir ileti yayınlanırken bir abone tarafından alınan iletiye kopyalanır.

### **ÖNCEKİ**

Bu özellik bir yanıt iletisine kopyalanır.

### **KOPRP**

Bu özellik bir rapor iletisine kopyalanır.

### **KOPYALA**

Bu özellik sonraki iletilerin tüm türlerine kopyalanır.

### **KOPYA**

Bu özellik bir iletiye kopyalanmaz.

**Varsayılan seçenek:** Varsayılan kopya seçenekleri kümesini sağlamak için aşağıdaki seçenek belirlenebilir:

### **KOPYA**

Bu özellik, iletilmekte olan bir iletiye, bir rapor iletisine ya da bir ileti yayınlanmakta olan bir abonenin aldığı iletiye kopyalanır.

Bu değer, COPFOR, artı COPRP ve COPPUB seçenekleri birleşiminin belirtilmesiyle eşdeğerdir.

Önceden açıklanan seçeneklerden hiçbiri gerekli değilse, aşağıdaki seçeneği kullanın:

### **KOPYA**

Bu değeri, başka bir kopyalama seçeneği belirtilmediğini belirtmek için kullanın; bu özellik ile sonraki iletiler arasında programlı olarak hiçbir ilişki yok. Bu, ileti tanımlayıcı özellikleri için her zaman döndürülür.

Bu, MQSETMP çağrısına ve MQINQMP çağrısından bir çıkış alanına giriş/çıkış alanıdır. Bu alanın ilk değeri COPDEF ' dir.

## **PDOPT (10 basamaklı işaretli tamsayı)**

Değer şu olmalıdır:

### **PDYO**

Seçenek belirtilmedi

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri PDNONE 'tır.

## **PDSID (10 basamaklı işaretli tamsayı)**

Bu, yapı tanıtcısıdır; değer şöyle olmalıdır:

### **PSIDV**

Özellik tanımlayıcı yapısına ilişkin tanıtcı.

Bu her zaman bir giriş alanıdır. Bu alanın başlangıç değeri **PSIDV** ' dir.

## **PDSUP (10 basamaklı işaretli tamsayı)**

Bu alanda, bu özelliği içeren iletinin kuyruğa konmasını sağlamak için kuyruk yöneticisinin ileti özelliğine ilişkin destek düzeyi ne kadar olmalıdır. Bu, yalnızca IBM MQ-tanımlı özellikler için geçerlidir; diğer tüm özellikler için destek isteğe bağlıdır.

The field is automatically set to the correct value when the IBM MQ-defined property is known by the queue manager. Özellik tanınmadıysa, PDSUPO atanır. Kuyruk yöneticisi, kuyruk yöneticisinin yanlış olarak tanıdığı IBM MQ-tanımlı bir özelliği içeren bir ileti aldığı anda. Kuyruk yöneticisi, *PDSUP* alanının değerini düzeltir.

CMNOVA seçeneğinin ayarlandığı bir ileti tanıtıcısı için MQSETMP çağrısını kullanarak IBM MQ-tanımlı bir özellik ayarlanırken, *PDSUP* bir giriş alanı olur. Bu, bir uygulamanın, doğru değerle, özelliğin bağlı kuyruk yöneticisi tarafından desteklenmediği, ancak iletinin başka bir kuyruk yöneticisiyle işlenmek üzere tasarlandığı bir IBM MQ-tanımlı özelliği koymasına olanak sağlar.

PSEPO değeri her zaman, IBM MQ-tanımlı özellikler olmayan özelliklere atanır.

İleti özelliklerini destekleyen bir IBM WebSphere MQ 7.0 kuyruk yöneticisi, tanınmayan bir *PDSUP* değeri içeren bir özellik alır; özellik aşağıdaki gibi işlenir:

- PDSUR, tanınmayan değerlerden herhangi biri PDRUM ' da içerilirse, bu değer belirtildi.
- PDAUXM ' de tanınmayan değerlerden herhangi biri varsa, PDSUPL belirtildi.
- Ters durumda, PDSUPO belirtildi.

MQINQMP çağrısı tarafından aşağıdaki değerlerden biri döndürülür ya da CMNOVA seçeneğinin belirlendiği bir ileti tanıtıcısında MQSETMP çağrısı kullanılırken, değerlerden biri belirtilebilir:

### **PDSSUPO**

Özellik, bir kuyruk yöneticisi tarafından desteklenmese de kabul edilir. İletinin ileti özelliklerini desteklemeyen bir kuyruk yöneticisine akabilmesi için özellik atılabilir. Bu değer, IBM MQ-tanımlı olmayan özelliklere de atanır.

### **PDSSPR**

Özelliğe ilişkin destek gereklidir. İleti, IBM MQ-tanımlı özelliği desteklemeyen bir kuyruk yöneticisi tarafından reddedilir. MQPUT ya da MQPUT1 çağrısı, tamamlanma kodu CCFAIL ve neden kodu RC2490 ile başarısız olur.

### **PDSSUPL**

İleti, yerel bir kuyruk için yazılmışsa, IBM MQ-tanımlı özelliği desteklemeyen bir kuyruk yöneticisi tarafından reddedilir. MQPUT ya da MQPUT1 çağrısı, tamamlanma kodu CCFAIL ve neden kodu RC2490 ile başarısız olur.

İleti uzak bir kuyruk yöneticisine gönderiliyorsa, MQPUT ya da MQPUT1 çağrısı başarılı olur.

Bu, MQINQMP çağrısındaki bir çıkış alanıdır ve MQSETMP çağrısıyla ilgili bir giriş alanı, ileti tanıtıcısı CMNOVA seçenek kümesiyle yaratıldıysa, bu alan çağrılır. Bu alanın ilk değeri PDSUPO 'dur.

## **PDSVER (10 basamaklı işaretli tamsayı)**

Bu, yapı sürüm numarasıdır; değer şu olmalıdır:

### **PDVER1**

Version-1 özellik tanımlayıcı yapısı.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

### **PDVERC**

Özellik tanımlayıcı yapısının yürürlükteki sürümü.

Bu her zaman bir giriş alanıdır. Bu alanın başlangıç değeri **PDVER1** ' dir.

## Başlangıçtaki değerler

Çizelge 716. MQPD 'deki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
PDSID	PDSIDV	'PD'
PDVER	PDVER1	1
PDOPT	PDYO	0
PDSUP	PDSSUPO	0
PDCT	PNYOC	0
PDCPYOPT	KOPYA	0

## RPG bildirimi

```
D* MQDMHO Structure
D*
D*
D* Structure identifier
D DMSID 1 4 INZ('DMHO')
D*
D* Structure version number
D DMVER 5 8I 0 INZ(1)
D*
D* Options that control the action of MQDLTMH
D DMOPT 9 12I 0 INZ(0)
```

## IBM i MQPMO (Put-message options) on IBM i

MQPMO yapısı, uygulamanın, iletilerin kuyruklara nasıl yerleştirileceğini ya da hangi konulara yayınlanmasını denetleyen seçenekleri belirlemesine olanak sağlar.

### Genel Bakış

#### Amaç

Yapı, MQPUT ve MQPUT1 çağrılarında bir giriş/çıkış değiştirir.

#### S\u00fcr\u00fcm

MQPMO 'nun yürürlükteki sürümü PMVER2' dir. Yalnızca yapının daha yeni sürümlerinde var olan alanlar, izleyen açıklamalarda olduğu gibi tanımlanır.

Sağlanan COPY dosyası, ortam tarafından desteklenen en son MQPMO sürümünü içerir; ancak, PMVER alanının ilk değeri PMVER1olarak ayarlıdır. version-1 yapısında mevcut olmayan alanları kullanmak için, uygulamanın PMVER alanını, gereken sürümün sürüm numarasına ayarlaması gerekir.

#### Karakter kümesi ve kodlama

Data in MQPMO must be in the character set given by the **CodedCharSetId** queue manager attribute and encoding of the local queue manager given by ENNAT. Ancak, uygulama bir IBM MQ istemcisi olarak çalışıyorsa, yapı, istemcinin karakter kümesinde ve kodlamasında olmalıdır.

- [“Alanlar” sayfa 1154](#)
- [“Başlangıçtaki değerler” sayfa 1167](#)
- [“RPG bildirimi” sayfa 1168](#)

### Alanlar

MQPMO yapısı aşağıdaki alanları içerir; alanlar alfabetik sırayla anlatılır:

### **PMCT (10 basamaklı işaretli tamsayı)**

Giriş kuyruğunun nesne tanıtıcısı.

PMPASI ya da PMPASA belirtilirse, bu alan, konulmakta olan iletiyle ilişkilendirilecek bağlam bilgilerinin yer aldığı giriş kuyruğu tanıtıcısını içermelidir.

MPMPASI ve PMPASA belirtilmiyorsa, bu alan yoksayılr.

Bu bir giriş alanıdır. Bu alanın başlangıç değeri 0' dir.

### **PMIDC (10 basamaklı işaretli tamsayı)**

Gönderilebilecek ileti sayısı.

Bu, dağıtım listesindeki kuyruklara gönderilebilecek iletilerin sayısıdır. Bu sayı, açıklamadı ve başarıyla açılan kuyruklar, ancak koyma işleminin başarısız olduğu kuyruklar içeriyor. Bu alan, dağıtım listesinde olmayan tek bir kuyruğa ileti yerleştirilirken de ayarlanır.

**Not:** Bu alan yalnızca, MQPUT ya da MQPUT1 çağrısındaki **CMPCOD** değıştirgesi CCOK ya da CCWARN olduğunda ayarlanır; **CMPCOD** parametresi CCFAIL ise, bu değeri belirlenmez.

Bu bir çıkış alanıdır. Bu alanın ilk değeri 0 'tır. *PMVER* , *PMVER2*değerinden küçükse bu alan ayarlanmaz.

### **PMKDC (10 basamaklı işaretli tamsayı)**

Yerel kuyruklara başarıyla gönderilen ileti sayısı.

Bu, yürürlükteki MQPUT ya da MQPUT1 çağrısının, dağıtım listesindeki yerel kuyruklar olan kuyruklara başarıyla gönderdiği ileti sayısıdır. Bu sayı, uzak kuyruklara çözülen kuyruklara gönderilen iletileri (yerel iletim kuyruğu, iletiyi saklamak için başlangıçta kullanıldığı halde) içermez. Bu alan, dağıtım listesinde olmayan tek bir kuyruğa ileti yerleştirilirken de ayarlanır.

Bu bir çıkış alanıdır. Bu alanın ilk değeri 0 'tır. *PMVER* , *PMVER2*değerinden küçükse bu alan ayarlanmaz.

### **PMOPT (10 basamaklı işaretli tamsayı)**

MQPUT ve MQPUT1işleminin denetlenmesini denetleyen seçenekler.

Aşağıdakilerin hiçbiri ya da hiçbiri belirlenebilir. Birden fazla değeri gerekliyse, değeri eklenebilir (aynı değeri bir kereden fazla eklemeyin). Geçerli olmayan birleşimler dikkat edilir; diğer birleşimler geçerli olur.

**Yayınlama seçenekleri:** Aşağıdaki seçenekler, iletilerin bir konuya yayınlanma şeklini denetler.

#### **PMSRTO**

Bu yayının MQMD ' ye ilişkin MDRQ ve MDRM alanlarına veri girildiğinde, abonelere geçilmez. Bu seçenek, ReplyToQ gerektiren bir rapor seçeneği ile kullanılırsa, arama RC2027 ile başarısız olur.

#### **PIPT**

Gönderilmekte olan yayın kuyruk yöneticisi tarafından alıkonabilecektir. Bu, bir abonenin, MQSUBRQ çağrısını kullanarak, yayımlandığı zamandan sonra bu yayının bir kopyasını istemesine olanak sağlar. Ayrıca, SONEWP seçeneğini kullanarak gönderilmemeyi seçmedikleri sürece, bu yayının yayınlanmasını sağlayan uygulamalara bir yayının gönderilmesine de olanak tanır. Bir uygulama alıkonan bir yayını gönderdiyse, o yayının mq.IsRetained ileti özelliği tarafından belirtilir.

Konu ağacının her düğümünde yalnızca bir yayın saklanabilir. Bu, başka bir uygulama tarafından yayınlanan bu konu için zaten alıkonan bir yayın varsa, bu yayının yerine bu yayınlı değıştirilir. Bu nedenle, aynı konu üzerinde birden fazla yayınlı tutma iletilisi bulundurmamak daha iyidir.

Alıkonan yayınlar bir abone tarafından istendiğinde, kullanılan abonelik, konuya genel arama karakteri içerebilir; bu durumda alıkonan yayınlar sayısı (konu ağacındaki çeşitli düğümlerde) ve istekte bulunan uygulamaya birkaç yayın gönderilebilir. Daha fazla ayrıntı için "MQSUBRQ-Abonelik isteği" sayfa 782 çağrısının açıklamasına bakın.

Bu seçenek kullanılırsa ve yayın saklanamazsa, ileti yayınlanmaz ve çağrı RC2479 ile başarısız olur.

**Syncpoint seçenekleri:** Aşağıdaki seçenekler, bir iş birimi içinde MQPUT ya da MQPUT1 çağrısının katılımlarıyla ilgilidir:

#### **PSYP**

İleti syncpoint denetimiyle birlikte girin.

Bu istek, olağan çalışma birimi protokolleri içinde işlem yapmak. İleti, iş birimi kesinleştirilinceye kadar iş biriminin dışında görünmez. İş birimi yedeklendiyse, ileti silinir.

Bu seçenek ve PMNSYP belirtilmediyse, koyma isteği bir iş birimi içinde değildir.

PMSYP, PMNSYP ile belirlenmemelidir.

#### **PMNSYP**

İletiyi eşitleme noktası denetimi olmadan yerleştirin.

Bu istek, olağan çalışma birimi protokollerinin dışında işlem yapmak. İleti hemen kullanılabilir ve bir iş birimi yedeklenerek silinemez.

Bu seçenek ve PMSYP belirtilmediyse, koyma isteği bir iş birimi içinde değildir.

PMNSYP, PMSYP ile belirlenmemelidir.

**İleti tanıtıcısı ve ilinti tanıtıcısı seçenekleri:** Aşağıdaki seçenekler kuyruk yöneticisinin yeni bir ileti tanıtıcısı ya da ilinti tanıtıcısı oluşturmasını ister:

#### **PMNMID**

Yeni bir ileti tanıtıcısı oluşturun.

Bu seçenek, kuyruk yöneticisinin MQMD ' deki *MDMID* alanının içeriğini yeni bir ileti tanıtıcısıyla değiştirmesine neden olur. Bu ileti tanıtıcısı iletilerle birlikte gönderilir ve MQPUT ya da MQPUT1 çağrısından çıkış için uygulamaya geri gönderilir.

Bu seçenek, ileti bir dağıtım listesine konduğunda da belirlenebilir; ayrıntılar için, MQPMR yapısındaki *PRMID* alanının tanımına bakın.

Bu seçeneğin kullanılması, her MQPUT ya da MQPUT1 çağrısından önce *MDMID* alanını MINONE olarak ilk durumuna getirme gereksinmesini giderir.

#### **PMNCID**

Yeni bir ilinti tanıtıcısı oluşturun.

Bu seçenek, kuyruk yöneticisinin MQMD ' deki *MDCID* alanının içeriğini yeni bir ilinti tanıtıcısıyla değiştirmesine neden olur. Bu ilinti tanıtıcısı iletilerle birlikte gönderilir ve MQPUT ya da MQPUT1 çağrısından çıkış için uygulamaya geri gönderilir.

Bu seçenek, ileti bir dağıtım listesine konduğunda da belirlenebilir; ayrıntılar için, MQPMR yapısındaki *PRCID* alanının tanımına bakın.

PMNCID, uygulamanın benzersiz bir ilinti tanıtıcısı gerektirdiği durumlarda kullanışlıdır.

**Grup ve bölüm seçenekleri:** Aşağıdaki seçenek, mantıksal iletilerin gruplarındaki ve kısımlarındaki iletilerin işlenmesiyle ilgilidir. Bu tanımlamalar, aşağıdaki seçeneğin anlaşılmasına yardımcı olabilir:

#### **Fiziksel ileti**

Bu, bir kuyruğa yerleştirilebilecek ya da kuyruktan kaldırılabilir en küçük bilgi birimidir; genellikle, tek bir MQPUT, MQPUT1 ya da MQGET çağrısında belirtilen ya da alınan bilgilere karşılık gelir. Her fiziksel ileti, kendi ileti tanımlayıcısına (MQMD) sahiptir. Genellikle, fiziksel iletiler ileti tanıtıcısı (MQMD ' de *MDMID* alanı) için farklı değerlere göre ayırt edilir; ancak bu, kuyruk yöneticisi tarafından zorunlu kılınmaz.

#### **Mantıksal ileti**

Bu, uygulama bilgilerinin tek bir birimidir. Sistem koşullarının yokluğunda, mantıksal bir ileti fiziksel bir iletiyle aynı olur. Ancak, mantıksal iletilerin büyük olduğu durumlarda, sistem kısıtlamaları, bir mantıksal iletinin *bölümler* olarak adlandırılan iki ya da daha çok fiziksel iletiye bölünmesini önerebilir ya da zorunlu kılabilir.



Kesimlere ayrılmış bir mantıksal ileti, aynı boş değerli grup tanıtıcısına (MQMD 'deMDGID alanı) ve aynı ileti dizisi numarasına (MQMD' deMDSEQ alanı) sahip iki ya da daha çok fiziksel iletten oluşur. Kesimler, mantıksal iletide verilerin başlangıcındaki fiziksel iletteki verilerin görel konumunu veren bölüm görel konumu (MQMD ' deMDOFF alanı) için farklı değerlere göre ayırt edilir. Her bir bölüm fiziksel bir ileti olduğu için, mantıksal bir iletteki kesimler genellikle farklı ileti tanıtıcılarına sahiptir.

Ayrılmamış bir mantıksal ileti; ancak, gönderme uygulaması tarafından bölümlenmeye izin verilen, boş değerli olmayan bir grup tanıtıcısı da vardır; ancak, bu durumda, mantıksal ileti bir ileti grubuna ait değilse, o grup tanıtıcısına sahip tek bir fiziksel ileti vardır. Gönderme uygulaması tarafından engellendiği mantıksal iletler, mantıksal ileti bir ileti grubuna ait değilse, boş değerli grup tanıtıcısına (GINONE) sahip olur.

### **İleti grubu**

Bu, aynı boş değerli grup tanıtıcısına sahip bir ya da daha fazla mantıksal ileti kümesidir. Gruptaki mantıksal iletler, 1-n aralığında bir tamsayı olan ileti sıra numarası için farklı değerlere göre ayırt edilir; burada n, gruptaki mantıksal iletlerin sayısıdır. Mantıksal iletlerden biri ya da daha fazlası bölümlendiye, grupta n fiziksel ileti sayısı daha fazla olur.

### **PLOGO**

Mantıksal iletlerin gruplarındaki ve kısımlarındaki iletler mantıksal sırada yer alıyor.

Bu seçenek, kuyruk yöneticisine uygulamanın, iletleri gruplar ve mantıksal iletler segmentlerine nasıl yerleştirdiğini bildirir. Yalnızca MQPUT çağrısında belirtilebilir; MQPUT1 çağrısında geçerli değildir.

PMLOGO belirtilirse, uygulamanın art arda gelen MQPUT çağrılarını kullandığını gösterir:

- Her bir mantıksal iletiye, boşluk olmadan, 0 'dan başlayarak, artan kesim görel konumu sırasına göre kesimler yerleştirin.
- Bölümleri sonraki mantıksal iletiye koymadan önce tüm bölümleri tek bir mantıksal iletiyle koyun.
- Herhangi bir boşluk olmadan, 1 'den başlayarak, ileti sıra sayısı artırımı sırasına göre, her ileti grubuna mantıksal iletleri yerleştirin.
- Sonraki ileti grubuna mantıksal iletler konmadan önce, tüm mantıksal iletleri bir ileti grubuna koyun.

Bu sipariş "mantıksal sipariş" olarak adlandırılır.

Uygulama, kuyruk yöneticisine, iletleri gruplar ve mantıksal iletler bölümlerine nasıl yerleştirdiğini anlattığından, kuyruk yöneticisi bunu yaptığı için, uygulamanın her bir MQPUT çağrısıyla ilgili grup ve bölüm bilgilerini korumak ve güncellemek zorunda kalmaması gerektiğini belirtti. Özellikle, kuyruk yöneticisi bunları uygun değerlere ayarlandığı için, uygulamanın MQMD ' de MDGID, MDSEQ ve MDOFF alanlarını ayarlamaya gerek olmadığı anlamına gelir. Uygulama, MQMD ' de yalnızca MDMFL alanının ayarına, iletlerin ne zaman gruplara ait olduğunu ya da mantıksal iletlerin kesimleri olduğunu ve bir mantıksal iletinin bir gruptaki ya da son kesiminde son iletiyi belirtmek için ayarlanacak.

Once a message group or logical message has been started, subsequent MQPUT calls must specify the appropriate MF\* flags in MDMFL in MQMD. Uygulama, sonlandırılmamış bir ileti grubu olduğunda ya da sonlandırılmamış bir mantıksal ileti olduğunda bir kısım olmayan bir ileti yerleştirmeye çalışırsa, çağrı neden kodu RC2241 ya da RC2242 ile uygun olduğu şekilde başarısız olur. Ancak, kuyruk yöneticisi yürürlükteki ileti grubuyla ya da yürürlükteki mantıksal iletiyle ilgili bilgileri saklar ve MQPUT çağrısını, grupta olmayan ya da bir kesim olmayan iletiyi koymak için yeniden yayınlamadan önce, uygun olduğu şekilde MFLMIG ya da MFLSEG belirtilerek bir ileti (uygulama iletisi verisi olmadan) göndererek bu bilgileri sonlandırabilir.

Çizelge 717 sayfa 1158 , geçerli olan seçenek ve işaretlerin birleşimlerini ve kuyruk yöneticisinin her bir durumda kullandığı MDGID, MDSEQ ve MDOFF alanlarının değerlerini gösterir. Tabloda gösterilmeyen seçenek ve işaret birleşimleri geçerli değil. Çizelgedeki kolonlar aşağıdaki anlamlara sahiptir:

**OTURUM KAPAT**

Çağrıda PMLOGO seçeneğinin belirtilip belirtilmediğini belirtir.

**MIG**

Çağrıda MFMIG ya da MFLMIG seçeneğinin belirlenip belirlenmediğini belirtir.

**GÇ**

Çağrıda MFSEG ya da MFLSEG seçeneğinin belirlenip belirlenmediğini belirtir.

**SEEG Tamam**

Çağrıda MFSEGA seçeneğinin belirlenip belirtilmediğini belirtir.

**Cur grp**

Çağrıdan önce geçerli bir ileti grubunun var olup olmadığını belirtir.

**Cur günlük iletisi**

Çağrıdan önce geçerli bir mantıksal iletinin var olup olmadığını belirtir.

**Diğer kolonlar**

Kuyruk yöneticisinin kullandığı değerleri gösterir. "Önceki" değeri, kuyruk tanıtıcısı için önceki iletide alan için kullanılan değeri belirtir.

**PMRLOC**

MQPMO yapısındaki PMRQN 'nin, iletinin yerleştirilecek olan yerel kuyruk adıyla tamamlanması gerektiğini belirtir. ResolvedQMGrAdı benzer şekilde, yerel kuyruğu barındıran yerel kuyruk yöneticisinin adıyla tamamlanır. Bunun ne anlama geldiğini görmek için OORLOQ 'ya bakın. Bir kullanıcının bir kuyruğa koyma yetkisi varsa, bu kullanıcının MQPUT çağrısında bu işareti belirtmesi için gereken yetkiye sahip olur. Özel bir yetki gerekli değildir.

Çizelge 717. Mantıksal iletilerin gruplarındaki ve kısımlarındaki iletilere ilişkin MQPUT seçenekleri								
Belirttiğiniz seçenekler				Aramadan önce grup ve günlük-msg durumu		Kuyruk yöneticisinin kullandığı değerler		
GÜNLÜK ORD	MIG	SEG	SEÇTAMAM	Cur Grp	Ccur günlük iletisi	MDGID	MDSEQ	MDOFF
Evet	Hayır	Hayır	Hayır	Hayır	Hayır	GINONE	1	0
Evet	Hayır	Hayır	Evet	Hayır	Hayır	Yeni grup tanıtıcısı	1	0
Evet	Hayır	Evet	Evet ya da Hayır	Hayır	Hayır	Yeni grup tanıtıcısı	1	0
Evet	Hayır	Evet	Evet ya da Hayır	Hayır	Evet	Önceki grup tanıtıcısı	1	Önceki görel konum + önceki bölüm uzunluğu
Evet	Evet	Evet ya da Hayır	Evet ya da Hayır	Hayır	Hayır	Yeni grup tanıtıcısı	1	0
Evet	Evet	Evet ya da Hayır	Evet ya da Hayır	Evet	Hayır	Önceki grup tanıtıcısı	Önceki sıra numarası + 1	0
Evet	Evet	Evet	Evet ya da Hayır	Evet	Evet	Önceki grup tanıtıcısı	Önceki sıra numarası	Önceki görel konum + önceki bölüm uzunluğu

Çizelge 717. Mantıksal iletilerin gruplarındaki ve kısımlarındaki iletilere ilişkin MQPUT seçenekleri (devamı var)

Belirttiğiniz seçenekler				Aramadan önce grup ve günlük-msg durumu		Kuyruk yöneticisinin kullandığı değerler		
Hayır	Hayır	Hayır	Hayır	Evet ya da Hayır	Evet ya da Hayır	GINONE	1	0
Hayır	Hayır	Hayır	Evet	Evet ya da Hayır	Evet ya da Hayır	GINONE ise yeni grup tanıtıcısı, alanındaki başka değer	1	0
Hayır	Hayır	Evet	Evet ya da Hayır	Evet ya da Hayır	Evet ya da Hayır	GINONE ise yeni grup tanıtıcısı, alanındaki başka değer	1	Alandaki değer
Hayır	Evet	Hayır	Evet ya da Hayır	Evet ya da Hayır	Evet ya da Hayır	GINONE ise yeni grup tanıtıcısı, alanındaki başka değer	Alandaki değer	0
Hayır	Evet	Evet	Evet ya da Hayır	Evet ya da Hayır	Evet ya da Hayır	GINONE ise yeni grup tanıtıcısı, alanındaki başka değer	Alandaki değer	Alandaki değer

**Not:**

- PMLOGO, MQPUT1 çağrısında geçerli değil.
- *MDMID* alanı için, kuyruk yöneticisi, *PMNMID* ya da *MINONE* belirtilirse yeni bir ileti tanıtıcısı oluşturur ve diğer bir değer olarak alandaki değeri kullanır.
- *MDCID* alanı için, kuyruk yöneticisi yeni bir ilinti tanıtıcısı oluşturur (*PMNCID* belirtildiyse) ve alandaki değeri başka bir biçimde kullanır.

PMLOGO belirtildiğinde, kuyruk yöneticisi bir gruptaki tüm iletilerin ve mantıksal bir iletide belirtilen tüm iletilerin, MQMD 'deki *MDPER* alanında aynı değere sahip olmasını gerektirir; yani, tüm bunların kalıcı olması ya da tümünün kalıcı olmaması gerekir. Bu koşul karşılanmazsa, MQPUT çağrısı başarısız olur. Neden kodu: RC2185 .

PMLOGO seçeneği, iş birimlerini aşağıdaki gibi etkiler:

- Bir gruptaki ilk fiziksel ileti ya da mantıksal ileti bir iş birimi içine konursa, aynı kuyruk tanıtıcısı kullanılsa, gruptaki ya da mantıksal iletteki diğer tüm fiziksel iletilerin bir iş birimi içine konması gerekir. Ancak, aynı iş birimi içinde yer almamaları gerekir. Bu, birçok fiziksel iletilerden oluşan bir ileti grubunun ya da mantıksal iletinin, kuyruk tanıtıcısı için iki ya da daha çok iş birimi arasında bölünmesine olanak tanır.
- Bir gruptaki ilk fiziksel ileti ya da mantıksal ileti bir iş birimi içine konmazsa, aynı kuyruk tanıtıcısı kullanılsa, gruptaki diğer fiziksel iletilerden hiçbiri ya da mantıksal ileti bir iş birimi içine yerleştirilebilir.

Bu koşullar karşılanmazsa, MQPUT çağrısı başarısız olur. Neden kodu: RC2245 .

PMLOGO belirtildiğinde, MQPUT çağrısında sağlanan MQMD MDVER2değerinden küçük olmamalıdır. Bu koşul karşılanmazsa, çağrı neden kodu RC2257 ile başarısız olur.

PMLOGO belirtilmemişse, mantıksal iletilerin gruplarındaki ve bölümlerindeki iletiler herhangi bir sıraya yerleştirilebilir ve tam ileti grupları ya da tam mantıksal iletiler koymak gerekmez. It is the

responsibility of the application to ensure that the *MDGID*, *MDSEQ*, *MDOFF*, and *MDMFL* fields have appropriate values.

Sistem başarısızlığı oluştuktan sonra, ortadaki bir ileti grubunu ya da mantıksal iletiyi yeniden başlatmak için kullanılabilir teknik budur. When the system restarts, the application can set the *MDGID*, *MDSEQ*, *MDOFF*, *MDMFL*, and *MDPER* fields to the appropriate values, and then issue the MQPUT call with PMSYP or PMNSYP set as *gerekli*, but without specifying PMLOGO. Bu çağrı başarılı olursa, kuyruk yöneticisi grup ve bölüm bilgilerini saklar ve bu kuyruk tanıtıcısı kullanılarak yapılan sonraki MQPUT çağrıları, PMLOGO ' yı olağan şekilde belirtebilir.

Kuyruk yöneticisinin MQPUT çağrısı için sakladığı grup ve kesim bilgileri, MQGET çağrısına ilişkin sakladığı grup ve bölüm bilgilerinden ayrıdır.

Herhangi bir kuyruk tanıtıcısı için uygulama, PMLOGO belirten MQPUT çağrılarında sahip olmayan MQPUT çağrılarını karıştırmaz; ancak, aşağıdaki noktalar dikkate alınmalıdır:

- PMLOGO belirtilmezse, her başarılı MQPUT çağrısı kuyruk yöneticisinin, kuyruk tanıtıcısı için grup ve kesim bilgilerini, uygulama tarafından belirlenen değerlere ayarlamasına neden olur; bu durum, kuyruk tanıtıcısı için kuyruk yöneticisi tarafından korunan varolan grup ve kesim bilgilerinin yerine geçer.
- PMLOGO belirtilmezse, geçerli bir ileti grubu ya da mantıksal ileti varsa arama başarısız olmaz; ancak çağrı bir CCWARN tamamlanma kodu ile başarılı olabilir. [Çizelge 718 sayfa 1160](#) , oluşabilecek çeşitli vakaları gösterir. Bu durumlarda, tamamlanma kodu CCOK değilse, neden kodu aşağıdakilerden biridir (uygun olduğu gibi):
  - RC2241
  - RC2242
  - RC2185
  - RC2245

**Not:** Kuyruk yöneticisi, MQPUT1 çağrısına ilişkin grup ve bölüm bilgilerini denetlemez.

Çizelge 718. MQPUT ya da MQCLOSE çağrısının grup ve bölüm bilgileriyle tutarlı olmadığı bir sonuç		
Yürürlükteki çağrı	Önceki arama PMLOGO ile MQPUT oldu	Önceki arama PMLOGO olmadan MQPUT oldu
PMLOGO ile MQPUT	CCCFAIL	CCCFAIL
PMLOGO olmadan MQPUT	CCWARN	CCOK
Sonlandırılmamış bir grupta ya da mantıksal iletiyle MQCLOSE	CCWARN	CCOK

Yalnızca iletileri ve bölümleri mantıksal sırayla koymak isteyen uygulamalar, bu kullanım için en basit seçenek olduğu için, PMLOGO ' yı belirtmeleri önerilir. Bu seçenek, kuyruk yöneticisi bu bilgileri yönettiği için, grup ve bölüm bilgilerini yönetme gereksiniminin uygulanını giderir. Ancak, özelleştirilmiş uygulamaların PMLOGO seçeneği tarafından sağlanandan daha fazla denetime gereksinimi olabilir ve bu, bu seçeneği belirtmeyerek elde edilebilir. Bu işlem yapılırsa, uygulamanın her MQPUT ya da MQPUT1 çağrısından önce, MQMD ' deki *MDGID*, *MDSEQ*, *MDOFF* ve *MDMFL* alanlarının doğru olarak ayarlandığından emin olmalıdır.

Örneğin, bu iletilerin gruplar halinde mi, yoksa mantıksal ileti bölümleri mi olduğunu dikkate almadan, aldığı fiziksel iletileri iletmek isteyen bir uygulama, PMLOGO belirtmemelidir. Bunun iki nedeni vardır:

- İletiler alınıp alınır, PMLOGO belirtildiğinde, iletilere yeni bir grup tanıtıcısı atanmasına neden olur ve bu ileti, ileti grubunun sonucundaki herhangi bir yanıtı ya da rapor iletilerini ilintilendirmek için iletilerin kaynağı için bu durumu zorlaştırabilir ya da olanaksız kılabilir.
- Kuyruk yöneticileri gönderme ve alma arasında birden çok yol içeren karmaşık bir ağda, fiziksel iletiler sıradan çıkabilirler. MQGET çağrısında PMLOGO ve karşılık gelen GMLOGO belirtmeyerek,

iletme uygulaması her fiziksel iletiyi alır almaz ve mantıksal sırada gelecek mantıksal sırada beklemeye gerek kalmadan alabilir ve iletebilirler.

Grup ya da mantıksal ileti bölümlerindeki iletiler için rapor iletileri oluşturan uygulamaların, rapor iletileri yerleştirilirken PMLOGO da belirtmemesi gerekir.

PMLOGO diğer PM\* seçeneklerinden herhangi biriyle belirtilebilir.

**Bağlam seçenekleri:** Aşağıdaki seçenekler ileti bağlamının işlenmesini denetler:

#### **PMSNOK**

İletiyile ilişkilendirilecek bağlam yok.

Hem kimlik, hem de kaynak bağlamı, bağlam olmadığını göstermek üzere ayarlanır. Bu, MQMD 'deki bağlam alanlarının şu şekilde ayarlanacak şekilde ayarlanandır:

- Karakter alanları için boşluk
- Byte alanları için boş değerler
- Sayısal alanlar için sıfırlar

#### **PMDEFC**

Varsayılan bağlamı kullanın.

İleti, hem kimlik hem de kaynak için, kendisiyle ilişkilendirilmiş varsayılan bağlam bilgilerine sahip olmak içindir. Kuyruk yöneticisi, ileti tanımlayıcısındaki bağlam alanlarını aşağıdaki gibi ayarlar:

*Çizelge 719. MQMD alanları için varsayılan bağlam bilgisi değerleri*

<b>MQMD ' de alan</b>	<b>Kullanılan değer</b>
<i>MDUID</i>	Olanaklıysa, ortamdan saptanır; tersi durumda boşluklara ayarlanır.
<i>MDACC</i>	Olanaklıysa, ortamdan belirlenir; tersi durumda ACNONE olarak ayarlanır.
<i>MDAID</i>	Boşluklara ayarlayın.
<i>MDPAT</i>	Ortamdan saptanır.
<i>MDPAN</i>	Olanaklıysa, ortamdan saptanır; tersi durumda boşluklara ayarlanır.
<i>MDPD</i>	İletinin konulduğu tarihe ayarlanır.
<i>MDPT</i>	İleti konduğunda zaman olarak ayarlayın.
<i>MDAOD</i>	Boşluklara ayarlayın.

İleti bağlamına ilişkin daha fazla bilgi için bkz. [İleti bağlamı](#) ve [Bağlam bilgilerini denetleme](#).

Bağlam seçeneği belirtilmediyse, bu varsayılan işlemdir.

#### **PMPASI**

Kimlik bağlamını bir giriş kuyruğu tanıtıcısından geçirin.

İleti, kendisiyle ilişkili bağlam bilgisi içermesini sağlar. Kimlik bağlamı, *PMCT* alanında belirtilen kuyruk tanıtıcısından alınır. Kaynak bağlamı bilgileri, kuyruk yöneticisi tarafından, PMDEFC için aynı şekilde oluşturulur (değerler için önceki çizelgeye bakın). İleti bağlamına ilişkin daha fazla bilgi için bkz. [İleti bağlamı](#) ve [Bağlam bilgilerini denetleme](#).

MOPUT çağrısı için, kuyruğun OOPASI seçeneğiyle (ya da bunu belirten bir seçenek) açılmış olması gerekir. MOPUT1 çağrısı için, OOPASI seçeneği ile MOPUT çağrısına ilişkin aynı yetki denetimi gerçekleştirilir.

#### **PMPASA**

Bir giriş kuyruğu tutamacından tüm bağlamı geçirin.

İleti, kendisiyle ilişkili bağlam bilgisi içermesini sağlar. Hem kimlik hem de kaynak bağlamı, *PMCT* alanında belirtilen kuyruk saptından alınır. İleti bağlamına ilişkin daha fazla bilgi için bkz. [İleti bağlamı](#) ve [Bağlam bilgilerini denetleme](#).

MQPUT çağrısı için, kuyruk OOPASA seçeneği (ya da bunu belirten bir seçenek) ile açılmış olmalıdır. MQPUT1 çağrısı için, aynı yetki denetimi, OOPASA seçeneği ile MQOP çağrısına ilişkin olarak yürütülür.

### **PMSSETI**

Kimlik bağlamını uygulamadan ayarlayın.

İleti, kendisiyle ilişkili bağlam bilgisi içermesini sağlar. Uygulama, MQMD yapısındaki kimlik bağlamını belirtir. Kaynak bağlamı bilgileri, kuyruk yöneticisi tarafından, PMDEFC için aynı şekilde oluşturulur (değerler için önceki çizelgeye bakın). İleti bağlamına ilişkin daha fazla bilgi için bkz. İleti bağlamı ve Bağlam bilgilerini denetleme.

MQPUT çağrısı için, kuyruğun OOSSETI seçeneği (ya da bunu belirten bir seçenek) ile açılmış olması gerekir. MQPUT1 çağrısı için, aynı yetki denetimi OOSSETI seçeneğiyle MQOP çağrısına göre yürütülür.

### **PMSETA**

Uygulamadaki tüm bağlamı ayarlayın.

İleti, kendisiyle ilişkili bağlam bilgisi içermesini sağlar. Uygulama, MQMD yapısındaki kimlik ve başlangıç bağlamını belirtir. İleti bağlamına ilişkin daha fazla bilgi için bkz. İleti bağlamı ve Bağlam bilgilerini denetleme.

MQPUT çağrısı için, kuyruğun OOSSETA seçeneğiyle açılmış olması gerekir. MQPUT1 çağrısı için, OOSSETA seçeneği ile MQOP çağrısına aynı yetki denetimi gerçekleştirilir.

PM\* bağlam seçeneklerinden yalnızca biri belirtilebilir. Bu seçeneklerden hiçbiri belirlenmezse, PMDEFC varsayılan değer olarak kabul edilir.

**Yanıt tiplerini yerleştirin.** Aşağıdaki seçenekler, bir MQPUT ya da MQPUT1 çağrısına döndürülen yanıtı denetler. Bu seçeneklerden yalnızca birini belirleyebilirsiniz. PMARES ve PMSRES belirlenmezse, PMRASQ ya da PMRAST varsayılır.

### **PMARES**

PMARES seçeneği, uygulama kuyruk yöneticisinin çağrısı tamamlamayı beklemesi olmadan, bir MQPUT ya da MQPUT1 işleminin tamamlandığı istekleri olur. Bu seçeneğin kullanılması, özellikle istemci bağ tanımlarını kullanan uygulamalar için ileti alışverişi başarımını yükseltebilirler. Bir uygulama, önceki zamanuyumsuz çağrılar sırasında bir hata oluşup oluşmadığını belirterek, MQSTAT komutunu kullanarak düzenli aralıklarla denetleyebilir.

Bu seçenekle, MQMD ' de yalnızca aşağıdaki alanların tamamlanabileceği garanti edilir;

- MDATNT
- MDPAT
- MDPAN
- MDAOD

Ayrıca, seçenekler olarak PMNMID ya da PMNCID ya da her ikisi de belirtilirse, döndürülen MDMID ve kullanıcı kimliği de de tamamlanır. (PMNMID, boş bir MDMID alanı belirtilerek örtük olarak belirtilebilir).

Yalnızca önceden belirlenen alanlar tamamlanır. Olağan durumda MQMD ya da MQPMO yapısında döndürülecek diğer bilgiler tanımsızdır.

MQPUT ya da MQPUT1 için zamanuyumsuz koyma yanıtı istenirken, CMPCOD ve REASON of CCOK ve RUNNONE, iletinin bir kuyruğa başarıyla konulduğu anlamına gelmeyebilir. Zamanuyumsuz put yanıtı kullanan ve iletilerin bir kuyruğa yerleştirilmesini gerektiren bir MQI uygulaması geliştirirken, put işlemlerinden hem CMPCOD, hem de NEDEN kodlarını denetlemeniz gerekir. Ayrıca zamanuyumsuz hata bilgilerini sorgulamak için MQSTAT işlevini de kullanabilirsiniz.

Her bir MQPUT/MQPUT1 çağrısının başarısı ya da başarısızlığı hemen döndürülme de, zamanuyumsuz bir çağrı altında oluşan ilk hata, daha sonra MQSTAT çağrısıyla belirlenebilecek bir durumda belirlenebilir.

Zamanuyumsuz koyma yanıtı kullanılarak eşitleme noktası altındaki kalıcı bir ileti teslim edilemezse ve hareketi kesinleştirme girişiminde bulunursanız, kesinleştirme başarısız olur ve işlem CCFAIL ' in tamamlanma koduyla ve RC2003 nedeniyle geriletir. Uygulama, önceki bir MQPUT ya da MQPUT1 hatasının nedenini saptamak için MQSTAT çağrısını yapabilir.

#### **PMSRES**

MQPMO yapısındaki bir put seçeneği için bu değerin belirlenmesi, MQPUT ya da MQPUT1 işleminin her zaman zamanuyumlu olarak yayınlanmasını sağlar. İşlem başarılı olursa, MQMD ve MQPMO 'daki tüm alanlar tamamlanır. Kuyruk ya da konu nesnesinde tanımlanan varsayılan değer koyma yanıt değeri bağımsız olarak zamanuyumlu bir yanıt verildiğinden emin olmak için sağlanmıştır.

#### **PMRASQ**

Bir MQPUT çağrısı için bu değer belirtilirse, kullanılan put yanıt tipi, uygulama tarafından açıldığı sırada kuyruğunda belirtilen DEFPRESP değerinden alınır. Bir istemci uygulaması IBM WebSphere MQ 7.0' dan önceki bir düzeyde bir kuyruk yöneticisine bağlıysa, PMSRES belirtilmiş gibi davranır.

Bir MQPUT1 çağrısı için bu seçenek belirlenirse, kuyruk tanımlamasından DEFPRESP değeri kullanılmamaktadır. MQPUT1 çağrısı PMSYP kullanıyorsa, PMARES için işlev görür ve PMNSYP kullanıyorsanız, PMSRES işlevi görür.

#### **PMRAST**

Bu, konu nesneleriyle kullanım için PMRASQ eşanlamlısıdır.

**Diğer seçenekler:** Aşağıdaki seçenekler yetkilendirme denetlemesini denetler ve kuyruk yöneticisi susturulduğunda ne olur:

#### **PLATTU**

Belirtilen kullanıcı kimliğiyle doğrulanıyor.

Bu, MQPUT1 çağrısının **OBJDSC** parametresindeki *ODAU* alanının, kuyrukta ileti koyma yetkisinin geçerliliğini denetlemek için kullanılacak bir kullanıcı kimliği içerdiğini belirtir. Bu çağrı, uygulamanın çalışmakta olduğu kullanıcı kimliğinin bu işlemi gerçekleştirme yetkisine sahip olup olmadığına bakılmaksızın, yalnızca bu *ODAU* kuyruğun belirtilen seçeneklerle açma yetkisine sahip olduğunda başarılı olabilir. (Bu seçenek, uygulamanın çalışmakta olduğu kullanıcı kimliğine her zaman geri verilen bağlam seçenekleri için geçerli değildir.)

Bu seçenek yalnızca MQPUT1 çağrısıyla geçerli olur.

#### **PFIQ**

Kuyruk yöneticisi susturulmuş durumda olduğunda başarısız olun.

Bu seçenek, kuyruk yöneticisi susturma durumdaysa, MQPUT ya da MQPUT1 çağrısını başarısız olarak zorlar.

Çağrı, RC2161 neden koduyla CCFAIL tamamlanma kodunu döndürür.

**Varsayılan seçenek:** Önceden açıklanan seçeneklerden hiçbiri gerekmiyorsa, aşağıdaki seçenek kullanılabilir:

#### **PMYOK**

Seçenek belirtilmedi.

Bu değer, başka bir seçeneğin belirlenmemesine işaret etmek için kullanılabilir; tüm seçenekler varsayılan değerlerini varsayar. PMNONE, program belgelerine yardımcı olmak üzere tanımlıdır; bu seçeneğin başka biriyle birlikte kullanılması amaçlanmaz, ancak değeri sıfır olduğu için, bu tür bir kullanım saptanamaz.

Bu bir giriş alanıdır. *PMOPT* alanının ilk değeri PMNONE 'dir.

#### **PMPRF (10 basamaklı işaretli tamsayı)**

Hangi MQPMR alanlarının var olduğunu gösteren işaretler.

Bu alan, uygulama tarafından sağlanan ileti kaydı kayıtlarında hangi MQPMR alanlarının bulunduğunu göstermek için ayarlanması gereken işaretleri içerir. *PMPRF* yalnızca ileti bir dağıtım listesine konduğunda kullanılır. *PMREC* sıfırsa ya da hem *PMPRO* , hem de *PMPRP* sıfırsa, alan yoksayılır.

Var olan alanlar için, kuyruk yöneticisi ilgili put iletisi kaydındaki alanlardan değerlerin her bir hedefi için kullanılır. Eksik olan alanlar için kuyruk yöneticisi, MQMD yapısından değerleri kullanır.

Koyma iletisi kayıtlarında hangi alanların bulunduğunu belirtmek için aşağıdaki işaretlerden biri ya da daha fazlası belirtilebilir:

**PFMID**

İleti tanıtıcısı alanı var.

**PFCID**

İlinti tanıtıcısı alanı var.

**PFA**

Grup tanıtıcısı alanı var.

**PFFB**

Geribildirim alanı mevcut.

**PFACC**

Muhasebe belirteci alanı var.

Bu işaret belirtilirse, *PMOPT* alanında *PMSETI* ya da *PMSETA* belirtilmelidir; bu koşul karşılanmazsa, çağrı neden kodu RC2158 ile başarısız olur.

Hiçbir MQPMR alanı yoksa, aşağıdaki değer belirlenebilir:

**FPNONE**

Hiçbir put-message kayıt alanı mevcut değil.

Bu değer belirtilirse, *PMREC* sıfır ya da hem *PMPRO* , hem de *PMPRP* sıfır olmalıdır.

FPNONE, program belgelerine yardımcı olacak şekilde tanımlıdır. Bu değişimin başka biriyle birlikte kullanılması amaçlanmaz, ancak değeri sıfır olduğu için, bu tür bir kullanım saptanamaz.

*PMPRF* geçersiz işaretler içeriyorsa ya da ileti kayıtları sağlandıysa, ancak *PMPRF* değeri FPNONE değerine sahipse, çağrı neden kodu RC2158 ile başarısız olur.

Bu bir giriş alanıdır. Bu alanın ilk değeri FPNONE 'dir. *PMVER* , *PMVER2* değerinden küçükse bu alan yok sayılır.

**PMPRO (10 basamaklı işaretli tamsayı)**

İlk olarak MQPMO ' nun başlangıcından ileti kaydının kayması.

Bu, MQPMO yapısının başlangıcındaki ilk MQPMR put ileti kaydının bayt cinsinden görel konutdur. Görel konum pozitif ya da negatif olabilir. *PMPRO* yalnızca ileti bir dağıtım listesine konduğunda kullanılır. *PMREC* sıfırsa, alan yoksayılır.

İleti bir dağıtım listesine konduğunda, her hedef için iletinin belirli özelliklerini belirtmek üzere bir ya da daha çok MQPMR içeren bir ileti kaydı dizisi sağlanabilir; bu özellikler şunlardır:

- İleti Tanıtıcısı
- ilinti tanıtıcısı
- grup tanıtıcısı
- geri bildirim değeri
- muhasebe simgesi

Bu özelliklerin tümünü belirtmek gerekli değildir, ancak hangi altküme seçiliyse, alanların doğru sırada belirtilmesi gerekir. Ek ayrıntılar için MQPMR yapısına ilişkin açıklamaya bakın.

Genellikle, dağıtım listesi açıldığında MQOD tarafından belirlenen nesne kayıtları olduğu için, birden çok ileti kaydı olması gerekir; her bir ileti kaydı, karşılık gelen nesne kaydı tarafından tanımlanan kuyruğa ilişkin ileti özelliklerini sağlar. Bu durumda ileti özellikleri dikkate alınmasa da, dağıtım listesindeki açıkta başarısız olan kuyruklar, dizideki uygun konumlarda kendilerine ayrılmış ileti kayıtlarını yerleştirmiş olmalıdır.

İleti kayıtlarının sayısı, nesne kayıtlarının sayısından farklı olacak şekilde olabilir. Nesne kayıtlarından daha az sayıda ileti kaydı varsa, ileti kaydı kayıtlarını yerleştirmeyen hedeflerin ileti özellikleri, MQMD



ileti tanımlayıcısındaki ilgili alanlardan alınır. Nesne kayıtlarından daha fazla ileti kaydı varsa, fazlalık kullanılmaz (bunlara erişebilmek için yine de mümkün olmalıdır). Put ileti kayıtları isteğe bağlıdır, ancak bunlar sağlanırsathem' un *PMREC* olması gerekir.

Put ileti kayıtları, *PMPRO* içindeki bir görelî konum belirtilerek ya da *PMPRP* 'ta bir adres belirterek, MÇOD' daki nesne kayıtlarına benzer şekilde sağlanabilir. Bunu nasıl yapacağına ilişkin ayrıntılar için, "IBM i üzerinde MÇOD (Nesne tanımlayıcı)" sayfa 1140 içinde açıklanan *ODORO* alanına bakın.

*PMPRO* ve *PMPRP* dışında birden fazla olamaz; her ikisi de sıfır değilse, arama başarısız olur. Neden kodu RC2159 .

Bu bir giriş alanıdır. Bu alanın ilk değeri 0 'tır. *PMVER* , *PMVER2* değerinden küçükse bu alan yok sayılır.

### **PMPRP (işaretçi)**

İlk ileti kaydının adresi.

Bu, ileti kaydının ilk MÇPMR ' nin adresidir. *PMPRP* yalnızca ileti bir dağıtım listesine konduğunda kullanılır. *PMREC* sıfırsa, alan yok sayılır.

İleti koyma kayıtlarını belirtmek için *PMPRP* ya da *PMPRO* kullanılabilir, ancak her ikisi de değil; ayrıntılar için *PMRRO* alanının tanımına bakın. *PMPRP* kullanılmazsa, boş (null) işaretçi ya da boş değerli (null) byte olarak ayarlanmalıdır.

Bu bir giriş alanıdır. Bu alanın başlangıç değeri, boş değerli göstergedir. *PMVER* , *PMVER2* değerinden küçükse bu alan yok sayılır.

### **PMREC (10 basamaklı işaretli tamsayı)**

Gönderilen ileti kayıtlarının ya da yanıt kayıtlarının sayısı.

Bu, uygulama tarafından sağlanan ileti kayıtlarını ya da MÇRR yanıt kayıtlarını içeren MÇPMR ' nin sayısıdır. Bu sayı, yalnızca ileti bir dağıtım listesine konulmakta olduğunda sıfırdan büyük olabilir. İleti kayıtları ve yanıt kayıtları isteğe bağlıdır-uygulamanın herhangi bir kayıt sağlamaması gerekir ya da yalnızca tek bir türe ait kayıtları sağlamayı seçebilir. Ancak, uygulama her iki tipteki kayıtları da sağlıyorsa, her tipte *PMREC* kaydı sağlamalıdır.

*PMREC* değerinin, dağıtım listesindeki hedeflerin sayısı ile aynı olması gerekmez. Çok sayıda kayıt sağlanırsa, fazlalık kullanılmaz; çok az kayıt sağlanırsa, ileti kaydı koyma değeri olmayan hedeflere ilişkin ileti özellikleri için varsayılan değerler kullanılır (bu konuda daha sonra *PMPRO* başlıklı konuya bakın).

*PMREC* sıfırdan küçükse ya da sıfırdan büyükse, ancak ileti bir dağıtım listesine konmadıysa, çağrı neden kodu RC2154 ile başarısız olur.

Bu bir giriş alanıdır. Bu alanın ilk değeri 0 'tır. *PMVER* , *PMVER2* değerinden küçükse bu alan yok sayılır.

### **PMRMN (48 baytlık karakter dizilimi)**

Hedef kuyruk yöneticisinin çözümlenmiş adı.

Bu ad, yerel kuyruk yöneticisi tarafından ad çözme işlemi gerçekleştirildikten sonra hedef kuyruk yöneticisinin adıdır. Döndürülen ad, *PMRQN* ile tanıtılan kuyruğa sahip olan kuyruk yöneticisinin adıdır ve yerel kuyruk yöneticisinin adı olabilir.

*PMRQN* , yerel kuyruk yöneticisinin ait olduğu kuyruk paylaşım grubunun sahip olduğu bir paylaşım kuyruksa, *PMRMN* kuyruk paylaşım grubunun adıdır. If the queue is owned by some other queue sharing group, *PMRQN* can be the name of the queue sharing group or the name of a queue manager that is a member of the queue sharing group (the nature of the value returned is determined by the queue definitions that exist at the local queue manager).

Boş olmayan bir değer yalnızca, nesne tek bir kuyruksa döndürülür; nesne bir dağıtım listesi ya da konuyla, döndürülen değer tanımsız olur.

Bu bir çıkış alanıdır. Bu alanın uzunluğu LNÇMN tarafından verilir. Bu alanın ilk değeri 48 boş karakterdir.

## PMRQN (48 baytlık karakter dizilimi)

Hedef kuyruğun çözümlenmiş adı.

Bu ad, yerel kuyruk yöneticisi tarafından ad çözme işlemi gerçekleştirildikten sonra hedef kuyruğunun adı. The name returned is the name of a queue that exists on the queue manager identified by *PMRMN*.

Boş olmayan bir değer yalnızca, nesne tek bir kuyruksa döndürülür; nesne bir dağıtım listesi ya da konuyla, döndürülen değer tanımsız olur.

Bu bir çıkış alanıdır. Bu alanın uzunluğu LNQN tarafından verilir. Bu alanın ilk değeri 48 boş karakterdir.

## PMRRO (10 basamaklı işaretli tamsayı)

MQPMO ' nun başlangıcından ilk yanıt kaydının kayması.

Bu, MQPMO yapısının başlangıcındaki ilk MQRR yanıt kaydının bayt cinsinden görelî konudur. Görelî konum pozitif ya da negatif olabilir. *PMRRO* yalnızca ileti bir dağıtım listesine konduğunda kullanılır. *PMREC* sıfırsa, alan yoksayılr.

İleti bir dağıtım listesine konduğunda, iletinin başarıyla gönderilmediği kuyrukları (MQRR 'deRRCC alanı) ve her başarısızlığın nedenini (MQRR' deRRREA alanı) tanımlamak için bir ya da daha fazla MQRR yanıt kaydı dizisi sağlanabilir. Kuyruk açılmadığından ya da koyma işlemi başarısız olduğu için ileti gönderilmemiş olabilir. Kuyruk yöneticisi yanıt kayıtlarını yalnızca, arama sonucu karışık olduğunda (yani, bazı iletiler başarılı bir şekilde başarısız olurken ya da tüm başarısız, ancak farklı nedenler için başarılı bir şekilde gönderildiğinde) ayarlar; bu çağrıdan RC2136 neden kodu bu vakayı belirtir. Aynı neden kodu tüm kuyruklar için geçerliyse, bu neden MQPUT ya da MQPUT1 çağrısının **REASON** parametresine döndürülür ve yanıt kayıtları belirlenmez.

Genellikle, dağıtım listesi açıldığında, MQOD tarafından belirlenen nesne kayıtları olduğu için, yanıt kayıtları da olmalıdır; gerektiğinde, her yanıt kaydı, ilgili nesne kaydı tarafından tanımlanan kuyruğa konmak üzere tamamlanma koduna ve neden koduna ayarlanır. Dağıtım listesindeki, açılmayan dağıtım listesindeki kuyruklar, dizideki uygun konumlarda kendilerine ayrılmış yanıt kayıtları olmalıdır; ancak, bunlar tamamlama kodu ve neden kodu için, put işlemi yerine, açık işlemde kaynaklanan neden kodlarına sahip olmalıdır.

Yanıt kayıtlarının sayı sayısı, nesne kayıtlarının sayısından farklı olabilir. Nesne kayıtlarından daha az sayıda yanıt kaydı varsa, uygulama, koyma işleminin başarısız olduğu tüm hedefleri ya da başarısızlıkların nedenlerini belirtmek için bu uygulama için mümkün olmayabilir. Nesne kayıtlarından daha fazla yanıt kaydı varsa, fazlalık kullanılmaz (bunlara erişebilmek için yine de olanaklı olması gerekir). Yanıt kayıtları isteğe bağlıdır, ancak bunlar belirtilirsethem' un *PMREC* olması gerekir.

Yanıt kayıtları, *PMRRO* içindeki bir görelî konum belirtilerek ya da *PMRRP* 'ta bir adres belirterek, MQOD' daki nesne kayıtlarına benzer şekilde sağlanabilir. Bunu nasıl yapacağına ilişkin ayrıntılar için, "[IBM i üzerinde MQOD \(Nesne tanımlayıcı\)](#)" sayfa 1140 içinde açıklanan *ODORO* alanına bakın. However, no more than one of *PMRRO* and *PMRRP* can be used; the call fails with reason code RC2156 if both are nonzero.

MQPUT1 çağrısı için bu alanın sıfır olması gerekir. Bunun nedeni, yanıt bilgilerinin (istendiye), MQOD nesne tanımlayıcısı tarafından belirtilen yanıt kayıtlarında döndürülmesinden kaynaklanır.

Bu bir giriş alanıdır. Bu alanın ilk değeri 0 'tır. *PMVER* , *PMVER2* değerinden küçükse bu alan yok sayılır.

## PMRRP (işaretçi)

İlk yanıt kaydının adresi.

Bu, ilk MQRR yanıt kaydının adresidir. *PMRRP* yalnızca ileti bir dağıtım listesine konduğunda kullanılır. *PMREC* sıfırsa, alan yoksayılr.

Yanıt kayıtlarını belirtmek için *PMRRP* ya da *PMRRO* kullanılabilir; ayrıntılar için her ikisi de değil; ayrıntılar için *PMRRO* alanının açıklamasını görebilirsiniz. *PMRRP* kullanılmazsa, boş (null) işaretçi ya da boş değerli (null) byte olarak ayarlanmalıdır.

MQPUT1 çağrısı için, bu alan boş değerli ya da boş değerli byte olmalıdır. Bunun nedeni, yanıt bilgilerinin (istendiye), MQOD nesne tanımlayıcısı tarafından belirtilen yanıt kayıtlarında döndürülmesinden kaynaklanır.

Bu bir giriş alanıdır. Bu alanın başlangıç değeri, boş değerli göstergedir. *PMVER* , *PMVER2*değerinden küçükse bu alan yok sayılır.

#### **PMSID (4 baytlık karakter dizgisi)**

Yapı tanıtıcısı.

Değer şu olmalıdır:

##### **PMSIDV**

Put-message options yapısına ilişkin tanıtıcı.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri PMSIDV ' dir.

#### **PMSL (MQUZA)**

Bu yayın tarafından hedeflenen abonelik düzeyi.

Yalnızca, bu değere eşit ya da bu değere eşit en yüksek *PMSL* aboneliği olan abonelikler bu yayını alır. Bu değer, sıfır ile 9 aralığında olmalıdır; sıfır, en düşük düzeydir.

Bu alanın ilk değeri 9 'tır.

#### **PMTO (10 basamaklı işaretli tamsayı)**

Ayrılmış.

Bu ayrılmış bir alandır; değeri anlamlı değildir. Bu alanın başlangıç değeri -1' dir.

#### **PMUDC (10 basamaklı işaretli tamsayı)**

Uzak kuyruklara başarıyla gönderilen ileti sayısı.

Bu, yürürlükteki MQPUT ya da MQPUT1 çağrısının uzak kuyruklara çözümleyen dağıtım listesindeki kuyruklara başarılı bir şekilde gönderdiği ileti sayısıdır. Kuyruk yöneticisinin dağıtım listesi biçiminde geçici olarak alıkonacağı iletiler, bu dağıtım listelerinin içerdiği hedef noktaların sayısı olarak sayılır. Bu alan, dağıtım listesinde olmayan tek bir kuyruğa ileti yerleştirilirken de ayarlanır.

Bu bir çıkış alanıdır. Bu alanın ilk değeri 0 'tır. *PMVER* , *PMVER2*değerinden küçükse bu alan ayarlanmaz.

#### **PMVER (10 basamaklı işaretli tamsayı)**

Yapı sürüm numarası.

Değer aşağıdakilerden biri olmalıdır:

##### **PMVER1**

Version-1 put-message options yapısı.

##### **PMVER2**

Version-2 put-message options structure.

Yalnızca yapının daha yeni sürümünde var olan alanlar, alanların açıklamalarında olduğu gibi tanımlanır. Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

##### **PMVERC**

Put-message options yapısının yürürlükteki sürümü.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri PMVER1' dir.

### **Başlangıçtaki değerler**

Çizelge 720. MQPMO 'daki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
<i>PMSID</i>	PMSIDV	' PMO→'
<i>PMVER</i>	PMVER1	1

Çizelge 720. MQPMO 'daki alanların ilk değerleri (devamı var)

Alan adı	Değişmezin adı	Değişmezin değeri
PMOPT	PMYOK	0
PMTO	Yok	-1
PMCT	Yok	0
PMKDC	Yok	0
PMUDC	Yok	0
PMIDC	Yok	0
PMRQN	Yok	Boşluklar
PMRMN	Yok	Boşluklar
PMREC	Yok	0
PMPRF	FPNONE	0
PMPRO	Yok	0
PMRRO	Yok	0
PMPRP	Yok	Boş değerli gösterge ya da boş değerli byte
PMRRP	Yok	Boş değerli gösterge ya da boş değerli byte

**Not:**

1. - simgesi tek bir boş karakteri temsil eder.

## RPG bildirimi

```

D*.1.....2.....3.....4.....5.....6.....7..
D* MQPMO Structure
D*
D* Structure identifier
D PMSID 1 4 INZ('PMO ')
D* Structure version number
D PMVER 5 8I 0 INZ(1)
D* Options that control the action of MQPUT and MQPUT1
D PMOPT 9 12I 0 INZ(0)
D* Reserved
D PMTO 13 16I 0 INZ(-1)
D* Object handle of input queue
D PMCT 17 20I 0 INZ(0)
D* Number of messages sent successfully to local queues
D PMKDC 21 24I 0 INZ(0)
D* Number of messages sent successfully to remote queues
D PMUDC 25 28I 0 INZ(0)
D* Number of messages that could not be sent
D PMIDC 29 32I 0 INZ(0)
D* Resolved name of destination queue
D PMRQN 33 80 INZ
D* Resolved name of destination queue manager
D PMRMN 81 128 INZ
D* Number of put message records or response records present
D PMREC 129 132I 0 INZ(0)
D* Flags indicating which MQPMR fields are present
D PMPRF 133 136I 0 INZ(0)
D* Offset of first put message record from start of MQPMO
D PMPRO 137 140I 0 INZ(0)
D* Offset of first response record from start of MQPMO
D PMRRO 141 144I 0 INZ(0)
D* Address of first put message record
    
```

D	PMPRP	145	160*	INZ(*NULL)
D*	Address of first response record			
D	PMRRP	161	176*	INZ(*NULL)
D*	Original message handle			
D	PMOMH	177	184I	0
D*	New message handle			
D	PMNMH	185	190I	0
D*	The action being performed			
D	PMACT	191	194I	0
D*	Reserved			
D	PMRE1	195	198I	0

## IBM i MQPMR (Put-message record) on IBM i

MQPMR yapısı, dağıtım listesine bir ileti konduğunda, tek bir hedefe ilişkin çeşitli ileti özelliklerini belirtmek için kullanılır.

### Genel Bakış

**Amaç:** MQPMR, MQPUT ve MQPUT1 çağrılarına ilişkin bir giriş/çıkış yapısıdır.

**Karakter kümesi ve kodlama:** Data in MQPMR must be in the character set given by the **CodedCharSetId** queue manager attribute and encoding of the local queue manager given by ENNAT. Ancak, uygulama bir IBM MQ istemcisi olarak çalışıyorsa, yapı, istemcinin karakter kümesinde ve kodlamasında olmalıdır.

**Kullanım:** MQPUT ya da MQPUT1 çağrısında bu yapılardan oluşan bir dizi sağlayarak, dağıtım listesindeki her hedef kuyruğu için farklı değerler belirtmek mümkündür. Alanların bazıları yalnızca giriş, diğerleri giriş/çıkış.

**Not:** Bu yapı, sabit bir yerleşim düzenine sahip olmadığı için olağan dışıdır. Bu yapıdaki alanlar isteğe bağlıdır ve her alanın varlığı ya da yokluğu, MQPMO 'daki *PMPRF* alanındaki işaretlerle gösterilir. Fields that are present **aşağıdaki sırada oluşmalıdır** :

- *PRMID*
- *PRCID*
- *PRGID*
- *PRFB*
- *PRACC*

Olmayan alanlar, kayıta yer kapmaz.

MQPMR 'nin değişmez yerleşim düzeni olmadığı için, COPY dosyasında bu bir tanım sağlanmaz. Uygulama programcısı, uygulamanın gerektirdiği alanları içeren bir bildirim yaratmalı ve var olan alanları belirtmek için *PMPRF* içinde işaretleri ayarlamalıdır.

- [“Alanlar” sayfa 1169](#)
- [“Başlangıçtaki değerler” sayfa 1171](#)
- [“RPG bildirim” sayfa 1171](#)

### Alanlar

MQPMR yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

#### PRACC (32 baytlık bit dizesi)

Muhasebe simgesi.

Bu, MQOPER ya da MQPUT1 çağrısında bulunan MQOR yapılarına karşılık gelen öge tarafından belirtilen bir adla kuyruğa gönderilen ileti için kullanılacak hesap simgesindeki bir simgedir. Bu, tek bir kuyruk için MQMD 'deki *MDACC* alanı ile aynı şekilde işlenir. Bu alanın içeriğiyle ilgili bilgi için [“MQMD \(Message descriptor\) on IBM i” sayfa 1092](#) içindeki *MDACC* açıklamalarına bakın.

Bu alan mevcut değilse, MQMD 'deki değer kullanılır.

Bu bir giriş alanıdır.

### **PRCID (24 baytlık bit dizgisi)**

İlinti tanıtıcısı.

Bu, MQOPER ya da MQPUT1 çağrısında bulunan MQOR yapılarının dizideki ilgili öge tarafından belirtilen adla kuyruğa gönderilen ileti için kullanılacak ilinti tanımlayıcısıdır. Bu, tek bir kuyruk için MQMD ' deki *MDCID* alanı ile aynı şekilde işlenir.

If this field is not present in the MQPMR record, or there are fewer MQPMR records than destinations, the value in MQMD is used for those destinations that do not have an MQPMR record containing a *PRCID* field.

PMNCID belirtilirse, bir *tek* yeni ilinti tanıtıcısı oluşturulur ve bunların MQPMR kayıtları olup olmadığına bakılmaksızın, dağıtım listesindeki tüm hedefler için kullanılır. Bu, PMNMID ' nin işleme şeklinden farklıdır ( *PRMID* alanına bakın).

Bu bir giriş/çıkış alanıdır.

### **PRFB (10 basamaklı işaretli tamsayı)**

Geri bildirim ya da neden kodu.

Bu, MQOPED ya da MQPUT1 çağrısında bulunan MQOR yapılarına karşılık gelen öge tarafından belirtilen adda kuyruğa gönderilen ileti için kullanılacak geri bildirim kodudur. Bu, tek bir kuyruk için MQMD ' deki *MDFB* alanı ile aynı şekilde işlenir.

Bu alan mevcut değilse, MQMD ' deki değer kullanılır.

Bu bir giriş alanıdır.

### **PRGID (24 baytlık bit dizesi)**

Grup tanıtıcısı.

Bu, MQOPER ya da MQPUT1 çağrısında bulunan MQOR yapılarının dizideki ilgili öge tarafından belirtilen adda kuyruğa gönderilen ileti için kullanılacak grup tanıtıcısıdır. Bu, tek bir kuyruk için MQMD ' deki *MDGID* alanı ile aynı şekilde işlenir.

If this field is not present in the MQPMR record, or there are fewer MQPMR records than destinations, the value in MQMD is used for those destinations that do not have an MQPMR record containing a *PRGID* field. Değer, Çizelge 717 sayfa 1158 içinde belgelendiği şekilde işlenir, ancak aşağıdaki farklar söz alır:

- Yeni bir grup tanıtıcısının kullanılabilmesi durumunda, kuyruk yöneticisi her hedef için farklı bir grup tanıtıcısı oluşturur (yani, iki hedef aynı grup tanıtıcısına sahip değildir).
- In those cases where the value in the field would be used, the call fails with reason code RC2258.

Bu bir giriş/çıkış alanıdır.

### **PRMID (24 baytlık bit dizesi)**

İleti tanıtıcısı.

Bu ileti, MQOPER ya da MQPUT1 çağrısında bulunan MQOR yapılarına karşılık gelen öge tarafından belirtilen adda kuyruğa gönderilen ileti için kullanılacak ileti tanıtıcısıdır. Bu, tek bir kuyruk için MQMD ' deki *MDMID* alanı ile aynı şekilde işlenir.

If this field is not present in the MQPMR record, or there are fewer MQPMR records than destinations, the value in MQMD is used for those destinations that do not have an MQPMR record containing a *PRMID* field. Bu değer MINONE ise, bu hedeflerin *her biri* için yeni bir ileti tanıtıcısı oluşturulur (yani, bu hedeflerin ikisi de aynı ileti tanıtıcısına sahip değildir).

PMNMID belirlenirse, dağıtım listesindeki tüm hedefler için, MQPMR kayıtları olup olmamasından bağımsız olarak yeni ileti tanıtıcıları yaratılır. Bu, PMNCID ' nin işleme şeklinden farklıdır ( *PRCID* alanına bakın).

Bu bir giriş/çıkış alanıdır.

## Başlangıçtaki değerler

Hiçbir yapı bildirimi sağlanmadığı için, bu yapı için herhangi bir başlangıç değeri tanımlanmamış. Aşağıdaki örnek bildiriminde, alanların tümü gerekiyorsa, uygulamanın uygulama programcısı tarafından nasıl bildirilmesi gerektiğini gösterir.

## RPG bildirimi

```
D*..1.....2.....3.....4.....5.....6.....7..
D* MQPMR Structure
D*
D* Message identifier
D PRMID 1 24
D* Correlation identifier
D PRCID 25 48
D* Group identifier
D PRGID 49 72
D* Feedback or reason code
D PRFB 73 76I 0
D* Accounting token
D PRACC 77 108
```

IBM i

## IBM üzerinde MQRFH (Kurallar ve biçimlendirme üstbilgisi)

MQRFH yapısı, kural ve biçimlendirme üstbilgisinin düzenini tanımlar.

## Genel Bakış

**Amaç:** Bu üstbilgi, dizgi verilerini ad-değer çiftleri biçiminde göndermek için kullanılabilir.

**Biçim adı:** FMRFH.

**Karakter kümesi ve kodlama:** MQRFH yapısındaki alanlar (*RFNVS* içinde olmak üzere), MQRFH 'den önce gelen üstbilgi yapısındaki *MDCSI* ve *MDENC* alanları tarafından verilen karakter kümesi ve kodlamadır. MQRFH, uygulama iletilisi verilerinin başlangıcındaki MQRFH' de yer alan bu alanlar tarafından belirlenir.

Karakter takımı, kuyruk adlarında geçerli olan karakterler için tek byte 'lık karakterler içeren bir karakter takımı olmalıdır.

- [“Alanlar” sayfa 1171](#)
- [“Başlangıçtaki değerler” sayfa 1173](#)
- [“RPG bildirimi” sayfa 1174](#)

## Alanlar

MQRFH yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

### RFCSI (10 basamaklı işaretli tamsayı)

Character set identifier of data that follows *RFNVS*.

Bu, *RFNVS* ' i izleyen verilerin karakter kümesi tanıtıcısını belirtir. MQRFH yapısındaki karakter verileri için geçerli değildir.

MQPUT ya da MQPUT1 çağrısında, uygulama bu alanı veri için uygun değere ayarlamalıdır. Aşağıdaki özel değer kullanılabilir:

### CINT

Bu yapının karakter kümesi tanıtıcısını devralır.

Character data in the data *takip edilen* this structure is in the same character set as this structure.

Kuyruk yöneticisi, bu değeri, iletide belirtilen yapının gerçek karakter kümesi tanıtıcısına göre değiştirir. Hata oluşmaması durumunda, MQGET çağrısıyla CSINHT değeri döndürülmez.

MQMD ' deki *MDPAT* alanının değeri ATBRKR ise, CSINHT kullanılamaz.

Bu alanın ilk değeri CSUNDF 'dir.

*RFNVS*' u izleyen verilerin sayısal kodlaması.

Bu, *RFNVS* ' i izleyen verilerin sayısal kodlamasını belirtir. MQRFH yapısındaki sayısal veriler için geçerli değildir.

MQPUT ya da MQPUT1 çağrısında, uygulama bu alanı veri için uygun değere ayarlamalıdır.

Bu alanın ilk değeri ENNAT ' tır.

### **RFFLG (10 basamaklı işaretli tamsayı)**

Bayraklar.

Şu öge belirtilebilir:

#### **YOK**

Bayrak yok.

Bu alanın ilk değeri RFNONE 'tır.

### **RFFMT (8 baytlık karakter dizilimi)**

*RFNVS*' i izleyen verilerin biçimi.

Bu, *RFNVS*' i izleyen verilerin biçim adını belirtir.

MQPUT ya da MQPUT1 çağrısında, uygulama bu alanı veri için uygun değere ayarlamalıdır. Bu alanı kodlamaya ilişkin kurallar, MQMD ' deki *MDFMT* alanı için yapılan kurallarla aynıdır.

Bu alanın ilk değeri FMNONE ' dir.

### **RFLEN (10 basamaklı işaretli tamsayı)**

*RFNVS* içinde olmak üzere toplam MQRFH uzunluğu.

Bu, yapının sonundaki *RFNVS* alanı da dahil olmak üzere, MQRFH yapısının bayt cinsinden uzunluğudur. Uzunluk, *RFNVS* alanını izleyen herhangi bir kullanıcı verisi içermez.

Bazı ortamlarda kullanıcı verilerinin veri dönüştürmesiyle ilgili sorunları önlemek için, *RFLEN* komutunu dört bir kereden fazla kullanmayı düşünün.

Aşağıdaki değişmez, yapının *sabit* bir kısmının uzunluğunu gösterir; yani, *RFNVS* alanı hariç olmak üzere uzunluk.

#### **RFLNV**

MQRFH yapısının değişmez kısmı uzunluğu.

Bu alanın ilk değeri RFLNV ' dir.

### **RFNVS (n-byte karakter dizgisi)**

Ad-değer çiftlerini içeren dizgi.

Bu, formdaki ad-değer çiftlerini içeren değişken uzunluklu bir karakter dizilimidir:

```
name1 value1 name2 value2 name3 value3 ...
```

Her ad ya da değer, bitişik ad ya da değerden bir ya da daha çok boşluk karakteriyle ayrılmalıdır; bu boşlukların boş olması gerekir. Bir ad ya da değer, adı ya da değeri tırnak işareti karakteriyle önceden düzeltilerek ve çözerek önemli boşluklar içerebilir; açılış tırnak işareti ile eşleşen kapanış tırnak işareti arasındaki tüm karakterler anlamlı olarak değerlendirilir. In the following example, the name is FAMOUS\_WORDS, and the value is Hello World:

```
FAMOUS_WORDS "Hello World"
```



Bir ad ya da deęer, boş karakter dışında ( *RFNVS* için sınırlayıcı olarak hareket eder) dışında herhangi bir karakter içerebilir. Ancak, birlikte çalışabilirliğe yardımcı olmak için, bir uygulamanın adları aşağıdaki karakterlerle kısıtlanmasını tercih edebilir:

- İlk karakter: Büyük harf ya da küçük harf alfabetik (A-Z, a-z) ya da altçizgi.
- Sonraki karakterler: büyük ya da küçük harf alfabetik, ondalık basamak (0-9), altçizgi, kısa çizgi ya da nokta.

Bir ad ya da deęer bir ya da daha çok tırnak işareti içeriyorsa, ad ya da deęer tırnak içine alınmalıdır ve dizedeki her tırnak işareti iki katına çıkarılmalıdır:

```
Famous_Words "The program displayed ""Hello World"""
```

Adlar ve deęerler büyük/küçük harf duyarlıdır, yani küçük harflerin büyük harflerle aynı olması dikkate alınmamaktadır. Örneğin, *FAMOUS\_WORDS* ve *Famous\_Words* iki farklı addır.

*RFNVS* bayt cinsinden uzunluk, *RFLEN* eksi *RFLENV* uzunluęuna eşittir. Bazı ortamlarda kullanıcı verilerinin veri dönüştürmesiyle ilgili sorunları önlemek için, bu uzunluęun dörtlerden biri olması önerilir. *RFNVS*, bu uzunluktaki boşluklarla doldurulmalı ya da dizedeki son anlamlı karakterin ardından boş bir karakter yerleştirilerek daha önce sonlandırılmalıdır. Boş deęer karakteri ve bu karakteri izleyen byte 'lar, belirtilen *RFNVS* uzunluęuna kadar yoksayıdır.

**Not:** Bu alanın uzunluęu deęişmez olduęu için, alan, desteklenen programlama dilleri için saęlanan yapının bildirimlerinden çıkarılır.

#### **RFSID (4 baytlık karakter dizilimi)**

Yapı tanıtıcısı.

Deęer şu olmalıdır:

##### **RFSİDT**

Kurallar ve biçimlendirme üstbilgisi yapısına ilişkin tanıtıcı.

Bu alanın ilk deęeri *RFSIDV* ' dir.

#### **RFVER (10 basamaklı işaretli tamsayı)**

Yapı sürüm numarası.

Deęer şu olmalıdır:

##### **RFVER1**

Version-1 kuralları ve biçimlendirme üstbilgisi yapısı.

Bu alanın ilk deęeri *RFVER1* ' dir.

### **Başlangıçtaki deęerler**

Çizelge 721. <i>MQRFH</i> içindeki alanların ilk deęerleri		
Alan adı	Deęişmezin adı	Deęişmezin deęeri
<i>RFSID</i>	RFSİDT	' RFH¬ '
<i>RFVER</i>	RFVER1	1
<i>RFLEN</i>	RFLENV	32
<i>RFENC</i>	ENNAT	Ortama göre deęişir
<i>RFCSI</i>	CSUNDF	0
<i>RFMT</i>	FMNONE	Boşluklar
<i>RFFLG</i>	YOK	0

Çizelge 721. MQRFH içindeki alanların ilk değerleri (devamı var)

Alan adı	Değişmezin adı	Değişmezin değeri
<b>Notlar:</b>		
1. - simgesi tek bir boş karakteri temsil eder.		

## RPG bildirimi

```
D*..1.....2.....3.....4.....5.....6.....7..
D* MQRFH Structure
D*
D* Structure identifier
D RFSID          1          4      INZ('RFH ')
D* Structure version number
D RFVER          5          8I 0  INZ(1)
D* Total length of MQRFH includingNameValueString
D RFLen          9          12I 0  INZ(32)
D* Numeric encoding of data that followsNameValueString
D RFENC          13         16I 0  INZ(273)
D* Character set identifier of data thatfollows NameValueString
D RFCSI          17         20I 0  INZ(0)
D* Format name of data that followsNameValueString
D RFFMT          21         28      INZ(' ')
D* Flags
D RFFLG          29         32I 0  INZ(0)
```

IBM i

## IBM üzerindeMQRFH2 (Kural ve biçimlendirme üstbilgisi 2)

MQRFH2 yapısı, version-2 kuralları ve biçimlendirme üstbilgisinin biçimini tanımlar.

### Genel Bakış

**Amaç:** Bu üstbilgi, XML benzeri bir sözdizimi kullanılarak kodlanan verileri göndermek için kullanılabilir. Bir ileti, serideki son MQRFH2 yapısının ardından isteğe bağlı olarak, kullanıcı verileri ile seride iki ya da daha fazla MQRFH2 yapısı içerebilir.

**Biçim adı:** FMRFH2.

**Karakter kümesi ve kodlama:** Özel kurallar, MQRFH2 yapısı için kullanılan karakter kümesi ve kodlama için geçerlidir:

- Fields other than *RF2NVD* are in the character set and encoding that are given by the *MDCSI* and *MDENC* fields in the header structure that precedes MQRFH2, or by those fields in the MQMD structure if the MQRFH2 is at the start of the application message data.

Karakter takımı, kuyruk adlarında geçerli olan karakterler için tek byte 'lık karakterler içeren bir karakter takımı olmalıdır.

MQGET çağrısında GMCONV belirtildiğinde, kuyruk yöneticisi bu alanları istenen karakter kümesine ve kodlamaya dönüştürür.

- *RF2NVD*, *RF2NVC* alanı tarafından verilen karakter kümesinde yer alıyor. Only certain Unicode character sets are valid for *RF2NVC* (see the description of *RF2NVC* for details).

Bazı karakter kümelerinin, kodlamaya bağımlı bir gösterimi vardır. *RF2NVC* bu karakter kümelerinden biriye, *RF2NVD*, MQRFH2' deki diğer alanlarla aynı kodlamada olmalıdır.

MQGET çağrısında GMCONV belirtildiğinde, kuyruk yöneticisi *RF2NVD* değerini istenen kodlamaya dönüştürür, ancak karakter kümesini değiştirmez.

- [“Alanlar” sayfa 1175](#)
- [“Başlangıçtaki değerler” sayfa 1179](#)
- [“RPG bildirimi” sayfa 1180](#)

## Alanlar

MQRFH2 yapısı aşağıdaki alanları içerir; alanlar alfabetik sırayla anlatılır:

### RF2CSI (10 basamaklı işaretli tamsayı)

Son *RF2NVD* alanını izleyen verilerin karakter kümesi tanıtıcısı.

Bu, son *RF2NVD* alanını izleyen verilerin karakter kümesi tanıtıcısını belirtir. MQRFH2 yapısının kendisindeki karakter verileri için geçerli değildir.

MQPUT ya da MQPUT1 çağrısında, uygulama bu alanı veri için uygun değere ayarlamalıdır. Aşağıdaki özel değer kullanılabilir:

#### CİNT

Bu yapının karakter kümesi tanıtıcısını devralır.

Character data in the data *takip edilen* this structure is in the same character set as this structure.

Kuyruk yöneticisi, bu değeri, iletide belirtilen yapının gerçek karakter kümesi tanıtıcısına göre değiştirir. Hata oluşmaması durumunda, MQGET çağrısıyla CSINHT değeri döndürülmez.

MQMD ' deki *MDPAT* alanının değeri ATBRKR ise, CSINHT kullanılamaz.

Bu alanın ilk değeri CSINHT ' dir.

### RF2ENC (10 basamaklı işaretli tamsayı)

Son *RF2NVD* alanını izleyen verilerin sayısal kodlaması.

Bu, son *RF2NVD* alanını izleyen verilerin sayısal kodlamasını belirtir; MQRFH2 yapısının kendisindeki sayısal veriler için geçerli değildir.

MQPUT ya da MQPUT1 çağrısında, uygulama bu alanı veri için uygun değere ayarlamalıdır.

Bu alanın ilk değeri ENNAT ' tır.

### RF2FLG (10 basamaklı işaretli tamsayı)

Bayraklar.

Aşağıdaki değer belirtilmelidir:

#### YOK

Bayrak yok.

Bu alanın ilk değeri RFNONE 'tır.

### RF2FMT (8 baytlık karakter dizesi)

Son *RF2NVD* alanını izleyen verilerin biçim adı.

Bu, son *RF2NVD* alanını izleyen verilerin biçim adını belirtir.

MQPUT ya da MQPUT1 çağrısında, uygulama bu alanı veri için uygun değere ayarlamalıdır. Bu alanı kodlamaya ilişkin kurallar, MQMD ' deki *MDFMT* alanı için yapılan kurallarla aynıdır.

Bu alanın ilk değeri FMNONE ' dir.

### RF2LEN (10 basamaklı işaretli tamsayı)

Total length of MQRFH2 including all *RF2NVL* and *RF2NVD* fields.

Bu uzunluk, yapının sonundaki *RF2NVL* ve *RF2NVD* alanları da içinde olmak üzere, MQRFH2 yapısının bayt cinsinden uzunluğudur. Bu, yapının sonunda birden çok sayıda *RF2NVL* ve *RF2NVD* alanı çifti olmak üzere sırayla geçerlidir:

```
length1, data1, length2, data2, ...
```

*RF2LEN* , yapının sonunda son *RF2NVD* alanını izleyebilecek herhangi bir kullanıcı verisi içermez.

Bazı ortamlarda kullanıcı verilerinin veri dönüştürmesiyle ilgili sorunları önlemek için, *RF2LEN* komutunu dört bir kereden fazla kullanmayı düşünün.

Aşağıdaki değişmez, yapının *sabit* bir kısmının uzunluğunu, yani *RF2NVL* ve *RF2NVD* alanları hariç olmak üzere, uzunluğun uzunluğunu verir:

#### **RFLEN2**

MQRFH2 yapısının sabit parçası uzunluğu.

Bu alanın ilk değeri RFLEN2' dir.

#### **RF2NVC (10 basamaklı işaretli tamsayı)**

Character set identifier of *RF2NVD*.

Bu, *RF2NVD* alanındaki verilerin kodlanmış karakter takımı tanıtıcısını belirtir. Bu, MQRFH2 yapısındaki diğer dizelerin karakter takımından farklıdır ve yapının sonundaki son *RF2NVD* alanını izleyen verilerden (varsa) farklı karakter takımından farklı olabilir.

*RF2NVC* , aşağıdaki CCSID değerlerinden birine sahip olmalıdır

#### **1200**

UTF-16, en son Unicode sürümü destekleniyor

#### **13488**

UTF-16, Unicode sürüm 2.0 atkümüsi

#### **17584**

UTF-16, Unicode sürüm 3.0 atkümüsi (Euro simgesini içerir)

#### **1208**

UTF-8, en son Unicode sürümü destekleniyor

UTF-16 karakter takımları için, *RF2NVD* kodlamasının kodlaması (byte sırası), MQRFH2 yapısındaki diğer alanların kodlamasıyla aynı olmalıdır. Vekil karakterler (X'D800'-X'DFFF') desteklenmez.

**Not:** If *RF2NVC* does not have one of the values listed previously, and the MQRFH2 structure requires conversion on the MQGET call, the call completes with reason code RC2111 and the message is returned unconverted.

Bu alanın ilk değeri 1208 'dir.

#### **RF2NVD (n-byte karakter dizgisi)**

Ad/değer verisi.

Bu, XML benzeri bir sözdizimi kullanılarak kodlanmış verileri içeren değişken uzunluklu bir karakter dizilimidir. Bu dizginin bayt cinsinden uzunluğu, *RF2NVD* alanının önünde yer alan *RF2NVL* alanı tarafından sağlanır; bu uzunluk, dörtten çok bir sayı olmalıdır.

*RF2NVL* ve *RF2NVD* alanları isteğe bağlıdır, ancak varsa çift olarak oluşmalıdır ve bitişik olmalıdır. Alan çifti, gerektiği kadar yinelenebilir; örneğin:

```
length1 data1 length2 data2 length3 data3
```

Bu alanlar isteğe bağlı olduğu için, desteklenen çeşitli programlama dilleri için sağlanan yapının bildirimlerinden çıkarılır.

*RF2NVD* is unusual because it is not converted to the character set specified on the MQGET call when the message is retrieved with the GMCONV option in effect; *RF2NVD* remains in its original character set. Ancak, *RF2NVD* , MQGET çağrısında belirlenen kodlamaya dönüştürülür.

**Ad/değer verisi sözdizimi:** Dizgi, sıfır ya da daha fazla özellik içeren tek bir "klasör" inden oluşur. Klasör, XML başlangıç ve bitiş etiketleriyle sınırlanır ve klasörle aynı adı taşıyan son etiketlerle sınırlanır:

```
<folder> property1 property2 ... </folder>
```

Klasör bitiş etiketini izleyen karakterler, *RF2NVL* tarafından tanımlanan uzunluğa kadar boş bırakılmalıdır. Klasör içinde, her özellik bir ad ve bir değerden oluşur ve isteğe bağlı olarak bir veri tipidir:

```
<name dt="datatype">value</name>
```

Bu örneklerde:

- Sınırlayıcı karakterler (<, =, ",/ve >) tam olarak gösterildiği gibi belirlenmelidir.
- name , özelliğin kullanıcı tarafından belirtilen adıdır; adlarla ilgili daha fazla bilgi için aşağıdaki örneğe bakın.
- datatype , özelliğe ilişkin isteğe bağlı bir kullanıcı tanımlı veri tipidir; geçerli veri tipleri için aşağıdaki örneğe bakın.
- value , özelliğin kullanıcı tarafından belirtilen değeridir; değerler hakkında daha fazla bilgi için aşağıdaki paragraflara bakın.
- Bir değerden önce gelen > karakteri ile değeri izleyen < karakteri arasında boşluklar vardır ve en az bir boşluk dt=' den önce gelmelidir. Diğer yerlerde, etiketler arasında ya da önceki ya da izleyen etiketler arasında serbestçe kodlanabilir (örneğin, okunabilirliği artırmak için); bu boşluklar anlamlı değildir.

Özellikler birbiriyle ilgiliyse, bunlar, XML başlangıç ve bitiş etiketleri içinde, grupla aynı adı taşıyan bir araya getirilerek gruplandırılabilir:

```
<folder> <group> property1 property2 ... </group> </folder>
```

Gruplar, sınır olmaksızın diğer gruplar içinde içiçe yerleştirilebilir ve bir grup, bir klasör içinde birden çok kez ortaya çıkabilir. Ayrıca, bir klasör için gruplarda olmayan bazı özellikleri ve gruplarda olmayan diğer özellikleri de içeren bir klasör için geçerlidir.

**Özellikler, gruplar ve klasörler adları:** Özelliklerin, grupların ve klasörlerin adları, bir özellik, grup ya da klasör adında izin verilmeyen iki nokta karakteri dışında geçerli bir XML etiketi adı olmalıdır. Özellikle:

- Adlar bir harfle ya da altçizgiyle başlamalıdır. Geçerli harfler, W3C XML belirtiminde tanımlanır ve temel olarak Unicode kategorileri Ll, Lu, Lo, Lt, ve Nl kategorilerinden oluşur.
- Bir addaki diğer karakterler harfler, ondalık basamaklar, altçizgi, kısa çizgi ya da noktalardan olabilir. Bunlar, Unicode kategorilerinin ll, lu, lo, lt, nl, mc, mn, lm ve nd kategorilerine karşılık gelir.
- Bir adın herhangi bir bölümünde Unicode uyumluluk karakterleri (X'F900' ve üstü) izin verilmez.
- Adlar, üst ya da küçük harflerin herhangi bir karışımında XML dizgisiyle başlamamalıdır.

Buna ek olarak:

- Adlar büyük ve küçük harfe duyarlıdır. Örneğin, ABC, abcve Abc üç farklı addır.
- Her klasörün ayrı bir ad alanı vardır. Sonuç olarak, bir klasördeki bir grup ya da özellik, başka bir klasörde aynı adı taşıyan bir grupla ya da bir özellikle çakışmıyor.
- Gruplar ve özellikler, bir klasör içinde aynı ad alanını kaplar. Sonuç olarak, bir özellik, o özelliği içeren klasör içindeki bir grupla aynı ada sahip olamaz.

Genel olarak, *RF2NVD* alanını çözümlen programlar, bu özelliklerin ya da grupların doğru biçimlendirilmeleri koşuluyla, programın tanımadığı özellikleri ya da adları yoksaymalıdır.

**Özellikler için veri tipleri:** Her özelliğin isteğe bağlı bir veri tipi olabilir. Belirtilirse, veri tipi, aşağıdaki değerlerden biri olmalıdır; üst, küçük harf ya da karışık büyük/küçük harf karışık olarak:

<i>Çizelge 722. Veri tipleri ve kullanımı</i>	
<b>Veri tipi</b>	<b>Kullanılma Amacı</b>
string	Herhangi bir karakter dizisi. Çıkış dizileri kullanılarak belirli karakterler belirtilmelidir.
boolean	Karakter 0 ya da 1 (1, TRUE).
bin.hex	Sekizlileri temsil eden onaltılık basamaklar.
i1	-128-+127 aralığındaki tamsayı sayısı, yalnızca ondalık basamaklar ve isteğe bağlı işaret kullanılarak ifade edilir.
i2	-32 768-+32 767 aralığındaki tamsayı sayısı, yalnızca ondalık basamaklar ve isteğe bağlı işaret kullanılarak ifade edilir.
i4	-2 147 483 648 ile + 2 147 483 647 aralığındaki tamsayı sayısı, yalnızca ondalık basamaklar ve isteğe bağlı işaret kullanılarak ifade edilir.
i8	-9 223 372 036 854 775 808 ile + 9 223 372 036 854 775 807 aralığındaki tamsayı sayısı, sadece ondalık basamakları ve isteğe bağlı işaret kullanılarak ifade edilir.
int	-9 223 372 036 854 775 808 ile + 9 223 372 036 854 775 807 aralığındaki tamsayı sayısı, sadece ondalık basamakları ve isteğe bağlı işaret kullanılarak ifade edilir. This can be used in place of i1, i2, i4, or i8 if the sender does not imply a particular precision.
r4	1.175E-37 ile 3.402 823 47E+38 aralığında bulunan kayan noktalı sayı, ondalık basamaklar, isteğe bağlı işaret, isteğe bağlı kesirli basamaklar ve isteğe bağlı üstel kullanılarak ifade edilir.
r8	Floating-point number with magnitude in the range 2.225E-307 through 1.797 693 134 862 3E+308 expressed using decimal digits, optional sign, optional fractional digits, and optional exponent.

**Özelliklerin değerleri:** Bir özelliğin değeri, zorunlu bir çıkış sırası olan özel karakterler dışında, herhangi bir karakterle oluşabilir. Aşağıdaki tabloda yer alan "zorunlu" olarak işaretlenen bir karakterin değerlerindeki her bir oluşum, ilgili çıkış sırasına göre değiştirilmelidir. Ayrıca, isteğe bağlı olarak ilişkili bir kaçış dizisi olan karakterler de gösterilir. "İsteğe bağlı" olarak işaretlenen bir karakterin değerindeki her bir oluşum, ilgili çıkış sırasına göre değiştirilebilir, ancak bu zorunlu değildir.

<i>Çizelge 723. Çıkış karakteri eklenmiş karakterler ve kullanımları</i>		
<b>Karakter</b>	<b>Çıkış sırası</b>	<b>Kullanım</b>
&	&amp;	Zorunlu
<	<	Zorunlu
>	&gt;	İsteğe Bağlı
"	&quot;	İsteğe Bağlı

Çizelge 723. Çıkış karakteri eklenmiş karakterler ve kullanımları (devamı var)		
Karakter	Çıkış sırası	Kullanım
'	&apos ;	İsteğe Bağlı

**Not:** Çıkış sırasının başlangıcındaki & karakteri, &amp ; ile değiştirilmemelidir.

Aşağıdaki örnekte, değerdeki boşluklar önemlidir; ancak, çıkış sıralarına gerek yoktur:

```
<Famous_Words>The program displayed "Hello World"</Famous_Words>
```

### RF2NVL (10 basamaklı işaretli tamsayı)

*RF2NVD* uzunluğu.

Bu, *RF2NVD* alanındaki verilerin bayt cinsinden uzunluğunu belirtir. *RF2NVD* alanını izleyen verilerin (varsa) veri dönüştürmesiyle ilgili sorunları önlemek için *RF2NVL* , dördünü birden çok kez olmalıdır.

**Not:** *RF2NVL* ve *RF2NVD* alanları isteğe bağlıdır, ancak varsa çift olarak oluşmalıdır ve bitişik olmalıdır. Alan çifti, gerektiği kadar yinelenebilir; örneğin:

```
length1 data1 length2 data2 length3 data3
```

Bu alanlar isteğe bağlı olduğu için, desteklenen çeşitli programlama dilleri için sağlanan yapının bildirimlerinden çıkarılır.

### RF2SID (4 baytlık karakter dizisi)

Yapı tanıtıcısı.

Değer şu olmalıdır:

#### RFSİDT

Kurallar ve biçimlendirme üstbilgisi yapısına ilişkin tanıtıcı.

Bu alanın ilk değeri RFSIDV ' dir.

### RF2VER (10 basamaklı işaretli tamsayı)

Yapı sürüm numarası.

Değer şu olmalıdır:

#### RFVER2

Version-2 kuralları ve biçimlendirme üstbilgisi yapısı.

Bu alanın ilk değeri RFVER2' dir.

## Başlangıçtaki değerler

Çizelge 724. MQRFH2 içindeki alanların ilk değerleri		
Alan adı	Değişmez adı	Değişmez değeri
<i>RF2SID</i>	RFSİDT	' RFH - '
<i>RF2VER</i>	RFVER2	2
<i>RF2LEN</i>	RFLN2	36
<i>RF2ENC</i>	ENNAT	Ortama göre değişir
<i>RF2CSI</i>	CİNT	-2
<i>RF2FMT</i>	FMNONE	Boşluklar

Çizelge 724. MQRFH2içindeki alanların ilk değerleri (devamı var)

Alan adı	Değişmezin adı	Değişmezin değeri
RF2FLG	YOK	0
RF2NVC	Yok	1208

#### Notlar:

1. - simgesi tek bir boş karakteri temsil eder.

#### RPG bildirim

```
D*..1.....2.....3.....4.....5.....6.....7..
D*
D* MQRFH2 Structure
D*
D* Structure identifier
D RF2SID          1          4      INZ('RFH ')
D* Structure version number
D RF2VER          5          8I 0 INZ(2)
D* Total length of MQRFH2 including allNameValueLength and
D* NameValueDatafields
D RF2LEN          9          12I 0 INZ(36)
D* Numeric encoding of data that followslast NameValueData field
D RF2ENC          13         16I 0 INZ(273)
D* Character set identifier of data thatfollows last NameValueData field
D RF2CSI          17         20I 0 INZ(-2)
D* Format name of data that follows lastNameValueData field
D RF2FMT          21         28      INZ(' ')
D* Flags
D RF2FLG          29         32I 0 INZ(0)
D* Character set identifier ofNameValueData
D RF2NVC          33         36I 0 INZ(1208)
```

IBM i

#### IBM üzerinde MQRMH (Başvuru iletisi üstbilgisi)

MQRMH yapısı, başvuru iletisi üstbilgisinin biçimini tanımlar.

#### Genel Bakış

**Amaç:** Bu üstbilgi, büyük miktarda veri göndermek için kullanıcı tarafından yazılan ileti kanalı çıkışlarıyla kullanılır ("toplu veri" olarak adlandırılır). bir kuyruk yöneticisinden diğerine. Normal ileti sistemine göre fark, toplu verilerin bir kuyruğun üzerinde saklanmaması; bunun yerine, kuyruğun üzerinde yalnızca bir *başvuru* 'nin toplu verileri saklanmasıdır. Bu, IBM MQ kaynaklarının birkaç büyük ileti tarafından tüketilmesinin olasılığını azaltır.

**Biçim adı:** FMRMH.

**Karakter kümesi ve kodlama:** MQRMH 'de karakter verileri ve görel konum alanları tarafından adreslenen dizgiler, yerel kuyruk yöneticisinin karakter kümesinde olmalıdır; bu, **CodedCharSetId** kuyruk yöneticisi özniteliği tarafından verilir. MQRMH 'deki sayısal veriler, yerel makine kodlamasında olmalıdır; bu, C programlama dili için ENNAT değeri tarafından verilir.

MQRMH 'nin karakter takımı ve kodlaması aşağıdaki gibi *MDCSI* ve *MDENC* alanlarında ayarlanmalıdır:

- MQMD (ileti verilerinin başlangıcındaki MQRMH yapısı ise) ya da
- MQRMH yapısından önce gelen üstbilgi yapısı (diğer tüm vakalar).

**Kullanım:** Bir uygulama, bir MQRMH 'den oluşan bir iletiyi koyar, ancak toplu verileri atlayarak bir ileti oluşturur. İleti kanal aracısı (MCA) tarafından iletim kuyruğundan okunduğunda, başvuru iletisi üstbilgisini işlemek için kullanıcı tarafından sağlanan bir ileti çıkışı çağrılır. Çıkış, MQRMH yapısı tarafından tanımlanan toplu verileri, MCA iletiyi kanaldan bir sonraki kuyruk yöneticisine göndermeden önce, başvuru iletisine sonuna eklenebilir.



Alıcı uçta, başvuru iletileri için bekleyen bir ileti çıkışı var olmalıdır. Bir başvuru iletileri alındığında, çıkış, iletteki MQRMH ' yi izleyen toplu verilerden nesneyi yaratmalı ve daha sonra, toplu veri olmadan başvuru iletisine geçmelidir. Başvuru iletileri daha sonra, bir kuyruktan başvuru iletileri (toplu veriler olmadan) okunarak bir uygulama tarafından alınabilir.

Olağan durumda, MQRMH yapısı iletelerde yer alan tüm yapıdır. Ancak, ileti bir iletim kuyruğalıysa, bir ya da daha çok ek üstbilgi MQRMH yapısından önce gelir.

Ayrıca, bir dağıtım listesine bir başvuru iletileri de gönderilebilir. Bu durumda, MQDH yapısı ve ilgili kayıtları, ileti bir iletim kuyruğunda olduğunda MQRMH yapısından önce gelir.

**Not:** İleti çıkışı doğru işleyemediğinden, bölümlenmiş bir ileti olarak bir başvuru iletileri gönderilmemelidir.

- “Veri dönüştürme” sayfa 1181
- “Alanlar” sayfa 1181
- “Başlangıçtaki değerler” sayfa 1185
- “RPG bildirim” sayfa 1186

## Veri dönüştürme

Veri dönüştürme işlemlerinde, MQRMH yapısının dönüştürülmesi, kaynak ortam verilerinin, kaynak nesne adının, hedef ortam verilerinin ve hedef nesne adının dönüştürülmesini içerir. Yapının başlangıcındaki *RMLen* byte 'ındaki diğer baytlar atılır ya da veri dönüştürmesinden sonra tanımsız değerler içerilir. Toplu veriler, aşağıdaki deyimlerin tümünün doğru olması koşuluyla dönüştürülecek:

- Toplu veriler, veri dönüştürme işlemi gerçekleştirildiğinde iletelerde bulunur.
- MQRMH ' deki *RMFMT* alanının değeri *FMNONE* dışında bir değer içeriyor.
- Kullanıcı tarafından yazılan bir veri dönüştürme çıkışı, belirtilen biçim adıyla var olur.

Ancak, ileti kuyrukta olduğunda genellikle toplu veri olmadığından ve sonuç olarak toplu verilerin *GMCONV* seçeneği tarafından dönüştürülemeyeceği konusunda dikkatli olun.

## Alanlar

MQRMH yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

### RMCSI (10 basamaklı işaretli tamsayı)

Toplu verilerin karakter kümesi tanıtıcısı.

Bu, toplu verilerin karakter kümesi tanıtıcısını belirtir; MQRMH yapısındaki karakter verileri için geçerli değildir.

*MQPUT* ya da *MQPUT1* çağrısında, uygulama bu alanı veri için uygun değere ayarlamalıdır. Aşağıdaki özel değer kullanılabilir:

#### CINT

Bu yapının karakter kümesi tanıtıcısını devralır.

Character data in the data *takip edilen* this structure is in the same character set as this structure.

Kuyruk yöneticisi, bu değeri, iletide belirtilen yapının gerçek karakter kümesi tanıtıcısına göre değiştirir. Hata oluşmaması durumunda, *MQGET* çağrısıyla *CSINHT* değeri döndürülmez.

*MQMD* ' deki *MDPAT* alanının değeri *ATBRKR* ise, *CSINHT* kullanılamaz.

Bu alanın ilk değeri *CSUNDF* 'dir.

### RMDEL (10 basamaklı işaretli tamsayı)

Hedef ortam verisinin uzunluğu.

Bu alan sıfırsa, hedef ortam verileri yoktur ve *RMDEO* yoksayıdır.

### **RMDEO (10 basamaklı işaretli tamsayı)**

Hedef ortam verilerinin görelî konumu.

Bu alan, MQRMH yapısının başlangıcından hedef ortam verilerinin kayma değerini belirtir. Hedef ortam verileri, bu veriler yaratan kişi tarafından biliniyorsa, başvuru iletisinin yaratıcısı tarafından belirtilebilir. Örneğin, hedef ortam verileri, toplu verilerin saklanabileceği nesne dizin yolu olabilir. Ancak, yaratan hedef ortam verilerini bilmiyorsa, gereken ortam bilgilerini belirlemek için kullanıcı tarafından sağlanan ileti çıkışının sorumluluğundadır.

Hedef ortam verilerinin uzunluğu *RMDEL* tarafından verilir; Bu uzunluk sıfırsa, hedef ortam verileri yoktur ve *RMDEO* yoksayılr. Bu durumda, hedef ortam verilerinin, yapının başlangıcından itibaren *RMLLEN* bayt içinde tamamen bulunması gerekir.

Uygulamalar, hedef ortam verilerinin *RMSEO*, *RMSNOve* *RMDNO* alanları tarafından adreslenen herhangi bir veriyle bitişik olduğunu varsaymamalıdır.

Bu alanın ilk değeri 0 'tır.

### **RMDL (10 basamaklı işaretli tamsayı)**

Toplu verilerin uzunluğu.

*RMDL* alanı, MQRMH yapısının gönderme yaptığı toplu verilerin uzunluğunu belirtir.

İletide toplu veri varsa, veriler, MQRMH yapısının başlangıcındaki *RMLLEN* byte 'tan itibaren bir görelî konum olarak başlar. Tüm ileti eksi *RMLLEN* uzunluğunun uzunluğu, toplu veri uzunluğunun uzunluğunu gösterir.

İletide veri varsa, *RMDL* , ilgili verilerin miktarını belirtir. Olağan durumda, *RMDL* için, iletide bulunan veri uzunluğunun aynı değere sahip olması gerekir.

MQRMH yapısı nesnedeki kalan verileri temsil ediyorsa (belirtilen mantıksal görelî konumdan başlayarak), iletide toplu veri yoksa, *RMDL* değeri için sıfır değeri kullanılabilir.

Veri yoksa, MQRMH ' nin sonu iletinin sonuna denk gelir.

Bu alanın ilk değeri 0 'tır.

### **RMDNL (imzalanmış 10 basamaklı tamsayı)**

Hedef nesne adı uzunluğu.

Bu alan sıfırsa, hedef nesne adı yoktur ve *RMDNO* yoksayılr.

### **RMDNO (10 basamaklı işaretli tamsayı)**

Hedef nesne adının görelî konumu.

Bu alan, MQRMH yapısının başlangıcından hedef nesne adının kayma değerini belirtir. Hedef nesne adı, bu veriler oluşturucuda biliniyorsa, başvuru iletisinin yaratıcısı tarafından belirtilebilir. Ancak, yaratan hedef nesne adını bilmiyorsa, yaratılacak ya da değiştirilecek nesneyi tanımlamak için kullanıcı tarafından sağlanan ileti çıkışının sorumluluğundadır.

Hedef nesne adının uzunluğu *RMDNL* tarafından verilir; Bu uzunluk sıfırsa, hedef nesne adı yoktur ve *RMDNO* yoksayılr. Varsa, hedef nesne adının, yapının başlangıcından itibaren *RMLLEN* bayt içinde tamamen bulunması gerekir.

Uygulamalar, hedef nesne adının *RMSEO*, *RMSNOve* *RMDEO* alanları tarafından adreslenen herhangi bir veriyle bitişik olduğunu varsaymamalıdır.

Bu alanın ilk değeri 0 'tır.

### **RMDO (10 basamaklı işaretli tamsayı)**

Toplu verilerin düşük görelî konumu.

Bu alan, toplu veri formlarının parçası olan nesnenin başlangıcından elde edilen toplu verilerin alt görelî konumunu belirtir. Nesnenin başlangıcından elde edilen toplu verilerin görelî konumu, *mantıksal*

*görelî konum* olarak adlandırılır. Bu, MQRMH yapısının başlangıcından elde edilen toplu verilerin fiziksel görelî konumu değildir; görelî konum *RMLEN* tarafından verilir.

Büyük nesnelerin başvuru iletileri kullanılarak gönderilmesine izin vermek için, mantıksal görelî konum iki alana ayrılır ve gerçek mantıksal görelî konum, bu iki alanın toplamını verir:

- *RMDO* , mantıksal görelî konum 1 000 000 000 olarak bölündüğünde elde edilen kalanı temsil eder. Böylece, 0 ile 999 999 999 aralığında bir değer vardır.
- *RMDO2* , mantıksal görelî konum 1 000 000 000 olarak bölündüğünde elde edilen sonucu temsil eder. Bu, mantıksal kayma içinde var olan 1 000 000 000 'in tam katının sayısıdır. Birden çok sayı, 0 ile 999 999 999 arasında bir aralıktaki sayıdır.

Bu alanın ilk değeri 0 'tır.

#### **RMDO2 (10 basamaklı işaretli tamsayı)**

Toplu verilerin yüksek görelî konumu.

Bu alan, toplu veri formlarının parçası olan nesnenin başlangıcından elde edilen toplu verilerin yüksek görelî konum değerini belirtir. Bu değer, 0 ile 999 999 999 arasındaki bir aralıktaki bir değerdir. Ayrıntılar için bkz. *RMDO* .

Bu alanın ilk değeri 0 'tır.

#### **RMENC (10 basamaklı işaretli tamsayı)**

Toplu verilerin sayısal kodlaması.

Bu, toplu verilerin sayısal kodlamasını belirtir; MQRMH yapısındaki sayısal veriler için geçerli değildir.

MQPUT ya da MQPUT1 çağrısında, uygulama bu alanı veri için uygun değere ayarlamalıdır.

Bu alanın ilk değeri ENNAT ' tır.

#### **RMFLG (10 basamaklı işaretli tamsayı)**

Başvuru ileti işaretleri.

Aşağıdaki işaretler tanımlanır:

##### **GEÇERLİK**

Başvuru ileti nesnenin son bölümünü içerir ya da gösterir.

Bu işaret, başvuru iletinin başvuruda bulunan nesnenin son bölümünü temsil ettiğini ya da içerdiğini gösterir.

##### **RMSNLST**

Başvuru ileti nesnenin son bölümünü içermiyor ya da göstermiyor.

RMNLST, program belgelerine yardımcı olmak için tanımlıdır. Bu seçeneğin başka biriyle birlikte kullanılması amaçlanmaz, ancak değeri sıfır olduğu için bu tür bir kullanım saptanamaz.

Bu alanın ilk değeri RMNLST ' dir.

#### **RMFMT (8 baytlık karakter dizilimi)**

Toplu verilerin adı biçimi.

Bu, toplu verilerin biçim adını belirtir.

MQPUT ya da MQPUT1 çağrısında, uygulama bu alanı veri için uygun değere ayarlamalıdır. Bu alanı kodlamaya ilişkin kurallar, MQMD ' deki *MDFMT* alanı için yapılan kurallarla aynıdır.

Bu alanın ilk değeri FMNONE ' dir.

#### **RMLEN (10 basamaklı işaretli tamsayı)**

Sabit alanların sonunda dizgiler de dahil olmak üzere MQRMH ' nin toplam uzunluğu, ancak toplu veriler değil.

Bu alanın ilk değeri sıfır.

### **RMOII (24 baytlık bit dizesi)**

Nesne eşgörünümü tanıtıcısı.

Bu alan, bir nesneye ilişkin belirli bir eşgörünümü tanımlamak için kullanılabilir. Gerekli değilse, aşağıdaki değer olarak ayarlanmalıdır:

#### **OIINON**

Nesne eşgörünümü tanıtıcısı belirtilmedi.

Değer, alanın uzunluğuna ilişkin ikili sıfır değeridir.

Bu alanın uzunluğu LNOIID tarafından verilir. Bu alanın ilk değeri OIINON 'tır.

### **RMOT (8 baytlık karakter dizilimi)**

Nesne tipi.

Bu ad, ileti çıkışı tarafından, desteklediği başvuru iletisi tiplerini tanımak için kullanılacak bir addir. Adın, *RMFMT* alanıyla aynı kurallara uygun olmasını dikkate alın.

Bu alanın ilk değeri 8 boşluktur.

### **RMSEL (10 basamaklı işaretli tamsayı)**

Kaynak ortam verilerinin uzunluğu.

Bu alan sıfırsa, kaynak ortam verileri yoktur ve *RMSEO* yoksayılr.

Bu alanın ilk değeri 0 'tır.

### **RMSEO (10 basamaklı işaretli tamsayı)**

Kaynak ortam verilerinin görelî konumu.

Bu alan, MQRMH yapısının başlangıcından kaynak ortam verilerinin kayma değerini belirtir. Kaynak ortam verileri, bu veriler yaratan kişi tarafından biliniyorsa, başvuru iletisinin yaratıcısı tarafından belirtilebilir. Örneğin, kaynak ortam verileri, toplu verileri içeren nesnenin dizin yolu olabilir. Ancak, yaratan kaynak ortam verilerini bilmiyorsa, gereken ortam bilgilerini belirlemek için kullanıcı tarafından sağlanan ileti çıkışının sorumluluğundadır.

Kaynak ortam verilerinin uzunluğu *RMSEL* tarafından verilir; Bu uzunluk sıfırsa, kaynak ortam verileri yoktur ve *RMSEO* yoksayılr. Bu durumda kaynak ortam verileri, yapının başlangıcından itibaren *RMLEN* bayt içinde tamamen bulunmalıdır.

Uygulamalar, yapıdaki son sabit alandan hemen sonra ortam verilerinin başlatıldığını ya da *RMSNO*, *RMDEO* ve *RMDNO* alanları tarafından adreslenen verilerin herhangi biriyle bitişik olduğunu varsaymamalıdır.

Bu alanın ilk değeri 0 'tır.

### **RMSID (4 baytlık karakter dizilimi)**

Yapı tanıtıcısı.

Değer şu olmalıdır:

#### **RMSIDR**

Başvuru iletisi üstbilgi yapısına ilişkin tanıtıcı.

Bu alanın ilk değeri RMSIDV ' dir.

### **RMSNL (10 basamaklı işaretli tamsayı)**

Kaynak nesne adı uzunluğu.

Bu alan sıfırsa, kaynak nesne adı yoktur ve *RMSNO* yoksayılr.

Bu alanın ilk değeri 0 'tır.

### **RMSNO (10 basamaklı işaretli tamsayı)**

Kaynak nesne adının görelî konumu.

Bu alan, MQRMH yapısının başlangıcından kaynak nesne adının kayma değerini belirtir. Kaynak nesne adı, bu veriler oluşturucuda biliniyorsa, başvuru iletisinin yaratıcısı tarafından belirtilebilir. Ancak, yaratan kaynak nesne adını bilmiyorsa, bu, erişilecek nesneyi tanımlamak için kullanıcı tarafından sağlanan ileti çıkışının sorumluluğundadır.

Kaynak nesne adının uzunluğu *RMSNL* tarafından verilir; Bu uzunluk sıfırsa, kaynak nesne adı yoktur ve *RMSNO* yoksayılır. If present, the source object name must reside completely within *RMLLEN* bytes from the start of the structure.

Uygulamalar, kaynak nesne adının *RMSEO*, *RMDEO* ve *RMDNO* alanları tarafından adreslenen herhangi bir veriyle bitişik olduğunu varsaymamalıdır.

Bu alanın ilk değeri 0 'tır.

### **RMVER (10 basamaklı işaretli tamsayı)**

Yapı sürüm numarası.

Değer şu olmalıdır:

#### **RMVER1**

Version-1 başvuru iletisi üstbilgi yapısı.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

#### **RMVERC**

Başvuru iletisi üstbilgi yapısının yürürlükteki sürümü.

Bu alanın ilk değeri *RMVER1* 'dir.

### **Başlangıçtaki değerler**

Çizelge 725. MQRMH 'deki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
<i>RMSID</i>	RMSIDR	'RMH→'
<i>RMVER</i>	RMVER1	1
<i>RMLLEN</i>	Yok	0
<i>RMENC</i>	ENNAT	Ortama göre değişir
<i>RMCSI</i>	CSUNDF	0
<i>RMFMT</i>	FMNONE	Boşluklar
<i>RMFLG</i>	RMSNLST	0
<i>RMOT</i>	Yok	Boşluklar
<i>RMOII</i>	OIINON	Boş Değerler
<i>RMSEL</i>	Yok	0
<i>RMSEO</i>	Yok	0
<i>RMSNL</i>	Yok	0
<i>RMSNO</i>	Yok	0
<i>RMDEL</i>	Yok	0
<i>RMDEO</i>	Yok	0
<i>RMDNL</i>	Yok	0
<i>RMDNO</i>	Yok	0

Çizelge 725. MQRMH 'deki alanların ilk değerleri (devamı var)

Alan adı	Değişmezin adı	Değişmezin değeri
RMDL	Yok	0
RMDO	Yok	0
RMDO2	Yok	0

**Notlar:**

1. - simgesi tek bir boş karakteri temsil eder.

```
D*..1.....2.....3.....4.....5.....6.....7..
D*
D* MQRMH Structure
D*
D* Structure identifier
D RMSID          1      4  INZ('RMH ')
D* Structure version number
D RMVER          5      8I 0 INZ(1)
D* Total length of MQRMH, including strings at end of fixed fields, but not
D* the bulk data
D RMLen          9      12I 0 INZ(0)
D* Numeric encoding of bulk data
D RMENC          13     16I 0 INZ(273)
D* Character set identifier of bulkdata
D RMCSI          17     20I 0 INZ(0)
D* Format name of bulk data
D RMFMT          21     28  INZ('      ')
D* Reference message flags
D RMFLG          29     32I 0 INZ(0)
D* Object type
D RMOT           33     40  INZ
D* Object instance identifier
D RMOII          41     64  INZ(X'00000000000000-
D                    00000000000000000000-
D                    000000000000')
D* Length of source environmentdata
D RMSEL          65     68I 0 INZ(0)
D* Offset of source environmentdata
D RMSEO          69     72I 0 INZ(0)
D* Length of source object name
D RMSNL          73     76I 0 INZ(0)
D* Offset of source object name
D RMSNO          77     80I 0 INZ(0)
D* Length of destination environmentdata
D RMDL           81     84I 0 INZ(0)
D* Offset of destination environmentdata
D RMDEO          85     88I 0 INZ(0)
D* Length of destination objectname
D RMDNL          89     92I 0 INZ(0)
D* Offset of destination objectname
D RMDNO          93     96I 0 INZ(0)
D* Length of bulk data
D RMDL           97     100I 0 INZ(0)
D* Low offset of bulk data
D RMDO           101    104I 0 INZ(0)
D* High offset of bulk data
D RMDO2          105    108I 0 INZ(0)
```

## RPG bildirim

### IBM i IBM üzerinde MQR (Yanıt kaydı)

MQR yapısı, hedef bir dağıtım listesi olduğunda, tek bir hedef kuyruk için açma ya da koyma işleminden kaynaklanan tamamlanma kodunu ve neden kodunu almak için kullanılır.

## Genel Bakış

**Amaç:** MQRR, MQOPEN, MQPUT ve MQPUT1 çağrılarını için bir çıkış yapısıdır.

**Karakter kümesi ve kodlama:** MQRR ' deki veriler, ENNAT tarafından verilen yerel kuyruk yöneticisinin **CodedCharSetId** kuyruk yöneticisi özneliği ve kodlaması tarafından verilen karakter kümesinde olmalıdır. Ancak, uygulama bir IBM MQ istemcisi olarak çalışıyorsa, yapı, istemcinin karakter kümesinde ve kodlamasında olmalıdır.

**Kullanım:** MQOPER ve MQPUT çağrılarında ya da MQPUT1 çağrısında bu yapılardan oluşan bir dizi sağlayarak, çağrının sonucu karma olduğunda bir dağıtım listesindeki tüm kuyruklara ilişkin tamamlanma kodları ve neden kodlarının belirlenmesi mümkündür. Bu durumda, çağrı listedeki bazı kuyruklar için başarılı olur, ancak diğerleri için başarısız olur. Çağrıdan RC2136 neden kodu, (uygulama tarafından sağlandıysa) yanıt kayıtlarının kuyruk yöneticisi tarafından ayarlandığını gösterir.

- “Alanlar” sayfa 1187
- “Başlangıçtaki değerler” sayfa 1187
- “RPG bildirim” sayfa 1187

## Alanlar

MQRR yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

### RRCC (imzalanmış 10 basamaklı tamsayı)

Kuyruğa ilişkin tamamlanma kodu.

Bu, MQOPED ya da MQPUT1 çağrısında bulunan MQOR yapılarının dizideki ilgili öge tarafından belirtilen ada ilişkin açma ya da koyma işleminin sonucundaki tamamlanma kodudur.

Bu her zaman bir çıkış alanıdır. Bu alanın ilk değeri CCOK 'tır.

### RRREA (10 basamaklı işaretli tamsayı)

Kuyruğun neden kodu.

Bu, MQOPED ya da MQPUT1 çağrısında bulunan MQOR yapılarının dizideki ilgili öge tarafından belirtilen ada ilişkin açma ya da koyma işleminin sonucundaki açık ya da koyma işleminin neden kodudur.

Bu her zaman bir çıkış alanıdır. Bu alanın ilk değeri RCNONE olur.

## Başlangıçtaki değerler

Çizelge 726. MQRR ' deki alanların ilk değerleri		
Alan adı	Değişmez adı	Değişmez değeri
RRCC	CCOK	0
RRREA	YOK	0

## RPG bildirim

```
D*..1.....2.....3.....4.....5.....6.....7..
D*
D* MQRR Structure
D*
D* Completion code for queue
D RRCC          1      4I 0 INZ(0)
D* Reason code for queue
D RRREA         5      8I 0 INZ(0)
```

MQSCO yapısı (MQCD yapısındaki TLS alanlarıyla birlikte), kanal protokolü TCP/IP olduğunda istemci bağlantısı için TLS kullanımını denetleyen yapılandırma seçeneklerini belirlemek için IBM MQ MQI client olarak çalışan bir uygulamanın kullanılmasını sağlar.

## Genel Bakış

**Amaç:** Yapı, MQCONNX çağrısına ilişkin bir giriş parametresidir.

İstemci kanalına ilişkin kanal protokolü TCP/IP değilse, MQSCO yapısı yoksayıdır.

**Karakter kümesi ve kodlama:** MQSCO 'daki veriler, ENNAT tarafından verilen yerel kuyruk yöneticisinin **CodedCharSetId** kuyruk yöneticisi özniteliği ve kodlaması tarafından verilen karakter kümesinde olmalıdır.

- [“Alanlar” sayfa 1188](#)
- [“Başlangıçtaki değerler” sayfa 1191](#)
- [“RPG bildirim” sayfa 1192](#)

## Alanlar

MQSCO yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

### SCAIC (10 basamaklı işaretli tamsayı)

Bu, SCAIP ya da SCAIO alanları tarafından adreslenen kimlik doğrulama bilgileri (MQAIR) kayıtlarından biri. Daha fazla bilgi için bkz. [“IBM üzerinde MQAIR \(Kimlik doğrulama bilgileri kaydı\)” sayfa 999](#). Değer sıfır ya da daha büyük olmalıdır. Değer geçerli değilse, arama başarısız olur ve neden kodu RC2383 ile başarısız olur.

Bu bir giriş alanıdır. Bu alanın ilk değeri 0 'tır.

### SCAIO (10 basamaklı işaretli tamsayı)

Bu, MQSCO yapısının başlangıcındaki ilk kimlik doğrulama bilgileri kaydının bayt cinsinden görel konutdur. Görel konum pozitif ya da negatif olabilir. SCAIC sıfırda, alan yoksayıdır.

MQAIR kayıtlarını belirtmek için SCAIO ya da SCAIP değerini kullanabilirsiniz, ancak her ikisini birden belirleyemez; ayrıntılar için SCAIP alanının tanımına bakın.

Bu bir giriş alanıdır. Bu alanın ilk değeri 0 'tır.

### SCAIP (10 basamaklı işaretli tamsayı)

Bu, ilk kimlik doğrulama bilgileri kaydının adresidir. SCAIC sıfırda, alan yoksayıdır.

MQAIR kayıtları dizisini aşağıdaki iki yoldan biriyle sağlayabilirsiniz:

- SCAIP işaretçi alanını kullanarak

Bu durumda uygulama, MQSCO yapısından ayrı olan bir MQAIR kaydı dizisini bildirebilir ve SCAIP 'ı dizinin adresine ayarlayabilir.

Farklı ortamlara (örneğin, C programlama dili gibi) taşınabilir bir şekilde işaretçi veri tipini destekleyen programlama dilleri için SCAIP 'ı kullanmayı düşünün.

- By using the offset field SCAIO

Bu durumda, uygulama MQAIR kayıtları dizisini izleyen bir MQSCO içeren bir bileşik yapı bildirmeli ve SCAIO değerini, MQSCO yapısının başlangıcındaki dizideki ilk kaydın kayma değerine ayarlamalıdır. Bu değer doğru olduğundan ve bir MQlong (en kısıtlayıcı programlama dili COBOL) içinde konaklayabilecek bir değere sahip olduğundan emin olun (geçerli aralık -999 999 999 ile +999 999 999 aralığında).



İşaretçi veri tipini desteklemeyen programlama dilleri için *SCAIO* komutunu kullanmayı ya da işaretçi veri tipini farklı ortamlara (örneğin, COBOL programlama dili gibi) portatif olmayan bir şekilde uygulamayı düşünün.

Hangi teknik seçiyorsanız, *SCAIP* ve *SCAIO* ' den yalnızca biri kullanılabilir; her ikisi de sıfır değilse, çağrı neden kodu RC2384 ile başarısız olur.

Bu bir giriş alanıdır. Bu alanın ilk değeri, göstergeleri destekleyen programlama dillerindeki boş değerli göstergedir ve tersi durumda, boş değerli byte dizilimidir.

**Not:** Programlama dilinin işaretçi veri tipini desteklemediği altyapılarda, bu alan uygun uzunluğun byte dizgisi olarak bildirilir.

### **SCCERLBL (10 basamaklı işaretli tamsayı)**

Bu alan, kullanılmakta olan sertifika etiketinin ayrıntılarını verir.

IBM MQ , SCCERLBL alanının değerini boşluk olarak başlatır. Gerekli değeri girin ya da varsayılan değeri kabul edin.

*ibmwebspheremquser\_id* is a valid value for this field for all versions of the product, and for MQSCO versions less than 5.0 it is the only valid value. Bu nedenle, bu alanın değeri çalıştırma zamanında yorumlanır ve gerekirse değiştirilir. 5.0değerinden küçük bir MQSCO sürümü belirtirseniz ya da SCCERLBL alanı için varsayılan boşluk değerini kabul ederseniz, sistem *ibmwebspheremquser\_id*değerini kullanır.

Bu bir giriş alanıdır.

### **SCCERTVPOL (10 basamaklı işaretli tamsayı)**

Bu alan, kullanılacak sertifika doğrulama ilkesinin tipini belirtir. Alan, aşağıdaki değerlerden birine ayarlanabilir:

#### **MQ\_CERT\_VAL\_POLICY\_ANY**

Güvenli yuva kitaplığı tarafından desteklenen sertifika geçerlilik denetimi ilkelerinin her birini uygulayın. Sertifika zincirini geçerli olarak gören ilkelerden biri varsa, sertifika zincirini kabul edin.

#### **MQ\_CERT\_VAL\_POLICY\_RFC5280**

Yalnızca RFC5280 uyumlu sertifika geçerlilik denetimi ilkesini uygulayın. Bu ayar ANY ayarından daha katı geçerlilik denetimi sağlar, ancak bazı eski dijital sertifikaları reddeder.

Bu alanın ilk değeri MQ\_CERT\_VAL\_POLICY\_ANY olarak geçerli olur.

### **SCCH (10 basamaklı işaretli tamsayı)**

Bu alan, istemci sistemine bağlı olan şifreleme donanımları için yapılandırma ayrıntılarını verir.

Alanı, aşağıdaki biçimde bir dizgiye ayarlayın ya da boş ya da boş bırakın:

```
GSK_PKCS11=the PKCS #11 driver path and file name;the PKCS #11 token label;the PKCS #11 token password;symmetric cipher setting>;
```

To use cryptographic hardware which conforms to the PKCS11 interface, for example, the IBM 4960 or IBM 4963, specify the PKCS11 driver path, PKCS11 token label, and PKCS11 token password strings, each terminated by a semi-colon.

PKCS #11 sürücüsü yolu, PKCS #11 kartı için destek sağlayan paylaşılan kitaplık için mutlak bir yoldur. PKCS #11 sürücüsü dosya adı, paylaşılan kitaplığın adıdır. PKCS #11 yolu ve dosya adı için gereken değer örneği:

```
/usr/lib/pkcs11/PKCS11_API.so
```

PKCStokenbelirteci etiketi tamamen küçük harfli olmalıdır. Donanımınızı büyük ya da büyük harfli bir simge etiketiyle yapılandırdıysanız, bu etiketi bu küçük harfli etiketle yeniden yapılandırın.

Herhangi bir şifreleme donanımı yapılandırması gerekmiyorsa, alanı boş ya da boş olarak ayarlayın.

Değer, alanın uzunluğundan kısaysa, değeri boş değerli bir karakterle sonlandırın ya da alanın uzunluğuna göre boşluklarla doldurur. Değer geçerli değilse ya da şifreleme donanımını yapılandırmak için kullanılan bir hataya yol açarsa, çağrı neden kodu RC2382 ile başarısız olur.

Bu bir giriş alanıdır. Bu alanın uzunluğu LNSSCH tarafından verilir. Bu alanın ilk değeri boşluk karakteridir.

### **AQUASSöiteb (10 basamaklı işaretli tamsayı)**

Bu alan, Suite B uyumlu şifrelemesinin kullanılıp kullanılmadığını ve hangi güç düzeyinin çalıştırılacağını belirtir. Değer bir ya da daha fazla olabilir:

- SCEPSUITEB0  
Suite B uyumlu şifreleme kullanılmıyor.
- SCEPSUITEB1  
B Suite 128 bit güvenlik düzeyi güvenliği kullanılır.
- SCEPSUITEB2  
B Grubu 192 bit güvenlik düzeyi güvenliği kullanılır.

**Not:** Bu alandaki diğer herhangi bir değerle SCEPSUITEB0 değerinin kullanılması geçersizdir.

### **SCFR (10 basamaklı işaretli tamsayı)**

IBM MQ , şifreleme donanımıyla, kullanılan şifreleme modüllerinin donanım ürünü tarafından sağlananlar olması için yapılandırılabilir; bunlar, kullanılmakta olan şifreleme donanımı ürününe bağlı olarak, belirli bir düzeye FIPS sertifikası almış olabilir.

Use this field to specify that only FIPS-certified algorithms are used if the cryptography is provided in IBM MQ-provided software.

IBM MQ kurulu olduğunda, bazı FIPS onaylı modüller sağlayan TLS şifreleme uygulaması da kurulur.

Değerler şöyle olabilir:

#### **MQSSL\_FIPS\_NO**

Bu varsayılan değerdir. Bu değere ayarlandığında:

- Belirli bir platform üzerinde desteklenen herhangi bir CipherSpec kullanılabilir.
- Şifreleme donanımı kullanılmadan çalıştırılırsa, aşağıdaki CipherSpecs , IBM MQ platformlarında FIPS 140-2 sertifikalı şifrelemeyi kullanarak çalışır:
  - TLS\_RSA\_WITH\_3DES\_EDE\_CBC\_SHA
  - TLS\_RSA\_WITH\_AES\_128\_CBC\_SHA
  - TLS\_RSA\_WITH\_AES\_256\_CBC\_SHA

#### **MQSSL\_FIPS\_YES**

Bu değere ayarlandığında, şifrelemeyi gerçekleştirmek için şifreleme donanımını kullanmıyorsanız, bu değer için

- Bu istemci bağlantısına uygulanan CipherSpec ' de yalnızca FIPS onaylı şifreleme algoritmaları kullanılabilir.
- Gelen ve giden TLS kanalı bağlantıları yalnızca aşağıdaki şifre belirteimlerinden biri kullanılırsa başarılı olur:
  - TLS\_RSA\_WITH\_3DES\_EDE\_CBC\_SHA
  - TLS\_RSA\_WITH\_AES\_128\_CBC\_SHA
  - TLS\_RSA\_WITH\_AES\_256\_CBC\_SHA

#### **Notlar:**

1. CipherSpec TLS\_RSA\_WITH\_3DES\_EDE\_CBC\_SHA kullanımdan kaldırıldı.
2. Mümkün olduğunda, FIPS-only CipherSpecs yapılandırıldıysa, MQI istemcisi, FIPS dışı CipherSpec withRC2393belirten bağlantıları reddeder. IBM MQ , bu tür bağlantıların tümünü reddetmeyi

garanti etmez ve IBM MQ yapılandırmanızın FIPS-uyumlu olup olmadığını belirlemek sizin sorumluluğunuzda.

### **SCKR (10 basamaklı işaretli tamsayı)**

Bu alan yalnızca AIX, Linux, and Windows sistemlerinde çalışan IBM MQ MQI clients için anlamlıdır. Anahtarların ve sertifikaların saklandığı anahtar veri tabanı dosyasının yerini belirtir. Anahtar veri tabanı dosyasının dosya adı zzz.kdbbiçiminde olmalıdır; burada zzz kullanıcı tarafından seçilebilir. SCKR alanı, dosya adı gövdesiyle birlikte bu dosyanın yolunu içerir (dosya adı, son .kdbdahil değil, ancak dosya adı dahil değildir). .kdb dosya soneki otomatik olarak eklenir.

Her anahtar veri tabanı dosyasının ilişkili bir *parola saklama dosyası kütüğü* vardır. Bu, anahtar veri tabanına programlı erişim izni vermek için kullanılan şifrelenmiş parolaları içerir. Parola şifreleme dosyası, aynı dizinde yer almalıdır ve anahtar veritabanı ile aynı dosya sapına sahip olmalıdır ve .sthsonakiyle sona ermelidir.

Örneğin, SCKR alanının değeri /xxx/yyy/keyise, anahtar veritabanı dosyası /xxx/yyy/key.kdbolmalı ve parola şifreleme dosyası /xxx/yyy/key.stholmalıdır; burada xxx ve yyy dizin adlarını gösterir.

Değer, alanın uzunluğundan kısaysa, değeri boş değerli bir karakterle sonlandırın ya da alanın uzunluğuna göre boşluklarla doldurur. Değer denetlenmez; anahtar havuzuna erişilirken bir hata varsa, arama başarısız olur ve neden kodu RC2381ile başarısız olur.

Bir IBM MQ MQI client' den TLS bağlantısı çalıştırmak için, SCKR seçeneğini geçerli bir anahtar veritabanı dosyası adına ayarlayın.

Bu bir giriş alanıdır. Bu alanın uzunluğu LNSSKR tarafından verilir. Bu alanın ilk değeri boşluk karakteridir.

### **SCSID (10 basamaklı işaretli tamsayı)**

Bu, yapı tanıtıcısıdır; değer şöyle olmalıdır:

#### **SCSIDV**

TLS yapılandırma seçenekleri yapısına ilişkin tanıtıcı.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri SCSIDV ' dir.

### **SCVER (10 basamaklı işaretli tamsayı)**

Bu, yapı sürüm numarasıdır; değer şu olmalıdır:

#### **SCVER1**

Version-1 TLS yapılandırma seçenekleri yapısı.

#### **SCVER2**

Version-2 TLS yapılandırma seçenekleri yapısı.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

#### **SCVERC**

TLS yapılandırma seçenekleri yapısının geçerli sürümü.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri SCVER2' dir.

## **Başlangıçtaki değerler**

Çizelge 727. MQSCO 'daki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
SCSID	SCSIDV	'SCO'
SCVER	SCVER5	1
SCKR	Yok	Dizgi ya da boşluk boş değerli

Çizelge 727. MQSCO 'daki alanların ilk değerleri (devamı var)

Alan adı	Değişmezin adı	Değişmezin değeri
SCCH	Yok	Dizgi ya da boşluk boş değerli
SCAIC	Yok	0
SCAIO	Yok	0
SCAIP	Yok	Boş değerli gösterge ya da boş değerli byte
SCKRC	Yok	Boş değerli gösterge ya da boş değerli byte
SCFR	Yok	Boş değerli gösterge ya da boş değerli byte
SCEPSUITEB	Yok	Boş değerli gösterge ya da boş değerli byte
SCCERTVPOL	Yok	Boş değerli gösterge ya da boş değerli byte
SCCERLBL	Yok	Boş değerli gösterge ya da boş değerli byte

**Notlar:**

1. ↵ simgesi tek bir boş karakteri temsil eder.
2. SCEPSUITEB seçenekleri için bkz. "RPG bildirim" sayfa 1192.

**RPG bildirim**

```

D*..1.....2.....3.....4.....5.....6.....7..
D* MQSCO Structure
D*
D* Structure identifier
D SCSID 1 4 INZ('SCO ')
D* Structure version number
D SCVER 5 8I 0 INZ(1)
D* Location of TLS key repository
D SCKR 9 264 INZ
D* Cryptographic hardware configuration string
D SCCH 265 520 INZ
D* Number of MQAIR records present
D SCAIC 521 524I 0 INZ(0)
D* Offset of first MQAIR record from start of MQSCO structure
D SCAIO 525 528I 0 INZ(0)
D* Address of first MQAIR record
D SCAIP 529 544* INZ(*NULL)
D* Ver:1 **
D* Number of unencrypted bytes sent/received before secret key is
D* reset
D SCKRC 545 548I 0 INZ(0)
D* Using FIPS-certified algorithms
D SCFR 549 552I 0 INZ(0)
D* Ver:2 **
* Use only Suite B cryptographic algorithms
D SCEPSUITEB0
D SCEPSUITEB1 553 556I 0 INZ(1)
D SCEPSUITEB2 557 560I 0 INZ(0)
D SCEPSUITEB3 561 564I 0 INZ(0)
D SCEPSUITEB4 565 568I 0 INZ(0)
D SCEPSUITEB 10I 0 DIM(4) OVERLAY(SCEPSUITEB0)
D* Ver:3 **
D* Certificate validation policy
    
```

## IBM i IBM i üzerinde MQSD (Abonelik tanımlayıcısı)

MQSD yapısı, yapılmakta olan abonelik ile ilgili ayrıntıları belirtmek için kullanılır.

### Genel Bakış

#### Amaç

Yapı, MQSUB çağrısında bir giriş/çıkış değiştirgesidir.

#### Yönetilen abonelikler

Bir uygulamanın belirli bir kuyruğu aboneliğiyle eşleşen yayınlar için hedef olarak kullanması gerekmiyorsa, yönetilen abonelik özelliğini kullanabilir. Bir uygulama yönetilen abonelik kullanmayı seçerse, kuyruk yöneticisi, MQSUB çağrısından çıkış olarak bir nesne tanımlayıcısı sağlayarak aboneyi yayınlanan iletilerin gönderildiği hedef hakkında bilgilendirir. Daha fazla bilgi için bkz. [HOBj \(10 basamaklı işaretli tamsayı\)-giriş/çıkış](#).

Abonelik kaldırıldığında, kuyruk yöneticisi aşağıdaki durumlarda yönetilen hedeften alınmayan iletileri temizlemeyi de taahhüt eder:

- Abonelik kaldırıldığında-CORMSB ile MQCLOSE kullanımı ile-ve yönetilen Hobj kapatılır.
- Sürekli olmayan bir abonelik (SONDUR) kullanılarak bir uygulamayla bağlantı kesildiğinde örtük olarak
- Süresi dolduğu ve yönetilen Hobj kapatıldığı için bir abonelik kaldırıldığında süre bitimine göre.

Kalıcı olmayan aboneliklerle yönetilen abonelikleri kullanmalısınız; böylece temizleme işlemi gerçekleştirilebilir ve böylece, kapalı kalıcı olmayan aboneliklere ilişkin iletiler kuyruk yöneticinizde yer kaplamaz. Sürekli abonelikler, yönetilen hedefleri de kullanabilir.

#### Karakter kümesi ve kodlama

MQSD 'deki veriler, ENNAT tarafından verilen yerel kuyruk yöneticisinin **CodedCharSetId** kuyruk yöneticisi özniteliği ve kodlaması tarafından belirtilen karakter kümesinde olmalıdır. Ancak, uygulama bir IBM MQ istemcisi olarak çalışıyorsa, yapı istemcinin karakter kümesinde ve kodlamasında olmalıdır.

- [“Alanlar” sayfa 1193](#)
- [“Başlangıçtaki değerler” sayfa 1205](#)
- [“RPG bildirim” sayfa 1206](#)

### Alanlar

MQSD yapısı aşağıdaki alanları içerir; alanlar alfabetik sırayla açıklanır:

#### SDAID (32 baytlık karakter dizilimi)

Bu değer, bu abonelik ile eşleşen tüm yayın iletilerinin İleti Tanımlayıcı (MQMD) *MDAID* alanında bulunur. *SDAID*, iletinin kimlik bağlamının bir parçasıdır. İleti bağlamıyla ilgili daha fazla bilgi için bkz. [İleti bağlamı](#).

*MDAID* hakkında daha fazla bilgi için bkz. [MDAID](#).

SOSETI seçeneği belirtilmezse, varsayılan bağlam bilgileri olarak, bu abonelik için yayınlanan her iletide ayarlanan *MDAID* boş olur.

SOSETI seçeneği belirtilirse, *SDAID* kullanıcı tarafından oluşturulur ve bu alan, bu abonelik için her yayında ayarlanacak *MDAID* değerini içeren bir giriş alanıdır.

Bu alanın uzunluğu LNAIDD tarafından verilir. Bu alanın ilk değeri 32 boş karakterdir.

Var olan bir abonelik SOALT seçeneğini kullanarak değiştiriliyorsa, gelecekteki yayın iletilerinin *SDAID* değeri değiştirilebilir.

SORES kullanılarak MQSUB çağrısından dönüş yapılırsa, bu alan abonelik için kullanılmakta olan *MDAID* olarak ayarlanır.

### **SDACC (32 baytlık karakter dizgisi)**

Bu değer, bu abonelikte eşleşen tüm yayın iletilerinin İleti Tanımlayıcı (MQMD) *MDACC* alanında bulunur. *MDACC* , iletinin kimlik bağlamının bir parçasıdır. İleti bağlamıyla ilgili daha fazla bilgi için bkz. İleti bağlamı.

*MDACC* hakkında daha fazla bilgi için bkz. MDACC.

*SDACC* alanı için aşağıdaki özel değeri kullanabilirsiniz:

#### **YOK**

Hesap simgesi belirtilmedi.

Değer, alanın uzunluğu için ikili sıfırdır.

SOSETI seçeneği belirtilmezse, hesap simgesi kuyruk yöneticisi tarafından varsayılan bağlam bilgileri olarak oluşturulur ve bu alan, bu abonelik için yayınlanan her iletide ayarlanan *MDACC* ögesini içeren bir çıkış alanıdır.

SOSETI seçeneği belirtilirse, muhasebe simgesi kullanıcı tarafından oluşturulur ve bu alan, bu abonelik için her yayında ayarlanacak *MDACC* değerini içeren bir giriş alanıdır.

Bu alanın uzunluğu LNACCT tarafından verilir. Bu alanın ilk değeri ACNONE' dır.

SOALT seçeneği kullanılarak var olan bir abonelik değiştiriliyorsa, ilerideki yayın iletilerinde *MDACC* değeri değiştirilebilir.

SORES kullanılarak bir MQSUB çağrısından dönüş yapılırsa, bu alan abonelik için kullanılmakta olan *MDACC* değerine ayarlanır.

### **SDASI (40 baytlık bit dizgisi)**

Bu, uygun yetki denetimlerinin gerçekleştirilmesine izin vermek için *SDAU* ile yetkilendirme hizmetine iletilen bir güvenlik tanıtıcısıdır.

*SDASI* yalnızca *SOALTU* belirtilirse ve *SDAU* alanı ilk boş karakter ya da alanın sonuna kadar tamamen boş değilse kullanılır.

SORES kullanılarak MQSUB çağrısından dönüş yapılıyorsa, bu alan değişmez.

Daha fazla bilgi için MQOD veri tipinde ODASI açıklamasına bakın.

### **SDAU (12 baytlık karakter dizilimi)**

*SOALTU* belirtirseniz, bu alan, uygulamanın çalışmakta olduğu kullanıcı kimliği yerine, aboneliğe ve hedef kuyruğa (MQSUB çağrısının **Hobj** değiştirilmesinde belirtilir) ilişkin yetkiyi denetlemek için kullanılan diğer bir kullanıcı kimliği içerir.

Başarılı olursa, bu alanda belirtilen kullanıcı kimliği, uygulamanın çalışmakta olduğu kullanıcı kimliği yerine, abonelik sahibi kullanıcı kimliği olarak kaydedilir.

*SOALTU* belirtilirse ve bu alan ilk boş karaktere ya da alanın sonuna kadar tamamen boşsa, abonelik ancak belirlenen seçeneklerle ya da çıkış için hedef kuyruk ile bu konuya abone olmak için kullanıcı yetkisi gerekmediğinde başarılı olabilir.

*SOALTU* belirtilmezse, bu alan yoksayılr.

SORES kullanılarak MQSUB çağrısından dönüş yapılıyorsa, bu alan değişmez.

Bu bir giriş alanıdır. Bu alanın uzunluğu LNUID tarafından verilir. Bu alanın ilk değeri 12 boş karakterdir.

### **SDCID (24 baytlık bit dizgisi)**

Bu abonelikte eşleşmesi için gönderilen tüm yayınlar, ileti tanımlayıcısında bu ilinti tanıtıcısını içerir. Birden çok abonelik yayınlarını almak için aynı kuyruğu kullanıyorsa, MQGET ilinti tanıtıcısıyla

kullanılması yalnızca belirli bir aboneliğe ilişkin yayınların elde edilmesine izin verir. Bu ilinti tanıtıcısı kuyruk yöneticisi ya da kullanıcı tarafından oluşturulabilir.

SOSCID seçeneği belirtilmezse, ilinti tanıtıcısı kuyruk yöneticisi tarafından üretilir ve bu alan, bu abonelik için yayınlanan her iletide ayarlanan ilinti tanıtıcısını içeren bir çıkış alanıdır.

SOSCID seçeneği belirtilirse, ilinti tanıtıcısı kullanıcı tarafından üretilir ve bu alan, bu abonelik için her yayında ayarlanacak ilinti tanıtıcısını içeren bir giriş alanıdır. Bu durumda, alan CINONE içeriyorsa, bu abonelik için yayınlanan her iletide ayarlanan ilinti tanıtıcısı, iletinin özgün konmasıyla yaratılan ilinti tanıtıcısıdır.

SOGRP seçeneği belirtilirse ve belirtilen ilinti tanıtıcısı, aynı kuyruk ve çakışan bir konu dizgisi kullanılarak var olan bir gruplanmış abonelikte aynıysa, yayının bir kopyasıyla yalnızca gruptaki en önemli abonelik sağlanır.

Bu alanın uzunluğu LNCID tarafından verilir. Bu alanın ilk değeri CINONE' dur.

SOALT seçeneği kullanılarak var olan bir abonelik değiştiriliyorsa ve bu alan bir giriş alanıysa, abonelik ilinti tanıtıcısı SOGRP seçeneği kullanılarak yaratılmadıysa değiştirilebilir.

SORES kullanılarak bir MQSUB çağrısından döndüğünüzde, bu alan abonelik için yürürlükteki ilinti tanıtıcısına ayarlanır.

### **SDEXP (10 basamaklı işaretli tamsayı)**

Bu, abonelik süresinin sona ermesinden sonra saniyenin onda biri olarak ifade edilen süredir. Bu aralık geçtikten sonra başka hiçbir yayın bu abonelikte eşleşmeyecek. Bu, bu aboneye gönderilen yayınların MQMD 'deki MDEXP alanında değer olarak da kullanılır.

Şu özel değer tanınır:

#### **EIULIM**

Aboneliğin sınırsız bir süre sonu vardır.

Var olan bir abonelik SOALT seçeneği kullanılarak değiştiriliyorsa, aboneliğin süresi değiştirilebilir.

SORES seçeneği kullanılarak bir MQSUB çağrısından dönüşte bu alan, aboneliğin geri kalan süre bitimine değil, özgün süre bitimine ayarlanır.

### **SDON (48 baytlık karakter dizgisi)**

Bu, yerel kuyruk yöneticisinde tanımlandığı şekilde, konu nesnesinin adıdır.

Ad aşağıdaki karakterleri içerebilir:

- Büyük harf alfabetik karakterler (A-Z)
- Küçük harf alfabetik karakterler (a-z)
- Sayısal basamaklar (0-9)
- Nokta (.), eğik çizgi (/), altçizgi (\_), yüzde (%)

Ad başta ya da gömülü boşluklar içermemelidir, ancak sonda boşluklar içerebilir. Addaki önemli verilerin sonunu göstermek için boş bir karakter kullanın; bu karakteri izleyen karakterler ve boş değer boşluk olarak değerlendirilir. Aşağıdaki kısıtlamalar söz konusudur:

- EBCDIC Katakana kullanan sistemlerde küçük harfli karakterler kullanılamaz.
- Komutlarda belirtildiğinde, küçük harf, eğik çizgi ya da yüzde içeren adlar tırnak içine alınmalıdır. Bu tırnak işaretleri, yapılarda alan olarak ya da çağrılarda parametre olarak oluşan adlar için belirlenmemelidir.

SDON , Tam konu adını oluşturmak için kullanılır.

Tam konu adı iki farklı alandan oluşturulabilir: SDON ve SDOS. Bu iki alanın nasıl kullanıldığına ilişkin ayrıntılar için [Konu dizgilerini birleştirme](#) başlıklı konuya bakın.

SORES seçeneği kullanılarak bir MQSUB çağrısından dönüş yapılıyorsa, bu alan değişmez.

Bu alanın uzunluğu LNTOPN tarafından verilir. Bu alanın ilk değeri 48 boş karakterdir.

SDALT seçeneğini kullanarak var olan bir abonelik değiştiriliyorsa, abone olunan konu nesnesinin adı değiştirilemez. Bu alan ve *SDOS* atlanabilir. Sağlanırsa, aynı tam konu adına çözümleri gerekir ya da arama RC2510 ile başarısız olur.

### SDOPT (10 basamaklı işaretli tamsayı)

Aşağıdaki seçeneklerden en az birini belirtmelisiniz:

- SOALT
- S1RALAR
- SOCRT

Değerler eklenebilir. Aynı sabiti bir kereden fazla eklemeyin. Tablo, bu seçenekleri nasıl birleştirebileceğinizi gösterir: Geçerli olmayan birleşimler not edildi; diğer birleşimler geçerlidir.

### Erişim ya da yaratma seçenekleri

Erişim ve yaratma seçenekleri, aboneliğin oluşturulup oluşturulmayacağını ya da var olan bir aboneliğin döndürülüp döndürülmeyeceğini ya da değiştirilip değiştirilmeyeceğini denetler. Bu seçeneklerden en az birini belirtmelisiniz. Çizelge, geçerli erişim ya da yaratma seçenekleri birleşimlerini görüntüler.

Çizelge 728. Geçerli erişim ve yaratma seçenekleri birleşimleri	
Seçenek birleşimi	Notlar
SOCRT	Yoksa bir abonelik oluşturur; abonelik varsa başarısız olur.
S1RALAR	Var olan bir aboneliği sürdürür, abonelik yoksa başarısız olur.
SOCRT + S1RALAMA	Yoksa bir abonelik oluşturur ve varsa, eşleşen bir aboneliği sürdürür. Birkaç kez çalıştırılacak bir uygulamada kullanılırsa kullanışlı bir birleşim.
SORES + SOALT (nota bakın)	Var olan bir aboneliği sürdürür ve MQSD ' de belirtilenlerle eşleşecek şekilde alanları değiştirir, abonelik yoksa başarısız olur.
SOCRT + SOALT (nota bakın)	Yoksa bir abonelik yaratır ve varsa, eşleşen bir aboneliği sürdürerek, MQSD ' de belirtilen alanlarla eşleşecek şekilde alanların değiştirilmesini sağlar. Devam etmeden önce aboneliğinin belirli bir durumda olduğundan emin olmak isteyen bir uygulamada kullanışlı birleşim.

### Not:

SOALT seçeneğini belirleyen seçenekler SORESdeğerini de belirleyebilir, ancak bu birleşim tek başına SOALT belirtiminin belirlenmesinde ek bir etki yaratmaz. SOALT , bir aboneliği değiştirmek için MQSUB 'nin çağrılması aboneliklerin de sürdürüldüğü anlamına geldiğinden SORES' i belirtir. Bunun tersi doğru değildir, ancak: bir aboneliğin sürdürülmesi değiştirileceği anlamına gelmez.

### SOCRT

Belirtilen konu için bir abonelik oluşturun. Aynı SDSN kullanan bir abonelik varsa, çağrı RC2432 ile başarısız olur. Bu hata, SOCRT seçeneği SORESile birleştirilerek önlenir. SDSN her zaman gerekli değildir. Daha fazla ayrıntı için bu alanın açıklamasına bakın.

SOCRT 'nin SORES ile birleştirilmesi, önce belirtilen SDSM için var olan bir abonelik olup olmadığını denetler ve önceden var olan aboneliğe bir tanıtıcı döndürürse; ancak var olan bir abonelik yoksa, MQSD' de sağlanan tüm alanlar kullanılarak yeni bir abonelik yaratılır.

SOCRT , benzer bir etki için SOALT ile de birleştirilebilir (bu konuda daha sonra SOALT ile ilgili ayrıntılara bakın).



## SIRALAR

*SDSN* tarafından belirtilenlerle eşleşen önceden var olan bir aboneliğe bir tanıtıcı döndürür. Eşleşen abonelik özniteliklerinde değişiklik yapılmadı ve bunlar MQSD yapısındaki çıkışta döndürülür. MQSD içeriğinin çoğu kullanılmaz: Kullanılan alanlar şunlardır: *SDSID*, *SDVER*, *SDOPT*, *SDAID*, *SDASIVe SDSN*.

Tam abonelik adıyla eşleşen bir abonelik yoksa, çağrı RC2428 neden koduyla başarısız olur. Bu hata, SOCRT seçeneği SORES ile birleştirilerek önlenir. SOCRT ile ilgili ayrıntılar için bkz. [SOCRT](#).

Aboneliğin kullanıcı kimliği, aboneliği oluşturan kullanıcı kimliğidir ya da daha sonra farklı bir kullanıcı kimliği tarafından değiştirildiyse, bu, en son, başarılı değişikliğin kullanıcı kimliğidir. Bir *SDAID* kullanılırsa ve o kullanıcı için diğer kullanıcı kimliklerinin kullanılmasına izin verilirse, *SDAID* aboneliğin yapıldığı kullanıcı kimliği yerine aboneliği oluşturan kullanıcı kimliği olarak kaydedilir.

Aboneliği oluşturan kullanıcı kimliği, bu alan kullanılıyorsa *SDAU* olarak kaydedilir ve o kullanıcı için diğer kullanıcı kimliklerinin kullanılmasına izin verilir.

SOAUID seçeneği olmadan oluşturulan eşleşen bir abonelik varsa ve aboneliğin kullanıcı kimliği, abonelik için tanıtıcı isteyen uygulamadan farklıysa, çağrı RC2434 neden koduyla başarısız olur.

Eşleşen bir abonelik varsa ve şu anda başka bir uygulama tarafından kullanılıyorsa, çağrı RC2429 neden koduyla başarısız olur. Şu anda aynı bağlantı tarafından kullanılıyorsa, çağrı başarısız olmaz ve abonelik için bir tanıtıcı döndürülür.

SubName içinde adı belirtilen abonelik, bir uygulamadan sürdürmek ya da değiştirmek için geçerli bir abonelik değilse, çağrı RC2523 ile başarısız olur.

SORES, SOALT tarafından örtük olarak belirtilir ve bu seçenekle birleştirilmesi gerekmez; ancak, bu iki seçeneğin birleştirilmesi bir hata değildir.

## SOALT

*SDSN* içinde belirtilenlerle eşleşen tam abonelik adıyla önceden var olan bir aboneliğe bir tanıtıcı döndürür. Aboneliğin MQSD ' de belirtilenlerden farklı olan öznitelikleri, o öznitelik için değişikliğe izin verilmediği sürece abonelikte değiştirilir. Ayrıntılar her özniteliğin tanımında belirtilir ve aşağıdaki tabloda özetlenir. Değiştirilemeyen bir özniteliği değiştirmeye çalışırsanız, çağrı aşağıdaki çizelgede gösterilen neden koduyla başarısız olur.

Tam abonelik adıyla eşleşen bir abonelik yoksa, çağrı RC2428 neden koduyla başarısız olur. Bu hata, SOCRT seçeneği SOALT ile birleştirilerek önlenir.

SOCRT 'nin SOALT ile birleştirilmesi, önce belirtilen tam abonelik adı için var olan bir abonelik olup olmadığını ve önceden ayrıntılı olarak yapılan değişikliklerle önceden var olan aboneliğin bir tanıtıcısı olup olmadığını denetler; ancak var olan abonelik yoksa, MQSD' de sağlanan tüm alanlar kullanılarak yeni bir abonelik yaratılır.

Aboneliğin kullanıcı kimliği, aboneliği oluşturan kullanıcı kimliğidir ya da daha sonra farklı bir kullanıcı kimliği tarafından değiştirildiyse, en son başarılı değişikliğin kullanıcı kimliğidir. *SDAU* kullanılırsa (ve bu kullanıcı için diğer kullanıcı kimliklerinin kullanılmasına izin verilirse), diğer kullanıcı kimliği, aboneliğin yapıldığı kullanıcı kimliği yerine, aboneliği yaratan kullanıcı kimliği olarak kaydedilir.

SOAUID seçeneği olmadan oluşturulan eşleşen bir abonelik varsa ve aboneliğin kullanıcı kimliği, abonelik için tanıtıcı isteyen uygulamadan farklıysa, çağrı başarısız olur. Neden kodu RC2434 .

Eşleşen bir abonelik varsa ve şu anda başka bir uygulama tarafından kullanılıyorsa, arama RC2429 ile başarısız olur. Şu anda aynı bağlantı tarafından kullanılıyorsa çağrı başarısız olmaz ve abonelik için bir tanıtıcı döndürülür.

SubName içinde adı belirtilen abonelik, bir uygulamadan sürdürmek ya da değiştirmek için geçerli bir abonelik değilse, çağrı RC2523 ile başarısız olur.

Aşağıdaki çizelgelerde, SOALT ile değiştirilebilecek abonelik öznitelikleri gösterilir.

Çizelge 729. MQSD ve MQSUB içindeki değiştirilebilecek öznitelikler			
Veri tipi tanımlayıcısı ya da işlev çağırısı	Alan adı	Bu öznitelik SOALT kullanılarak değiştirilebilir mi?	Neden Kodu
MQSD	Dayanıklılık seçenekleri	Hayır	RC2509
MQSD	Hedef Seçenekleri	Evet	Yok
MQSD	Kayıt seçenekleri	Evet (bkz. not 1 )	RC2515 SOGRP ' yi değiştirmeye çalışırsanız
MQSD	Yayınlama seçenekleri	Evet (bkz. not 2 )	Yok
MQSD	Genel arama karakteri seçenekleri	Hayır	RC2510
MQSD	Diğer seçenekler	Hayır (bkz. not 3 )	Yok
MQSD	ObjectName	Hayır	RC2510
MQSD	SDAU	Hayır (bkz. not 4 )	Yok
MQSD	SDASI	Hayır (bkz. not 4 )	Yok
MQSD	SDEXP	Evet	Yok
MQSD	SDOS	Hayır	RC2510
MQSD	SDSN	Hayır (bkz. not 5 )	Yok
MQSD	SDSUD	Evet	Yok
MQSD	SDCID	Evet (bkz. not 6 )	RC2515 gruplanmış abonelikteyken
MQSD	SDPRI	Evet	Yok
MQSD	SDACC	Evet	Yok
MQSD	SDAID	Evet	Yok
MQSD	SDSL	Hayır	RC2512
MQSUB	Hobj.	Evet (bkz. not 6 )	RC2515 gruplanmış abonelikteyken

**Notlar:**

1. SOGRP değiştirilemiyor.
2. SONEWP , aboneliğin bir parçası olmadığı için değiştirilemiyor
3. Bu seçenekler aboneliğin bir parçası değildir
4. Bu öznitelik, aboneliğin bir parçası değil
5. Bu öznitelik, değiştirilmekte olan aboneliğin tanıtıcısıdır
6. Gruplanmış alt ( SOGRP ) ögesinin bir parçası dışında değiştirilebilen

**Dayanıklılık seçenekleri:** Aşağıdaki seçenekler, aboneliğin ne kadar dayanıklı olduğunu denetler. Bu seçeneklerden yalnızca birini belirleyebilirsiniz. Var olan bir aboneliği SOALT seçeneğini kullanarak değiştiriyorsanız, aboneliğin dayanıklılığını değiştiremezsiniz. SORESkullanılarak bir MQSUB çağırısından döndüğünüzde, uygun dayanıklılık seçeneği ayarlanır.

**SODUR.**

Bu konuya ilişkin aboneliğin, CORMSB seçeneğiyle MQCLOSE kullanılarak belirtik olarak kaldırılincaya kadar kalmasını isteyin. Bu abonelik belirtik olarak kaldırılmazsa, bu uygulama kuyruk yöneticisine bağlandıktan sonra da kalır.

Sürekli aboneliğe izin vermeyen bir konuya sürekli abonelik istenirse, çağrı RC2436 ile başarısız olur.

**SONDUR.**

Önceden belirtik olarak kaldırılmadıysa, kuyruk yöneticisine yönelik uygulama bağlantısı kapatıldığında bu konuya ilişkin aboneliğin kaldırılmasını isteyin. SONDUR , SODUR seçeneğinin tersidir ve program belgelerine yardımcı olmak üzere tanımlanır. İkisi de belirtilmezse, varsayılan değer budur.

**Hedef seçenekleri:** Aşağıdaki seçenekler, abone olunan bir konuya ilişkin yayınların gönderileceği hedefi denetler. SOALT seçeneğini kullanarak var olan bir abonelik değiştiriliyorsa, abonelik için yayınlar için kullanılan hedef değiştirilebilir. SORES kullanılarak bir MQSUB çağrısından dönüş yapılırsa, bu seçenek uygunsa ayarlanır.

**SOMAN.**

Yayınların gönderildiği hedefin kuyruk yöneticisi tarafından yönetilmesini isteyin.

*HOBJ* içinde döndürülen nesne tanıtıcısı, kuyruk yöneticisi tarafından yönetilen bir kuyruğu temsil eder ve sonraki MQGET, MQCB, MQINQ ya da MQCLOSE çağrılarıyla birlikte kullanılır.

Önceki bir MQSUB çağrısından döndürülen bir nesne tanıtıcısı, SOMAN belirtilmediğinde **Hobj** değiştirilmesinde sağlanamaz.

**Kayıt seçenekleri:** Aşağıdaki seçenekler, bu abonelik için kuyruk yöneticisine yapılan kayıt ayrıntılarını denetler. SOALT seçeneği kullanarak var olan bir abonelik değiştiriliyorsa, bu kayıt seçenekleri değiştirilebilir. SORES kullanılarak bir MQSUB çağrısından döndüğünüzde, uygun kayıt seçenekleri ayarlanır.

**SOGRP.**

Bu abonelik, aynı kuyruk kullanılarak aynı *SDSL* abonelikleriyle gruplanır ve aynı ilinti tanıtıcısı belirtilerek, birden çok yayın iletisinin abonelik grubuna sağlanmasına neden olacak yayınların, üst üste gelen konu dizeleri kümesi nedeniyle kuyruğa tek bir iletinin gönderilmesine neden olur. Bu seçenek kullanılmazsa, eşleşen her benzersiz abonelik ( *SDSN* ile tanıtılır), yayının bir kopyasıyla birlikte sağlanır. Bu, yayının birden çok kopyasının, birden çok abonelik tarafından paylaşılan kuyruğa yerleştirileceği anlamına gelebilir.

Yalnızca gruptaki en önemli abonelik, yayının bir kopyasıyla birlikte sağlanır. En önemli abonelik, genel arama karakterinin bulunduğu noktaya kadar Tam konu adını temel alır. Grup içinde genel arama karakteri şemalarının bir karışımı kullanılırsa, yalnızca genel arama karakterinin konumu önemlidir. Aynı kuyruğu paylaşan abonelikler grubunda farklı genel arama karakteri şemalarını birleştirmemeniz önerilir.

Yeni bir gruplanmış abonelik oluştururken yine de benzersiz bir *SDSN* olmalıdır, ancak gruptaki var olan bir aboneliğin tam konu adıyla eşleşirse, arama RC2514 ile başarısız olur.

Gruptaki en önemli abonelik SON0LC belirtiyorsa ve bu aynı uygulamadan bir yayınsa, kuyruğa herhangi bir yayın teslim edilmez.

Bu seçenekle yapılan bir abonelik değiştirilirken, MQSUB çağrısında *Hobj* (kuyruk ve kuyruk yöneticisi adını gösterir) gruplamayı belirten alanlar ve *SDCID* değiştirilemez. Bunları değiştirme girişimi, RC2515 ile çağrı başarısız olmasına neden olur.

Bu seçenek, CINONE olarak ayarlanmamış bir *SDCID* ile SOSCID ile birleştirilmeli ve SOMAN ile birleştirilmemelidir.

**SOAUID**

SOAUID belirtildiğinde, abonenin kimliği tek bir kullanıcı kimliğiyle sınırlı değildir. Bu, herhangi bir kullanıcının uygun yetkiye sahip olduğunda aboneliği değiştirmesini ya da sürdürmesini sağlar.

Aynı anda yalnızca tek bir kullanıcı aboneliğe sahip olabilir. Başka bir uygulama tarafından kullanılmakta olan bir aboneliği kullanmaya devam etme girişimi, RC2429 ile yapılan aramanın başarısız olmasına neden olur.

Bu seçeneği var olan bir aboneliğe eklemek için, SOALTkullanan MQSUB çağrısı, özgün aboneliğin kendisiyle aynı kullanıcı kimliğinden gelmelidir.

Bir MQSUB çağrısı SOAUID ile var olan bir aboneliğe başvurursa ve kullanıcı kimliği özgün abonelikten farklıysa, çağrı yalnızca yeni kullanıcı kimliğinin konuya abone olma yetkisi varsa başarılı olur. İşlem hatasız olarak tamamlandığında, bu abonenin gelecekteki yayınları, yayın iletilerinde yeni kullanıcı kimliği ayarlanmış olarak abonenin kuyruğuna yerleştirilir.

Hem SOAUID , hem de SOFUIDbelirtmeyin. Hiçbiri belirtilmezse, varsayılan değer SOFUID' dir.

## **SOFUID**

SOFUID belirtildiğinde, abonelik yalnızca aboneliği değiştiren son kullanıcı kimliği tarafından değiştirilebilir ya da sürdürülebilir. Abonelik değiştirilmediyse, aboneliği yaratan kullanıcı kimliği olur.

Bir MQSUB komutu SOAUID ile var olan bir aboneliğe gönderme yaparsa ve SOFUID' yi kullanmak için SOALT kullanarak aboneliği değiştirirse, aboneliğin kullanıcı kimliği artık bu yeni kullanıcı kimliğinde düzeltilmiştir. Arama, yalnızca yeni kullanıcı kimliğinin konuya abone olma yetkisi varsa başarılı olur.

Aboneliğin sahibi olarak kaydedilenden başka bir kullanıcı kimliği SOFUID aboneliğini sürdürmeye ya da değiştirmeye çalışırsa, arama RC2434 ile başarısız olur. Bir aboneliğin sahip olan kullanıcı kimliği, **DISPLAY SBSTATUS** komutu kullanılarak görüntülenebilir.

Hem SOAUID , hem de SOFUIDbelirtmeyin. Hiçbiri belirtilmezse, varsayılan değer SOFUID' dir.

**Yayın seçenekleri:** Aşağıdaki seçenekler, yayınların bu aboneye gönderilme şeklini denetler. Var olan bir abonelik SOALT seçeneği kullanılarak değiştiriliyorsa, bu yayın seçenekleri değiştirilebilir.

## **SONOLC**

Aracıya, uygulamanın kendi yayınlarından hiçbirini görmek istemediğini bildirir. Bağlantı tanıtıcıları aynıysa, yayınların aynı uygulamadan kaynaklandığı kabul edilir. SORES kullanılarak bir MQSUB çağrısından dönüşte, bu seçenek uygunsuz ayarlanır.

## **SONEWP**

Bu abonelik oluşturulduğunda, şu anda alıkonan hiçbir yayın gönderilmez, yalnızca yeni yayınlar gönderilir. Bu seçenek yalnızca SOCRE belirtildiğinde geçerlidir. Abonelikte yapılan sonraki değişiklikler, yayınların akışını değiştirmez ve bu nedenle bir konuda tutulan yayınlar, aboneye yeni yayınlar olarak gönderilmiştir.

Bu seçenek SOCRE olmadan belirtilirse, çağrı RC2046 ile başarısız olur. SORES kullanılarak bir MQSUB çağrısından dönüş yapılsa, abonelik bu seçenek kullanılarak yaratılmış olsa da bu seçenek ayarlanmaz.

Bu seçenek kullanılmazsa, önceden alıkonan iletiler sağlanan hedef kuyruğa gönderilir. RC2525 ya da RC2526 hatası nedeniyle bu işlem başarısız olursa, abonelik oluşturma işlemi başarısız olur.

Bu seçenek SOPUBRile birlikte geçerli değildir.

## **SOPUBR**

Bu seçeneğin ayarlanması, abonenin özellikle gerektiğinde bilgi isteyeceğini gösterir. Kuyruk yöneticisi aboneye istenmeyen iletiler göndermez. Önceki bir MQSUB çağrısından Hsub tanıtıcısı kullanılarak bir MQSUBRQ çağrısı her yapıldığında, alıkonan yayın (ya da bir joker karakter belirtilirse birden çok yayın) aboneye gönderilir. Bu seçenek kullanılarak MQSUB çağrısının sonucu olarak hiçbir yayın gönderilmez. SORES kullanılarak bir MQSUB çağrısından dönüşte, bu seçenek uygunsuz ayarlanır.

Bu seçenek, SONEWPile birlikte geçerli değildir.

**Genel arama karakteri seçenekleri:** Aşağıdaki seçenekler, MQSD ' nin SDOS alanında sağlanan dizgide genel arama karakterlerinin nasıl yorumlanacağını denetler. Bu seçeneklerden yalnızca birini belirleyebilirsiniz. Var olan bir abonelik SOALT seçeneği kullanılarak değiştiriliyorsa, bu genel arama

karakteri seçenekleri değiştirilemez. SORES kullanılarak bir MQSUB çağrısından döndüğünüzde, uygun genel arama karakteri seçeneği ayarlanır.

### SOWCHR

Genel arama karakterleri yalnızca konu dizesi içindeki karakterlerde çalışır. SOWCHR alanı, eğik çizgiyi (/) özel bir anlamı olmayan başka bir karakter olarak işler.

SOWCHR tarafından tanımlanan davranış aşağıdaki çizelgede gösterilir:

Özel Karakter	Davranış
*	Joker karakter, sıfır ya da daha fazla karakter
?	Genel arama karakteri, bir karakter
%	'*', '?' ya da '%' karakterlerinin bir dizgide kullanılmasına ve özel karakter olarak yorumlanmamasına izin vermek için çıkış karakteri kullanın; örneğin, '% *', '%?' ya da '% % '.

Örneğin, şu konuda yayınlama:

```
/level0/level1/level2/level3/level4
```

aboneleri aşağıdaki konuları kullanarak eşleştirir:

```
*  
/*  
/ level0/level1/level2/level3/*  
/ level0/level1/*/level3/level4  
/ level0/level1/le?e12/level3/level4
```

**Not:** Bu joker karakter kullanımı, Yayınlama/Abone Olma için MQRFH1 biçimlendirilmiş iletileri kullanırken, IBM MQ V6 ve WebSphere MB V6 ' da sağlanan anlamı tam olarak sağlar. Bunun yeni yazılan uygulamalar için kullanılmaması ve yalnızca daha önce o sürüm için çalışan uygulamalar için kullanılması ve SOWTOP içinde açıklandığı gibi varsayılan genel arama karakteri davranışını kullanacak şekilde değiştirilmemiş olması önerilir.

### SOWTOP

Joker karakterler yalnızca konu dizesi içindeki konu öğelerinde çalışır. Hiçbiri seçilmezse, bu varsayılan davranıştır.

SOWTOP için gereken davranış aşağıdaki çizelgede gösterilmiştir:

Özel Karakter	Davranış
/	Konu düzeyi ayırıcısı
#	Genel arama karakteri: birden çok konu düzeyi
+	Genel arama karakteri: tek konu düzeyi

### Not:

'+' ve '#', bir konu düzeyinde diğer karakterlerle (kendileri dahil) karıştırıldıysa genel arama karakteri olarak kabul edilmez. Aşağıdaki dizgide, '#' ve '+' karakterleri olağan karakterler olarak işlenir.

```
level0/level1/#+/level3/level#
```

Örneğin, şu konuda yayınlama:

```
/level0/level1/level2/level3/level4
```

aboneleri aşağıdaki konuları kullanarak eşleştirir:

```
#  
/#  
/ level0/level1/level2/level3/#  
/ level0/level1/+/level3/level4
```

**Not:** Bu joker karakter kullanımı, Yayınlama/Abone Olma için MQRFH2 biçimlendirilmiş iletileri kullanırken WebSphere Message Broker 6 içinde sağlanan anlamı sağlar.

**Diğer seçenekler:** Aşağıdaki seçenekler, abonelik yerine API çağrısını nasıl yayınlanacağına denetler. SORES kullanılarak bir MQSUB çağrısından dönüşte bu seçenekler değişmez.

### SOALTU

SDAU alanı, bu MQSUB çağrısının geçerliliğini denetlemek için kullanılacak bir kullanıcı kimliği içerir. Arama, yalnızca bu SDAU ' nun belirtilen erişim seçenekleriyle nesneyi açma yetkisi varsa, uygulamanın altında çalıştığı kullanıcı kimliğinin bunu yapma yetkisi olup olmadığına bakılmaksızın başarılı olabilir.

### SOSCID

Abonelik, *SDCID* alanında sağlanan ilinti tanıtıcısını kullanmaktır. Bu seçenek belirtilmezse, abonelik sırasında kuyruk yöneticisi tarafından otomatik olarak bir ilinti tanıtıcısı yaratılır ve *SDCID* alanında uygulamaya döndürülür. Daha fazla bilgi için bkz. [SDCID \(24 baytlık bit dizisi\) SDCID](#) .

### SOSETI

Abonelik, *SDACC* ve *SDAID* alanlarında sağlanan muhasebe simgesini ve uygulama kimlik verilerini kullanmaktır.

Bu seçenek belirtilirse, hedef kuyruğa 00SETI ile MQOPEN çağrısı kullanılarak erişildiği gibi, SOMAN seçeneğinin de kullanıldığı durumlar dışında, aynı yetki denetimi gerçekleştirilir; bu durumda hedef kuyrukta yetki denetimi yoktur.

Bu seçenek belirtilmezse, bu aboneye gönderilen yayınlarla ilişkili varsayılan bağlam bilgileri aşağıdaki gibi olur:

Çizelge 732. Bu aboneye gönderilen yayınlar için varsayılan bağlam bilgileri	
MQMD ' deki Alan	Kullanılan değer
MDUID	Abonelik yapıldığında aboneliğe ilişkilendirilen kullanıcı kimliği.
MDACC	Olanaklıysa ortamdan saptanır; yoksa ACNONE olarak ayarlanır.
MDAID	Boşluklara ayarla

Bu seçenek yalnızca SOCRE ve SOALT ile geçerlidir. SORES ile kullanılırsa, *SDACC* ve *SDAID* alanları yoksayılır, bu nedenle bu seçeneğin etkisi olmaz.

Abonelik, daha önce aboneliğin sağladığı kimlik bağlamı bilgilerinin bulunduğu bu seçenek kullanılmadan değiştirilirse, değiştirilen abonelik için varsayılan bağlam bilgileri oluşturulur.

Farklı kullanıcı kimliklerinin bunu SOAUID seçeneğiyle kullanmasına izin veren bir abonelik farklı bir kullanıcı kimliği tarafından sürdürüldüğünde, artık aboneliğe sahip olan yeni kullanıcı kimliği için varsayılan kimlik bağlamı oluşturulur ve yeni kimlik bağlamını içeren sonraki yayınlar teslim edilir.

## **SOFIQ**

Kuyruk yöneticisi susturma durumundaysa MQSUB çağrısı başarısız olur. z/OS sistemlerinde, bir CICS ya da IMS uygulaması için bu seçenek, bağlantı susturma durumundaysa MQSUB çağrısına da başarısız olmasına neden olur.

### **SDAU (12 baytlık karakter dizilimi)**

SOALTU belirtirseniz, bu alan, uygulamanın çalışmakta olduğu kullanıcı kimliği yerine, aboneliğe ve hedef kuyruğa (MQSUB çağrısının **Hobj** değiştirilmesinde belirtilir) ilişkin yetkiyi denetlemek için kullanılan diğer bir kullanıcı kimliği içerir.

Başarılı olursa, bu alanda belirtilen kullanıcı kimliği, uygulamanın çalışmakta olduğu kullanıcı kimliği yerine, abonelik sahibi kullanıcı kimliği olarak kaydedilir.

SOALTU belirtilirse ve bu alan ilk boş karaktere ya da alanın sonuna kadar tamamen boşsa, abonelik ancak belirtilen seçeneklerle bu konuya ya da çıkış için hedef kuyruğa hiçbir kullanıcı yetkisinin abone olması gerekmediğinde başarılı olabilir.

SOALTU belirtilmezse, bu alan yoksayıdır.

SORESkullanılarak MQSUB çağrısından dönüş yapılıyorsa, bu alan değişmez.

Bu bir giriş alanıdır. Bu alanın uzunluğu LNUID tarafından verilir. Bu alanın ilk değeri 12 boş karakterdir.

### **SDPRI (10 basamaklı işaretli tamsayı)**

Bu değer, bu abonelikte eşleşen tüm yayın iletilerinin ileti tanımlayıcısının (MQMD) MQPRI alanında bulunan değerdir. MQMD 'deki MQPRI alanıyla ilgili daha fazla bilgi için bkz. [MDPRI](#).

Değer sıfırdan büyük ya da sıfıra eşit olmalıdır; sıfır en düşük önceliktir. Aşağıdaki özel değerler de kullanılabilir:

#### **PRQDEF**

MQSUB çağrısındaki Hobj alanında bir abonelik kuyruğu sağlandığında ve yönetilen bir tanıtıcı olmadığında, iletinin önceliği bu kuyruğun **DefPriority** özniteliğinden alınır. Bu şekilde tanımlanan kuyruk bir küme kuyruğuyorsa ya da kuyruk adı çözme yolunda birden çok tanımlama varsa, [MDPRI](#) için açıklandığı gibi, yayınlama iletisi kuyruğa konduğunda öncelik belirlenir.

MQSUB çağrısı yönetilen bir tanıtıcı kullanıyorsa, iletinin önceliği, abone olunan konuyla ilişkili model kuyruğunun **DefPriority** özniteliğinden alınır.

#### **PRPUB**

İletinin önceliği, özgün yayının önceliğidir. Bu, alanın ilk değeridir.

SOALT seçeneği kullanılarak var olan bir abonelik değiştiriliyorsa, ilerideki yayın iletilerinin MQPRI değeri değiştirilebilir.

SORESkullanılarak bir MQSUB çağrısından döndüğünüzde, bu alan abonelik için kullanılmakta olan yürürlükteki önceliğe ayarlanır.

### **SDRO (MQCHARV)**

SDRO, kuyruk yöneticisi SDON içinde sağlanan adı çözdükten sonra uzun nesne adıdır.

Uzun nesne adı SDOS içinde sağlanırsa ve SDON içinde hiçbir şey sağlanmazsa, bu alanda döndürülen değer, SDOS içinde sağlananla aynıdır.

Bu alan atlanırsa ( SDRO.VSBufSize sıfır), SDRO döndürülmez, ancak uzunluk SDRO.VSLength. Uzunluk tam SDROdeğerinden kısaysa, kesilir ve sağlanan uzunluğa sığabilecek en sağdaki karakterlerin sayısını döndürür.

SDRO yanlış belirtilirse, MQCHARV yapısının nasıl kullanılacağına ilişkin açıklamaya göre ya da uzunluk üst sınırını aşarsa, çağrı RC2520 neden koduyla başarısız olur.

### **SDSID (4 baytlık karakter dizisi)**

Bu yapı tanıtıcısıdır; değer şu olmalıdır:

## **SDSIDV**

Abonelik Tanımlayıcı yapısına ilişkin tanıtıcı.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri SDSIDV ' dir.

## **SDSL (10 basamaklı işaretli tamsayı)**

Bu, abonelik ile ilişkili düzeydir. Yayınlar, yalnızca yayın sırasında kullanılan PubLevel düzeyinden küçük ya da bu değere eşit en yüksek *SDSL* değerine sahip abonelikler kümesinde yer alıyorsa bu aboneliğe teslim edilir.

Değer, sıfır ile 9 aralığında olmalıdır. Sıfır en düşük düzeydir.

Bu alanın ilk değeri 1 'dir.

Var olan bir abonelik *SOALT* seçeneği kullanılarak değiştiriliyorsa, *SDSL* değiştirilemez.

## **SDSN (MQCHARV)**

SDSN abonelik adını belirtir.

Bu alan yalnızca *SDOPT* *SODUR* seçeneğini belirtiyorsa, ancak sağlandıysa *SONDUR* kuyruk yöneticisi tarafından da kullanılır. Belirtilirse, *SDSN* kuyruk yöneticisi içinde benzersiz olmalıdır; bu alan abonelikleri tanımlamak için kullanılır.

*SDSN* uzunluk üst sınırı 10240 'tır.

Bu alan iki amaca hizmet eder. *SODUR* aboneliği için, abonelik tanıtıcısını kapattıysanız ( *COKPSB* seçeneğini kullanarak) ya da kuyruk yöneticisiyle bağlantınız kesildiyse, bu abonelik oluşturulduktan sonra sürdürülecek bir aboneliği tanımlama aracıdır. Yaratıldıktan sonra kaldırılacak aboneliğin tanımlanması, *SORES* seçeneğiyle *MQSUB* çağrısı kullanılarak gerçekleştirilir. *SDSN* alanı, *DISPLAY* *SBSTATUS* içindeki *SDSN* alanındaki aboneliklerin yönetim görünümünde de görüntülenir.

*SDSN* yanlış belirtilirse, *MQCHARV* yapısının nasıl kullanılacağına ya da uzunluk üst sınırını aştığına ya da gerektiğinde atlandığına ( *SDSN*) göre. *VCHRL* sıfır) ya da uzunluk üst sınırını aşarsa, çağrı *RC2440* neden koduyla başarısız olur.

Bu bir giriş alanıdır. Bu yapıdaki alanların başlangıç değerleri, *MQCHARV* yapısındaki değerlerle aynıdır.

Var olan bir abonelik *SOALT* seçeneği kullanılarak değiştiriliyorsa, abonelik adı, aboneliği tanımlamak için kullanılan alan olduğundan değiştirilemez. *SORES* seçeneğiyle bir *MQSUB* çağrısının çıkışında değiştirilmez.

## **SDSS (MQCHARV)**

*SDSS* , bir konudan gelen iletilere abone olurken kullanılan seçim ölçütlerini sağlayan dizedir.

Bu değişken uzunluklu alan, *SORES* seçeneği kullanılarak bir *MQSUB* çağrısının çıkışında, bir arabellek sağlandıysa ve *VSBufSize* içinde de pozitif bir arabellek uzunluğu varsa döndürülür. Çağrıda arabellek sağlanmazsa, *MQCHARV* ' nin *VSL* kalınlık alanında yalnızca seçim dizgisinin uzunluğu döndürülür. Sağlanan arabellek alanı döndürmek için gereken alandan küçükse, sağlanan arabellekte yalnızca *VSBufSize* byte döndürülür.

*SDSS* yanlış belirtilirse, *MQCHARV* yapısının nasıl kullanılacağına ilişkin açıklamaya göre ya da uzunluk üst sınırını aşarsa, çağrı *RC2519* neden koduyla başarısız olur.

## **SDSUD (MQCHARV)**

Bu alandaki abonelikte sağlanan veriler, bu aboneliğe gönderilen her yayının *mq.SubUserData* ileti özelliği olarak içerilir.

*SDSUD* uzunluk üst sınırı 10240 'tır.

*SDSUD* yanlış belirtilirse, *MQCHARV* yapısının nasıl kullanılacağına ilişkin açıklamaya göre ya da uzunluk üst sınırını aşarsa, çağrı başarısız olur; neden kodu *RC2431*.

Bu bir giriş alanıdır. Bu yapıdaki alanların başlangıç değerleri, *MQCHARV* yapısındaki değerlerle aynıdır.

Var olan bir abonelik *SOALT* seçeneği kullanılarak değiştiriliyorsa, abonelik kullanıcı verileri değiştirilebilir.



Bu deęişken uzunluklu alan, bir arabellek saęlandıysa ve *VSBufleni* içinde pozitif bir arabellek uzunluęu varsa, SORES seçeneęi kullanılarak MQSUB çağrısının çıkışında döndürülür. Çaęrıda arabellek saęlanmazsa, MQCHARV ' nin *VCHRL* alanında yalnızca abonelik kullanıcı verilerinin uzunluęu döndürülür. Saęlanan arabellek, alanı döndürmek için gereken alandan küçükse, saęlanan arabellekte yalnızca *VSBuflen* byte döndürülür.

### SDVER (10 basamaklı işaretli tamsayı)

Bu, yapı sürümü numarasıdır; deęer řu olmalıdır:

#### SDVER1

Version-1 Abonelik Tanımlayıcı yapısı.

Ařaęıdaki deęişmez, yürürlükteki sürümün sürüm numarasını belirtir:

#### SDVERC

Abonelik Tanımlayıcı yapısının yürürlükteki sürümü.

Bu her zaman bir giriş alanıdır. Alanın ilk deęeri SDVER1' dir.

### Başlangıçtaki deęerler

Çizelge 733. MQSD ' deki alanların başlangıç deęerleri		
Alan adı	Deęişmezin adı	Deęişmezin deęeri
SDSID	SDSIDV	'SD↯↯'
SDVER	SDVER1	1
SDOPT	SONDUR.	0
SDON	Yok	Boşluklar
SDAU	Yok	Boşluklar
SDASI	SINONE.	Boş Deęerler
SDEXP	EIULIM	-1
SDOS	MQCHARV için tanımlanan adlar ve deęerler	
SDSN	MQCHARV için tanımlanan adlar ve deęerler	
SDSUD	MQCHARV için tanımlanan adlar ve deęerler	
SDCID	TAMAMDıR.	Boş Deęerler
SDPRI	PRQDEF	-3
SDACC	YOK	Boş Deęerler
SDAID	Yok	Boşluklar
SDSL	Yok	1
SDRO	MQCHARV ' de tanımlandıęı řekilde adlar ve deęerler	
<b>Not:</b>		
1. ↯ simgesi, tek bir boş karakteri temsil eder.		

## RPG bildirimi

```
D*..1....:....2....:....3....:....4....:....5....:....6....:....7..
D* MQSD Structure
D*
D* Structure identifier
D SDSID          1          4
D* Structure version number
D SDVER          5          8I 0
D* Options associated with subscribing
D SDOPT          9          12I 0
D* Object name
D SDON           13         60
D* Alternate user identifier
D SDAU           61         72
D* Alternate security identifier
D SDASI          73         112
D* Expiry of Subscription
D SDEXP          113        116I 0
D* Object Long name
D SDOSP          117        132*
D SDOSO          133        136I 0
D SDOSS          137        140I 0
D SDOSL          141        144I 0
D SDOSC          145        148I 0
D* Subscription name
D SDSNP          149        164*
D SDSNO          165        168I 0
D SDSNS          169        172I 0
D SDSNL          173        176I 0
D SDSNC          177        180I 0
D* Subscription User data
D SDSUDP         181        196*
D SDSUDO         197        200I 0
D SDSUDS         201        204I 0
D SDSUDL         205        208I 0
D SDSUDC         209        212I 0
D* Correlation Id related to this subscription
D SDCID          213        236
D* Priority set in publications
D SDPRI          237        240I 0
D* Accounting Token set in publications
D SDACC          241        272
D* Appl Identity Data set in publications
D SDAID          273        304
D* Message Selector
D SDSSP          305        320*
D SDSSO          321        324I 0
D SDSSS          325        328I 0
D SDSSL          329        332I 0
D SDSSC          333        336
D* Subscription level
D SDSL           337        340 0
D* Resolved Long object name
D SDROP          341        356*
D SDR00          357        360I 0
D SDR0S          361        364I 0
D SDR0L          365        368I 0
D SDR0C          369        372I 0
```

IBM i

## IBM üzerinde MQSMPO (İleti özelliği seçeneklerini belirle)

**MQSMPO** yapısı, uygulamaların, iletilerin özelliklerini nasıl ayarlayacağını denetleyen seçenekleri belirlemesine olanak sağlar.

### Genel Bakış

**Amaç:** Yapı, **MQSETMP** çağrısında bir giriş parametresidir.

**Karakter kümesi ve kodlama:** **MQSMPO** içindeki veriler, uygulamanın uygulama ve kodlamasının (ENNAT) karakter kümesinde yer almalıdır.

- [“Alanlar” sayfa 1207](#)
- [“Başlangıçtaki değerler” sayfa 1208](#)

- “RPG bildirim” sayfa 1208

## Alanlar

MQSMPO yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

### **SPOPT (10 basamaklı işaretli tamsayı)**

**Konum seçenekleri:** Aşağıdaki seçenekler, özelliğin özellik imleciyle karşılaştırılan görelî konularıyla ilgilidir:

#### **SSPSETF**

Belirtilen adla eşleşen ilk özelliğin değerini ayarlar ya da yoksa, eşleşen bir sıradüzene sahip diğer tüm özelliklerden sonra yeni bir özellik ekler.

#### **SSPSETC**

Özellik imlecine işaret edilen özelliğin değerini ayarlar. İmlecın gösterdiği özellik, IPINQF ya da IPINQN seçeneğini kullanarak en son sorgulanana işaret eder.

İleti tanıtıcısı yeniden kullanıldığında ya da bir MQGET çağrısındaki MQGMO yapısının *HMSG* alanında ya da MQPUT çağrısındaki MQPMO yapısındaki ileti tanıtıcısı belirtildiğinde, özellik geçici çizelgesi sıfırlanır.

Özellik imleci henüz oluşturulmadığında ya da özellik imlecının gösterdiği özellik silindiğinde bu seçenek kullanılırsa, çağrı tamamlanma kodu CCFAIL ve neden kodu RC2471 ile başarısız olur.

#### **SPSETA**

Özellik imlecine işaret eden özellik sonrasında yeni bir özellik ayarlar. İmlecın gösterdiği özellik, IPINQF ya da IPINQO seçeneğini kullanarak en son sorgulanana işaret eder.

İleti tanıtıcısı yeniden kullanıldığında ya da bir MQGET çağrısındaki MQGMO yapısının *HMSG* alanında ya da MQPUT çağrısındaki MQPMO yapısındaki ileti tanıtıcısı belirtildiğinde, özellik geçici çizelgesi sıfırlanır.

Özellik imleci henüz oluşturulmadığında ya da özellik imlecının gösterdiği özellik silindiğinde bu seçenek kullanılırsa, çağrı tamamlanma kodu CCFAIL ve neden kodu RC2471 ile başarısız olur.

Tanımlanan seçeneklerden hiçbirine gerek duyarsanız, aşağıdaki seçeneği kullanın:

#### **SİSTAN**

Seçenek belirtilmedi.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri SPSETF 'dir.

### **SPSID (10 basamaklı işaretli tamsayı)**

Bu, yapı tanıtıcısıdır; değer şöyle olmalıdır:

#### **SPSIDV**

İleti özelliği seçenekleri yapısının belirlenmesine ilişkin tanıtıcı.

Bu her zaman bir giriş alanıdır. Bu alanın başlangıç değeri **SPSIDV** 'dir.

### **SPVAKCSI (10 basamaklı işaretli tamsayı)**

Değer bir karakter dizilimiye, ayarlanacak özellik değerinin karakter kümesi.

Bu her zaman bir giriş alanıdır. Bu alanın başlangıç değeri **CSAPL** 'dir.

### **SVPALENC (10 basamaklı işaretli tamsayı)**

Değer sayıysa, ayarlanacak özellik değerinin kodlaması.

Bu her zaman bir giriş alanıdır. Bu alanın başlangıç değeri **ENNAT** 'dir.

### **SPVER (10 basamaklı işaretli tamsayı)**

Bu, yapı sürüm numarasıdır; değer şu olmalıdır:

## SPVER1

Version-1 , ileti özelliği seçenekleri yapısını ayarlar.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

## SİSTEK

İleti özelliği seçenekleri yapısının yürürlükteki sürümü.

Bu her zaman bir giriş alanıdır. Bu alanın başlangıç değeri **SPVER1**' dir.

## Başlangıçtaki değerler

Çizelge 734. MQSMPO içindeki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
SPSID	SPSIDV	' SMPO '
SPVER	SPVER1	1
SPOPT	SİSTAN	0
SPVALENC	ENNAT	Ortama göre değişir
SPVALCSI	CSAPL	-3

## RPG bildirimi

```
D* MQSMPO Structure
D*
D*
D* Structure identifier
D  SPSID          1      4  INZ('SMPO')
D*
D* Structure version number
D  SPVER          5      8I 0 INZ(1)
D*
** Options that control the action of
D* MQSETMP
D  SPOPT          9      12I 0 INZ(0)
D*
D* Encoding of Value
D  SPVALENC      13      16I 0 INZ(273)
D*
D* Character set identifier of Value
D  SPVALCSI      17      20I 0 INZ(-3)
```

## IBM i IBM üzerinde MQSRO (Abonelik İsteği Seçenekleri)

MQSRO yapısı, uygulamanın bir abonelik isteğinin nasıl yapıldığını denetleyen seçenekleri belirlemesine olanak sağlar.

## Genel Bakış

**Amaç:** Yapı, MQSUBRQ çağrısındaki bir giriş/çıkış parametresidir.

**Sürüm:** MQSRO ' nun yürürlükteki sürümü: SRVER1.

- “Alanlar” sayfa 1208
- “Başlangıçtaki değerler” sayfa 1209
- “RPG bildirimi” sayfa 1210

## Alanlar

MQSRO yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

### **SRNMP (10 basamaklı işaretli tamsayı)**

Bu bir çıkış alanıdır ve bu çağrının sonucu olarak abonelik kuyruğuna gönderilen yayın sayısını belirtmek için uygulamaya geri döndürülür. bu çağrı sayısının bir sonucu olarak bu sayıda yayın gönderilmesine rağmen, bu kadar çok mesajın, özellikle de kalıcı olmayan mesajlar olması halinde, uygulamanın elde edilmesi için bu kadar çok mesaja ulaşabileceğine dair bir garanti yoktur.

Abone olunması gereken bir genel arama karakteri varsa, birden çok yayın olabilir. If no wildcards were present in the topic string when the subscription represented by *HSUB* was created, then at most one publication is sent as a result of this call.

### **SROPT (10 basamaklı işaretli tamsayı)**

Aşağıdaki seçeneklerden biri belirtilmeli. Tek bir seçenek belirtilebilir.

**Diğer seçenekler:** Aşağıdaki seçenek, kuyruk yöneticisi susturulduğunda ne olacağını denetler:

#### **SRFIQ**

Kuyruk yöneticisi susturma durumdaysa, MQSUBRQ çağrısı başarısız olur.

**Varsayılan seçenek:** Önceden açıklanan seçenek gerekmiyorsa, aşağıdaki seçeneğin kullanılması gerekir:

#### **YOK**

Başka bir seçenek belirtilmemeyi belirtmek için bu değeri kullanın; tüm seçenekler varsayılan değerlerini kabul eder.

SRNONE, program belgelerine yardımcı olur. Bu seçeneğin başka biriyle birlikte kullanılması amaçlanmasa da, değeri sıfır olduğu için bu kullanım saptanamaz.

### **SRSID (4 baytlık karakter dizilimi)**

Bu, yapı tanıtıcısıdır; değer şöyle olmalıdır:

#### **SİRSİS**

Abonelik İsteği SROPT yapısına ilişkin tanıtıcı.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri SRSIDV ' dir.

### **SRVER (10 basamaklı işaretli tamsayı)**

Bu, yapı sürüm numarasıdır; değer şu olmalıdır:

#### **SRVER1**

Version-1 Abonelik İsteği Seçenekleri yapısı.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

#### **SİLAC**

Abonelik İsteği Seçenekleri yapısının geçerli sürümü.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri SRVER1' dir.

### **Başlangıçtaki değerler**

<i>Çizelge 735. MQSRO ' daki alanların ilk değerleri</i>		
<b>Alan adı</b>	<b>Değişmezin adı</b>	<b>Değişmezin değeri</b>
<i>SRSID</i>	SİRSİS	' SRO→ '
<i>SRVER</i>	SRVER1	1
<i>SROPT</i>	YOK	0
<i>SRNMP</i>	Yok	0

Çizelge 735. MQSTRO 'daki alanların ilk değerleri (devamı var)

Alan adı	Değişmezin adı	Değişmezin değeri
<b>Notlar:</b>		
1. - simgesi tek bir boş karakteri temsil eder.		
2. Boş değerli dizgi ya da boşluklar, C içindeki boş değerli dizeyi ve diğer programlama dillerindeki boş karakterleri belirtir.		

## RPG bildirimi

```
D*..1....:....2.....3.....4.....5.....6.....7..
D* MQSTRO Structure
D*
D* Structure identifier
D SRSID 1 4
D* Structure version number
D SRVER 5 8I 0
D* Options that control the action of MQSUBRQ
D SROPT 9 12I 0
D* Number of publications sent
D SRNMP 13 16I 0
```

IBM i

## MQSTS (Status reporting structure) on IBM i

MQSTS yapısı, MQSTAT komutu tarafından döndürülen durum yapısındaki verileri açıklar.

### Genel Bakış

**Karakter kümesi ve kodlama:** MQSTS 'deki karakter verileri, yerel kuyruk yöneticisinin karakter takımında; bu, *CodedCharSetId* kuyruk yöneticisi özneteliği tarafından verilir. MQSTS içindeki sayısal veriler yerel makine kodlamasıdır; bu *ENNAT* tarafından verilir.

**Kullanım:** MQSTAT komutu, durum bilgilerini almak için kullanılır. Bu bilgiler bir MQSTS yapısıyla döndürülür. MQSTAT ile ilgili bilgi için bkz. [“IBM üzerinde MQSTAT \(Durum bilgilerini al\)” sayfa 1336.](#)

- [“Alanlar” sayfa 1210](#)
- [“Başlangıçtaki değerler” sayfa 1213](#)
- [“RPG bildirimi” sayfa 1214](#)

### Alanlar

MQSTS yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

#### STSCC (10 basamaklı işaretli tamsayı)

Bu, MQSTS yapısındaki bildirilen ilk hatadan kaynaklanan tamamlama kodudur.

Bu her zaman bir çıkış alanıdır. Bu alanın ilk değeri CCOK 'tır.

#### STSTFC (10 basamaklı işaretli tamsayı)

Bu, başarısız olan zamanuyumsuz çağrılarının sayısıdır.

Bu bir çıkış alanıdır. Bu alanın ilk değeri 0 'tır.

#### STSOBJN (48 baytlık karakter dizilimi)

Bu, ilk hatada yer alan nesnenin yerel adıdır.

Bu bir çıkış alanıdır. Bu alanın ilk değeri 48 boş karakterdir.

### **STSOQMGR (48 baytlık karakter dizilimi)**

Bu ad, *STSOBJN* nesnesinin tanımlandığı kuyruk yöneticisinin adıdır. İlk boş karakteri ya da alanın sonuna kadar boş olan bir ad, uygulamanın bağlı olduğu kuyruk yöneticisini belirtir (yerel kuyruk yöneticisi).

Bu bir çıkış alanıdır. Bu alanın ilk değeri 48 boş karakterdir.

### **STSOO (10 basamaklı işaretli tamsayı)**

The STSOO used to open the object being reported upon. Yalnızca MQSTS ya da daha sonraki bir sürüm olan sürüm 2 'de bulunur.

STSOO değeri, MQSTAT **STYPE** parametresinin değerine bağlıdır.

#### **STATAPT**

Sfır.

#### **DURDURC**

Sfır.

#### **DURDURUR**

Hata ortaya çıktığında STSOO kullanılır. Başarısızlığın nedeni, MQSTS yapısındaki *STSCC* ve *STSRC* alanlarında raporlanır.

STSOO , bir çıkış alanıdır. Başlangıç değeri sıfır.

### **STSOS (MQCHARV)**

Raporlanmakta olan başarısız nesnenin uzun nesne adı. Yalnızca MQSTS ya da daha sonraki bir sürüm olan sürüm 2 'de bulunur.

STSOS , uzunluk üst sınırı 10240 olan bir MQCHARV alanıdır. MQCHARV yapısının nasıl kullanılacağını görmek için [MQCHARV](#) başlıklı konuya bakın.

STSOS ' in yorumu, MQSTAT **STYPE** parametresinin değerine bağlıdır.

#### **STATAPT**

This is the long object name of the queue or topic used in the MQPUT operation, which failed.

#### **DURDURC**

Sfır uzunluklu dizgi

#### **DURDURUR**

Bu, yeniden bağlanmanın başarısız olmasına neden olan nesnenin uzun nesne adıdır.

STSOS , bir çıkış alanıdır. Başlangıç değeri sıfır uzunluklu bir dizilimdir.

### **STSOT (10 basamaklı işaretli tamsayı)**

*ObjectName* içinde adlandırılmakta olan nesnenin tipi. Olası değerler şunlardır:

#### **OTALSQ**

Diğer ad kuyruğu.

#### **OTLOCQ**

Yerel kuyruk.

#### **OTMODQ**

Model kuyruğu.

#### **OTQ**

Kuyruk.

#### **OTREMQ**

Uzak kuyruk.

## **OTTOP**

Konu.

Bu her zaman bir çıkış alanıdır. Bu alanın ilk değeri OTQ ' dur.

## **STSRC (10 basamaklı işaretli tamsayı)**

Bu, MQSTS yapısında bildirilen ilk hatadan kaynaklanan neden kodudur.

Bu her zaman bir çıkış alanıdır. Bu alanın ilk değeri RCNONE olur.

## **STSRBJN (48 baytlık karakter dizgisi)**

Yerel kuyruk yöneticisi, adı çözdükten sonra *STSOBJN* içinde belirtilen hedef kuyruğunun adı. The name returned is the name of a queue that exists on the queue manager identified by *STSRQMGR*.

Boş olmayan bir değer yalnızca, nesne göz atma, giriş ya da çıkış (ya da herhangi bir birleşim) için açılmış tek bir kuyruksa döndürülür. Açılan nesne aşağıdakilerden herhangi biriye, *STSRBJN* boşluklara ayarlanır:

- Konu
- Bir kuyruk, ancak göz atma, giriş ya da çıkış için açılmadı

Bu bir çıkış alanıdır. Bu alanın ilk değeri 48 boş karakterdir.

## **STSRQMGR (48 byte 'lık karakter dizgisi)**

Bu ad, yerel kuyruk yöneticisinin adı çözdükten sonra hedef kuyruk yöneticisinin adıdır. Döndürülen ad, *STSRBJN* ile tanımlanan kuyruğa sahip olan kuyruk yöneticisinin adıdır. *STSRQMGR* , yerel kuyruk yöneticisinin adı olabilir.

*STSRBJN* , yerel kuyruk yöneticisinin ait olduğu kuyruk paylaşım grubunun sahip olduğu bir paylaşılan kuyruksa, *STSRQMGR* kuyruk paylaşım grubunun adıdır. If the queue is owned by some other queue sharing group, *STSRBJN* can be the name of the queue sharing group or the name of a queue manager that is a member of the queue sharing group (the nature of the value returned is determined by the queue definitions that exist at the local queue manager).

Boş olmayan bir değer yalnızca, nesne göz atma, giriş ya da çıkış (ya da herhangi bir birleşim) için açılmış tek bir kuyruksa döndürülür. Açılan nesne aşağıdakilerden herhangi biriye, *STSRQMGR* boşluklara ayarlanır:

- Konu
- Bir kuyruk, ancak göz atma, giriş ya da çıkış için açılmadı
- Belirtilen OOBNDN değerine sahip bir küme kuyruğu (ya da **DefBind** kuyruk özneliği OOBNDN değerine sahip olduğunda OOBNDQ ile geçerli olur)

Bu bir çıkış alanıdır. Bu alanın ilk değeri 48 boş karakterdir.

## **STSSC (10 basamaklı işaretli tamsayı)**

Bu, başarılı olan zamanuyumsuz koyma çağrılarının sayısıdır.

Bu bir çıkış alanıdır. Bu alanın ilk değeri 0 'tır.

## **STSSID (4 baytlık karakter dizilimi)**

Bu, yapı tanıtıcısıdır. Değer şu olmalıdır:

### **STSSID**

Durum raporlama yapısına ilişkin tanıtıcı.

Bu alanın ilk değeri STSSID ' dir.

## **STSSO (10 basamaklı işaretli tamsayı)**

The STSSO used to open the failing subscription. Yalnızca MQSTS ya da daha sonraki bir sürüm olan sürüm 2 'de bulunur.

STSSO ' in yorumu, MQSTAT **STYPE** parametresinin değerine bağlıdır.



**STATAPT**

Sfır.

**DURDURC**

Sfır.

**DURDURUR**

Hata ortaya çıktığında STSS0 kullanılır. Başarısızlığın nedeni, MÖSTS yapısındaki STSCC ve STSRC alanlarında raporlanır. Hata bir konuya abone olma ile ilgili değilse, döndürülen değer sıfır olur.

STSS0 , bir çıkış alanıdır. Başlangıç değeri sıfır.

**STSSUN (MÖCHARV)**

Başarısız olan aboneliğin adı. Yalnızca MÖSTS ya da daha sonraki bir sürüm olan sürüm 2 'de bulunur.

STSSUN , maximum uzunluğuna 10240 uzunluğunda bir MÖCHARV alanıdır. MÖCHARV yapısının nasıl kullanılacağını görmek için MÖCHARV başlıklı konuya bakın.

STSSUN ' in yorumu, MÖSTAT **STYPE** parametresinin değerine bağlıdır.

**STATAPT**

Sfır uzunluklu dizgi.

**DURDURC**

Sfır uzunluklu dizgi.

**DURDURUR**

Yeniden bağlantı başarısız olmasına neden olan aboneliğin adı. Herhangi bir abonelik adı yoksa ya da başarısızlık bir abonelikle ilgili değilse, bu sıfır uzunluklu bir dizgidir.

STSSUN , bir çıkış alanıdır. Başlangıç değeri sıfır uzunluklu bir dizilimdir.

**STSVR (10 basamaklı işaretli tamsayı)**

Bu, yapı sürüm numarasıdır. Değer şu olmalıdır:

**STSVR1**

Durum raporlama yapısına ilişkin sürüm numarası.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

**STSVRC**

Durum raporlama yapısının geçerli sürümü.

Bu alanın ilk değeri STSVR1' dir.

**STSWC (10 basamaklı işaretli tamsayı)**

Bu, bir uyarıyla tamamlanan zamanuyumsuz koyma çağrılarının sayısıdır.

Bu bir çıkış alanıdır. Bu alanın ilk değeri 0 'tır.

**Başlangıçtaki değerler**

Çizelge 736. MÖSTS ' deki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
STSSID	STSID	
STSVR	STSVRC	STSVR1
STSCC	CCOK	0
STSRC	YOK	0

Çizelge 736. MQSTS 'deki alanların ilk değerleri (devamı var)

Alan adı	Değişmezin adı	Değişmezin değeri
STSSC	Yok	0
STSWC	Yok	0
STSFC	Yok	0
STSOT	Yok	0
STSOBJN	Yok	Boşluklar
STSOQMGR	Yok	Boşluklar
STSR OBJN	Yok	Boşluklar
STSRQMGR	Yok	Boşluklar
STSOS	MQCHARV için tanımlandığı şekilde adlar ve değerler	
STSSUN	MQCHARV için tanımlandığı şekilde adlar ve değerler	
STS00	Yok	0
STSS0	Yok	0

## RPG bildirim

```

D*..1.....2.....3.....4.....5.....6.....7..
D* MQSTS Structure
D*
D* Structure identifier
D STSSID 1 4
D* Structure version number
D STSVER 5 8I 0
D* Completion code
D STSCC 9 12I 0
D* Reason code
D STSRC 13 16I 0
D* Success count
D STSSC 17 20I 0
D* Warning count
D STSWC 21 24I 0
D* Failure count
D STSFC 25 28I 0
D* Object type
D STSOT 29 32I 0
D* Object name
D STSOBJN 33 80
D* Object queue manager
D STSOQMGR 81 128
D* Resolved object name
D STSR OBJN 129 176
D* Resolved object queue manager name
D STSRQMGR 177 224
D* Ver:1 **
D* Failing object long name
D* Address of variable length string
D STSOSCHRP 225 240*
D* Offset of variable length string
D STSOSCHRO 241 244I 0
D* Size of buffer
D STSOSVSBS 245 248I 0
D* Length of variable length string
D STSOSCHRL 249 252I 0
D* CCSID of variable length string
D STSOSCHRC 253 256I 0
D* Failing subscription name
D* Address of variable length string
D STSSUNCHRP 257 272*

```

```
D* Offset of variable length string
D STSSUNCHRO      273    276I 0
D* Size of buffer
D STSSUNVSB      277    280I 0
D* Length of variable length string
D STSSUNCHRL     281    284I 0
D* CCSID of variable length string
D STSSUNCHRC     285    288I 0
D* Failing open options
D STS00          289    292I 0
D* Failing subscription options
D STSS0         293    296I 0
D* Ver:2 **
```

## MQTM-Tetikleme iletisi

MQTM yapısı, bir kuyruk için tetikleme olayı ortaya çıktığında kuyruk yöneticisi tarafından bir tetikleme izleme uygulamasına gönderilen tetikleyici iletisinde verileri tanımlar.

### Genel Bakış

**Amaç:** Bu yapı, IBM MQ Framework arabirimlerinden biri olan IBM MQ Trigger Monitor Interface (TMI) ürününün bir parçasıdır.

**Biçim adı:** FMTM.

**Karakter kümesi ve kodlama:** MQTM 'deki karakter verileri, MQTM' yi oluşturan kuyruk yöneticisinin karakter takımında yer alıyor. MQTM 'deki sayısal veriler, MQTM' yi oluşturan kuyruk yöneticisinin makine kodlamasında yer alıyor.

The character set and encoding of the MQTM are given by the *MDCSI* and *MDENC* fields in:

- MQMD (MQTM yapısı ileti verilerinin başlangıcındaki bir yapıysa) ya da
- MQTM yapısından önce gelen üstbilgi yapısı (diğer tüm durumlar).

**Kullanım:** Bir tetikleme izleme programı uygulamasının tetikleyici iletisinde bilgilerin bir kısmını ya da tümünü, tetikleme izleme uygulaması tarafından başlatılan uygulamaya geçirmesi gerekebilir. Başlatılan uygulama için gerekli olabilecek bilgiler, *TMQN*, *TMTD* ve *TMUD*' yi içerir. Tetikleme izleme programı, MQTM yapısını doğrudan başlatılan uygulamaya geçirebilir ya da ortam tarafından izin verilen uygulamaya göre ve başlatılan uygulama için uygun olan bir MQTMC2 yapısını geçirebilir. MQTMC2 ile ilgili bilgi için bkz. [“IBM üzerinde MQTMC2 \(Tetikleyici iletisi 2-karakter biçimi\)” sayfa 1219](#).

- IBM üzerinde, IBM MQ ile verilen tetikleme izleme uygulaması, başlatılan uygulamaya bir MQTMC2 yapısı iletir.

Tetikleyiciyle ilgili bilgi için [Tetikleme önkoşulları](#) başlıklı konuya bakın.

- [“Tetikleme iletisi için MQMD” sayfa 1215](#)
- [“Alanlar” sayfa 1216](#)
- [“Başlangıçtaki değerler” sayfa 1218](#)
- [“RPG bildirimi” sayfa 1219](#)

### Tetikleme iletisi için MQMD

*Çizelge 737. Kuyruk yöneticisi tarafından oluşturulan bir tetikleme iletisinin MQMD 'deki alanlarına ilişkin ayarlar*

MQMD ' de alan	Kullanılan değer
MDSID	MDSIDV
MDVER	MDVER1
MDREP	ROYOK
MDMT	MTDGRM

Çizelge 737. Kuyruk yöneticisi tarafından oluşturulan bir tetikleme iletisinin MQMD ' deki alanlarına ilişkin ayarlar (devamı var)

MQMD ' de alan	Kullanılan değer
MDEXP	EIULIM
MDFB	YOK
MDENC	ENNAT
MDCSI	Kuyruk yöneticisinin <b>CodedCharSetId</b> özniteliği
MDFMT	FMTM
MDPRI	Başlatma kuyruğunun <b>DefPriority</b> özniteliği
MDPER	PENPER
MDMID	Benzersiz bir değer
MDCID	CINONE
MDBOC	0
MDRQ	Boşluklar
MDRM	Kuyruk yöneticisinin adı
MDUID	Boşluklar
MDACC	YOK
MDAID	Boşluklar
MDPAT	ATQM ya da ileti kanalı aracısına uygun olan
MDPAN	Kuyruk yöneticisi adının ilk 28 byte 'ı
MDPD	Tetikleme iletisinin gönderildiği tarih
MDPT	Tetikleme iletisinin gönderildiği saat
MDAOD	Boşluklar

Aşağıdaki durumlar dışında, benzer değerleri ayarlamak için bir tetikleme iletisi oluşturan bir uygulama önerilir:

- *MDPRI* alanı PRQDEF olarak ayarlanabilir (kuyruk yöneticisi, ileti konduğunda, başlangıç kuyruğu için bu değeri varsayılan önceliğe göre değiştirir).
- *MDRM* alanı boşluklara ayarlanabilir (kuyruk yöneticisi, iletiyi koyduğunda bu değeri yerel kuyruk yöneticisinin adıyla değiştirecektir).
- Bağlam alanları, uygulama için uygun ayarlanmalıdır.

## Alanlar

MQTM yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

### TMAI (256 baytlık karakter dizilimi)

Uygulama tanıtıcısı.

Bu, başlatılacak uygulamayı tanıtan ve tetikleme iletisini alan tetikleme izleme programı uygulaması tarafından kullanılan bir karakter dizilimidir. Kuyruk yöneticisi, bu alanı, *TMPN* alanı tarafından tanımlanan süreç nesnesinin **App1Id** özniteliğinin değeriyle birlikte başlatır; bu öznitelige ilişkin ayrıntılar için [“IBM üzerindeki süreç tanımlamalarına ilişkin öznitelikler” sayfa 1374 ' e bakın](#). Bu verilerin içeriği kuyruk yöneticisi için bir önem göstermiyor.

The meaning of *TMAI* is determined by the trigger-monitor application. The trigger monitor provided by IBM MQ requires *TMAI* to be the name of an executable program.

Bu alanın uzunluğu LNPROA tarafından verilir. Bu alanın ilk değeri 256 boş karakterdir.

### **TMT (10 basamaklı işaretli tamsayı)**

Uygulama tipi.

Bu, başlatılacak programın niteliyi tanıtır ve tetikleme iletisini alan tetikleme izleme programı uygulaması tarafından kullanılır. Kuyruk yöneticisi, bu alanı, *TMPN* alanı tarafından tanımlanan süreç nesnesinin **App1Type** özniteliğinin değeriyle birlikte başlatır; bu özniteliğe ilişkin ayrıntılar için [“IBM üzerindeki süreç tanımlamalarına ilişkin öznitelikler” sayfa 1374](#) ' e bakın. Bu verilerin içeriği kuyruk yöneticisi için bir önem göstermiyor.

*TMAT* , aşağıdaki standart değerlerden birine sahip olabilir. Kullanıcı tanımlı tipler de kullanılabilir, ancak ATUFST ile ATULST arasındaki değerlerle sınırlandırılmalıdır:

#### **SAATCICS**

CICS işlemi.

#### **ATVSE**

CICS/VSE işlemi.

#### **AT400**

IBM i uygulaması.

#### **ATUFST**

Kullanıcı tanımlı uygulama tipi için en düşük değer.

#### **ATULST**

Kullanıcı tanımlı uygulama tipi için en yüksek değer.

Bu alanın ilk değeri 0 'tır.

### **TMED (128 baytlık karakter dizilimi)**

Ortam verileri.

Bu, başlatılacak uygulamaya ilişkin ortamla ilgili bilgileri içeren ve tetikleme iletisini alan tetikleyici izleme uygulaması tarafından kullanılan bir karakter dizilimidir. Kuyruk yöneticisi, bu alanı, *TMPN* alanı tarafından tanımlanan süreç nesnesinin **EnvData** özniteliğinin değeriyle birlikte başlatır; bu özniteliğe ilişkin ayrıntılar için [“IBM üzerindeki süreç tanımlamalarına ilişkin öznitelikler” sayfa 1374](#) ' e bakın. Bu verilerin içeriği kuyruk yöneticisi için bir önem göstermiyor.

Bu alanın uzunluğu LNPROE tarafından verilir. Bu alanın ilk değeri 128 boş karakterdir.

### **TMPN (48 baytlık karakter dizilimi)**

Süreç nesnesinin adı.

Bu, tetiklenen kuyruk için belirtilen kuyruk yöneticisi işlemi nesnesinin adıdır ve tetikleme iletisini alan tetikleyici izleme programı tarafından kullanılabilir. Kuyruk yöneticisi, bu alanı, *TMQN* alanı tarafından tanımlanan kuyruğun **ProcessName** özniteliğinin değeriyle birlikte başlatır; bu özniteliğin ayrıntıları için [“Kuyruklara ilişkin öznitelikler” sayfa 1345](#) ' e bakın.

Alanın tanımlanan uzunluğundan daha kısa olan adlar her zaman boşluklarla sağa doldurulur; boş bir karakter karakterinden önce sona erdirilmez.

Bu alanın uzunluğu LNPRON tarafından verilir. Bu alanın ilk değeri 48 boş karakterdir.

### **TMQN (48 baytlık karakter dizisi)**

Tetiklenen kuyruğun adı.

Tetikleme olayının olduğu kuyruğun adı ve uygulama tarafından başlatılan uygulama tarafından tetikleme izleme uygulaması tarafından kullanılır. Kuyruk yöneticisi bu alanı, tetiklenen kuyruğun

**QName** özniteliğinin değeriyle kullanıma hazırlar; bu özniteliğe ilişkin ayrıntılar için [“Kuyruklara ilişkin öznitelikler” sayfa 1345 ' e bakın.](#)

Alanın tanımlanan uzunluğundan daha kısa olan adlar boşluklarla doldurulur ve boş bir karakter karakteriyle sona erdirilmez.

Bu alanın uzunluğu LNQN tarafından verilir. Bu alanın ilk değeri 48 boş karakterdir.

#### **TMSID (4 baytlık karakter dizilimi)**

Yapı tanıtıcısı.

Değer şu olmalıdır:

##### **TMSIDV**

Tetikleyici ileti yapısına ilişkin tanıtıcı.

Bu alanın ilk değeri TMSIDV ' dir.

#### **TMTD (64 baytlık karakter dizilimi)**

Verileri tetikler.

Bu, tetikleme iletisini alan, tetikleme izleme uygulaması tarafından kullanılacak serbest biçimli verilerdir. Kuyruk yöneticisi, bu alanı, *TMQN* alanı tarafından tanımlanan kuyruğun **TriggerData** özniteliğinin değeriyle birlikte başlatır; bu özniteliğin ayrıntıları için [“Kuyruklara ilişkin öznitelikler” sayfa 1345 ' e bakın.](#) Bu verilerin içeriği kuyruk yöneticisi için bir önem göstermiyor.

Bu alanın uzunluğu INTRGD tarafından verilir. Bu alanın ilk değeri 64 boş karakterdir.

#### **TMUD (128 baytlık karakter dizilimi)**

Kullanıcı verileri.

Bu, başlatılacak uygulamayla ilgili kullanıcı bilgilerini içeren ve tetikleme iletisini alan tetikleyici izleme uygulaması tarafından kullanılan bir karakter dizilimidir. Kuyruk yöneticisi, bu alanı, *TMPN* alanı tarafından tanımlanan süreç nesnesinin **UserData** özniteliğinin değeriyle birlikte başlatır; bu özniteliğe ilişkin ayrıntılar için [“IBM üzerindeki süreç tanımlamalarına ilişkin öznitelikler” sayfa 1374 ' e bakın.](#) Bu verilerin içeriği kuyruk yöneticisi için bir önem göstermiyor.

Bu alanın uzunluğu LNPROU tarafından verilir. Bu alanın ilk değeri 128 boş karakterdir.

#### **TMVER (10 basamaklı işaretli tamsayı)**

Yapı sürüm numarası.

Değer şu olmalıdır:

##### **TMVER1**

Tetikleyici ileti yapısına ilişkin sürüm numarası.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

##### **TMVERC**

Tetikleyici ileti yapısının yürürlükteki sürümü.

Bu alanın ilk değeri TMVER1' dir.

### **Başlangıçtaki değerler**

Çizelge 738. MQTM ' deki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
<i>TMSID</i>	TMSIDV	' TM <sub>1-1</sub> '
<i>TMVER</i>	TMVER1	1
<i>TMQN</i>	Yok	Boşluklar

Çizelge 738. MQTM 'deki alanların ilk değerleri (devamı var)

Alan adı	Değişmezin adı	Değişmezin değeri
TMPN	Yok	Boşluklar
TMTD	Yok	Boşluklar
TMAT	Yok	0
TMAI	Yok	Boşluklar
TMED	Yok	Boşluklar
TMUD	Yok	Boşluklar

**Notlar:**

1. - simgesi tek bir boş karakteri temsil eder.

## RPG bildirim

```
D*..1.....2.....3.....4.....5.....6.....7..
D*
D* MQTM Structure
D*
D* Structure identifier
D TMSID 1 4 INZ('TM ')
D* Structure version number
D TMVER 5 8I 0 INZ(1)
D* Name of triggered queue
D TMQN 9 56 INZ
D* Name of process object
D TMPN 57 104 INZ
D* Trigger data
D TMTD 105 168 INZ
D* Application type
D TMAT 169 172I 0 INZ(0)
D* Application identifier
D TMAI 173 428 INZ
D* Environment data
D TMED 429 556 INZ
D* User data
D TMUD 557 684 INZ
```

IBM i

## IBM üzerindeMQTMC2 (Tetikleyici iletisi 2-karakter biçimi)

Bir tetikleme izleme programı bir başlatma kuyruğundan bir tetikleme iletisi (MQTM) aldığıında, tetikleme izleyicisinin tetikleyici iletisinde bulunan bilgilerin bazılarını ya da tümünü tetikleme izleyicisinin başlattığı uygulamaya geçirmesi gerekebilir.

### Genel Bakış

**Amaç:** Başlatılan uygulama için gerekli olabilecek bilgiler TC2QN, TC2TDve TC2UDIçerir. Tetikleme izleme uygulaması, MQTM yapısını doğrudan başlatılan uygulamaya geçirebilir ya da ortam tarafından izin verilen uygulamaya ve başlatılan uygulamaya uygun olarak, bunun yerine bir MQTMC2 yapısının iletilmesine neden olabilir.

This structure is part of the IBM MQ Trigger Monitor Interface (TMI), which is one of the IBM MQ framework interfaces.

**Karakter kümesi ve kodlama:** MQTMC2 'deki karakter verileri, yerel kuyruk yöneticisinin karakter takımında yer alıyor; bu, **CodedCharSetId** kuyruk yöneticisi özneliği tarafından verilir.

**Kullanım:** MQTMC2 yapısı, MQTM yapısının biçimi gibidir. Bu fark, MQTM'deki karakter olmayan alanların MQTMC2 'de aynı uzunluktaki karakter alanlarına çevrilir ve yapının sonuna kuyruk yöneticisi adı eklenmektedir.

- IBM üzerinde, IBM MQ ile verilen tetikleyici izleyicisi uygulaması, başlatılan uygulamaya bir MQTMC2 yapısı iletir.
- [“Alanlar” sayfa 1220](#)
- [“Başlangıçtaki değerler” sayfa 1221](#)
- [“RPG bildirim” sayfa 1221](#)

## Alanlar

MQTMC2 yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

### **TC2AI (256 baytlık karakter dizgisi)**

Uygulama tanıtıcısı.

MQTM yapısındaki *TMAI* alanına bakın.

### **TC2AT (4 baytlık karakter dizgisi)**

Uygulama tipi.

Bu alanda her zaman boşluk, özgün tetikleyici iletilisinin MQTM yapısındaki *TMAT* alanındaki değer ne olursa olsun bulunur.

### **TC2ED (128 baytlık karakter dizgisi)**

Ortam verileri.

MQTM yapısındaki *TMED* alanına bakın.

### **TC2PN (48 baytlık karakter dizgisi)**

Süreç nesnesinin adı.

MQTM yapısındaki *TMPN* alanına bakın.

### **TC2QMN (48 baytlık karakter dizgisi)**

Kuyruk yöneticisi adı.

Bu, tetikleme olayının ortaya çıktığı kuyruk yöneticisinin adıdır.

### **TC2QN (48 baytlık karakter dizgisi)**

Tetiklenen kuyruğun adı.

MQTM yapısındaki *TMQN* alanına bakın.

### **TC2SID (4 baytlık karakter dizgisi)**

Yapı tanıtıcısı.

Değer şu olmalıdır:

#### **TCSIDV**

Tetikleyici ileti (karakter biçimi) yapısına ilişkin tanıtıcı.

### **TC2TD (64 baytlık karakter dizgisi)**

Verileri tetikler.

MQTM yapısındaki *TMTD* alanına bakın.

### **TC2UD (128 baytlık karakter dizgisi)**

Kullanıcı verileri.

MQTM yapısındaki *TMUD* alanına bakın.

### **TC2VER (4 baytlık karakter dizgisi)**

Yapı sürüm numarası.



Değer şu olmalıdır:

### TCVER2

Sürüm 2 tetikleme iletisi (karakter biçimi) yapısı.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

### TCEVERC

Tetikleyici iletisi (karakter biçimi) yapısının yürürlükteki sürümü.

## Başlangıçtaki değerler

Çizelge 739. MQTMC2' deki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
TC2SID	TCSIDV	' TMC↵ '
TC2VER	TCVER2	' ↵↵2 '
TC2QN	Yok	Boşluklar
TC2PN	Yok	Boşluklar
TC2TD	Yok	Boşluklar
TC2AT	Yok	Boşluklar
TC2AI	Yok	Boşluklar
TC2ED	Yok	Boşluklar
TC2UD	Yok	Boşluklar
TC2QMN	Yok	Boşluklar

**Notlar:**

- ↵ simgesi tek bir boş karakteri temsil eder.

## RPG bildirim

```
D*.1.....2.....3.....4.....5.....6.....7..
D* MQTMC2 Structure
D*
D* Structure identifier
D TC2SID 1 4
D* Structure version number
D TC2VER 5 8
D* Name of triggered queue
D TC2QN 9 56
D* Name of process object
D TC2PN 57 104
D* Trigger data
D TC2TD 105 168
D* Application type
D TC2AT 169 172
D* Application identifier
D TC2AI 173 428
D* Environment data
D TC2ED 429 556
D* User data
D TC2UD 557 684
D* Queue manager name
D TC2QMN 685 732
```

MQWIH yapısı, z/OS iş yükü yöneticisi tarafından işlenecek bir iletinin başlangıcında var olması gereken bilgileri açıklar.

## Genel Bakış

**Biçim adı:** FMWIH.

**Karakter kümesi ve kodlama:** MQWIH yapısındaki alanlar, MQWIH ' den önce gelen üstbilgi yapısındaki *MDCSI* ve *MDENC* alanları tarafından verilen karakter kümesi ve kodlamadır ve MQWIH uygulama iletisi verilerinin başlangıcındaki bu alanlar tarafından MQWIH yapısındaki alanlar tarafından verilir.

Karakter takımı, kuyruk adlarında geçerli olan karakterler için tek byte 'lık karakterler içeren bir karakter takımı olmalıdır.

**Kullanım:** If a message is to be processed by the z/OS workload manager, the message must begin with an MQWIH structure.

- [“Alanlar” sayfa 1222](#)
- [“Başlangıçtaki değerler” sayfa 1224](#)
- [“RPG bildirimi” sayfa 1224](#)

## Alanlar

MQWIH yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

### WICSI (10 basamaklı işaretli tamsayı)

MQWIH ' ı izleyen verilerin karakter kümesi tanıtıcısı.

Bu, MQWIH yapısını izleyen verilerin karakter kümesi tanıtıcısını belirtir; bu, MQWIH yapısındaki karakter verileri için geçerli değildir.

MQPUT ya da MQPUT1 çağrısında, uygulama bu alanı veri için uygun değere ayarlamalıdır. Aşağıdaki özel değer kullanılabilir:

#### CINT

Bu yapının karakter kümesi tanıtıcısını devralır.

Character data in the data *takip edilen* this structure is in the same character set as this structure.

Kuyruk yöneticisi, bu değeri, iletide belirtilen yapının gerçek karakter kümesi tanıtıcısına göre değiştirir. Hata oluşmaması durumunda, MQGET çağrısıyla CSINHT değeri döndürülmez.

MQMD ' deki *MDPAT* alanının değeri ATBRKR ise, CSINHT kullanılamaz.

Bu alanın ilk değeri CSUNDF 'dir.

### WIENC (10 basamaklı işaretli tamsayı)

MQWIH ' ı izleyen verilerin sayısal kodlaması.

Bu, MQWIH yapısını izleyen verilerin sayısal kodlamasını belirtir; bu, MQWIH yapısındaki sayısal veriler için geçerli değildir.

MQPUT ya da MQPUT1 çağrısında, uygulama bu alanı veri için uygun değere ayarlamalıdır.

Bu alanın ilk değeri 0 'tır.

### WIFLG (10 basamaklı işaretli tamsayı)

İşaretler

Değer şu olmalıdır:

#### KAZANAN

Bayrak yok.

Bu alanın ilk deęeri WINONE deęeridir.

#### **WIFMT (8 baytlık karakter dizilimi)**

MQWIH ' ı izleyen verilerin adını biçimle.

Bu, MQWIH yapısından sonra gelen verilerin biçim adını belirler.

MQPUT ya da MQPUT1 çağrısında, uygulama bu alanı veri için uygun deęere ayarlamalıdır. Bu alanı kodlamaya ilişkin kurallar, MQMD ' deki *MDFMT* alanı için yapılan kurallarla aynıdır.

Bu alanın uzunluęu LNFMT tarafından verilir. Bu alanın ilk deęeri FMNONE ' dir.

#### **WILEN (10 basamaklı işaretili tamsayı)**

MQWIH yapısının uzunluęu.

Deęer řu olmalıdır:

##### **WILEN1**

version-1 iş bilgileri üstbilgisi yapısı uzunluęu.

Ařaęıdaki deęişmez, yürürlükteki sürümün uzunluęunu belirtir:

##### **WILENC**

İş bilgisi üstbilgisi yapısının geçerli sürümü uzunluęu.

Bu alanın ilk deęeri WILEN1' dir.

#### **WIRSV (32 baytlık karakter dizilimi)**

Ayrılmıř.

Bu ayrılmıř bir alandır; boş bırakılmalıdır.

#### **WISID (4 baytlık karakter dizilimi)**

Yapı tanıtıcısı.

Deęer řu olmalıdır:

##### **WISIDV**

İş bilgileri üstbilgi yapısına ilişkin tanıtıcı.

Bu alanın ilk deęeri WISIDV ' dir.

#### **WISNM (32 baytlık karakter dizilimi)**

Hizmet adı.

Bu, iletiyi işlemek için kullanılan hizmetin adıdır.

Bu alanın uzunluęu LNSVNM tarafından verilir. Bu alanın ilk deęeri 32 boş karakterdir.

#### **WISST (8 baytlık karakter dizilimi)**

Hizmet adımı adı.

Bu ad, iletinin ilişkilendirdięi *WISNM* adımının adıdır.

Bu alanın uzunluęu LNSVST tarafından verilir. Bu alanın ilk deęeri 8 boş karakterdir.

#### **WITOK (16 baytlık bit dizesi)**

İleti simgesi.

Bu, iletiyi benzersiz bir şekilde tanımlayan bir ileti belirticidir.

MQPUT ve MQPUT1 çağrıları için bu alan yok sayılır. Bu alanın uzunluęuna LNMTOK deęeri verilir. Bu alanın ilk deęeri MTKON deęeridir.

#### **WIVER (10 basamaklı işaretili tamsayı)**

Yapı sürüm numarası.

Değer şu olmalıdır:

### WIVER1

Version-1 iş bilgileri üstbilgi yapısı.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

### WIVERC

İş bilgisi üstbilgi yapısının yürürlükteki sürümü.

Bu alanın ilk değeri WIVER1' dir.

## Başlangıçtaki değerler

Çizelge 740. MQWIH içindeki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
WISID	WISIDV	'WIH↵'
WIVER	WIVER1	1
WILEN	WILEN1	120
WIENC	Yok	0
WICSI	CSUNDF	0
WIFMT	FMNONE	Boşluklar
WIFLG	KAZANAN	0
WISNM	Yok	Boşluklar
WISST	Yok	Boşluklar
WITOK	MTKNON	Boş Değerler
WIRSV	Yok	Boşluklar

**Notlar:**

1. ↵ simgesi tek bir boş karakteri temsil eder.

## RPG bildirimi

```
D*..1.....2.....3.....4.....5.....6.....7..
D*
D* MQWIH Structure
D*
D* Structure identifier
D  WISID          1      4    INZ('WIH ')
D* Structure version number
D  WIVER          5      8I 0 INZ(1)
D* Length of MQWIH structure
D  WILEN          9      12I 0 INZ(120)
D* Numeric encoding of data that followsMQWIH
D  WIENC         13      16I 0 INZ(0)
D* Character-set identifier of data thatfollows MQWIH
D  WICSI         17      20I 0 INZ(0)
D* Format name of data that followsMQWIH
D  WIFMT         21      28    INZ('      ')
D* Flags
D  WIFLG         29      32I 0 INZ(0)
D* Service name
D  WISNM         33      64    INZ
D* Service step name
D  WISST         65      72    INZ
D* Message token
D  WITOK         73      88    INZ(X'00000000000000-
D                                0000000000000000')
```

## IBM i IBM üzerinde MQXQH (Transmission-queue header)

MQXQH yapısı, ileti kuyruklarında uygulama ileti verilerinin başına önek olarak eklenen bilgileri açıklar.

### Genel Bakış

**Amaç:** İletim kuyruğu, uzak kuyruklara ilişkin iletileri geçici olarak tutan (yerel kuyruk yöneticisine ait olmayan kuyruklar için yazılmış olan) özel bir yerel kuyruk tipidir. İletim kuyruğu, USTRAN değerine sahip **Usage** kuyruk özniteliği tarafından belirlenmektedir.

**Biçim adı:** FMXQH.

**Karakter kümesi ve kodlama:** MQXQH içindeki veriler, C programlama dili için ENNAT tarafından verilen yerel kuyruk yöneticisinin **CodedCharSetId** kuyruk yöneticisi özniteliği ve kodlaması tarafından verilen karakter kümesinde yer almalıdır.

The character set and encoding of the MQXQH must be set into the *MDCSI* and *MDENC* fields in:

- Ayrı MQMD (MQXQH yapısı ileti verilerinin başlangıcındaki ise) ya da
- MQXQH yapısından önce gelen üstbilgi yapısı (diğer tüm vakalar).

**Kullanım:** İletim kuyruğunda bulunan bir ileti *iki* ileti tanımlayıcısıdır:

- Bir ileti tanımlayıcısı ileti verilerinden ayrı olarak saklanır; bu, *ayrı ileti tanımlayıcısı* olarak adlandırılır ve ileti, ileti iletim kuyruğuna yerleştirildiğinde kuyruk yöneticisi tarafından oluşturulur. Ayrı ileti tanımlayıcısındaki bazı alanlar, MQPUT ya da MQPUT1 çağrısında uygulama tarafından sağlanan ileti tanımlayıcısından kopyalanır.

İleti tanımlayıcısı, ileti iletim kuyruğundan kaldırıldığında, MQGET çağrısının **MSGDSC** parametresindeki uygulamaya döndürülen ayrı ileti tanımlayıcısına sahip olur.

- İleti verilerinin bir parçası olarak, MQXQH yapısı içinde ikinci bir ileti tanımlayıcısı saklanır; bu, *yerleşik ileti tanımlayıcısı* olarak adlandırılır ve uygulamanın MQPUT ya da MQPUT1 çağrısında (ikincil varyasyonlarla) sağladığı bir ileti tanımlayıcısının bir kopyasıdır.

Katıştırılmış ileti tanımlayıcısı her zaman bir version-1 MQMD ' dir. Uygulamanın koyduğu ileti, MQMD 'deki bir ya da daha çok version-2 alanı için varsayılan olmayan değerlere sahipse, MQXQH' yi izleyen bir MQMDE yapısı ve ardından uygulama ileti verileri (varsa) sırası gelir. MQMDE aşağıdakilerden biri olabilir:

- Kuyruk yöneticisi tarafından üretilir (uygulama iletiyi koymak için bir version-2 MQMD kullanıyorsa) ya da
- Uygulama ileti verilerinin başında zaten var (uygulama iletiyi koymak için bir version-1 MQMD kullanıyorsa).

Katıştırılmış ileti tanımlayıcısı, ileti son hedef kuyruğundan kaldırıldığında, MQGET çağrısının **MSGDSC** parametresindeki uygulamaya döndürülen bir tanımlayıcıdır.

- [“Ayrı ileti tanımlayıcısındaki alanlar” sayfa 1226](#)
- [“Katıştırılmış ileti tanımlayıcısındaki alanlar” sayfa 1227](#)
- [“İletileri Uzak Kuyruklara Koyma” sayfa 1227](#)
- [“İletilerin doğrudan iletim kuyruklarına koyma” sayfa 1227](#)
- [“İletim kuyruklarından ileti alınması” sayfa 1228](#)
- [“Alanlar” sayfa 1228](#)
- [“Başlangıçtaki değerler” sayfa 1229](#)
- [“RPG bildirimi” sayfa 1229](#)

## Ayrı ileti tanımlayıcısındaki alanlar

Ayrı ileti tanımlayıcısındaki alanlar, kuyruk yöneticisi tarafından aşağıdaki listede gösterildiği gibi ayarlanır. Kuyruk yöneticisi version-2 MQMD ' yi desteklemiyorsa, işlev kaybı olmadan bir version-1 MQMD kullanılır.

Çizelge 741. Ayrı ileti tanımlayıcısındaki ve kullanılan değerlerdeki alanlar

Ayrı MQMD ' de alan	Kullanılan değer
MDSID	MDSIDV
MDVER	MDVER2
MDREP	Katıştırılmış ileti tanımlayıcısından kopyalandı, ancak ROAUXM tarafından tanımlanan bitler sıfıra ayarlandı. (Bu işlem, bir ileti yerleştirildiğinde ya da iletim kuyruğundan kaldırıldığında, COA ya da COD (COD) rapor iletilsinin oluşturulmasını önler.)
MDMT	Katıştırılmış ileti tanımlayıcısından kopyalandı.
MDEXP	Katıştırılmış ileti tanımlayıcısından kopyalandı.
MDFB	Katıştırılmış ileti tanımlayıcısından kopyalandı.
MDENC	ENNAT
MDCSI	Kuyruk yöneticisinin <b>CodedCharSetId</b> özneliği.
MDFMT	FMXQH
MDPRI	Katıştırılmış ileti tanımlayıcısından kopyalandı.
MDPER	Katıştırılmış ileti tanımlayıcısından kopyalandı.
MDMID	Kuyruk yöneticisi yeni bir değer üretti. Bu ileti tanıtıcısı, kuyruk yöneticisinin katıştırılmış ileti tanımlayıcısı için oluşturduğu <i>MDMID</i> ' den farklıdır (daha önce açıklanan şekilde bakın).
MDCID	Yerleşik ileti tanımlayıcısından <i>MDMID</i> .
MDBOC	0
MDRQ	Katıştırılmış ileti tanımlayıcısından kopyalandı.
MDRM	Katıştırılmış ileti tanımlayıcısından kopyalandı.
MDUID	Katıştırılmış ileti tanımlayıcısından kopyalandı.
MDACC	Katıştırılmış ileti tanımlayıcısından kopyalandı.
MDAID	Katıştırılmış ileti tanımlayıcısından kopyalandı.
MDPAT	ATQM
MDPAN	Kuyruk yöneticisi adının ilk 28 byte 'ı.
MDPD	İletinin iletim kuyruğuna konduğu tarih.
MDPT	İletinin iletim kuyruğuna konduğu saat.
MDAOD	Boşluklar
MDGID	GINONE
MDSEQ	1
MDOFF	0
MDMFL	MFYOK
MDOLN	OUNDF

## Katıştırılmış ileti tanımlayıcısındaki alanlar

Gömülü ileti tanımlayıcısındaki alanların değerleri, aşağıdaki durumlar dışında, MQPUT ya da MQPUT1 çağrısının **MSGDSC** değiştirgesindeki değerlerle aynı olur:

- **MDVER** alanı her zaman MDVER1 değerine sahiptir.
- **MDPRI** alanında PRQDEF değeri varsa, bu değer, kuyruğun **DefPriority** özniteliğinin değeriyle değiştirilir.
- **MDPER** alanında PEQDEF değeri varsa, bu değer, kuyruğun **DefPersistence** özniteliğinin değeriyle değiştirilir.
- **MDMID** alanında MINONE değeri varsa ya da PMNMID seçeneği belirtildiyse ya da ileti bir dağıtım listesi iletirse, **MDMID** , kuyruk yöneticisi tarafından oluşturulan yeni bir ileti tanıtıcısıyla değiştirilir.

Bir dağıtım listesi iletisi, farklı iletim kuyruklarına yerleştirilen daha küçük dağıtım listesi iletilerine bölündüğünde, yeni yerleşik ileti tanımlayıcılarının her birindeki **MDMID** alanı, özgün dağıtım listesi iletisinde olduğu gibi aynıdır.

- **PMNCID** seçeneği belirtilmişse, **MDCID** , kuyruk yöneticisi tarafından oluşturulan yeni bir ilinti tanıtıcısıyla değiştirilir.
- Bağlam alanları, **PMO** değiştirgesinde belirtilen PM\* seçenekleri tarafından belirtildiği gibi ayarlanır; bağlam alanları şunlardır:
  - **MDACC**
  - **MDAID**
  - **MDAOD**
  - **MDPAN**
  - **MDPAT**
  - **MDPD**
  - **MDPT**
  - **MDUID**
- version-2 alanları (varsa) MQMD ' den kaldırılır ve version-2 alanlarından biri ya da daha fazlası varsayılan olmayan bir değere sahipse, MQMD yapısına taşınmıştır.

## İletileri Uzak Kuyruklara Koyma

: Bir uygulama uzak kuyruğa (doğrudan uzak kuyruğun adını belirterek ya da uzak kuyruğun yerel tanımlamasını kullanarak) uzak bir kuyruğa ileti koyduğunda, yerel kuyruk yöneticisi:

- Yerleşik ileti tanımlayıcısını içeren bir MQXQH yapısı yaratır
- Bir MQMDE gerekiyorsa, bir MQMDE ekler ve önceden var değilse
- Uygulama iletisi verilerini ekler
- İletiyi uygun bir iletim kuyruğuna yerleştirir

## İletilerin doğrudan iletim kuyruklarına koyma

Ayrıca, bir uygulamanın doğrudan iletim kuyruğuna bir ileti koyması da mümkündür. Bu durumda, uygulama, uygulama iletisi verilerinin bir MQXQH yapısıyla önekinde ve alanları uygun değerlerle kullanıma hazırlamalıdır. Buna ek olarak, MQPUT ya da MQPUT1 çağrısının **MSGDSC** parametresindeki **MDFMT** alanı FMXQH değerine sahip olmalıdır.

Uygulama tarafından yaratılan MQXQH yapısındaki karakter verileri, yerel kuyruk yöneticisinin karakter kümesinde ( **CodedCharSetId** kuyruk yöneticisi özniteliği tarafından tanımlanır) olmalıdır ve tamsayı verileri yerli makine kodlamasında olmalıdır. Ayrıca, MQXQH yapısındaki karakter verileri, alanın tanımlı uzunluğuna kadar boşluklarla doldurulmalıdır; kuyruk yöneticisi boş değerli ve sonraki karakterleri MQXQH yapısındaki boşluklara dönüştürmediği için, boş bir karakter kullanılarak önceden sonlandırılmamalı.

Ancak, kuyruk yöneticisinin bir MQXQH yapısının var olduğunu ya da alanlar için geçerli değerlerin belirtildiğini göz önünde bulundurun.

## İletim kuyruklarından ileti alınması

Bir iletim kuyruğundan ileti alan uygulamaların, MQXQH yapısındaki bilgileri uygun bir şekilde işlemesi gerekir. Uygulama iletileri verilerinin başında MQXQH yapısının bulunması, MQGET çağrısının **MSGDSC** parametresindeki *MDFMT* alanında döndürülmekte olan FMXQH değeri ile belirtilir. **MSGDSC** parametresindeki *MDCSI* ve *MDENC* alanlarında döndürülen değerler, MQXQH yapısındaki karakter ve tamsayı verisinin karakter kümesini ve kodlamasını gösterir. Uygulama iletileri verilerinin karakter takımı ve kodlaması, yerleşik ileti tanımlayıcısında *MDCSI* ve *MDENC* alanları tarafından tanımlanır.

## Alanlar

MQXQH yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

### XQMD (MQMD1)

Özgün ileti tanımlayıcısı.

Bu, yerleşik ileti tanımlayıcısıdır ve ileti ilk olarak uzak kuyruğa konduğunda, MQPUT ya da MQPUT1 çağrısında **MSGDSC** değiştirgesi olarak belirtilen ileti tanımlayıcı MQMD 'nin bir kopyasını içerir.

**Not:** Bu bir version-1 MQMD 'dir.

Bu yapıdaki alanların ilk değerleri, MQMD yapısındaki alanlarla aynıdır.

### XQRQ (48 baytlık karakter dizilimi)

Hedef kuyruğun adı.

This is the name of the message queue that is the apparent eventual destination for the message (this may prove not to be the actual eventual destination if, for example, this queue is defined at *XQRQM* to be a local definition of another remote queue).

İleti bir dağıtım listesi iletiyse (yani, yerleşik ileti tanımlayıcısındaki *MDFMT* alanı *FMDH* 'dir), *XQRQ* boş olur.

Bu alanın uzunluğu LNQN tarafından verilir. Bu alanın ilk değeri 48 boş karakterdir.

### XQRQM (48 baytlık karakter dizgisi)

Hedef kuyruk yöneticisinin adı.

Bu ad, ileti için nihai hedef olan kuyruğa sahip kuyruk yöneticisi ya da kuyruk paylaşım grubunun adıdır.

İleti bir dağıtım listesi iletiyse, *XQRQM* boş olur.

Bu alanın uzunluğu LNQM tarafından verilir. Bu alanın ilk değeri 48 boş karakterdir.

### XQSID (4 baytlık karakter dizilimi)

Yapı tanıtıcısı.

Değer şu olmalıdır:

#### XQSIDV

İletim kuyruğu üstbilgi yapısına ilişkin tanıtıcı.

Bu alanın ilk değeri XQSIDV 'dir.

### XQVER (10 basamaklı işaretli tamsayı)

Yapı sürüm numarası.

Değer şu olmalıdır:

#### XQVER1

İletim kuyruğu üstbilgi yapısına ilişkin sürüm numarası.



Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

### XQVERC

İletim kuyruğu üstbilgi yapısının yürürlükteki sürümü.

Bu alanın ilk değeri XQVER1' dir.

## Başlangıçtaki değerler

Çizelge 742. MQXQH içindeki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
XQSİD	XQSİDV	'XQH↵'
XQVER	XQVER1	1
XQRQ	Yok	Boşluklar
XQRQM	Yok	Boşluklar
XQMD	MQMD ile aynı adlar ve değerler; bkz. Çizelge 710 sayfa 1132	-

**Notlar:**

- ↵ simgesi tek bir boş karakteri temsil eder.

## RPG bildirim

```
D*..1.....2.....3.....4.....5.....6.....7..
D*
D* MQXQH Structure
D*
D* Structure identifier
D XQSİD 1 4 INZ('XQH ')
D* Structure version number
D XQVER 5 8I 0 INZ(1)
D* Name of destination queue
D XQRQ 9 56 INZ
D* Name of destination queue manager
D XQRQM 57 104 INZ
D* Original message descriptor
D XQ1SID 105 108 INZ('MD ')
D XQ1VER 109 112I 0 INZ(1)
D XQ1REP 113 116I 0 INZ(0)
D XQ1MT 117 120I 0 INZ(8)
D XQ1EXP 121 124I 0 INZ(-1)
D XQ1FB 125 128I 0 INZ(0)
D XQ1ENC 129 132I 0 INZ(273)
D XQ1CSI 133 136I 0 INZ(0)
D XQ1FMT 137 144 INZ(' ')
D XQ1PRI 145 148I 0 INZ(-1)
D XQ1PER 149 152I 0 INZ(2)
D XQ1MID 153 176 INZ(X'00000000000000-
000000000000000000-
00000000000000')
D XQ1CID 177 200 INZ(X'00000000000000-
000000000000000000-
00000000000000')
D XQ1BOC 201 204I 0 INZ(0)
D XQ1RQ 205 252 INZ
D XQ1RM 253 300 INZ
D XQ1UID 301 312 INZ
D XQ1ACC 313 344 INZ(X'00000000000000-
000000000000000000-
000000000000000000-
000000')
D XQ1AID 345 376 INZ
D XQ1PAT 377 380I 0 INZ(0)
D XQ1PAN 381 408 INZ
D XQ1PD 409 416 INZ
```

D	XQ1PT	417	424	INZ
D	XQ1AOD	425	428	INZ

## IBM i Function calls on IBM i

IBM i programlamasındaki işlev çağrılarını hakkında bilgi edinmek için bu bilgileri kullanın.

### IBM i' da arama açıklamalarında kullanılan kurallar

Her çağrı için, bu konu derlemi, çağrıya ilişkin parametrelerin ve kullanımın açıklamasını verir. Bu, çağrıyı tipik çağrılar ve RPG programlama dilinde parametrelerinin tipik bildirimlerine uymaktadır.

**Önemli:** When coding IBM MQ API calls you must ensure that all relevant parameters (as described in the following sections) are provided. Bunun yapmaması, önceden kestirilemeyen sonuçlar üretebilir.

Her bir çağrıya ilişkin açıklama aşağıdaki bölümleri içerir:

#### Arama adı

Arama adı ve ardından arama amacına ilişkin kısa bir açıklama vardır.

#### Parametreler

Her parametre için, adın arkasından parantez içinde veri tipi gelir () ve yönü; örneğin:

*CMPCOD* (9-basamaklı ondalık tamsayı)-çıkış

There is more information about the structure data types in [“Temel veri tipleri” sayfa 982](#).

Parametrenin yönü şu şekilde olabilir:

#### Giriş

Siz (programcı) bu değıştirgeyi sağlamalısınız.

#### Çıkış

Çağrı bu parametreyi döndürür.

#### Giriş/çıkış

Bu parametreyi sağlamanız gerekir, ancak bu parametre çağrıyla değıştirilir.

Parametrenin, parametrenin alabileceği herhangi bir değer listesiyle birlikte, parametrenin amacını açıklayan kısa bir açıklama da vardır.

Her çağrıdaki son iki parametre bir tamamlanma kodu ve neden kodudur. Tamamlama kodu, çağrının başarıyla tamamlanıp tamamlanmadığını, kısmen mi, yoksa hiç mi tamamlanmadığını gösterir. Kısmi başarıyla ilgili daha fazla bilgi ya da çağrı başarısızlığı neden kodunda verilmiştir.

#### Kullanım notları

Çağrıyla ilgili ek bilgi, bunu nasıl kullanacağını ve kullanımıyla ilgili herhangi bir kısıtlamayı tanımlıyor.

#### RPG çağrısı

Çağrıyı tipik olarak çağırma ve parametrelerinin bildirimi, RPG ' de.

Diğer giriş kuralları şunlardır:

#### Değişmezler

Değişmezlerin adları büyük harfle gösterilir; örneğin, OOUT.

#### Diziler

Bazı çağrılarda, parametreler değışmez büyük-küçük boyutlu karakter dizgileri dizileridir. Bu parametrelerin açıklamalarında, küçük harf *n* bir sayısal değışmezi temsil eder. When you code the declaration for that parameter, replace the *n* with the numeric value you require.

## IBM i IBM üzerindeki MQBACK (Geri al değışiklikleri)

MQBACK çağrısı, kuyruk yöneticisine tüm iletinin aldığı ve son eşitleme noktasının geriletmesinden bu yana gerçekleşen tüm çağrılarının oluştuğunu gösterir. Bir iş biriminin bir parçası olarak gönderilen iletiler silinir; iş biriminin bir parçası olarak alınan iletiler kuyruktan geri alınır.

- Bu arama aşağıdaki ortamlarda desteklenir:

-  AIX
-  IBM i
-  Windows

- [“Sözdizimi” sayfa 1231](#)
- [“Kullanım notları” sayfa 1231](#)
- [“Parametreler” sayfa 1232](#)
- [“RPG Bildirimi” sayfa 1233](#)

## Sözdizimi

MQBACK (*Hconn, CompCode, Reason*)

## Kullanım notları

Bu kullanım notlarını, MQBACK kullanırken dikkate alın.

1. Bu arama, yalnızca kuyruk yöneticisinin iş birimini koordine ettiğinde kullanılabilir. Bu, değişikliklerin yalnızca IBM MQ kaynaklarını etkilediği yerel bir iş birimidir.
2. Kuyruk yöneticisinin iş birimini koordine etmediği ortamlarda, MQBACK yerine uygun geri alma çağrısı kullanılmalıdır. Ortam, uygulamanın olağan dışı bir şekilde sonlandırılmasına neden olan örtük bir geri dönüş özelliğini de destekleyebilir.
  - IBM i' ta bu çağrı, kuyruk yöneticisi tarafından koordine edilen yerel iş birimleri için kullanılabilir. This means that a commitment definition must not exist at job level, that is, the STRCMTCTL command with the **CMTSCOPE (\*JOB)** parameter must not have been issued for the job.
3. Bir uygulama, bir iş biriminde kesinleştirilmemiş değişikliklerle sona ererse, bu değişikliklerin yok edilmesi, uygulamanın olağan ya da olağan dışı bir şekilde sona ermesine bağlıdır. Ek ayrıntılar için [“IBM üzerinde MQDISC \(Bağlantı kesme kuyruğu yöneticisi\)” sayfa 1268](#) içindeki kullanım notlarına bakın.
4. Bir uygulama, gruplara ya da mantıksal ileti bölümlerine ileti yerleştirdiğinde ya da ileti aldığı anda, kuyruk yöneticisi, son başarılı MQPUT ve MQGET çağrılarına ilişkin ileti grubuyla ve mantıksal iletiyle ilgili bilgileri saklar. Bu bilgiler kuyruk tanıtıcısı ile ilişkilendirilir ve aşağıdaki gibi şeyler içerir:
  - MQMD ' de *MDGID, MDSEQ, MDOFF* ve *MDMFL* alanlarının değerleri.
  - İletinin bir iş biriminin parçası olup olmadığı.
  - MQPUT çağrısına ilişkin: İletinin kalıcı mı, yoksa kalıcı değil mi olduğu.

The queue manager keeps üç sets of group and segment information, one set for each of the following:

- Son başarılı MQPUT çağrısı (bu, bir iş biriminin bir parçası olabilir).
- Kuyruktan ileti kaldıran son başarılı MQGET çağrısı (bu, bir iş biriminin parçası olabilir).
- Kuyruktan ileti alan son başarılı MQGET çağrısı (bu, bir iş biriminin parçası olamaz).

Uygulama, iletileri bir iş biriminin bir parçası olarak koyduysa ya da alıyorsa ve uygulama iş birimini yedeklemeye karar verirse, grup ve bölüm bilgileri daha önce sahip olduğu değere geri yüklenir:

- MQPUT çağrısıyla ilişkili bilgiler, yürürlükteki iş biriminde o kuyruk tanıtıcısı için ilk başarılı MQPUT çağrısından önce sahip olduğu değere geri yüklendi.
- MQGET çağrısıyla ilişkili bilgiler, yürürlükteki iş biriminde o kuyruk tanıtıcısı için ilk başarılı MQGET çağrısından önce sahip olduğu değere geri yüklenir.

İş birimi başlatıldıktan sonra uygulama tarafından güncellenen, ancak iş biriminin kapsamı dışında olan kuyruklar, iş birimi getirilirse, grup ve bölüm bilgilerini geri yüklemeyebilir.

Bir iş birimi yedeklendiğinde grup ve kesim bilgilerinin önceki değerine geri yüklenmesi, uygulamanın çok sayıda iş biriminden oluşan büyük bir ileti grubu ya da büyük mantıksal ileti yayınlamasını ve iş birimlerinden biri başarısız olursa, ileti grubunun ya da mantıksal iletinin doğru noktasında yeniden başlatılmasına olanak tanır. Yerel kuyruk yöneticisinin kuyruk depolaması sınırlı olması durumunda, birkaç iş birimi kullanılması avantajlı olabilir. Ancak, bir sistem hatası ortaya çıkarsa, uygulama, iletileri yeniden başlatabilmek ya da iletileri doğru noktaya getirmek için yeterli bilgileri sağlamalıdır. Bir sistem hatasından sonra doğru noktada yeniden başlatılabilmeye ilişkin ayrıntılar için, "[MQPMO \(Put-message options\) on IBM i](#)" sayfa 1154'te açıklanan PMLOGO seçeneğine ve "[IBM üzerinde MQGMO \(get-message options\)](#)" sayfa 1060'ta açıklanan GMLOGO seçeneğine bakın.

Kalan kullanım notları, yalnızca kuyruk yöneticisi iş birimlerini koordine ettiğinde geçerlidir:

1. Bir iş birimi, bağlantı tanıtıcısı ile aynı kapsama sahiptir. Başka bir deyişle, belirli bir iş birimini etkileyen tüm IBM MQ çağrılarını, aynı bağlantı tanıtıcısı kullanılarak gerçekleştirilmelidir. Farklı bir bağlantı tanıtıcısı kullanılarak verilen çağrılar (örneğin, başka bir uygulama tarafından yayınlanan çağrılar) farklı bir iş birimini etkiler. Bağlantı tutamaçlarının kapsamı hakkında bilgi için "[IBM üzerinde MQCONN \(Connect kuyruk yöneticisi\)](#)" sayfa 1256 içinde açıklanan **HCONN** parametresine bakın.
2. Bu çağrıdan etkilenenler, yalnızca yürürlükteki iş biriminin bir parçası olarak alınan ya da alınan iletiler etkilenir.
3. Bir iş birimi içinde MQGET, MQPUT ya da MQPUT1 tarafından çağrılan, ancak bir kesinleştirme ya da geriletme çağrısını hiçbir zaman sorun etmemeyen, uzun süredir çalışan bir uygulama, diğer uygulamalar tarafından kullanılanmayan iletileri doldurabilecek kuyruklara neden olabilir. Bu olasılığa karşı koruma sağlamak için, yönetici, **MaxUncommittedMsgs** kuyruk yöneticisi öznelikliğini, kuyrukları dolduran kaçak uygulamaları önleyecek kadar düşük bir değere ayarlamalıdır, ancak beklenen ileti sistemi uygulamalarının doğru şekilde çalışmasına izin verecek kadar yüksek bir değere ayarlanmalıdır.

## Parametreler

MQBACK çağrısı aşağıdaki deęiřtirgeleri içerir:

### **HCONN (10 basamaklı iřaretli tamsayı)-giriř**

Baęlantı tanıtıcısı.

Bu tanıtıcı, kuyruk yöneticisiyle olan baęlantıyı gösterir. The value of *HCONN* was returned by a previous MQCONN or MQCONNX call.

### **CMPCOD (10 basamaklı iřaretli tamsayı)-çıkıř**

Tamamlanma kodu.

Bu, aşağıdakilerden biridir:

#### **CCOK**

İřlem başarıyla tamamlandı.

#### **CCCFAIL**

Arama başarısız oldu.

### **REASON (10 basamaklı iřaretli tamsayı)-çıkıř**

Neden kodu ön eleme *COMCOD*.

*COMCOD* CCOK ise:

#### **YOK**

(0, X'000 ') Raporlamak için bir neden yok.

*COMCOD* CCCFAIL ise:

#### **RC2219**

(2219, X'8AB') MQI çağrısı, önceki arama tamamlanmadan önce yeniden girildi.

**RC2009**

(2009, X'7D9') Kuyruk yöneticisine yönelik bağlantı kaybedildi.

**RC2018**

(2018, X'7E2') Bağlantı tanıtıcısı geçerli değil.

**RC2101**

(2101, X'835 ') Nesne zarar gördü.

**RC2123**

(2123, X'84B') Kesinleştirme ya da geri dönüş işlemi sonucu karışık.

**RC2162**

(2162, X'872 ') Kuyruk yöneticisi sona erdiriliyor.

**RC2102**

(2102, X'836 ') Yetersiz sistem kaynakları var.

**RC2071**

(2071, X'817 ') Kullanılabilir bellek yetersiz.

**RC2195**

(2195, X'893 ') Beklenmeyen bir hata oluştu.

**RPG Bildirimi**

```
C*..1.....2.....3.....4.....5.....6.....7..
C          CALLP          MQBACK(HCONN : COMCOD : REASON)
```

Çağrıya ilişkin prototip tanımı:

```
D*..1.....2.....3.....4.....5.....6.....7..
DMQBACK          PR          EXTPROC('MQBACK')
D* Connection handle
D HCONN          10I 0 VALUE
D* Completion code
D COMCOD          10I 0
D* Reason code qualifying COMCOD
D REASON          10I 0
```

**IBM i IBM üzerinde MQBEGIN (İş birimini başlat)**

MQBEGIN çağırısı, kuyruk yöneticisi tarafından eşgüdümlü bir iş birimi başlatır ve bu, dış kaynak yöneticilerini içermeyebilir.

- Bu arama aşağıdaki ortamlarda desteklenir:

-  AIX
-  IBM i
-  Windows

- [“Sözdizimi” sayfa 1233](#)
- [“Kullanım notları” sayfa 1234](#)
- [“Parametreler” sayfa 1235](#)
- [“RPG Bildirimi” sayfa 1236](#)

**Sözdizimi**

MQBEGIN (HCONN, BEGOP, CMPCOD, REASON)

## Kullanım notları

1. MQBEGIN çağrısı, kuyruk yöneticisi tarafından eşgüdümlü bir iş birimi başlatmak için ve diğer kaynak yöneticilerinin sahip olduğu kaynaklara ilişkin değişiklikleri içerebilecek bir iş birimi başlatmak için kullanılabilir. Kuyruk yöneticisi üç tip iş birimi tipini destekler:

### **Kuyruk yöneticisi-eşgüdümlü yerel iş birimi**

Bu, kuyruk yöneticisinin katılan tek kaynak yöneticisi olduğu bir iş birimidir ve kuyruk yöneticisi iş birimi eşgüdümcüsü olarak işlev görür.

- Bu iş birimi tipini başlatmak için, ilk MQPUT, MQPUT1 ya da MQGET çağrısında PMSYP ya da GMSYP seçeneği belirtilmelidir.

Uygulama, iş birimini başlatmak için MQBEGIN çağrısını yayınlaması gerekmez, ancak MQBEGIN kullanılırsa, çağrıya CCWARN ve neden kodu RC2121 ile arama tamamlanır.

- Bu iş birimi tipini kesinleştirmek ya da yedeklemek için, MQCMIT ya da MQBACK çağrısının kullanılması gerekir.

### **Kuyruk yöneticisi-eşgüdümlü genel iş birimi**

This is a unit of work in which the queue manager acts as the unit-of-work coordinator, both for IBM MQ resources ve for resources belonging to other resource managers. Bu kaynak yöneticileri, iş birimindeki kaynaklarda yapılan tüm değişikliklerin kesinleştirildiğinden ya da yedeklendiğinden emin olmak için kuyruk yöneticisiyle işbirliği yapar.

- Bu tip bir iş birimi başlatmak için, MQSTART çağrısının kullanılması gerekir.
- Bu iş birimi tipini kesinleştirmek ya da yedeklemek için, MQCMIT ve MQBACK çağrılarının kullanılması gerekir.

### **Dışardan eşgüdümlü genel iş birimi**

Bu, kuyruk yöneticisinin bir katılımcı olduğu bir iş birimidir, ancak kuyruk yöneticisi iş birimi eşgüdümcüsü olarak işlev görmez. Bunun yerine, kuyruk yöneticisinin işbirliği yaptığı bir iş birimi eşgüdümcüsü var.

- Bu tip bir çalışma birimi başlatmak için, dış birim iş koordinatörünün sağladığı ilgili çağrı kullanılmalıdır.

İş birimini başlatmayı denemek için MQBEGIN çağrısı kullanılırsa, arama başarısız olur ve neden kodu RC2012 ile başarısız olur.

- Bu tip bir iş birimini kesinleştirmek ya da yedeklemek için, dış birim iş koordinatörünün sağladığı kesinleştirme ve geri gönderme çağrıları kullanılmalıdır.

MQCMIT ya da MQBACK çağrısı çalışma birimini kesinleştirmeyi ya da yedeklemeyi denemek için kullanılırsa, arama başarısız olur ve neden kodu RC2012 ile başarısız olur.

2. Uygulama bir iş biriminde kesinleştirilmemiş değişikliklerle sona ererse, bu değişikliklerin yok edilmesi, uygulamanın olağan ya da olağandışı bir şekilde sona ermesine bağlıdır. Ek ayrıntılar için ["IBM üzerinde MQDISC \(Bağlantı kesme kuyruğu yöneticisi\)" sayfa 1268](#) içindeki kullanım notlarına bakın.
3. Bir uygulama, aynı anda yalnızca bir iş birimine katılabilir. Uygulama için var olan bir iş birimi varsa, bu iş birimi zaten ne olursa olsun, MQBEGIN çağrısı, uygulama için var olan bir iş birimi varsa, neden kodu RC2128 ile başarısız olur.
4. The MQBEGIN call is not valid in an IBM MQ client environment. Çağrıyı kullanma girişimi başarısız oldu; neden kodu RC2012.
5. Kuyruk yöneticisi, genel iş birimleri için iş birimi eşgüdümcüsü olarak hareket ederken, iş birimine katılabilen kaynak yöneticileri, kuyruk yöneticisinin yapılandırma dosyasında tanımlanır.
6. IBM i' ta, üç tip iş birimi aşağıdaki şekilde desteklenir:
  - **Kuyruk yöneticisi-eşgüdümlü yerel iş birimleri** yalnızca, bir kesinleştirme tanımı iş düzeyinde bulunmadığında kullanılabilir. Bu durumda, iş için **CMTSCOPE (\*JOB)** parametresine sahip STRCMTCTL komutu verilmemiş olmalıdır.
  - **Kuyruk yöneticisi-eşgüdümlü genel iş birimleri** desteklenmiyor.

- **Dışarıdan eşgüdümlü genel iş birimleri** yalnızca, bir kesinleştirme tanımı iş düzeyinde olduğunda kullanılabilir; bu durumda, iş için **CMTSCOPE (\*JOB)** parametresiyle STRCMTCTL parametresine sahip bir komut verilmelidir. Bu işlem yapıldıysa, IBM i COMMIT ve ROLLBACK işlemleri, diğer katılımcı kaynak yöneticilerine ait kaynakların yanı sıra IBM MQ kaynakları için de geçerlidir.

## Parametreler

MQBEGIN çağrısında aşağıdaki deęiřtirgeler vardır:

### HCONN (10 basamaklı iřaretli tamsayı)-giriř

Baęlantı tanıtıcısı.

Bu tanıtıcı, kuyruk yöneticisiyle olan baęlantıyı gösterir. The value of *HCONN* was returned by a previous MQCONN or MQCONNX call.

### BEGOP (MQBO)-giriř/çıkıř

MQBEGIN iřlemini denetleyen seęenekler.

Ayrıntılar için bkz. "MQBO (Begin options) on IBM i" sayfa 1003.

Herhangi bir seęenek gerekmiyorsa, C ya da S/390 çevirici programında yazılan programlar, bir MQBO yapısının adresini belirtmek yerine, boş deęerli bir parametre adresi belirtebilir.

### CMPCOD (10 basamaklı iřaretli tamsayı)-çıkıř

Tamamlanma kodu.

Bu, aşağıdakilerden biridir:

#### CCOK

İřlem başarıyla tamamlandı.

#### CCWARN

Uyarı (kısmi tamamlama).

#### CCCFAIL

Arama başarısız oldu.

### REASON (10 basamaklı iřaretli tamsayı)-çıkıř

Neden kodu ön eleme *CMPCOD*.

*CMPCOD* CCOK ise:

#### YOK

(0, X'000 ') Raporlamak için bir neden yok.

*CMPCOD* CCWARN ise:

#### RC2121

(2121, X'849 ') Katılan kaynak yöneticisi kayıtlı deęil.

#### RC2122

(2122, X'84A') Katılan kaynak yöneticisi kullanılmıyor.

*CMPCOD* CCFAIL ise:

#### RC2134

(2134, X'856 ') Begin-options yapısı geçerli deęil.

#### RC2219

(2219, X'8AB') MQI çağrısı, önceki arama tamamlanmadan önce yeniden girildi.

#### RC2009

(2009, X'7D9') Kuyruk yöneticisine yönelik baęlantı kaybedildi.

#### RC2012

(2012, X'7DC') Ortamda arama geçerli deęil.

**RC2018**

(2018, X'7E2') Bağlantı tanıtıcısı geçerli değil.

**RC2046**

(2046, X'7FE') Seçenekler geçerli değil ya da tutarlı değil.

**RC2162**

(2162, X'872 ') Kuyruk yöneticisi sona erdiriliyor.

**RC2102**

(2102, X'836 ') Yetersiz sistem kaynakları var.

**RC2071**

(2071, X'817 ') Kullanılabilir bellek yetersiz.

**RC2195**

(2195, X'893 ') Beklenmeyen bir hata oluştu.

**RC2128**

(2128, X'850 ') İş birimi zaten başlatıldı.

**RPG Bildirimi**

```
C*.1.....2.....3.....4.....5.....6.....7..
C          CALLP      MQBEGIN(HCONN : BEGOP : CMPCOD :
C                                REASON)
```

Çağrıya ilişkin prototip tanımı:

```
D*.1.....2.....3.....4.....5.....6.....7..
DMQBEGIN      PR          EXTPROC('MQBEGIN')
D* Connection handle
D HCONN              10I 0 VALUE
D* Options that control the action of MQBEGIN
D BEGOP              12A
D* Completion code
D CMPCOD              10I 0
D* Reason code qualifying CMPCOD
D REASON              10I 0
```

**IBM i****MQBUFMMH (Arabelleği ileti tutamaçına dönüştür) IBM üzerinde**

MQBUFMMH işlev çağrısı bir arabelleği ileti tanıtıcısı olarak dönüştürür ve MQMHBUF çağrısının tersi olur.

Bu çağrı, arabelleğindeki bir ileti tanımlayıcısını ve MQRFH2 özelliklerini alır ve bunları bir ileti tanıtıcısı aracılığıyla kullanılabilir kılar. İsteğe bağlı olarak, ileti verilerindeki MQRFH2 özellikleri kaldırılır. Özellikler kaldırıldıktan sonra arabelleğin içeriğini doğru şekilde tanımlamak için gerekirse, ileti tanımlayıcısının *Encoding, CodedCharSetIdve Format* alanları güncellenir.

- “Sözdizimi” sayfa 1236
- “Kullanım notları” sayfa 1236
- “Parametreler” sayfa 1237
- “RPG Bildirimi” sayfa 1238

**Sözdizimi**

MQBUFMMH (*Hconn, Hmsg, BufMsgHOpts, MsgDesc, Buffer, BufferLength, DataLength, CompCode, Reason*)

**Kullanım notları**

MQBUFMMH çağrıları API çıktılarıyla algılanamaz; arabellek, uygulama alanındaki bir ileti tanıtıcısı içine dönüştürülür; çağrı kuyruk yöneticisine ulaşmaz.



## Parametreler

MQBUMFH çağrısında şu deęiřtirgeler vardır:

### HCONN (10 basamaklı iřaretli tamsayı)-giriř

Bu tanıtıcı, kuyruk yöneticiyle olan baęlantıyı gösterir. *HCONN* deęeri, **Hmsg** deęiřtirgesinde belirtilen ileti tanıtıcısını yaratmak için kullanılan baęlantı tanıtıcısı ile eřleşmelidir.

İleti tanıtıcısı HCUNAS kullanılarak yaratıldıysa, bir arabelleęi ileti tutamaçına dönüřtüren iř parçacıęıda geçerli bir baęlantı kurulmalıdır. Geçerli bir baęlantı kurulmamıřsa, arama RC2009 ile başarısız olur.

### HMSG (20 basamaklı iřaretli tamsayı)-giriř

Bu tanıtıcı, bir arabelleęin doldurulması gereken ileti tanıtıcısıdır. Deęer önceki bir MQCRTMH çağrısı tarafından döndürüldü.

### BMHOPT (MQBMHO)-giriř

MQBMHO yapısı, uygulamaların, ileti tutamaçlarının arabelleklerden nasıl üretildięini denetleyen seçenekleri belirlemesine olanak saęlar.

Ayrıntılar için bkz. [“IBM üzerinde MQBMHO \(İleti iřleme seçenekleri için arabellek\)” sayfa 1002.](#)

### MSGDSC (MQMD)-giriř/çıkıř

*MSGDSC* yapısı, ileti tanımlayıcı özelliklerini içerir ve arabellek alanının içerięini açıklar.

Çaęrıdan çıktıda, özellikler isteęe baęlı olarak arabellek alanından kaldırılır ve bu durumda, arabellek alanını doęru şekilde tanımlamak için ileti tanımlayıcısı güncellenir.

Bu yapıdaki veriler, uygulamanın karakter kümesi ve kodlamasında olmalıdır.

### BUFLEN (10 basamaklı iřaretli tamsayı)-giriř

*BUFLEN*, arabellek alanının bayt cinsinden uzunluęudur.

Sıfır baytlık bir *BUFLEN* deęeri geçerlidir ve arabellek alanının veri içermedięi anlamına gelir.

### BUFFER (1-byte bit dizgi x BUFLEN)-giriř/çıkıř

*BUFFER*, ileti arabelleęinin bulunduęu alanı tanımlar. Çoęu veri için, arabelleęi 4 byte 'lık bir sınır üzerinde hizalamanız gerekir.

*BUFFER*, karakter ya da sayısal veri içeriyorsa, **MSGDSC** parametresindeki *CodedCharSetId* ve *Encoding* alanlarını, veriye uygun deęerlere ayarlayın; bu, gerekirse verilerin dönüřtürülmesini saęlar.

Özellikler ileti arabelleęinde bulunursa, bunlar isteęe baęlı olarak kaldırılır; daha sonra, çağrıdan dönüřte ileti tanıtıcısından kullanılabilir duruma gelir.

C programlama dilinde, parametre, parametre olarak belirtilebilecek herhangi bir veri tipinin adresi anlamına gelen, iřaretçi (pointer-to-void) olarak bildirilir.

**BUFLEN** parametresi sıfırda, *BUFFER* ifadesine başvurulmaz. Bu durumda, C ya da System/390 çevirici programlarında yazılan programlarla geçirilen parametre adresi boş deęerli olabilir.

### DATLEN (10 basamaklı iřaretli tamsayı)-çıkıř

*DATLEN*, özelliklerin kaldırıldıęı, arabelleęin bayt cinsinden uzunluęudur.

### CMPCOD (10 basamaklı iřaretli tamsayı)-çıkıř

#### CCOK

İřlem başarıyla tamamlandı.

#### CCCFAIL

Arama başarısız oldu.

### REASON (10 basamaklı iřaretli tamsayı)-çıkıř

Neden kodu niteleyici *CMPCOD*.

CMPCOD CCOK ise:

**YOK**

(0, X'000 ') Raporlamak için bir neden yok.

CMPCOD CCFAIL ise:

**RC2204**

(2204, X'089C') Bağdaştırıcı kullanılmıyor.

**RC2130**

(2130, X'852 ') Bağdaştırıcı hizmeti modülü yüklenemiyor.

**RC2157**

(2157, X'86D') Birincil ve ana sunucu ASID değerleri farklı.

**RC2489**

(2489, X'09B9') Arabellek için ileti işleme seçenekleri yapısı geçerli değil.

**RC2004**

(2004, X'07D4') Arabellek parametresi geçerli değil.

**RC2005**

(2005, X'07D5') Arabellek uzunluğu parametresi geçerli değil.

**RC2219**

(2219, X'08AB') Önceki arama tamamlanmadan önce girilen MQI çağırısı.

**RC2009**

(2009, X'07D9') Kuyruk yöneticisine yönelik bağlantı kaybedildi.

**RC2460**

(2460, X'099C') İleti tanıtıcısı geçerli değil.

**RC2026**

(2026, X'07EA') İleti tanımlayıcısı geçerli değil.

**RC2499**

(2499, X'09C3') İleti tanıtıcısı zaten kullanılıyor.

**RC2046**

(2046, X'07FE') Seçenekler geçerli değil ya da tutarlı değil.

**RC2334**

(2334, X'091E') MQRFH2 yapısı geçerli değil.

**RC2421**

(2421, X'0975 ') Özellikler içeren bir MQRFH2 klasörü ayrıştırılmadı.

**RC2195**

(2195, X'893 ') Beklenmeyen bir hata oluştu.

## RPG Bildirimi

```
C*.1.....2.....3.....4.....5.....6.....7..
C                               CALLP      MQBUFMH(HCONN : HMSG : BMHOPT :
                               MSGDSC : BUFLN : BUFFER :
                               DATLEN : CMPCOD : REASON)
```

Çağrıya ilişkin prototip tanımı:

```
DMQBUFMH          PR              EXTPROC('MQBUFMH')
D* Connection handle
D HCONN              10I 0
D* Message handle
D HMSG              10I 0
D* Options that control the action of MQBUFMH
D BMHOPT            12A  VALUE
D* Message descriptor
D MSGDSC              364A
D* Length in bytes of the Buffer area
```

D BUFLN	10I 0
D* Area to contain the message buffer	
D BUFFER	* VALUE
D* Length of the output buffer	
D DATLEN	10I 0
D* Completion code	
D CMPCOD	10I 0
D* Reason code qualifying CompCode	
D REASON	10I 0

## IBM i IBM üzerinde MQCB (Geri çağırma yönet)

MQCB çağırısı, belirtilen nesne tanıtcısı için geri çağırma yeniden kaydettirir ve geri bildirmeye ilişkin etkinleştirme ve değişiklikleri denetler.

Geri çağırma, belirli olaylar oluştuğunda IBM MQ tarafından çağırılan bir kod parçasıdır (dinamik olarak bağlantı oluşturulabilen ya da işlev göstergesi olarak belirtilen).

Bir V7 istemcisinde MQCB ve MQCTL ' yi kullanmak için, bir V7 sunucusuna bağlanmanız gerekir ve kanalın **SHARECNV** parametresine sıfır dışında bir değer olmalıdır.

Genel iş birimleri ile ilgili bilgi için bkz. [Genel çalışma birimleri](#).

Tanımlanabilen geri çağırma tipleri şunlardır:

### İleti tüketicisi

Bir ileti tüketicisi geri bildirme işlevi, bir ileti, belirtilen seçim ölçütlerine uygun olarak bir nesne tanıtcısında kullanılabilir olduğunda çağırılır.

Her nesne tanıtcısı için yalnızca bir geri çağırma işlevi kaydedilebilir. Tek bir kuyruk birden çok seçim ölçütüyle okunacaksa, kuyruğun birden çok kez açılması ve her bir tutamaçla ilgili bir tüketicisi işlevi kaydedilmelidir.

### Olay işleyici

Olay işleyici, tüm geri bildirme ortamını etkileyen koşullar için çağırılır.

Bu işlev, bir olay koşulu oluştuğunda (örneğin, bir kuyruk yöneticisi ya da bağlantı durdurma ya da susturucu) çağırılır.

Bu işlev, tek bir ileti tüketicisi için belirli koşullar (örneğin, RC2016; ) için çağırılmaz, ancak geri çağırma işlevi olağan şekilde bitmezse çağırılır.

- [“Sözdizimi” sayfa 1239](#)
- [“MQCB için kullanım notları” sayfa 1239](#)
- [“MQCB için deęiřtirgeler” sayfa 1241](#)
- [“RPG Bildirimi” sayfa 1247](#)

### Sözdizimi

MQCB (*HCONN, OPERATN, HOBJ, CBDSC, MSGDSC, GMO, CMPCOD, REASON*)

### MQCB için kullanım notları

1. MQCB, kuyruğun kullanılabilir olduğuna belirtilen ölçütlerle eşleşen her ileti için çağırılacak işlemi tanımlamak için kullanılır. İşlem işlendiğinde, ileti kuyruktan kaldırılır ve tanımlı ileti tüketicisine geçirilir ya da iletiyi almak için kullanılan bir ileti simgesi sağlanır.
2. MQCB, MQCTL ile tüketime başlamadan önce geri çağırma yordamlarını tanımlamak için kullanılabilir ya da bir geri çağırma yordamından kullanılabilir.
3. MQCB ' yi bir geri bildirme yordamından uzak kullanmak için önce MQCTL kullanarak ileti tüketimini askıya almanız ve daha sonra tüketime devam etmeniz gerekir.

## İleti tüketicisi geri çağırma sırası

Tüketiciyi, tüketicinin yaşam çevrimi boyunca anahtar noktalarında geri çağırma için bir tüketici yapılandırabilirsiniz. Örneğin:

- tüketicinin ilk kayıt yaptırdığı zaman,
- bağlantı başlatıldığında,
- bağlantı durdurulduğunda ve
- Tüketici, bir MQCLOSE ile belirttik olarak ya da örtük olarak silindiğinde.

Çizelge 743. MQCTL yüklemi tanımlamaları	
Komut	Anlamı
MQCTL (BAŞLAT)	CTLSR İşlemi kullanılarak MQCTL çağırısı
MQCTL (DURDUR)	CTLSP İşlemi 'ni kullanarak MQCTL çağırısı
MQCTL (BEKLE)	CTLSW İşlemine kullanarak MQCTL çağırısı

Tüketicinin, tüketiciyle ilişkili durumu korumasını sağlar. Bir uygulama tarafından bir geri çağırma istendiğinde, tüketici çağırısına ilişkin kurallar şunlardır:

### Kaydettir

Her zaman geri çağırma çağırısının ilk çağırısıdır.

Her zaman MQCB (CBREG) çağırısıyla aynı iş parçacığıda çağırılır.

### START

Her zaman MQCTL (START) komutu ile zamanuyumlu olarak çağırılır.

- MQCTL (START) komutu döndürmeden önce, tüm START çağrıları tamamlanır.

CTLTHR istenirse, ileti teslimiyle aynı iş parçacığıdır.

Örneğin, önceki bir geri çağrı MQCTL (STOP) sırasında MQCTL (STOP) gibi bir çağrıya, başlatma ile çağrılan arama garanti edilmez.

### DUR

Bağlantı yeniden başlatılıncaya kadar, bu çağrıdan sonra başka ileti ya da olay teslim edilmez.

Uygulama, daha önce START ya da bir ileti ya da bir olay için çağrıldıysa, STOP garantilendir.

### DEREGISTER

Her zaman geri çağırma çağırısının son tipi olur.

Uygulamanızın START ve STOP geri çağrılarında iş parçacığı tabanlı kullanıma hazırlama ve temizleme gerçekleştirdiğinden emin olun. İş parçacığı tabanlı olmayan kullanıma hazırlama ve temizleme işlemini REGISTER ve DEREGISTER geri çağrıları ile yapabilirsiniz.

Belirtilenler dışında iş parçacığının yaşam ve kullanılabilirliği hakkında herhangi bir varsayımda bulunmaz. Örneğin, DEREGISTER ' a yapılan son çağrıdan sonra canlı olarak kalan bir iş parçacığına güvenmeyin. Benzer bir şekilde, CTLTHR kullanmamayı seçtiyseniz, bağlantı her başlatıldığında iş parçacığının var olduğunu varsaymayın.

Uygulamanızın iş parçacığı özellikleri için belirli gereksinimleri varsa, her zaman bir iş parçacığı buna uygun olarak yaratılabilir, sonra MQCTL (WAIT) seçeneğini kullanabilirsiniz. Bu adım, zamanuyumsuz ileti teslimi için iş parçacığını IBM MQ ' e eşleştirir .

## İleti tüketici bağlantısı kullanımı

Olağan durumda, bir uygulama olağanüstken başka bir MQI çağırısı yayınlarken, arama başarısız olur ve neden kodu RC2219 ile başarısız olur.

Ancak, önceki arama tamamlanmadan önce uygulamanın başka bir MQI çağırısı yayınlaması gerekirken, özel durumlar da vardır. Örneğin, CBRE içeren bir MQCB çağırısı sırasında tüketici çağrılabilir.

Böyle bir durumda, uygulamanın bir MÖCB ya da MÖCTL komutunu veren uygulamanın sonucu olarak, uygulama geri çağrıldığında, uygulamanın başka bir MQI çağrısı yayınlamaya izin verilir. Bu yönetim ortamı, CBCTRC tipi CBCTRC ile çağrıldığında, tüketici işlevinde bir MQOPEN çağrısı (örneğin, bir MQOPEN çağrısı) yayınlayabileceğiniz anlamına gelir. MQDISC dışında herhangi bir MQI çağrısına izin verilir.

## **MÖCB için deęiřtirgeler**

MÖCB çağrısında ařaęıdaki deęiřtirgeler vardır:

### **HCONN (10 basamaklı iřaretli tamsayı)-giriř**

Geri bildirme iřlevini yönet-HCONN parametresi.

Bu tanıtıcı, kuyruk yöneticisiyle olan baęlantıyı gösterir. The value of *HCONN* was returned by a previous MQCONN or MQCONNX call.

### **OPERATN (10 basamaklı iřaretli tamsayı)-giriř**

Geri bildirme iřlevini yönet-OPERATN parametresi.

Belirtilen nesne tanıtıcısı için tanımlanan geri bildirme iřleminde iřlem yapılıyor. Ařaęıdaki seeneklerden birini belirlemeniz gerekir; birden fazla seenek gerekliyse, deęerler eklenebilir (aynı deęiřmezi bir kereden fazla eklemeyin) ya da bit ya da iřlemi kullanarak (programlama dili, bit iřlemleri destekliyorsa).

Geerli olmayan birleřimler dikkat edilir; dięer tüm birleřimler geerli olur.

### **KESME**

Belirtilen nesne tanıtıcısı için geri bildirme iřlevini tanımlayın. Bu iřlem, çağrılacak iřlevi ve kullanılacak seim ölçütlerinin tanımlarını tanımlar.

Nesne tanıtıcısı için önceden tanımlanmış bir geri bildirme iřlevi tanımlanır, tanım deęiřtirilir. Geri çağırma deęiřtirilirken bir hata saptanırsa, iřlev kayıttan kaldırılır.

Bir geri çağrı, daha önce kaydı kaldırıldığı aynı geri çağırma iřlevinde kaydedilirse, bu bir deęiřtirme iřlemi olarak iřlem görür; ilk ya da son çağrılar çağrılmaz.

CCBREG, CTLSU ya da CTLRE ile birlikte kullanılabilir.

### **CCBUNR**

Nesne tanıtıcısı için ileti tüketmeyi durdurun ve geri bildirme için uygun olan tanıtıcıyı kaldırır.

İliřkili tanıtıcı kapatılırsa, geri çağırma otomatik olarak kayıttan kaldırılır.

CCBUNR bir tüketici içinden çağrılırsa ve geri bildirme için bir durdurma çağrısı tanımlandıysa, bu çağrı tüketiciden geri döndükten sonra çağrılır.

Bu iřlem kayıtlı bir tüketicisi olmayan bir *Hobj* için verilirse, çağrı RC2448ile döner.

### **CTLSU**

Nesne tanıtıcısı için iletilerin tüketilmesinin askıya alınması.

Bu iřlem bir olay iřleyicisine uygulanır, olay iřleyici askıya alındığında olay almaz ve askıya alınan durumdaysa kaçırılan olaylar devam edildiğinde iřleme saęlanmaz.

Askıya alma iřlemi askıya alındığında, tüketici iřlevi denetim tipi geri çağrılarını almaya devam eder.

### **CCTLRE**

Nesne tanıtıcısı için ileti tüketmeye devam edin.

Bu iřlem bir olay iřleyicisine uygulanır, olay iřleyici askıya alındığında olay almaz ve askıya alınan durumdaysa kaçırılan olaylar devam edildiğinde iřleme saęlanmaz.

### **CBDSC (MÖCBD)-giriř**

Geri bildirme iřlevini yönet-CBDSC parametresi.

Bu yapı, uygulama tarafından kaydedilmekte olan geri bildirme işlevini ve kaydı kaydettirirken kullanılan seçenekleri tanımlayan bir yapıdır.

Yapıyla ilgili ayrıntılar için bkz. "[MQCBD-Geri çağırma tanımlayıcısı](#)" sayfa 287 .

Geri çağırma tanımlayıcısı yalnızca CBREG seçeneği için gereklidir; tanımlayıcı gerekli değilse, geçirilen parametre adresi boş değerli olabilir.

### **HOBJ (10 basamaklı işaretli tamsayı)-giriş**

Geri bildirme işlevini yönet-HOBJ parametresi.

Bu tanıttıcı, bir iletinin tüketileceği nesne için kurulmuş olan erişimi gösterir. Bu, önceki bir [MQOPED](#) ya da [MQSUB](#) çağrısından ( **HOBJ** parametresindeki) döndürülen bir tanıttıcıdır.

Bir olay işleyici yordamı (CBTEH) tanımlanırken **HOBJ** gerekli değildir ve HONONE olarak belirtilmeli.

Bu *Hobj* bir [MQOPER](#) çağrısından döndürülürse, kuyruk aşağıdaki seçeneklerden biriyle ya da birkaçında açılmış olmalıdır:

- OOINPS
- OOINPX
- OOINPQ
- OOBROW

### **MSGDSC (MQMD)-giriş**

Geri bildirme işlevini yönet -MSGDSC parametresi.

Bu yapı, gerekli iletinin özniteliklerini ve alınan iletinin özniteliklerini açıklamaz.

**MsgDesc** parametresi, tüketicinin gerektirdiği iletilerin özniteliklerini ve ileti tüketicisine geçirilecek MQMD ' nin sürümünü tanımlar.

MQMD ' deki *MsgId*, *CorrelId*, *GroupId*, *MsgSeqNumber* ve *Offset* , **GetMsgOpts** parametresindeki seçeneklere bağlı olarak, ileti seçimi için kullanılır.

The *Encoding* and *CodedCharSetId* are used for message conversion if you specify the GMCONV option.

Ayrıntılar için bkz. [MQMD](#) .

*MsgDesc* yalnızca CBREG için kullanılır ve herhangi bir alan için varsayılan değer dışında bir değer gerektiriyorsa, *MsgDesc* , bir olay işleyicisi için kullanılmaz.

Tanımlayıcı gerekli değilse, geçirilen parametre adresi boş değerli olabilir.

Birden çok tüketici çakışan seçimlerle aynı kuyruğa kayıt yaptıysa, her ileti için seçilen tüketici tanımsız olduğunu unutmayın.

### **GMO (MQGMO)-giriş**

Geri bildirme işlevini yönet-GMO parametresi.

İletinin iletilerin nasıl iletileceğini denetleyen seçenekler.

All options have the meaning as described in "[IBM üzerinde MQGMO \(get-message options\)](#)" sayfa 1060, when used on an MQGET call, except:

#### **GSSIG**

Bu seçeneğe izin verilmez.

#### **GMBRWF, GMBRWN, GMmbh, GMMBC**

Bir göz atma tüketicisine teslim edilen iletilerin sırası, bu seçeneklerin birleşimleri tarafından belirlenir. Önemli birleşimler şunlardır:

#### **GMBRWF**

Kuyruktaki ilk ileti tüketicisiye sürekli olarak teslim edilir. Bu, tüketicinin yok edici olarak geri bildirimde iletiyi tüketmesi yararlı olur. Bu seçeneği dikkatli kullanın.

**GMBRWN**

Kuyruktaki her iletiye, yürürlükteki imleç konumundan, kuyruğun sonuna ulaşıncaya kadar her ileti verilir.

**GMBRWF + GMBRWN**

İmleç, kuyruğun başlangıcına sıfırlanır. Daha sonra, imleç kuyruğun sonuna ulaşıncaya kadar tüketici her iletiyi verilir.

**GMBRWF + GMMBH ya da GMMBC**

Kuyruğun başlangıcından başlayarak, tüketiciye kuyrukta ilk işaretlenmemiş ilk ileti verilir ve bu ileti bu tüketici için işaretlenir. Bu birleşim, tüketicinin geçerli imleç noktasının arkasına eklenen yeni iletileri alabilmesini sağlar.

**GMBRWN + GMMBH ya da GMMBC**

İmlecin konumundan başlayarak, tüketiciye kuyrukta bir sonraki işaretli olmayan ileti verilir ve bu ileti tüketici için işaretlenir. İletiler geçerli imleç konumunun ardındaki kuyruğa eklenebildiğinden, bu bileşimi dikkatli kullanın.

**GMBRWF + GMBRWN + GMMBH ya da GMMBC**

This combination is not permitted, if used the call returns RC2046.

**GMNWT, GMWT ve GMWI**

Bu seçenekler, tüketicinin nasıl çağrılacağını denetler.

**GDNWT**

Tüketici hiçbir zaman RC2033ile çağrılır. Tüketici yalnızca iletiler ve olaylar için çağrılır

**Sıfır GMWI ile GMWT**

RC2033 kodu yalnızca ileti olmadığında tüketiciye iletilir ve

- tüketici başlatıldı
- son mesaj neden kodundan bu yana tüketicinin en az bir mesaj teslim edildi.

Bu, sıfır bekleme aralığı belirtildiğinde, tüketicinin meşgul bir döngüde yoklaşmasını önler.

**GMWT ve pozitif bir GMWI**

Kullanıcı, belirtilen bekleme aralığından sonra çağrılır; neden kodu RC2033. Bu arama, herhangi bir iletinin tüketiciye teslim edilip edilmediği dikkate alınmaksızın yapılır. Bu, kullanıcının sağlıklı işletim bildirimini ya da toplu iş tipi işleme gerçekleştirmesini sağlar.

**GMWT ve WIULIM ' dan GMWI**

Bu, RC2033döndürmeden önce sonsuz bekleme değerini belirtir. Tüketici hiçbir zaman RC2033ile çağrılır.

*GMO* yalnızca CBREG için kullanılır ve herhangi bir alan için varsayılan değer dışında bir değer gerektiriyorsa, *GMO* , bir olay işleyicisi için kullanılmaz.

Seçenekler gerekmiyorsa, geçirilen parametre adresi boş değerli olabilir.

MQGMO yapısında bir ileti özellikleri tanıtıcısı sağlandıysa, tüketici geri bildiriminde geçirilen MQGMO yapısında bir kopya sağlanır. MQCB çağrısından geri dönerek, uygulama ileti özellikleri tanıtıcısını silebilir.

**CMPCOD (10 basamaklı işaretli tamsayı)-çıkış**

Geri bildirme işlevini yönet-CMPCOD parametresi.

Tamamlanma kodu; bu kod aşağıdakilerden biridir:

**CCOK**

İşlem başarıyla tamamlandı.

**CCWARN**

Uyarı (kısmi tamamlama).

**CCCFail**

Arama başarısız oldu.

## REASON (10 basamaklı işaretli tamsayı)-çıkış

Geri bildirme işlevini yönet-REASON parametresi.

Aşağıdaki neden kodları, kuyruk yöneticisinin **REASON** parametresi için döndürülebileceği kodlardır.

*CMPCOD* CCOK ise:

### YOK

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* CCFAIL ise:

### RC2204

(2204, X'89C') Bağdaştırıcı kullanılmıyor.

### RC2133

(2133, X'855 ') Veri dönüştürme hizmetleri modüllerinin yüklenmesi gerçekleştirilemiyor.

### RC2130

(2130, X'852 ') Bağdaştırıcı hizmeti modülü yüklenemiyor.

### RC2374

(2374, X' 946 ') API çıkışı başarısız oldu.

### RC2183

(2183, X'887 ') API çıkışı yüklenemiyor.

### RC2157

(2157, X'86D') Birincil ve ana sunucu ASID değerleri farklı.

### RC2005

(2005, X'7D5') Arabellek uzunluğu parametresi geçerli değil.

### RC2219

(2219, X'8AB') Önceki arama tamamlanmadan önce girilen MQI çağrısı.

### RC2487

(2487, X'9B7') Geri çağrı tipi alanı yanlış.

### RC2448

(2448, X' 990 ') Kayıtlı bir geri çağırma olmadığı için, kayıttan kaldırma, askıya alma ya da sürdürme işlemi yapılamıyor.

### RC2486

(2486, X'9B6') *CallbackFunction* ya da *CallbackName* belirtilmeli, ancak her ikisi belirtilmemelidir.

### RC2483

(2483, X'9B3') Yanlış geri çağırma tipi alanı.

### RC2484

(2484, X'9B4') MQCBD seçenekleri alanı yanlış.

### RC2140

(2140, X'85C') Bekleme isteği CICStarafından reddedildi.

### RC2009

(2009, X'7D9') Kuyruk yöneticisine yönelik bağlantı kaybedildi.

### RC2217

(2217, X'8A9') Bağlantı için yetkili değil.

### RC2202

(2202, X'89A') Bağlantı susturulmuş durumda.

### RC2203

(2203, X'89B') Bağlantı sona erdiriliyor.

### RC2207

(2207, X'89F') İtiltilendirme tanıtıcısı hatası.



**RC2010**

(2010, X'7DA') Veri uzunluęu parametresi geęerli deęil.

**RC2016**

(2016, X'7E0') Kuyruęun engellenmesini saęlar.

**RC2351**

(2351, X'92F') Genel iř akıřmaları birimleri.

**RC2186**

(2186, X'88A') Al-ileti seęenekleri yapısı geęerli deęil.

**RC2353**

(2353, X' 931 ') Genel iř birimi iin kullanılan tanıtıcı.

**RC2018**

(2018, X'7E2') Baęlantı tanıtıcısı geęerli deęil.

**RC2019**

(2019, X'7E3') Nesne tanıtıcısı geęerli deęil.

**RC2259**

(2259, X'8D3') Tutarsız gz atma belirtimi.

**RC2245**

(2244, X'8C5') Tutarsız iř birlięi-iř belirtimi.

**RC2246**

(2246, X'8C6') İmle altındaki ileti alma iin geęerli deęil.

**RC2352**

(2352, X' 930 ') Genel iř birimi yerel iř birimi ile akıřıyor.

**RC2247**

(2247, X'8C7') Eřleřtirme seęenekleri geęerli deęil.

**RC2485**

(2485, X'9B4') Yanlıř *MaxMsgLength* alanı.

**RC2026**

(2026, X'7EA') İleti tanımlayıcısı geęerli deęil.

**RC2497**

(2497, X'9C1') Belirtilen iřlev giriř noktası modlde bulunamadı.

**RC2496**

(2496, X'9C0') Modl bulundu, ancak yanlıř tipte; 32 bit, 64 bit deęil ya da geęerli bir dinamik baęlantı kitaplıęı.

**RC2495**

(2495, X'9BF') Modl, arama yolunda bulunamadı ya da ykleme yetkisine sahip deęil.

**RC2250**

(2250, X'8CA') İleti sıra numarası geęerli deęil.

**RC2331**

(2331, X'91B') İleti simgesinin kullanımı geęerli deęil.

**RC2033**

(2033, X'7F1') İleti yok.

**RC2034**

(2034, X'7F2') Browse imleci iletide konumlandırılmamıř.

**RC2036**

(2036, X'7F4') Kuyruk gz atma iin aık deęil.

**RC2037**

(2037, X'7F5') Kuyruk giriř iin aık deęil.

**RC2041**

(2041, X'7F9') Nesne tanımlaması aıldıęından beri deęiřtirildi.

- RC2101**  
(2101, X'835 ') Nesne zarar gördü.
- RC2206**  
(2206, X'89E') API Çağrısında yanlış işlem kodu.
- RC2046**  
(2046, X'7FE') Seçenekler geçerli değil ya da tutarlı değil.
- RC2193**  
(2193, X'891 ') Sayfa kümesi veri kümesine erişilirken hata oluştu.
- RC2052**  
(2052, X'804 ') Kuyruk silinmiştir.
- RC2394**  
(2394, X'95A') Kuyruk yanlış dizin tipi içeriyor.
- RC2058**  
(2058, X'80A') Kuyruk yöneticisi adı geçerli değil ya da bilinmiyor.
- RC2059**  
(2059, X'80B') Kuyruk yöneticisi bağlantı için kullanılabilir değil.
- RC2161**  
(2161, X'871 ') Kuyruk yöneticisi susturuyor.
- RC2162**  
(2162, X'872 ') Kuyruk yöneticisi sona erdiriliyor.
- RC2102**  
(2102, X'836 ') Yetersiz sistem kaynakları var.
- RC2069**  
(2069, X'815 ') Bu tutamaç için bekleyen sinyal.
- RC2071**  
(2071, X'817 ') Kullanılabilir bellek yetersiz.
- RC2109**  
(2109, X'83D') Çıkış programı tarafından çağrı engellendi.
- RC2024**  
(2024, X'7E8') Yürürlükteki iş birimi içinde başka ileti işlenemez.
- RC2072**  
(2072, X'818 ') Syncpoint desteği yok.
- RC2195**  
(2195, X'893 ') Beklenmeyen bir hata oluştu.
- RC2354**  
(2354, X' 932 ') Genel iş birimi içindeki bilgiler başarısız oldu.
- RC2355**  
(2355, X' 933 ') İşlerin birim içi çağrıları karışımı desteklenmiyor.
- RC2255**  
(2255, X'8CF') Kuyruk yöneticisinin kullanması için kullanılacak iş birimi.
- RC2090**  
(2090, X'82A') MQGMO' da bekleme aralığı geçerli değil.
- RC2256**  
(2256, X'8D0') MQGMO' nun yanlış sürümü sağlandı.
- RC2257**  
(2257, X'8D1') MQMD' nin yanlış sürümü sağlandı.
- RC2298**  
(2298, X'8FA') İstenen işlev geçerli ortamda yok.

## RPG Bildirimi

```
C*.1.....2.....3.....4.....5.....6.....7..
C          CALLP      MQCB(HCONN : OPERATN : CBDSC :
                   HOBJ : MSGDSC : GMO :
                   DATLEN : CMPCOD : REASON)
```

Çağrıya ilişkin prototip tanımı:

```
DMQCB          PR          EXTPROC('MQCB')
D* Connection handle
D HCONN          10I 0 VALUE
D* Operation
D OPERATN        10I 0 VALUE
D* Callback descriptor
D CBDSC          180A
D* Object handle
D HOBJ          10I 0 VALUE
D* Message Descriptor
D MSGDSC          364A
D* Get options
D GMO          112A
D* Completion code
D CMPCOD          10I 0
* Reason code qualifying CompCode
D REASON          10I 0
```

## IBM i MQCLOSE (Close object) on IBM i

MQCLOSE çağrısı bir nesneye yeniden bağlantı verir ve MQOPEN çağrısının tersi olur.

- [“Sözdizimi” sayfa 1247](#)
- [“Kullanım notları” sayfa 1247](#)
- [“Parametreler” sayfa 1248](#)
- [“RPG Bildirimi” sayfa 1253](#)

### Sözdizimi

MQCLOSE (*HCONN*, *HOBJ*, *OPTS*, *CMPCOD*, *REASON*)

### Kullanım notları

1. Bir uygulama MQDISC çağrısını yayınlarken ya da olağan ya da olağan dışı sona erdirildiğinde, uygulama tarafından açılmış olan ve hala açık olan nesnelere otomatik olarak CONONE seçeneği ile kapatılır.
2. Kapatılmakta olan nesne bir *kuyruksaise* aşağıdaki noktalar geçerlidir:
  - Kuyruktaki işlemler bir iş biriminin bir parçası olarak gerçekleştirilirse, eşitleme noktası, syncpoint sonucunu etkilemeden önce ya da daha sonra kapatılabilir.
  - Kuyruk OOBROW seçeneği ile açıldıysa, göz atma imleci yok edilir. Kuyruk daha sonra OOBROW seçeneğiyle yeniden açılırsa, yeni bir imleç yaratılır (MQOPEN ' da açıklanan OOBROW seçeneğiyle birlikte bakın).
  - MQCLOSE çağrısı sırasında bu tanıtıcı için şu anda bir ileti kilitliyse, kilit serbest bırakılır ( “IBM üzerinde MQGMO (get-message options)” sayfa 1060 ' ta açıklanan GMLK seçeneğine bakın).
3. Kapatılan nesne bir *dinamik kuyruksa* (kalıcı ya da geçici) ise, aşağıdaki noktalar geçerlidir:
  - Dinamik bir kuyruk için, ilgili MQOPED çağrısında belirlenen seçeneklerden bağımsız olarak CODEL ya da COPURG seçenekleri belirtilebilir.

- Dinamik bir kuyruk silindiğinde, kuyruğa karşı üstün olan GMWT seçeneğiyle tüm MQGET çağrılarını iptal edilir ve RC2052 neden kodunun döndürülmesini sağlar. “[IBM üzerinde MQGMO \(get-message options\)](#)” sayfa 106’da açıklanan GMWT seçeneğine bakın.

Dinamik bir kuyruk silindikten sonra, önceden edinilmiş bir *HOB*J tanıtıcısı kullanılarak kuyruğa gönderme yapma girişiminde bulunan herhangi bir çağrı (MQCLOSE dışında), neden kodu RC2052 ile başarısız olur.

Silinmiş bir kuyruğa uygulamalar tarafından erişilememesine rağmen, kuyruğun sistemden kaldırılmamasına ve ilişkili kaynakların serbest bırakılmamasına, kuyruğa gönderme yapan tüm işler kapatılıncaya kadar ve kuyruğu etkileyen iş birimlerinin tümünün kesinleştirildiğini ya da yedekleneceğini unutmayın.

- Kalıcı bir dinamik kuyruk silindiğinde, MQCLOSE çağrısında belirtilen *HOB*J tanıtıcısı, kuyruğu yaratan MQOPER çağrısının döndürdüğü bir tanıtıcı değilse, MQOPER çağrısının geçerliliğini denetlemek için kullanılan kullanıcı kimliğinin kuyruğu silme yetkisi olduğu bir denetim yapılır. MQOPEN çağrısında OOALTU seçeneği belirtilmişse, denetlenen kullanıcı kimliği *ODAU* adını işaretler.

Bu denetim, aşağıdaki durumlarda gerçekleştirilmez:

- Belirlenen tanıtıcı, kuyruğu yaratan MQOPEN çağrısının döndürdüğü tanıtıcıdır.
- Silinmekte olan kuyruk, geçici bir dinamik kuyruğudur.
- Geçici bir dinamik kuyruk kapatıldığında, MQCLOSE çağrısında belirlenen *HOB*J tanıtıcısı, kuyruğu yaratan MQOPER çağrısı tarafından döndürülen bir çağrıysa, kuyruğun silinir. Bu durum, MQCLOSE çağrısında belirtilen kapatma seçeneklerinden bağımsız olarak gerçekleşir. Kuyruktaki iletiler atılırsa, bunlar atılır; rapor iletileri oluşturulmaz.

Kuyruğu etkileyen kesinleştirilmemiş iş birimleri varsa, kuyruk ve iletileri silinmeye devam eder, ancak bu işlem, iş birimlerinin başarısız olmasına neden olmaz. Ancak, daha önce açıklandığı gibi, iş birimleriyle ilişkili kaynaklar, iş birimlerinin her biri kesinleştirilinceye ya da yedekleninceye kadar serbest bırakılmaz.

#### 4. Kapatılacak nesne bir *dağıtım listesi* ise, aşağıdaki noktalar geçerlidir:

- Dağıtım listesi için geçerli olan tek seçenek *CONTEN* ' dir; başka bir seçenek belirtilirse, çağrı neden kodu RC2046 ya da RC2045 ile başarısız olur.
- When a distribution list is closed, individual completion codes and reason codes are not returned for the queues in the list - only the **CMPCOD** and **REASON** parameters of the call are available for diagnostic purposes.

Kuyruklardan biri kapatılırsa, kuyruk yöneticisi işleme devam eder ve dağıtım listesindeki kalan kuyrukları kapatmayı dener. Çağrıya ilişkin **CMPCOD** ve **REASON** parametreleri, başarısızlığı açıklayan bilgileri döndürmek için ayarlanır. Bu nedenle, kuyrukların çoğu başarıyla kapatılmış olsa da, tamamlanma kodunun *CCFAIL* olması mümkündür. Hatayla karşılaştıran kuyruk saptanmaz.

Birden çok kuyruktaki hata varsa, **CMPCOD** ve **REASON** parametrelerinde hangi hatanın raporlanmadığı tanımlanmaz.

## Parametreler

MQCLOSE çağrısında şu deęiřtirgeler var:

### **HCONN (10 basamaklı iřaretli tamsayı)-giriř**

Baęlantı tanıtıcısı.

Bu tanıtıcı, kuyruk yöneticisiyle olan baęlantıyı gösterir. The value of *HCONN* was returned by a previous MQCONN or MQCONNX call.

### **HOBJ (10 basamaklı iřaretli tamsayı)-giriř/çıkıř**

Nesne tanıtıcısı.

Bu tanıtıcı, kapatılmakta olan nesneyi gösterir. Nesne herhangi bir tipte olabilir. The value of *HOB*J was returned by a previous MQOPEN call.

Çağrı başarıyla tamamlandıktan sonra, kuyruk yöneticisi bu parametreyi, ortam için geçerli bir tanıtıcı olmayan bir değere ayarlar. Bu değer:

## **HOUNUH**

Kullanılamaz nesne tanıtıcısı.

## **OPTS (10 basamaklı işaretli tamsayı)-giriş**

MQCLOSE işleminin denetlenmesini denetleyen seçenekler.

**OPTS** parametresi, nesnenin kapatılma şeklini denetler. Yalnızca kalıcı dinamik kuyruklar ve abonelikler birden çok yolla kapatılabilir. Kalıcı dinamik kuyruklar alıkonabilir ya da silinebilir; bunlar, QDPERM değerine sahip bir **DefinitionType** öznitelige sahip kuyruklar ( "Kuyruklara ilişkin öznitelikler" sayfa 1345 içinde açıklanan **DefinitionType** öznitelğine bakın). Kapatma seçenekleri, bu konudaki daha sonraki bir tabloda özetlenmiştir.

Sürekli abonelikler alıkonabilir ya da kaldırılabilir; bunlar SODUR seçeneğiyle MQSUB çağrısıyla yaratılır.

Tutamacı yönetilen bir hedefe kapatırken (SOMAN seçeneğini kullanan bir MQSUB çağrısına döndürülen **Hobj** parametresidir) kuyruk yöneticisi, ilişkili abonelik de kaldırıldığında, alınmamış yayınları temizleyecek. Bu işlem, bir MQSUB çağrısında döndürülen **Hsub** parametresindeki CORMSB seçeneği kullanılarak yapılır. Kalıcı olmayan bir abonelik için CORMSB 'nin MQCLOSE' de varsayılan davranış olduğunu unutmayın.

Yönetilmeyen bir hedefe yönelik bir tutamacı kapattığınızda, yayınların gönderildiği kuyruğu temizlemekten sorumlu olduğunuz bir hedef vardır. Önce CORMSB olanağını kullanarak aboneliği kapatmanız ve daha sonra, geri kalan hiçbir şey kalmayınca kadar iletileri kuyruktan çıkarma önerildiniz.

Aşağıdakilerden biri (ve yalnızca bir tanesi) belirtilmelidir:

### **Dinamik kuyruk kapatma seçenekleri**

Bu seçenekler, kalıcı dinamik kuyrukların nasıl kapatılacağını denetler:

#### **CODEL**

Kuyruğu silin.

Aşağıdakilerden biri geçerliyse, kuyruk silinir:

- Bu, önceki bir MQOPER çağrısı tarafından oluşturulan kalıcı bir dinamik kuyruktır ve kuyruksuz ve işlenmemiş bir alma ya da kuyruğa alma isteği olmayan iletiler (yürürlükteki görev için ya da başka bir görev için) yoktur.
- Bu, **HOB**Jdöndüren MQOPEN çağrısıyla yaratılan geçici dinamik kuyruktır. Bu durumda, kuyrukta bulunan tüm iletiler temizlenir.

Diğer tüm durumlarda, **Hobj** MQSUB çağrısında döndürüldüğü durum da içinde olmak üzere, çağrı neden kodu RC2045ile başarısız olur ve nesne silinmez.

#### **KOPÇA**

Kuyruğun silinmesi, üzerindeki iletilerin temizlenmesi.

Aşağıdakilerden biri geçerliyse, kuyruk silinir:

- Bu, önceki bir MQOPER çağrısı tarafından yaratılan kalıcı bir dinamik kuyruktır ve kuyruk için kesinleştirilmemiş bir alma ya da koyma isteği yoktur (yürürlükteki görev ya da başka bir görev için).
- Bu, **HOB**Jdöndüren MQOPEN çağrısıyla yaratılan geçici dinamik kuyruktır.

Diğer tüm durumlarda, **Hobj** MQSUB çağrısında döndürüldüğü durum da içinde olmak üzere, çağrı neden kodu RC2045ile başarısız olur ve nesne silinmez.

Sonraki çizelge, hangi kapanış seçeneklerinin geçerli olduğunu ve nesnenin alıkonulup tutulmadığını ve silinip silinmeyeceğini gösterir.

<i>Çizelge 744. Alıkonan ya da silinen nesnelere kullanmak için geçerli kapanış seçenekleri</i>			
<b>Nesne ya da kuyruk tipi</b>	<b>CONONE</b>	<b>CODEL</b>	<b>KOPÇA</b>
Kuyruktan başka nesne	Alıkonan	Geçerli değil	Geçerli değil
Önceden tanımlı kuyruk	Alıkonan	Geçerli değil	Geçerli değil
Kalıcı dinamik kuyruk	Alıkonan	Boş ve bekleyen güncelleme yok ise silindi	Silinen iletiler; bekleyen güncelleme yoksa kuyruk silindi
Geçici dinamik kuyruk (kuyruğun yaratıcısı tarafından çağrılan çağrı)	Silindi	Silindi	Silindi
Geçici dinamik kuyruk (kuyruğun yaratıcısı tarafından çağrılmama çağrısı)	Alıkonan	Geçerli değil	Geçerli değil
Dağıtım listesi	Alıkonan	Geçerli değil	Geçerli değil
Yönetilen abonelik hedefi	Alıkonan	Geçerli değil	Geçerli değil
Dağıtım listesi (abonelik kaldırıldı)	İletiler silindi; kuyruk silindi	Geçerli değil	Geçerli değil

#### **Abonelik kapatma seçenekleri**

Bu seçenekler, tutamaç kapatıldığında dayanıklı aboneliklerin kaldırılıp kaldırılmayacağını ve uygulamanın hala uygulama tarafından okunmayı bekleyerek temizlenip temizlenmediğini denetler. Bu seçenekler, yalnızca bir MQSUB çağrısının **HSUB** parametresindeki bir nesne tanıtıcısı ile kullanılmak üzere geçerlidir.

#### **COKPSB**

Aboneliğin tanıtıcısı kapatıldı, ancak yapılan abonelik alıkonur. Yayınlar, abonelikte belirtilen hedefe gönderilmeye devam eder. Bu seçenek yalnızca, abonelik SODUR seçeneğiyle birlikte yapıldıysa geçerlidir. Abonelik dayanıklıysa, COKPSB varsayılan değerdir

#### **CORMSB**

Abonelik kaldırılır ve abonelik tanıtıcısı kapatılır.

MQSUB çağrısının **Hobj** parametresi, **Hsub** parametresinin kapatılarak geçersiz kılınmaz ve geri kalan yayınları almak için MQGET ya da MQCB için kullanılmaya devam edebilir. MQSUB çağrısının **Hobj** parametresi de kapatıldığında, yönetilen bir hedef alındıysa, alınmamış yayınlar kaldırılır.

Abonelik, dayanıklı değilse, varsayılan değer CORMSB 'dir.

Bu abonelik kapatma seçenekleri aşağıdaki tablolarda özetlenmiştir:

Sürekli abonelik tanıtıcısını kapatmak, ancak aboneliği bırakmak için aşağıdaki abonelik kapatma seçeneklerini kullanın:

<i>Çizelge 745. Dayanıklı bir abonelik tanıtıcısını kapatma ve aboneliğin etrafında bırakma görevi seçenekleri</i>	
<b>Görev</b>	<b>Abonelik kapatma seçeneği</b>
Yayınları MQOPENed tanıtıcısında tut	COKPSB
Bir MQOPENed tanıtıcısındaki yayınları kaldır	Eyleme izin verilmiyor
Yayınları, SOMAN ile bir tanıtıcı üzerinde tutun	COKPSB
Yayınları SOMAN ile bir tanıtıcı üzerinde kaldırma	Eyleme izin verilmiyor

Bir kalıcı abonelik tanıtıcısını kapatarak ya da kalıcı olmayan bir abonelik tanıtıcısını kapatarak aboneliği kaldırmak için, aşağıdaki abonelik kapatma seçeneklerini kullanın:

Çizelge 746. Aboneliği geri almak için görev seçenekleri	
Görev	Abonelik kapatma seçeneği
Yayınları MÇOPENed tanıtıcısında tut	CORMSB
Bir MÇOPENed tanıtıcısındaki yayınları kaldır	Eyleme izin verilmiyor
Yayınları, SOMAN ile bir tanıtıcı üzerinde tutun	CORMSB
Yayınları SOMAN ile bir tanıtıcı üzerinde kaldırma	KOPYA B

### önden okuma seçenekleri

Aşağıdaki seçenekler, bir uygulama tarafından istenmeden önce istemciye gönderilen ve henüz uygulama tarafından tüketilmemiş olan, kalıcı olmayan iletilerin ne olacağını denetler. Bu iletiler, uygulama tarafından istekte bulunulmasını bekleyen istemci okuma önokuma arabelleğinde saklanır ve MÇCLOSE işlemi tamamlanmadan önce atılabilir ya da kuyruktan tüketilebilir.

#### KOIMM

Nesne hemen kapatılır ve bir uygulama istenmeden önce istemciye gönderilen iletiler atılır ve herhangi bir uygulama tarafından tüketilemez. Bu varsayılan değerdir.

#### KOQSC

Nesneyi kapatma isteği yapıldı, ancak bir uygulama tarafından istekte bulunmadan önce istemciye gönderilen herhangi bir ileti, istemci okuma yazma arabelleğinde bulunmaya devam ederse, MÇCLOSE çağrısı bir uyarı kodu RC2458 ile geri döner ve nesne tanıtıcısı geçerli kalır.

Bundan sonra, uygulama daha fazla kullanılabilir oluncaya kadar iletileri almak için nesne tanıtıcısını kullanmaya devam edebilir ve sonra nesneyi yeniden kapatmaya devam edebilir. Daha sonra istekte bulunan bir uygulama öncesinde istemciye ileti gönderilmeyecek, daha sonra okunacak ileti kapatılır.

Bir ileti son MÇGET çağrısıyla ve COIMM kullanıldıysa atılacak olan MÇCLOSE arasında bir ileti gelebileceği için, uygulamaların ileriye doğru okuma arabelleğindeki başka ileti olmadığı bir noktaya ulaşmaya çalışmak yerine COQSC ' yi kullanmamız önerilir.

Zamanuyumsuz bir geri bildirme işlevinden COQSC ile bir MÇCLOSE komutu verilirse, ileriki iletilerin okuma davranışı geçerli olur. RC2458 uyarı kodu döndürülürse, geri bildirme işlevi en az bir kez daha çağrılır. Önceden okunan son ileti geri bildirme işlevine geçirildiğinde, CBCFLG alanı CBCFBE olarak ayarlanır.

### Varsayılan seçenek

Daha önce açıklanan seçeneklerden hiçbirine gerek duymuyorsanız, aşağıdaki seçeneği kullanabilirsiniz:

#### CONONE

İsteğe bağlı kapatma işlemi gerekli değil.

Bunun için belirtilmesi gerekir:

- Kuyruktan başka nesnelere
- Önceden Tanımlı Kuyruk
- Geçici dinamik kuyruklar (ancak, bu durumda HOBJ , kuyruğu yaratan MÇAN çağrısının döndürdüğü tanıtıcı değildir).
- Dağıtım listeleri

Önceki durumların tümünde, nesne korunur ve silinmez.

Bu seçenek geçici bir dinamik kuyruk için belirtilirse:

- The queue is deleted, if it was created by the MÇOPEN call that returned HOBJ ; any messages that are on the queue are purged.

- Diğer tüm durumlarda, kuyruk (ve üzerindeki iletiler) korunur.

Bu seçenek kalıcı bir dinamik kuyruk için belirtilirse, kuyruk korunur ve silinmez.

### **CMPCOD (10 basamaklı işaretli tamsayı)-çıkış**

Tamamlanma kodu.

Bu, aşağıdakilerden biridir:

#### **CCOK**

İşlem başarıyla tamamlandı.

#### **CCWARN**

Uyarı (kısmi tamamlama).

#### **CCCFAIL**

Arama başarısız oldu.

### **REASON (10 basamaklı işaretli tamsayı)-çıkış**

Neden kodu ön eleme *CMPCOD*.

*CMPCOD* CCOK ise:

#### **YOK**

(0, X'000 ') Raporlamak için bir neden yok.

*CMPCOD* CCWARN ise:

#### **RC2241**

(2241, X'8C1') İleti grubu tamamlanmadı.

#### **RC2242**

(2242, X'8C2') Mantıksal ileti tamamlanmadı.

*CMPCOD* CCFAIL ise:

#### **RC2219**

(2219, X'8AB') MQI çağırısı, önceki arama tamamlanmadan önce yeniden girildi.

#### **RC2009**

(2009, X'7D9') Kuyruk yöneticisine yönelik bağlantı kaybedildi.

#### **RC2018**

(2018, X'7E2') Bağlantı tanıtıcısı geçerli değil.

#### **RC2019**

(2019, X'7E3') Nesne tanıtıcısı geçerli değil.

#### **RC2035**

(2035, X'7F3') Erişim için yetkili değil.

#### **RC2101**

(2101, X'835 ') Nesne zarar gördü.

#### **RC2045**

(2045, X'7FD') Seçenek, nesne tipi için geçerli değil.

#### **RC2046**

(2046, X'7FE') Seçenekler geçerli değil ya da tutarlı değil.

#### **RC2058**

(2058, X'80A') Kuyruk yöneticisi adı geçerli değil ya da bilinmiyor.

#### **RC2059**

(2059, X'80B') Kuyruk yöneticisi bağlantı için kullanılabilir değil.

#### **RC2162**

(2162, X'872 ') Kuyruk yöneticisi sona erdiriliyor.

#### **RC2055**

(2055, X'807 ') Kuyruk, bir ya da daha fazla ileti içeriyor ya da kesinleştirilmemiş bir put ya da alma isteği içeriyor.



## RC2102

(2102, X'836 ') Yetersiz sistem kaynakları var.

## RC2063

(2063, X'80F') Güvenlik hatası oluştu.

## RC2071

(2071, X'817 ') Kullanılabilir bellek yetersiz.

## RC2195

(2195, X'893 ') Beklenmeyen bir hata oluştu.

## RPG Bildirimi

```
C*..1.....2.....3.....4.....5.....6.....7..
C                               CALLP      MQCLOSE(HCONN : HOBJ : OPTS :
C                               CMPCOD : REASON)
```

Çağrıya ilişkin prototip tanımı:

```
D*..1.....2.....3.....4.....5.....6.....7..
DMQCLOSE      PR                EXTPROC('MQCLOSE')
D* Connection handle
D HCONN                10I 0 VALUE
D* Object handle
D HOBJ                10I 0
D* Options that control the action of MQCLOSE
D OPTS                10I 0 VALUE
D* Completion code
D CMPCOD                10I 0
D* Reason code qualifying CMPCOD
D REASON                10I 0
```

IBM i

## IBM üzerinde MQCMIT (değişiklikleri kesinleştir)

MQCMIT çağrısı, kuyruk yöneticisine uygulamanın bir uyumluluk noktasına ulaştığını ve son eşitleme noktasının kalıcı kılınacağı için tüm ileti alkonacağını ve aldığını gösterir. Bir iş biriminin bir parçası olarak sunulan iletiler, diğer uygulamalar tarafından kullanılabilir kılınır; bir iş biriminin bir parçası olarak alınan iletiler silinir.

- [“Sözdizimi” sayfa 1253](#)
- [“Kullanım notları” sayfa 1253](#)
- [“Parametreler” sayfa 1254](#)
- [“RPG Bildirimi” sayfa 1255](#)

## Sözdizimi

MQCMIT (HCONN, COMCOD, REASON)

## Kullanım notları

Bu kullanım notlarını MQCMIT kullanırken dikkate alın.

1. Bu arama, yalnızca kuyruk yöneticisinin iş birimini koordine ettiğinde kullanılabilir. Bu, değişikliklerin yalnızca IBM MQ kaynaklarını etkilediği yerel bir iş birimidir.
2. Kuyruk yöneticisinin iş birimini koordine etmediği ortamlarda, MQCMIT yerine uygun kesinleştirme çağrısı kullanılmalıdır. Ortam, uygulamanın olağan şekilde sonlanmasına neden olan örtük bir kesinleştirmeyi de destekleyebilir.

- IBM i' ta bu çağrı, kuyruk yöneticisi tarafından koordine edilen yerel iş birimleri için kullanılabilir. This means that a commitment definition must not exist at job level, that is, the STRCMTCTL command with the **CMTSCOPE (\*JOB)** parameter must not have been issued for the job.
3. Bir uygulama, bir iş biriminde kesinleştirilmemiş değişikliklerle sona ererse, bu değişikliklerin yok edilmesi, uygulamanın olağan ya da olağan dışı bir şekilde sona ermesine bağlıdır. Ek ayrıntılar için [“IBM üzerinde MQDISC \(Bağlantı kesme kuyruğu yöneticisi\)” sayfa 1268](#) içindeki kullanım notlarına bakın.
  4. Bir uygulama, gruplara ya da mantıksal ileti bölümlerine ileti yerleştirdiğinde ya da ileti aldığı anda, kuyruk yöneticisi, son başarılı MQPUT ve MQGET çağrılarına ilişkin ileti grubuyla ve mantıksal iletiyle ilgili bilgileri saklar. Bu bilgiler kuyruk tanıtıcısı ile ilişkilendirilir ve aşağıdaki gibi şeyler içerir:
    - MQMD ' de *MDGID*, *MDSEQ*, *MDOFF* ve *MDMFL* alanlarının değerleri.
    - İletinin bir iş biriminin parçası olup olmadığı.
    - MQPUT çağrısına ilişkin: İletinin kalıcı mı, yoksa kalıcı değil mi olduğu.

Bir iş birimi kesinleştirildiğinde, kuyruk yöneticisi grup ve bölüm bilgilerini saklar ve uygulama yürürlükteki ileti grubuna ya da mantıksal iletiye ileti yerleştirmeye ya da ileti almaya devam edebilir.

Bir iş birimi kesinleştirildiğinde, grubun ve kesim bilgilerinin saklanması, uygulamanın büyük bir ileti grubu ya da çok sayıda iş birimi üzerinden birçok kesimden oluşan büyük bir mantıksal ileti yayınlamasına olanak tanır. Yerel kuyruk yöneticisinin kuyruk depolaması sınırlı olması durumunda, birkaç iş birimi kullanılması avantajlı olabilir. Ancak, bir sistem hatası ortaya çıkarsa, uygulama, iletileri yeniden başlatabilmek ya da iletileri doğru noktaya getirmek için yeterli bilgileri sağlamalıdır. Bir sistem hatasından sonra doğru noktada yeniden başlatılabilmeye ilişkin ayrıntılar için, [“MQPMO \(Put-message options\) on IBM i” sayfa 1154](#) içinde açıklanan PMLOGO seçeneğine ve [“IBM üzerinde MQGMO \(get-message options\)” sayfa 1060](#)' ta açıklanan GMLOGO seçeneğine bakın.

Kalan kullanım notları, yalnızca kuyruk yöneticisi iş birimlerini koordine ettiğinde geçerlidir:

1. Bir iş birimi, bağlantı tanıtıcısı ile aynı kapsama sahiptir. Başka bir deyişle, belirli bir iş birimini etkileyen tüm IBM MQ çağrıları, aynı bağlantı tanıtıcısı kullanılarak gerçekleştirilmelidir. Farklı bir bağlantı tanıtıcısı kullanılarak verilen çağrılar (örneğin, başka bir uygulama tarafından yayınlanan çağrılar) farklı bir iş birimini etkiler. Bağlantı tutamaçlarının kapsamı hakkında bilgi için MQCONN ' de açıklanan **HCONN** parametresine bakın.
2. Bu çağrıdan etkilenenler, yalnızca yürürlükteki iş biriminin bir parçası olarak alınan ya da alınan iletiler etkilenir.
3. Bir iş birimi içinde MQGET, MQPUT ya da MQPUT1 tarafından çağrılan, ancak bir kesinleştirme ya da geri alma çağrısını hiçbir zaman sorun etmemeyen, uzun süredir çalışan bir uygulama, diğer uygulamalar tarafından kullanılanmayan iletileri doldurabilecek kuyruklara neden olabilir. Bu olasılığa karşı koruma sağlamak için, yönetici, **MaxUncommittedMsgs** kuyruk yöneticisi özniteliğini, kuyrukları dolduran kaçak uygulamaları önleyecek kadar düşük bir değere ayarlamalıdır, ancak beklenen ileti sistemi uygulamalarının doğru şekilde çalışmasına izin verecek kadar yüksek bir değere ayarlanmalıdır.

## Parametreler

MQCMIT çağrısında aşağıdaki deęiřtirgeler vardır:

### **HCONN (10 basamaklı iřaretli tamsayı)-giriř**

Baęlantı tanıtıcısı.

Bu tanıtıcı, kuyruk yöneticisiyle olan baęlantıyı gösterir. The value of *HCONN* was returned by a previous MQCONN or MQCONNX call.

### **COMCOD (10 basamaklı iřaretli tamsayı)-çıkıř**

Tamamlanma kodu.

Bu, aşağıdakilerden biridir:

#### **CCOK**

İřlem başarıyla tamamlandı.

**CCWARN**

Uyarı (kısmi tamamlama).

**CCCFAIL**

Arama başarısız oldu.

**REASON (10 basamaklı işaretli tamsayı)-çıkış**

Neden kodu ön eleme *COMCOD*.

*COMCOD* CCOK ise:

**YOK**

(0, X'000 ') Raporlamak için bir neden yok.

*COMCOD* CCWARN ise:

**RC2003**

(2003, X'7D3') İş birimi yedeklendi.

**RC2124**

(2124, X'84C') Kesinleştirme işleminin sonucu beklemede.

*COMCOD* CCFAIL ise:

**RC2219**

(2219, X'8AB') MQI çağrısı, önceki arama tamamlanmadan önce yeniden girildi.

**RC2009**

(2009, X'7D9') Kuyruk yöneticisine yönelik bağlantı kaybedildi.

**RC2018**

(2018, X'7E2') Bağlantı tanıtıcısı geçerli değil.

**RC2101**

(2101, X'835 ') Nesne zarar gördü.

**RC2123**

(2123, X'84B') Kesinleştirme ya da geri dönüş işlemi sonucu karışık.

**RC2162**

(2162, X'872 ') Kuyruk yöneticisi sona erdiriliyor.

**RC2102**

(2102, X'836 ') Yetersiz sistem kaynakları var.

**RC2071**

(2071, X'817 ') Kullanılabilir bellek yetersiz.

**RC2195**

(2195, X'893 ') Beklenmeyen bir hata oluştu.

**RPG Bildirimi**

```
C*..1.....2.....3.....4.....5.....6.....7..
C                               CALLP      MQCMIT(HCONN : COMCOD : REASON)
```

Çağrıya ilişkin prototip tanımı:

```
D*..1.....2.....3.....4.....5.....6.....7..
DMQCMIT          PR          EXTPROC('MQCMIT')
D* Connection handle
D HCONN          10I 0 VALUE
D* Completion code
D COMCOD         10I 0
D* Reason code qualifying COMCOD
D REASON         10I 0
```

MQCONN çağrısı bir uygulama programını kuyruk yöneticisine bağlar. Sonraki ileti kuyruklama çağrılarında uygulama tarafından kullanılan bir kuyruk yöneticisi bağlantı tanıtıcısı sağlar.

- Uygulamaların kuyruk yöneticisine bağlanmak için MQCONN ya da MQCONNX çağrısını ya da kuyruk yöneticisinden bağlantıyı kesmek için MQDISC çağrısını kullanmaları gerekir.

IBM MQ for Multiplatforms' ta, bir uygulamadaki her iş parçacığı farklı kuyruk yöneticilerine bağlanabilir. Diğer sistemlerde, bir işlem içindeki koşut zamanlı tüm bağlantıların aynı kuyruk yöneticisi olması gerekir.

- [“Sözdizimi” sayfa 1256](#)
- [“Kullanım notları” sayfa 1256](#)
- [“Parametreler” sayfa 1256](#)
- [“RPG Bildirimi” sayfa 1259](#)

## Sözdizimi

MQCONN (*QMNAME, HCONN, CMPCOD, REASON*)

## Kullanım notları

1. MQCONN çağrısını kullanarak bağlantının yapıldığı kuyruk yöneticisi *yerel kuyruk yöneticisi* olarak adlandırılır.
2. Yerel kuyruk yöneticisinin iyeliğindeki kuyruklar, uygulamayı yerel kuyruklar olarak görmektedir. Bu kuyruklardan ileti koymak ve bu kuyruklardan ileti almak mümkündür.

Yerel kuyruk yöneticisinin ait olduğu kuyruk paylaşım grubunun sahip olduğu paylaşılan kuyruklar, uygulama için yerel kuyruklar olarak görüntülenir. Bu kuyruklardan ileti koymak ve bu kuyruklardan ileti almak mümkündür.

Uzak kuyruk yöneticilerinin sahip olduğu kuyruklar, uzak kuyruklar olarak görüntülenir. Bu kuyruklara ileti koymak mümkündür, ancak bu kuyruklardan ileti almak mümkün değildir.

3. Bir uygulama çalışırken kuyruk yöneticisi başarısız olursa, sonraki IBM MQ çağrılarında kullanmak üzere yeni bir bağlantı tanıtıcısı elde etmek için uygulamanın MQCONN çağrısını yeniden yayınlaması gerekir. Uygulama, çağrı başarılı oluncaya kadar MQCONN çağrısını düzenli olarak yayınlatabilir.

Bir uygulama kuyruk yöneticisine bağlı olup olmadığından emin değilse, bir bağlantı tanıtıcısı elde etmek için uygulama bir MQCONN çağrısını güvenle yayınlatabilir. Uygulama önceden bağlandıysa, döndürülen tanıtıcı, önceki MQCONN çağrısının döndürdüğü, ancak tamamlanma kodu CCWARN ve neden kodu RC2002 ile aynı olur.

4. Uygulama IBM MQ çağrılarını kullanmayı bitirdiğinde, uygulama kuyruk yöneticisinden bağlantıyı kesmek için MQDISC çağrısını kullanmalıdır.
5. IBM i' ta, olağan dışı sona erdirilen programlarda kuyruk yöneticisinden otomatik olarak bağlantı kesilmez. Bu nedenle, MQCONN ya da MQCONNX çağrısının tamamlanma kodu CCWARN ve neden kodu RC2002 döndürülmesi olasılığına izin vermek için uygulamalar yazılmalıdır. Bu durumda döndürülen bağlantı tanıtıcısı olağan biçimde kullanılabilir.

## Parametreler

MQCONN çağrısında şu deęiřtirgeler vardır:

### QMNAME (48 baytlık karakter dizilimi)-giriř

Kuyruk yöneticisinin adı.

Uygulamanın bağlanmak istedięi kuyruk yöneticisinin adıdır. Ad ařaęıdaki karakterleri içerebilir:

- Büyük harf alfabetik karakterler (A-Z)

- Küçük harf alfabetik karakterler (a-z)
- Sayısal basamaklar (0-9)
- Nokta (.), eğik çizgi (/), alt çizgi (\_), yüzde (%)

Ad, baştaki ya da gömülü boşluklar içermemeli, ancak sondaki boşlukları içermeyebilir. Addaki önemli verilerin sonunu göstermek için boş karakter kullanılabilir; bunun ardından boş değer ve izleyen karakterler boşluk olarak işlenir. Belirtilen ortamlarda aşağıdaki kısıtlamalar geçerlidir:

- IBM üzerinde, komutlar üzerinde belirtildiğinde, küçük harf karakterleri, eğik çizgi ya da yüzde işareti tırnak işareti içine alınmalıdır. Bu tırnak işaretleri **QMNAME** parametresinde belirtilmemelidir.

Ad tümüyle boşluklardan oluşuyorsa, *varsayılan* kuyruk yöneticisinin adı kullanılır.

*QMNAME* için belirtilen ad, *bağlanabilir* kuyruk yöneticisinin adı olmalıdır.

**Kuyruk paylaşım grupları:** Birkaç kuyruk yöneticisinin var olduğu ve bir kuyruk paylaşım grubu oluşturmak üzere yapılandırıldığı sistemlerde, kuyruk yöneticisi adı yerine *QMNAME* için kuyruk paylaşım grubunun adı belirtilebilir. Bu, uygulamanın, kuyruk paylaşım grubunda bulunan *any* kuyruk yöneticisine bağlanmasını sağlar. Sistem, boş bir *QMNAME* ' un varsayılan kuyruk yöneticisi yerine kuyruk paylaşım grubuna bağlanmasına neden olacak şekilde de yapılandırılabilir.

*QMNAME* , kuyruk paylaşım grubunun adını belirtiyorsa, ancak sistemde o adı taşıyan bir kuyruk yöneticisi de varsa, eski olarak bu adı tercih eden ikinciye bağlantı yapılır. Yalnızca bu bağlantı başarısız olursa, kuyruk paylaşım grubundaki kuyruk yöneticilerinden biriyle bağlantı girişiminde bulunmaya çalışılır.

If the connection is successful, the handle returned by the MQCONN or MQCONNX call can be used to access *Tümü* of the resources (both shared and nonshared) that belong to the particular queue manager to which connection has been made. Bu kaynaklara erişim, tipik yetki denetimlerine tabidir.

Uygulama, koşut zamanlı bağlantı kurmak için iki MQCONN ya da MQCONNX çağrısını yayınlarsa ve bir ya da her ikisi de kuyruk paylaşım grubunun adını belirtiyorsa, ikinci çağrı tamamlanma kodu CCWARN ve neden kodu RC2002 döndürülebilir. Bu durum, ikinci çağrı ilk çağrıyla aynı kuyruk yöneticisine bağlandığında oluşur.

Kuyruk paylaşım grupları yalnızca z/OS üzerinde desteklenir. Kuyruk paylaşım grubuna yönelik bağlantı yalnızca toplu, RRS toplu işi ve TSO ortamlarında desteklenir.

**IBM MQ istemci uygulamaları:** IBM MQ MQI client uygulamaları için, belirtilen kuyruk yöneticisi adına sahip her bir istemci bağlantısı kanal tanımlaması için bir bağlantı girişiminde bulunuluncaya kadar, bir bağlantı girişiminde bulunulması denir. Ancak kuyruk yöneticisi, belirtilen adla aynı ada sahip olmalıdır. Tümü boş bir ad belirtilirse, her istemci bağlantısı kanalı, her biri başarılı oluncaya kadar kuyruk yöneticisi adını taşıyan her bir istemci bağlantı kanalı denir; bu durumda kuyruk yöneticisinin gerçek adına ilişkin herhangi bir denetim yoktur.

**IBM MQ istemcisi kuyruk yöneticisi grupları:** Belirtilen ad bir yıldız işaretiyle (\*) başlıyorsa, bağlantının yapıldığı gerçek kuyruk yöneticisinin, uygulama tarafından belirtilenizden farklı bir adı olabilir. Belirtilen ad (yıldız işareti olmadan), bağlantı için uygun olan kuyruk yöneticilerine ilişkin bir *grup* tanımlar. Uygulama, bir bağlantının gerçekleştirilebileceği bulunana kadar, her birini sırayla, alfabetik sırada deneyerek gruptan bir tane seçer. Gruptaki kuyruk yöneticilerinden hiçbiri bağlantı için kullanılamıyorsa, arama başarısız olur. Her kuyruk yöneticisi yalnızca bir kez denir. Ad için tek başına bir yıldız işareti belirtilirse, somutlama tanımlı bir varsayılan kuyruk yöneticisi grubu kullanılır.

Kuyruk yöneticisi grupları yalnızca, MQ-istemci ortamında çalışan uygulamalar için desteklenir; istemci dışı bir uygulama, yıldız işaretiyle başlayan kuyruk yöneticisi adını belirlerse, çağrı başarısız olur. Bir grup, gruptaki kuyruk yöneticileriyle iletişim kurmak için aynı kuyruk yöneticisi adına (yıldız işareti olmadan belirtilen ad) sahip birkaç istemci bağlantı kanalı tanımlaması sağlayarak tanımlanır. Varsayılan grup, her biri boş kuyruk yöneticisi adına sahip bir ya da daha çok istemci bağlantı kanalı tanımlaması sağlayarak tanımlanır (bu nedenle, bir istemci uygulamasının adı için tek bir yıldız işareti belirtilmesiyle aynı etkiye sahip olur).

Bir uygulamanın bir kuyruk yöneticisine bağlandıktan sonra, bir uygulama, ileti ve nesne tanımlayıcılarındaki kuyruk yöneticisi adı alanlarındaki tipik şekilde, uygulamanın gerçekten bağlı

olduğu kuyruk yöneticisinin adını ( *yerel kuyruk yöneticisi* ) belirtmek için boşluk belirtebilir. Uygulamanın bu adı bilmesi gerekiyorsa, **QMgrName** kuyruk yöneticisi özneliğini sorgulamak için MQINQ çağrısı yayınlanabilir.

Bağlantı adına bir yıldız imi önizlendiğinde, uygulamanın gruptaki belirli bir kuyruk yöneticisine bağlanmaya bağımlı olmadığını belirtir. Uygun uygulamalar şöyle olacaktır:

- İleti içeren, ancak ileti alamayan uygulamalar.
- İstek iletilerini yerleştiren ve daha sonra, yanıt iletilerini bir *geçici dinamik* kuyruktan alan uygulamalar.

Uygun olmayan uygulamalar, belirli bir kuyruk yöneticisinde belirli bir kuyruktan ileti almaya gereksinim duyanlar olacaktır; bu tür uygulamaların adı yıldız işaretiyle önlememelidir.

Bir yıldız işareti belirtilirse, adın geri kalanının uzunluk üst sınırı 47 karakterdir.

Bu parametrenin uzunluğu LNQMN tarafından verilir.

### **HCONN (10 basamaklı işaretli tamsayı)-çıkış**

Bağlantı tanıtıcısı.

Bu tanıtıcı, kuyruk yöneticisiyle olan bağlantıyı gösterir. Uygulama tarafından yayınlanan sonraki tüm ileti kuyruklama çağrılarında bu değer belirtilmelidir. MQDISC çağrısı yayınlandığında ya da tutamaç işleminin kapsamını tanımlayan işlem birimi sonlandığında geçerli olmak üzere sona erer.

Tutamaç kapsamı, en küçük birimle sınırlıdır koşut işleme, uygulamanın çalıştığı altyapı tarafından desteklenir; tanıtıcı, MQCONN çağrısının yayınlandığı koşut işleme birimi dışında geçerli değildir.

- IBM üzerinde, tutamaç kapsamı, aramayı yayınlayan işidir.

### **CMPCOD (10 basamaklı işaretli tamsayı)-çıkış**

Tamamlanma kodu.

Bu, aşağıdakilerden biridir:

#### **CCOK**

İşlem başarıyla tamamlandı.

#### **CCWARN**

Uyarı (kısmi tamamlama).

#### **CCFAIL**

Arama başarısız oldu.

### **REASON (10 basamaklı işaretli tamsayı)-çıkış**

Neden kodu ön eleme *CMPCOD*.

*CMPCOD* CCOK ise:

#### **YOK**

(0, X'000 ') Raporlamak için bir neden yok.

*CMPCOD* CCWARN ise:

#### **RC2002**

(2002, X'7D2') Uygulaması zaten bağlı.

*CMPCOD* CCFAIL ise:

#### **RC2219**

(2219, X'8AB') MQI çağrısı, önceki arama tamamlanmadan önce yeniden girildi.

#### **RC2267**

(2267, X'8DB') Küme iş yükü çıkışı yüklenemiyor.

#### **RC2009**

(2009, X'7D9') Kuyruk yöneticisine yönelik bağlantı kaybedildi.

**RC2018**

(2018, X'7E2') Bağlantı tanıtıcısı geçerli değil.

**RC2035**

(2035, X'7F3') Erişim için yetkili değil.

**RC2137**

(2137, X'859 ') Nesne başarıyla açılmadı.

**RC2058**

(2058, X'80A') Kuyruk yöneticisi adı geçerli değil ya da bilinmiyor.

**RC2059**

(2059, X'80B') Kuyruk yöneticisi bağlantı için kullanılabilir değil.

**RC2161**

(2161, X'871 ') Kuyruk yöneticisi susturuyor.

**RC2162**

(2162, X'872 ') Kuyruk yöneticisi sona erdiriliyor.

**RC2102**

(2102, X'836 ') Yetersiz sistem kaynakları var.

**RC2063**

(2063, X'80F') Güvenlik hatası oluştu.

**RC2071**

(2071, X'817 ') Kullanılabilir bellek yetersiz.

**RC2195**

(2195, X'893 ') Beklenmeyen bir hata oluştu.

**RPG Bildirimi**

```
C*..1.....2.....3.....4.....5.....6.....7..
C          CALLP      MQCONN(QMNAME : HCONN : CMPCOD :
C                               REASON)
```

Çağrıya ilişkin prototip tanımı:

```
D*..1.....2.....3.....4.....5.....6.....7..
DMQCONN      PR          EXTPROC('MQCONN')
D* Name of queue manager
D QMNAME          48A
D* Connection handle
D HCONN          10I 0
D* Completion code
D CMPCOD          10I 0
D* Reason code qualifying CMPCOD
D REASON          10I 0
```

**IBM i IBM üzerinde MQCONNX (Connect kuyruk yöneticisi (genişletilmiş))**

MQCONNX çağrısı bir uygulama programını kuyruk yöneticisine bağlar. Bu, sonraki IBM MQ çağrılarında uygulama tarafından kullanılan bir kuyruk yöneticisi bağlantı tanıtıcısı sağlar.

MQCONNX çağrısı, MQCONNX çağrısının, çağrılarının çalışma şeklini denetlemek için belirtilmesine izin vermesi dışında, MQCONN çağrısına benzer.

IBM MQ for Multiplatforms' ta, bir uygulamadaki her iş parçacığı farklı kuyruk yöneticilerine bağlanabilir. Diğer sistemlerde, bir işlem içindeki koştuzamanlı tüm bağlantıların aynı kuyruk yöneticisi olması gerekir.

- “Sözdizimi” sayfa 1260

- “Parametreler” sayfa 1260
- “RPG Bildirimi” sayfa 1260

## Sözdizimi

MQCONN (QMNAME, CNOPT, HCONN, CMPCOD, REASON)

## Parametreler

MQCONN çağrısında şu deęiřtirgeler vardır:

### QMNAME (48 baytlık karakter dizilimi)-giriř

Kuyruk yöneticisinin adı.

Ayrıntılar için “IBM üzerinde MQCONN (Connect kuyruk yöneticisi)” sayfa 1256 içinde açıklanan **QMNAME** parametresine bakın.

### CNOPT (MQCNO)-giriř/çıkıř

MQCONN 'in iřlemine denetleyen seenekler.

Ayrıntılar için bkz. “IBM i üzerinde MQCNO (Baęlantı seenekleri)” sayfa 1032.

### HCONN (10 basamaklı iřaretli tamsayı)-çıkıř

Baęlantı tanıtıcısı.

Ayrıntılar için “IBM üzerinde MQCONN (Connect kuyruk yöneticisi)” sayfa 1256 içinde açıklanan **HCONN** parametresine bakın.

### CMPCOD (10 basamaklı iřaretli tamsayı)-çıkıř

Tamamlanma kodu.

Ayrıntılar için “IBM üzerinde MQCONN (Connect kuyruk yöneticisi)” sayfa 1256 içinde açıklanan **CMPCOD** parametresine bakın.

### REASON (10 basamaklı iřaretli tamsayı)-çıkıř

Neden kodu ön eleme *CMPCOD*.

Olası neden kodlarına iliřkin ayrıntılar için “IBM üzerinde MQCONN (Connect kuyruk yöneticisi)” sayfa 1256 içinde açıklanan **REASON** parametresine bakın.

İzleyen ek neden kodları, MQCONN çağrısı tarafından döndürülebilir:

*CMPCOD* CCFAIL ise:

#### RC2278

(2278, X'8E6') İstemci baęlantı alanları geerli deęil.

#### RC2139

(2139, X'85B') Connect-options yapısı geerli deęil.

#### RC2046

(2046, X'7FE') Seenekler geerli deęil ya da tutarlı deęil.

## RPG Bildirimi

```
C*..1.....2.....3.....4.....5.....6.....7..
C          CALLP      MQCONN(QMNAME : HCONN : CMPCOD :
C                                REASON)
```

Çaęrıya iliřkin prototip tanımı:

```
D*..1.....2.....3.....4.....5.....6.....7..
```



```

DMQCONN          PR          EXTPROC('MQCONN')
D* Name of queue manager
D QMNAME          48A
D* Options that control the action of MQCONN
D HCONN          224A
D* Connection handle
D HCONN          10I 0
D* Completion code
D CMPCOD         10I 0
D* Reason code qualifying CMPCOD
D REASON         10I 0

```

## IBM i IBM üzerinde MQCRTMH (İleti tanıtıcısı yarat)

MQCRTMH çağrısı bir ileti tanıtıcısı döndürür.

Bir uygulama bunu sonraki ileti kuyruklama çağrılarında kullanabilir:

- İleti tutamacının bir özelliğini ayarlamak için [MQSETMP](#) çağrısını kullanın.
- İleti tutamacındaki bir özelliğin değerini öğrenmek için [MQINQMP](#) çağrısını kullanın.
- İleti tutamacının bir özelliğini silmek için [MQDLTMP](#) çağrısını kullanın.

İleti tanıtıcısı, MQPUT ve MQPUT1 çağrılarında kullanılabilir. İleti tanıtıcısı özelliklerini, ileti konması gereken iletinin özellikleriyle ilişkilendirir. Benzer şekilde, MQGET çağrısında bir ileti tanıtıcısı belirleyerek, MQGET çağrısı tamamlandığında ileti tanıtıcısı kullanılarak, alınmakta olan iletinin özelliklerine erişilebilir.

İleti tanıtıcısını silmek için [MQDLTMH](#) ögesini kullanın.

- “Sözdizimi” sayfa 1261
- “Parametreler” sayfa 1261
- “RPG Bildirimi” sayfa 1263

## Sözdizimi

MQCRTMH (*Hconn, CrtMsgHOpts, Hmsg, CompCode, Reason*)

## Parametreler

MQCRTMH çağrısı aşağıdaki değişirgeleri içerir:

### HCNN (10 basamaklı işaretli tamsayı)-giriş

Bu tanıtıcı, kuyruk yöneticisiyle olan bağlantıyı gösterir. The value of *HCONN* was returned by a previous *MQCONN* or *MQCONNX* call. Kuyruk yöneticisine yönelik bağlantı geçerli olmak üzere sona erdirilirse ve ileti tanıtıcısı üzerinde IBM MQ çağrısı çalışmıyorsa, iletiyi silmek için örtük olarak [MQDLTMH](#) çağrılır.

Diğer bir seçenek olarak, aşağıdaki değeri de belirleyebilirsiniz:

### HCUNAS

Bağlantı tanıtıcısı, herhangi bir kuyruk yöneticisine yönelik bir bağlantıyı göstermiyor.

When this value is used, the message handle must be deleted with an explicit call to [MQDLTMH](#) in order to release any storage allocated to it; IBM MQ never implicitly deletes the message handle.

İleti tanıtıcısını yaratan iş parçacısında bulunan bir kuyruk yöneticisine en az bir geçerli bağlantı olmalıdır; tersi durumda, arama RC2018 ile başarısız olur.

### CRTOPT (MQCMHO)-giriş

MQCRTMH işleminin denetlenmesini denetleyen seçenekler. Ayrıntılar için bkz. [MQCMHO](#).

### HMSG (20 basamaklı işaretli tamsayı)-çıkış

Çıkışta, ileti tutamacındaki özellikleri ayarlamak, sorgulamak ve silmek için kullanılacak bir ileti tanıtıcısı döndürülür. Başlangıçta ileti tanıtıcısı özellik içermez.

İleti tanıtıcısı, ilişkili bir ileti tanımlayıcısına da sahiptir. Başlangıçta bu ileti tanımlayıcısı varsayılan değerleri içerir. İlişkili ileti tanımlayıcısı alanlarının değerleri, MQSETMP ve MQINQMP çağruları kullanılarak ayarlanabilir ve sorgulanabilir. MQDLTMP çağrısı, ileti tanımlayıcısının bir alanını varsayılan değerine geri döndürür.

*HCONN* değiştirgesi HCUNAS değeri olarak belirtilirse, döndürülen ileti tanıtıcısı MQGET, MQPUT ya da MQPUT1 çağrılarında işlem birimi içindeki herhangi bir bağlantıyla kullanılabilir, ancak aynı anda yalnızca tek bir IBM MQ çağrısı tarafından kullanılabilir. Tanıtıcı, ikinci bir IBM MQ çağrısı aynı ileti tanıtıcısını kullanma girişiminde bulunursa, ikinci IBM MQ çağrısı neden kodu RC2499 ile başarısız olur.

*HCONN* parametresi HCUNAS değilse, döndürülen ileti tanıtıcısı yalnızca belirtilen bağlantıda kullanılabilir.

Aynı *HCONN* değiştirgesi değerinin, bu ileti tanıtıcısı kullanıldığı sonraki MQI çağrılarında kullanılması gerekir:

- MQDLTMH
- MQSETMP
- MQINQMP
- MQDLTMP
- MQMHBUF
- MQBUFMH

Döndürülen ileti tanıtıcısı, ileti tanıtıcısı için MQDLTMH çağrısı yayınlandığında ya da tutamaç işleminin kapsamını tanımlayan işlem birimi sonlandığında geçerli olacak şekilde durdurulur. MQDLTMH, ileti tanıtıcısı yaratıldığında belirli bir bağlantı sağlanırsa ve kuyruk yöneticisiyle bağlantı geçerliyse, örtük olarak çağrılır; örneğin, MQDBC çağrılırsa.

#### **CMPCOD (10 basamaklı işaretli tamsayı)-çıkış**

Tamamlanma kodu; bu kod aşağıdakilerden biridir:

##### **CCOK**

İşlem başarıyla tamamlandı.

##### **CCCFAIL**

Arama başarısız oldu.

#### **REASON (10 basamaklı işaretli tamsayı)-çıkış**

Neden kodu niteleyici *CMPCOD*.

*CMPCOD* CCOK ise:

##### **YOK**

(0, X'000 ') Raporlamak için bir neden yok.

*CMPCOD* CCCFAIL ise:

##### **RC2204**

(2204, X'089C') Bağdaştırıcı kullanılmıyor.

##### **RC2130**

(2130, X'852 ') Bağdaştırıcı hizmeti modülü yüklenemiyor.

##### **RC2157**

(2157, X'86D') Birincil ve ana sunucu ASID değerleri farklı.

##### **RC2219**

(2219, X'08AB') Önceki arama tamamlanmadan önce girilen MQI çağrısı.

##### **RC2461**

(2461, X'099D') İleti tanıtıcısı yaratma seçenekleri yapısı geçerli değil.

##### **RC2273**

(2273, X'7D9') Kuyruk yöneticisine yönelik bağlantı kaybedildi.

**RC2017**

(2017, X'07E1') Kullanılabilir başka tanıtıcı yok.

**RC2018**

(2018, X'7E2') Bağlantı tanıtıcısı geçerli değil.

**RC2460**

(2460, X'099C') İleti tanıtıcısı göstergesi geçerli değil.

**RC2046**

(2046, X'07FE') Seçenekler geçerli değil ya da tutarlı değil.

**RC2071**

(2071, X'817 ') Kullanılabilir bellek yetersiz.

**RC2195**

(2195, X'893 ') Beklenmeyen bir hata oluştu.

Daha ayrıntılı bilgi için bkz. [“IBM i için dönüş kodları \(ILE RPG\)” sayfa 1403](#) .

**RPG Bildirimi**

```
C*.1.....2.....3.....4.....5.....6.....7..
C          CALLP      MQCRTMH(HCONN : CRTOPT : HMSG :
                          CMPCOD : REASON)
```

Çağrıya ilişkin prototip tanımı:

```
DMQCRTMH          PR          EXTPROC('MQCRTMH')
D* Connection handle
D HCONN           10I 0 VALUE
D* Options that control the action of MQCRTMH
D CRTOPT          12A
D* Message handle
D HMSG            20I 0
D* Completion code
D CMPCOD          10I 0
D* Reason code qualifying CompCode
D REASON          10I 0
```

**IBM i IBM üzerinde MQCTL (geri çağırma denetimi)**

MQCTL çağırısı, bağlantı için açılan nesne tanıtıcılarında denetleme işlemlerini gerçekleştirir.

- [“Sözdizimi” sayfa 1263](#)
- [“Kullanım notları” sayfa 1263](#)
- [“Parametreler” sayfa 1263](#)
- [“RPG Bildirimi” sayfa 1268](#)

**Sözdizimi**

MQCTL (*Hconn, Operation, ControlOpts, CompCode, Reason*)

**Kullanım notları**

1. Geri bildirme yordamları, çağırdıkları tüm hizmetlerden gelen yanıtları denetlemeli ve yordam çözülemeyen bir koşul saptarsa, geri çağırma yordamlarına yinelenen çağrıların önlenmesi için bir MQCB (CBREG) komutu yayınlamalıdır.

**Parametreler**

MQCTL çağırısı aşağıdaki parametrelere sahiptir:

## **HCONN (10 basamaklı işaretli tamsayı)-giriş**

Bu tanıtıcı, kuyruk yöneticisiyle olan bağlantıyı gösterir. The value of *HCONN* was returned by a previous MQCONN or MQCONNX call.

## **OPERATN (10 basamaklı işaretli tamsayı)-giriş**

Belirtilen nesne tanıtıcısı için tanımlanan geri bildirme işleminde işlem yapılıyor. Aşağıdaki seçeneklerden birini ve yalnızca birini belirtmeniz gerekir:

### **CTLSR**

Belirtilen bağlantı tanıtıcısı için, tanımlı tüm ileti tüketicisi işlevleri için iletilerin tütütümesini başlatın.

Geri çağrılar, sistem tarafından başlatılan bir iş parçacığının üzerinde çalışır; bu, uygulama iş parçacıklarının hiçbirinden farklıdır.

Bu işlem, sistem tarafından sağlanan bağlantı tanıtıcısı denetimini verir. Tüketici iş parçacığı dışındaki bir iş parçacığı tarafından yayınlanabilen yalnızca MQI çağrıları şunlardır:

- CTLSR İşlemi ile MQCTL
- CTLSU İşlemi ile MQCTL
- MQDISC-HConn bağlantısı kesilmeden önce, bu işlem CTLSR işlemi ile MQCTL ' yi gerçekleştirir.

Bağlantı tanıtıcısı başlatıldığında bir IBM MQ API çağrısı yayınlanırsa ve çağrı bir ileti tüketicisi işlevinden kaynaklanmazsa,RC2500 döndürülür.

Bir bağlantı başarısız olursa, bu bağlantı mümkün olan en kısa sürede sona ermektedir. Bu nedenle, ana iş parçacığında bir süre için RC2500 dönüş kodunu almak üzere ana iş parçacığında yayınlanan bir IBM MQ API çağrısı için, bağlantı durdurulan duruma geri döndüğünde RC2009 dönüş kodunu alır.

Bu, bir tüketici işlevinde yayınlanabilir. Geri çağırma yordamıyla aynı bağlantı için, tek amacı önceden verilmiş bir CTLSR işlemi iptal etmek olabilir.

Uygulama iş parçacıklı bir IBM MQ kitaplığından bağlıysa, bu seçenek desteklenmez.

### **CTLSW**

Belirtilen bağlantı tanıtıcısı için, tanımlı tüm ileti tüketicisi işlevleri için iletilerin tütütümesini başlatın.

İleti tüketicileri aynı iş parçacığı üzerinde çalışır ve denetim MQCTL ' nin çağrısına döndürülünceye kadar döndürülmez:

- MQCTL CTLSR ya da CTLSU işlemleri tarafından serbest bırakıldı ya da
- Tüm tüketici yordamları kayıttan kaldırıldı ya da askıya alındı.

Tüm tüketiciler kayıttan kaldırılırsa ya da askıya alınırsa, örtük bir CTLSR işlemi yayınlanır.

Bu seçenek, yürürlükteki bağlantı tanıtıcısı ya da başka bir bağlantı tanıtıcısı için, geri bildirme yordamından kullanılamaz. Arama girişiminde bulunulursa, RC2012ile birlikte geri döndürülür.

Bir CTLSW işlemi sırasında herhangi bir zamanda kayıtlı olmayan, askıya alınmış olmayan tüketicilere arama başarısız olur ve bir RC2446neden kodu ile başarısız olur.

Bir CTLSW işlemi sırasında bağlantı askıya alınırsa, MQCTL çağrısı RC2521; için bir uyarı neden kodu döndürür; bağlantı 'başlatıldı' kalır.

Uygulama CTLSR ya da CTLRE komutunu yayınlamayı seçebilir. Bu örnekte, CTLRE işlem öbekleri.

Tek bir iş parçacıklı istemcide bu seçenek desteklenmez.

### **CTLSP**

İletilerin tüketmesini durdurun ve tüm tüketicilerin bu seçenek tamamlanmadan operasyonlarını tamamlaması için bekleyin. Bu işlem bağlantı tanıtıcısını serbest bırakır.

Bu seçenek, bir geri çağrı yordamından yayınlandıysa, yordam çıkılıncaya kadar bu seçenek yürürlüğe girmez. Zaten okunan iletiler için tüketici yordamlarından sonra ve geri çağırma yordamlarıyla çağrılan çağrılar (istendiyse) sonrasında, başka ileti tüketici yordamı çağrılmaz.

Bir geri çağrı yordamı dışında yayınlanırsa, denetim, önceden okunan iletiler için tüketici rutinleri tamamlanincaya kadar ve çağrılar durdurulduktan sonra (istendiyse) çağrılara geri dönmez (istenirse). Ancak geri çağrılar, kayıtlı olarak kalmaya devam ediyor.

Bu işlevin önünde okuma yazma hiçbir etkisi yoktur. Teslim edilecek başka iletilerin olup olmadığını belirlemek için geri bildirme işlevi içinden MQCLOSE (COQSC) programını çalıştırdığınızdan emin olmalısınız.

#### **CTLSU**

İletilerin tüketilmesini duraklatın. Bu işlem bağlantı tanıtıcısını serbest bırakır.

Bu, uygulamaya ilişkin iletilerin önden okuma işlemini etkilemez. Uzun süre boyunca iletileri tüketmeyi durdurmak istiyorsanız, kuyruğun kapatılıp tüketimin devam etmesi gerekirken yeniden açmayı düşünün.

Bir geri çağrı yordamından yayınlanırsa, yordam çıkışlarına kadar yürürlüğe girmez. Yürürlükteki yordam çıkışlarından sonra ileti tüketici yordamlarından başka bir yordam çağrılacak.

Bir geri çağırma dışında yayınlanırsa, denetim, yürürlükteki tüketici yordamı tamamlanincaya kadar arayanın çağrısına dönmez ve daha fazla çağrılmaz.

#### **CCTLRE**

İletilerin tüketilmesine devam edin.

Bu seçenek genellikle ana uygulama iş parçacığından verilir, ancak aynı yordamda yayınlanan daha önceki bir askıya alma isteğini iptal etmek için bir geri çağırma yordamından da kullanılabilir.

CTLRE, bir CTLSW ' yi sürdürmek için kullanılırsa, işlem öbekleri kullanılır.

#### **PCTLOP (MQCTLO)-giriş**

MQCTL ' nin işlemini denetleyen seçenekler

Yapıyla ilgili ayrıntılar için [MQCTLO](#) başlıklı konuya bakın.

#### **CMPCOD (10 basamaklı işaretli tamsayı)-çıkış**

Tamamlanma kodu; bu kod aşağıdakilerden biridir:

##### **CCOK**

İşlem başarıyla tamamlandı.

##### **CCWARN**

Uyarı (kısmi tamamlama).

##### **CCCFAIL**

Arama başarısız oldu.

#### **REASON (10 basamaklı işaretli tamsayı)-çıkış**

Aşağıdaki neden kodları, kuyruk yöneticisinin **Reason** parametresi için döndürebileceği kodlardır.

CMPCOD CCOK ise:

##### **YOK**

(0, X'000 ') Raporlamak için bir neden yok.

CMPCOD CCFAIL ise:

##### **RC2133**

(2133, X'855 ') Veri dönüştürme hizmetleri modüllerinin yüklenmesi gerçekleştirilemiyor.

##### **RC2204**

(2204, X'89C') Bağdaştırıcı kullanılmıyor.

##### **RC2130**

(2130, X'852 ') Bağdaştırıcı hizmeti modülü yüklenemiyor.

**RC2374**

(2374, X' 946 ') API çıkışı başarısız oldu.

**RC2183**

(2183, X'887 ') API çıkışı yüklenemiyor.

**RC2157**

(2157, X'86D') Birincil ve ana sunucu ASID değerleri farklı.

**RC2005**

(2005, X'7D5') Arabellek uzunluğu parametresi geçerli değil.

**RC2487**

(2487, X'9B7') Geri çağırma yordamı çağırılmıyor

**RC2448**

(2448, X' 990 ') Kayıtlı bir geri çağırma olmadığı için Deregister, Suspend ya da Sürdürme işlemi gerçekleştirilemiyor

**RC2486**

(2486, X'9B6') Bir CBREG çağrısında hem CallbackFunction hem de CallbackName belirtildi ya da CallbackFunction ya da CallbackName ' dan biri belirtildi, ancak kayıtlı geri bildirme işleviyle eşleşmiyor.

**RC2483**

(2483, X'9B3') Yanlış CallBackTip alanı.

**RC2219**

(2219, X'8AB') Önceki arama tamamlanmadan önce girilen MQI çağrısı.

**RC2444**

(2444, X'98C') Seçenek bloğu yanlış.

**RC2484**

(2484, X'9B4') MQCBD seçenekleri alanı yanlış.

**RC2140**

(2140, X'85C') Bekleme isteği CICStarafından reddedildi.

**RC2009**

(2009, X'7D9') Kuyruk yöneticisine yönelik bağlantı kaybedildi.

**RC2217**

(2217, X'8A9') Bağlantı için yetkili değil.

**RC2202**

(2202, X'89A') Bağlantı susturulmuş durumda.

**RC2203**

(2203, X'89B') Bağlantı sona erdiriliyor.

**RC2207**

(2207, X'89F') İntilendirme tanıtıcısı hatası.

**RC2016**

(2016, X'7E0') Kuyruğun engellenmesini sağlar.

**RC2351**

(2351, X'92F') Genel iş çakışmaları birimleri.

**RC2186**

(2186, X'88A') Al-ileti seçenekleri yapısı geçerli değil.

**RC2353**

(2353, X' 931 ') Genel iş birimi için kullanılan tanıtıcı.

**RC2018**

(2018, X'7E2') Bağlantı tanıtıcısı geçerli değil.

**RC2019**

(2019, X'7E3') Nesne tanıtıcısı geçerli değil.

**RC2259**

(2259, X'8D3') Tutarsız göz atma belirtimi.

**RC2245**

(2244, X'8C5') Tutarsız iş birliği-iş belirtimi.

**RC2246**

(2246, X'8C6') İmleç altındaki ileti alma için geçerli değil.

**RC2352**

(2352, X' 930 ') Genel iş birimi yerel iş birimi ile çakışıyor.

**RC2247**

(2247, X'8C7') Eşleştirme seçenekleri geçerli değil.

**RC2485**

(2485, X'9B5') Yanlış MaxMsgUzunluk alanı

**RC2026**

(2026, X'7EA') İleti tanımlayıcısı geçerli değil.

**RC2497**

(2497, X'9C1') Belirtilen işlem giriş noktası modülde bulunamadı.

**RC2496**

(2496, X'9C0') Modül bulundu, ancak yanlış tipte (32 bit ya da 64 bit) ya da geçerli bir dll değil.

**RC2495**

(2495, X'9BF') Modül, arama yolunda bulunamadı ya da yükleme yetkisine sahip değil.

**RC2206**

(2206, X'89E') İleti-tanımlayıcısı hatası.

**RC2250**

(2250, X'8CA') İleti sıra numarası geçerli değil.

**RC2331**

(2331, X'91B') İleti simgesinin kullanımı geçerli değil.

**RC2036**

(2036, X'7F4') Kuyruk göz atma için açık değil.

**RC2037**

(2037, X'7F5') Kuyruk giriş için açık değil.

**RC2041**

(2041, X'7F9') Nesne tanımlaması açıldığından beri değiştirildi.

**RC2101**

(2101, X'835 ') Nesne zarar gördü.

**RC2488**

(2488, X'9B8') API Çağrısında Hatalı İşlem kodu

**RC2046**

(2046, X'7FE') Seçenekler geçerli değil ya da tutarlı değil.

**RC2193**

(2193, X'891 ') Sayfa kümesi veri kümesine erişilirken hata oluştu.

**RC2052**

(2052, X'804 ') Kuyruk silinmiştir.

**RC2394**

(2394, X'95A') Kuyruk yanlış izin tipi içeriyor.

**RC2058**

(2058, X'80A') Kuyruk yöneticisi adı geçerli değil ya da bilinmiyor.

**RC2059**

(2059, X'80B') Kuyruk yöneticisi bağlantı için kullanılabilir değil.

**RC2161**

(2161, X'871 ') Kuyruk yöneticisi susturuyor.

**RC2162**

(2162, X'872 ') Kuyruk yöneticisi sona erdiriliyor.

**RC2102**

(2102, X'836 ') Yetersiz sistem kaynakları var.

**RC2069**

(2069, X'815 ') Bu tutamaç için bekleyen sinyal.

**RC2071**

(2071, X'817 ') Kullanılabilir bellek yetersiz.

**RC2109**

(2109, X'83D') Çıkış programı tarafından çağrı engellendi.

**RC2072**

(2072, X'818 ') Syncpoint desteği yok.

**RC2195**

(2195, X'893 ') Beklenmeyen bir hata oluştu.

**RC2354**

(2354, X' 932 ') Genel iş birimi içindeki bilgiler başarısız oldu.

**RC2355**

(2355, X' 933 ') İşlerin birim içi çağrıları karışımı desteklenmiyor.

**RC2255**

(2255, X'8CF') Kuyruk yöneticisinin kullanması için kullanılabilir iş birimi.

**RC2090**

(2090, X'82A') MQGMO' da bekleme aralığı geçerli değil.

**RC2256**

(2256, X'8D0') MQGMO' nun yanlış sürümü sağlandı.

**RC2257**

(2257, X'8D1') MQMD' nin yanlış sürümü sağlandı.

**RC2298**

(2298, X'8FA') İstenen işlev geçerli ortamda yok.

**RPG Bildirimi**

```
C*..1.....2.....3.....4.....5.....6.....7..  
C          CALLP      MQCTL(HCONN : OPERATN : PCTLOP :  
                          CMPCOD : REASON)
```

Çağrıya ilişkin prototip tanımı:

```
DMQCTL          PR          EXTPROC('MQCTL')  
D* Connection handle  
D HCONN          10I 0 VALUE  
D* Operation  
D OPERATN        10I 0 VALUE  
D* Control options  
D PCTLOP          32A  
D* Completion code  
D CMPCOD          10I 0  
D* Reason code qualifying CompCode  
D REASON          10I 0
```

**IBM i****IBM üzerinde MQDISC (Bağlantı kesme kuyruğu yöneticisi)**

MQDISC çağrısı, kuyruk yöneticisi ile uygulama programı arasındaki bağlantıyı keser ve MQCONN ya da MQCONNX çağrısının tersi olur.



- “Sözdizimi” sayfa 1269
- “Kullanım notları” sayfa 1269
- “Parametreler” sayfa 1269
- “RPG Bildirimi” sayfa 1270

## Sözdizimi

MQDISC (*HCONN*, *CMPCOD*, *REASON*)

## Kullanım notları

1. Bir MQDISC çağrısı, uygulama hala nesnelere açıkken yayınlanırsa, bu nesnelere kuyruk yöneticisi tarafından kapatılır ve kapatma seçenekleri CONONE olarak ayarlanır.
2. Uygulama bir iş birimindeki kesinleştirilmemiş değişikliklerle sona ererse, bu değişikliklerin yok edilmesi, uygulamanın sona erme şekline bağlıdır:
  - a. Uygulama, bitmeden önce MQDISC çağrısını yayınlarsa:
    - Kuyruk yöneticisi eşgüdümlü iş birimi için, kuyruk yöneticisi uygulama adına MQCMIT çağrısını yayınlar. İş birimi olanaklıysa, kesinleştirilir ve desteklenmiyorsa yedeklenir.
    - Dışarıdan eşgüdümlü bir iş birimi için, iş biriminin durumunda bir değişiklik yoktur; ancak kuyruk yöneticisi, iş birimi eşgüdümcüsü tarafından istendiğinde, iş biriminin kesinleştirilmesinin gerektiğini gösterir.
  - b. Uygulama olağan bir şekilde sona ererse, ancak MQDISC çağrısını yayınlamadan iş birimi geriletilir.
  - c. Uygulama, MQDISC çağrısını yayınlamadan *anormal* ' i sona erdirirse, iş birimi geriletilir.

## Parametreler

MQDISC çağrısında aşağıdaki değişirgeler vardır:

### HCONN (10 basamaklı işaretli tamsayı)-giriş/çıkış

Bağlantı tanıtıcısı.

Bu tanıtıcı, kuyruk yöneticisiyle olan bağlantıyı gösterir. The value of *HCONN* was returned by a previous MQCONN or MQCONNX call.

Çağrı başarıyla tamamlandığında, kuyruk yöneticisi *HCONN* değerini, ortam için geçerli bir tanıtıcı olmayan bir değere ayarlar. Bu değer:

#### HCUNUH

Kullanılmayan bağlantı tanıtıcısı.

### CMPCOD (10 basamaklı işaretli tamsayı)-çıkış

Tamamlanma kodu.

Bu, aşağıdakilerden biridir:

#### CCOK

İşlem başarıyla tamamlandı.

#### CCWARN

Uyarı (kısmi tamamlama).

#### CCCFAIL

Arama başarısız oldu.

### REASON (10 basamaklı işaretli tamsayı)-çıkış

Neden kodu ön eleme *CMPCOD*.

*CMPCOD* CCOK ise:

## YOK

(0, X'000 ') Raporlamak için bir neden yok.

CMPCOD CCFAIL ise:

## RC2219

(2219, X'8AB') MQI çağrısı, önceki arama tamamlanmadan önce yeniden girildi.

## RC2009

(2009, X'7D9') Kuyruk yöneticisine yönelik bağlantı kaybedildi.

## RC2018

(2018, X'7E2') Bağlantı tanıtıcısı geçerli değil.

## RC2058

(2058, X'80A') Kuyruk yöneticisi adı geçerli değil ya da bilinmiyor.

## RC2059

(2059, X'80B') Kuyruk yöneticisi bağlantı için kullanılabilir değil.

## RC2162

(2162, X'872 ') Kuyruk yöneticisi sona erdiriliyor.

## RC2102

(2102, X'836 ') Yetersiz sistem kaynakları var.

## RC2071

(2071, X'817 ') Kullanılabilir bellek yetersiz.

## RC2195

(2195, X'893 ') Beklenmeyen bir hata oluştu.

## RPG Bildirimi

```
C*..1.....2.....3.....4.....5.....6.....7..
C          CALLP      MQDISC(HCONN : CMPCOD : REASON)
```

Çağrıya ilişkin prototip tanımı:

```
D*..1.....2.....3.....4.....5.....6.....7..
DMQDISC      PR          EXTPROC('MQDISC')
D* Connection handle
D HCONN          10I 0
D* Completion code
D CMPCOD          10I 0
D* Reason code qualifying CMPCOD
D REASON          10I 0
```

## IBM i MQDLTMH (Delete message handle) on IBM i

MQDLTMH çağrısı bir ileti tanıtıcısını siler ve MQCRTMH çağrısının tersi olur.

- [“Sözdizimi” sayfa 1270](#)
- [“Kullanım notları” sayfa 1271](#)
- [“Parametreler” sayfa 1272](#)
- [“RPG Bildirimi” sayfa 1273](#)

## Sözdizimi

MQDLTMH ((Hconn, Hmsg, DltMsgHOpts, CompCode, Reason)

## Kullanım notları

1. Bu çağrıyı, yalnızca kuyruk yöneticisinin iş birimini koordine ettiğinde kullanabilirsiniz. Bu durumda şunlar olabilir:
  - Değişikliklerin yalnızca IBM MQ kaynaklarını etkilediği yerel bir iş birimi.
  - Değişikliklerin, diğer kaynak yöneticilerine ait kaynakları etkileyebileceği gibi, IBM MQ kaynaklarını etkilediği genel bir iş birimi.

Yerel ve genel çalışma birimleriyle ilgili daha fazla ayrıntı için bkz. [“IBM üzerinde MQBEGIN \(İş birimini başlat\)” sayfa 1233.](#)
2. Kuyruk yöneticisinin iş birimini koordine etmediği ortamlarda, MQBACK yerine uygun geri çağırma çağrısını kullanın. Ortam, uygulamanın olağan dışı bir şekilde sonlandırılmasına neden olan örtük bir geri dönüş özelliğini de destekleyebilir.
  - z/OS' ta aşağıdaki çağrıları kullanın:
    - İş birimi yalnızca IBM MQ kaynaklarını etkilediğinde, toplu iş programları ( IMS toplu DL/I programları da içinde olmak üzere) MQBACK çağrısını kullanabilir. However, if the unit of work affects both IBM MQ resources and resources belonging to other resource managers (for example, Db2 ), use the SRRBACK call provided by the z/OS Recoverable Resource Service (RRS). SRRBACK çağrısı, RRS eşgüdümü için etkinleştirilen kaynak yöneticilerine ait kaynaklarda yapılan değişiklikleri destekliyor.
    - CICS applications must use the EXEC CICS SYNCPOINT ROLLBACK command to back out the unit of work. Do not use the MQBACK call for CICS applications.
    - IMS uygulamaları (toplu DL/I programlarından farklı), iş birimini yedeklemek için ROLB gibi IMS çağrıları kullanılmalıdır. Do not use the MQBACK call for IMS applications (other than batch DL/I programs).
  - IBM i' ta, kuyruk yöneticisi tarafından koordine edilen yerel iş birimleri için bu çağrıyı kullanın. This means that a commitment definition must not exist at job level, that is, the STRCMTCTL command with the **CMTSCOPE (\*JOB)** parameter must not have been issued for the job.
3. Bir uygulama, bir iş biriminde kesinleştirilmemiş değişikliklerle sona ererse, bu değişikliklerin yok edilmesi, uygulamanın olağan ya da olağan dışı bir şekilde sona ermesine bağlıdır. Ek ayrıntılar için [“IBM üzerinde MQDISC \(Bağlantı kesme kuyruğu yöneticisi\)” sayfa 1268](#) içindeki kullanım notlarına bakın.
4. Bir uygulama, gruplara ya da mantıksal ileti bölümlerine ileti yerleştirdiğinde ya da ileti aldığı anda, kuyruk yöneticisi, son başarılı MQPUT ve MQGET çağrılarına ilişkin ileti grubuyla ve mantıksal iletiyle ilgili bilgileri saklar. Bu bilgiler kuyruk tanıtıcısı ile ilişkilendirilir ve aşağıdaki gibi şeyler içerir:
  - MQMD ' de *GroupId*, *MsgSeqNumber*, *Offset* ve *MsgFlags* alanlarının değerleri.
  - İletinin bir iş biriminin parçası olup olmadığı.
  - MQPUT çağrısına ilişkin: İletinin kalıcı mı, yoksa kalıcı değil mi olduğu.

Kuyruk yöneticisi, her biri için bir küme olmak üzere üç grup grup ve bölüm bilgisi kümesi tutar:

  - Son başarılı MQPUT çağrısı (bu, bir iş biriminin bir parçası olabilir).
  - Kuyruktan ileti kaldıran son başarılı MQGET çağrısı (bu, bir iş biriminin parçası olabilir).
  - Kuyruktan ileti alan son başarılı MQGET çağrısı (bu, bir iş biriminin parçası olamaz).

Uygulama, iletileri bir iş biriminin bir parçası olarak koyduysa ya da alıyorsa ve uygulama iş birimini yedeklediyse, grup ve bölüm bilgileri daha önce sahip olduğu değere geri yüklenir:

  - MQPUT çağrısıyla ilişkili bilgiler, yürürlükteki iş biriminde o kuyruk tanıtıcısı için ilk başarılı MQPUT çağrısından önce sahip olduğu değere geri yüklendi.
  - MQGET çağrısıyla ilişkili bilgiler, yürürlükteki iş biriminde o kuyruk tanıtıcısı için ilk başarılı MQGET çağrısından önce sahip olduğu değere geri yüklenir.

Çalışma birimi başladıktan sonra uygulama tarafından güncellenen, ancak iş biriminin kapsamı dışında olan kuyruklar, iş birimi getirilirse, grup ve bölüm bilgilerini geri yüklemez.

Bir iş birimi yedeklendiğinde grup ve kesim bilgilerinin önceki değerine geri yüklenmesi, uygulamanın çok sayıda iş biriminden oluşan büyük bir ileti grubu ya da büyük mantıksal ileti yayınlamasını ve iş birimlerinden biri başarısız olursa, ileti grubunun ya da mantıksal iletinin doğru noktasında yeniden başlatılmasına olanak tanır. Yerel kuyruk yöneticisinin kuyruk depolaması sınırlı olması durumunda, birkaç iş birimi kullanılması avantajlı olabilir. Ancak, bir sistem hatası ortaya çıkarsa, uygulama, doğru noktada iletileri yeniden başlatabilmek ya da iletileri almak için yeterli bilgileri sağlamalıdır.

Sistem hatasından sonra doğru noktada yeniden başlatılabilmeye ilişkin ayrıntılar için, [PMOPT \(10 basamaklı imzalı tamsayı\)](#) içinde açıklanan PMLOGO seçeneğine ve [GMOPT \(10 basamaklı işaretli tamsayı\)](#) içinde açıklanan GMLOGO seçeneğini kullanın.

Kalan kullanım notları, yalnızca kuyruk yöneticisi iş birimlerini koordine ettiğinde geçerlidir:

5. Bir iş birimi, bağlantı tanıtıcısı ile aynı kapsama sahiptir. Belirli bir iş birimini etkileyen tüm IBM MQ çağrılarını, aynı bağlantı tanıtıcısı kullanılarak gerçekleştirilmelidir. Farklı bir bağlantı tanıtıcısı kullanılarak verilen çağrılar (örneğin, başka bir uygulama tarafından yayınlanan çağrılar) farklı bir iş birimini etkiler. Bağlantı tutamaçlarının kapsamına ilişkin bilgi için [HCONN \(10 basamaklı işaretli tamsayı\)](#)-çıkış başlıklı konuya bakın.
6. Bu çağrıdan etkilenenler, yalnızca yürürlükteki iş biriminin bir parçası olarak alınan ya da alınan iletiler etkilenir.
7. Bir iş birimi içinde MQGET, MQPUT ya da MQPUT1 tarafından çağrılan, ancak bir kesinleştirme ya da geriletme çağrısını hiçbir zaman yayınlamayan, uzun süredir çalışan bir uygulama, diğer uygulamaların kullanımına sunulmayan iletilerle kuyrukları doldurabilir. Bu olasılığa karşı koruma sağlamak için, denetiminin **MaxUncommittedMsgs** kuyruk yöneticisi özneliğini, kuyrukları dolduran kaçak uygulamaları önleyecek kadar düşük bir değere ayarlaması gerekir, ancak beklenen ileti sistemi uygulamalarının doğru şekilde çalışmasına izin verecek kadar yüksek olmalıdır.

## Parametreler

MQDLTMH çağrısı aşağıdaki deęiřtirgeleri içerir:

### **HCONN (10 basamaklı işaretli tamsayı)-giriř**

Bu tanıtıcı, kuyruk yöneticisiyle olan bağlantıyı gösterir.

Deęerin, **HMSG** parametresinde belirtilen ileti tanıtıcısını yaratmak için kullanılan bağlantı tanıtıcısı ile eřleşmesi gerekir.

İleti tanıtıcısı HCUNAS kullanılarak yaratıldıysa, ileti tanıtıcısını silmeye ilişkin iş parçasısında geçerli bir bağlantı oluşturulmalıdır; tersi durumda, çağrı RC2009 ile başarısız olur.

### **HMSG (20 basamaklı işaretli tamsayı)-giriř/çıkıř**

Bu, silinmek üzere ileti tanıtıcısıdır. Deęer önceki bir MQCRTMH çağrısı tarafından döndürüldü.

Çaęrıyı başarıyla tamamlamaya ilişkin tanıtıcı, ortam için geçersiz bir değere ayarlıdır. Bu deęer:

#### **HMUNUH**

Kullanılmayan ileti tanıtıcısı.

The message handle cannot be deleted if another IBM MQ call is in progress that was passed the same message handle.

### **DLTOPT (MQDMHO)-giriř**

Ayrıntılar için bkz. [MQDMHO](#) .

### **CMPCOD (10 basamaklı işaretli tamsayı)-çıkıř**

Tamamlanma kodu; bu kod aşağıdakilerden biridir:

#### **CCOK**

İřlem başarıyla tamamlandı.

#### **CCCFAIL**

Arama başarısız oldu.

## REASON (10 basamaklı işaretli tamsayı)-çıkış

Neden kodu niteleyici *CMPCOD*.

*CMPCOD* CCOK ise:

### YOK

(0, X'000 ') Raporlamak için bir neden yok.

*CMPCOD* değeri CCFAIL ise:

### RC2204

(2204, X'089C') Bağdaştırıcı kullanılmıyor.

### RC2130

(2130, X'852 ') Bağdaştırıcı hizmeti modülü yüklenemiyor.

### RC2157

(2157, X'86D') Birincil ve ana sunucu ASID değerleri farklı.

### RC2219

(2219, X'08AB') Önceki arama tamamlanmadan önce girilen MQI çağrısı.

### RC2009

(2009, X'07D9') Kuyruk yöneticisine yönelik bağlantı kaybedildi.

### RC2462

(2462, X'099E') İleti tanıtıcısı seçeneklerini silme yapısı geçerli değil.

### RC2460

(2460, X'099C') İleti tanıtıcısı göstergesi geçerli değil.

### RC2499

(2499, X'09C3') İleti tanıtıcısı zaten kullanılıyor.

### RC2046

(2046, X'07FE') Seçenekler geçerli değil ya da tutarlı değil.

### RC2071

(2071, X'817 ') Kullanılabilir bellek yetersiz.

### RC2195

(2195, X'893 ') Beklenmeyen bir hata oluştu.

Daha ayrıntılı bilgi için bkz. ["IBM i için dönüş kodları \(ILE RPG\)" sayfa 1403](#) .

## RPG Bildirimi

```
C*.1.....2.....3.....4.....5.....6.....7..
C          CALLP      MQDLTMH(HCONN : HMSG : DLTOPT :
                      CMPCOD : REASON)
```

Çağrıya ilişkin prototip tanımı:

```
DMQDLTMH          PR          EXTPROC('MQDLTMH')
D* Connection handle
D HCONN           10I 0 VALUE
D* Message handle
D HMSG           20I 0
D* Options that control the action of MQDLTMH
D DLTOPT         12A
D* Completion code
D CMPCOD         10I 0
D* Reason code qualifying CompCode
D REASON         10I 0
```

## MQDLTMP-İleti özelliğini sil

MQDLTMP çağrısı, bir özelliği ileti tanıtıcısından siler ve MQSETMP çağrısının tersi olur.

- “Sözdizimi” sayfa 1274
- “Parametreler” sayfa 1274
- “RPG Bildirimi” sayfa 1275

## Sözdizimi

MQDLTMP (*Hconn, Hmsg, DltPropOpts, Name, CompCode, Reason*)

## Parametreler

MQDLTMP çağrısı aşağıdaki deęiřtirgeleri içerir:

### HCONN (10 basamaklı iřaretli tamsayı)-Giriř

Bu tanıtıcı, kuyruk yöneticiyle olan baęlantıyı gösterir. Deęerin, **HMSG** parametresinde belirtilen ileti tanıtıcısını yaratmak için kullanılan baęlantı tanıtıcısı ile eřleşmesi gerekir.

İleti tanıtıcısı HCUNAS kullanılarak yaratıldıysa, ileti tanıtıcısını silmeye iliřkin iř parçacısında geçerli bir baęlantı oluşturulmalıdır; tersi durumda, çağrı RC2009 ile başarısız olur.

### HMSG (20 basamaklı iřaretli tamsayı)-giriř

Bu, silinecek özellięi içeren ileti tanıtıcısıdır. Deęer önceki bir MQCRTMH çağrısı tarafından döndürüldü.

### DLTOPT (MQDMPO)-Giriř

Ayrıntılı bilgi için [MQDMPO](#) veri tipine bakın.

### PRNAME (MQCHARV)-giriř

Silinecek özellięin adı. Özellik adlarıyla ilgili ek bilgi için [Özellik adları](#) konusuna bakın.

Özellik adında genel arama karakterlerine izin verilmez.

### CMPCOD (10 basamaklı iřaretli tamsayı)-çıkıř

Tamamlanma kodu; bu kod aşağıdakilerden biridir:

#### CCOK

İřlem başarıyla tamamlandı.

#### CCWARN

Uyarı (kısmi tamamlama).

#### CCCFAIL

Arama başarısız oldu.

### REASON (10 basamaklı iřaretli tamsayı)-çıkıř

Neden kodu niteleyici *CMPCOD*.

*CMPCOD* CCOK ise:

#### YOK

(0, X'000 ') Raporlamak için bir neden yok.

*CMPCOD* CCWARN ise:

#### RC2471

(2471, X'09A7') Özellik yok.

#### RC2421

(2421, X'0975 ') Özellikler içeren bir MQRFH2 klasörü ayrıştırılmadı.

*CMPCOD* CCCFAIL ise:

#### RC2204

(2204, X'089C') Baędařtırıcı kullanılmıyor.

**RC2130**

(2130, X'0852 ') Baędařtırıcı hizmeti modülü yüklenemiyor.

**RC2157**

(2157, X'086D') Birincil ve ana ASID deęerleri farklı.

**RC2219**

(2219, X'08AB') Önceki arama tamamlanmadan önce girilen MQI çağırısı.

**RC2009**

(2009, X'07D9') Kuyruk yöneticisine yönelik baęlantı kaybedildi.

**RC2481**

(2481, X'09B1') İleti silme seçenekleri yapısı geçerli deęil.

**RC2460**

(2460, X'099C') İleti tanıtıcısı geçerli deęil.

**RC2499**

(2499, X'09C3') İleti tanıtıcısı zaten kullanılıyor.

**RC2046**

(2046, X'07FE') Seçenekler geçerli deęil ya da tutarlı deęil.

**RC2442**

(2442, X'098A') Özellik adı geçersiz.

**RC2111**

(2111, X'083F') Özellik adı kodlanmış karakter takımı tanıtıcısı geçerli deęil.

**RC2195**

(2195, X'0893 ') Beklenmeyen bir hata oluştu.

Bu kodlarla ilgili daha fazla bilgi için bkz. [API tamamlama ve neden kodları](#).

**RPG Bildirimi**

```
C*..1.....2.....3.....4.....5.....6.....7..
C                                CALLP      MQDLTMP(HCONN : HMSG : DLTOPT :
                                PRNAME : CMPCOD : REASON)
```

Çaęrıya ilişkin prototip tanımı:

```
DMQDLTMP          PR                EXTPROC('MQDLTMP')
D* Connection handle
D HCONN                10I 0 VALUE
D* Message handle
D HMSG                20I 0 VALUE
D* Options that control the action of MQDLTMP
D DLTOPT                12A
D* Property name
D PRNAME                32A
D* Completion code
D CMPCOD                10I 0
D* Reason code qualifying CompCode
D REASON                10I 0
```

**IBM i IBM üzerinde MQGET (İleti al)**

MQGET çağırısı, MQOPEN çağırısı kullanılarak açılmış olan bir yerel kuyruktan bir ileti alır.

- [“Sözdizimi” sayfa 1276](#)
- [“Kullanım notları” sayfa 1276](#)
- [“Parametreler” sayfa 1278](#)
- [“RPG Bildirimi” sayfa 1283](#)

## Sözdizimi

MQGet (HCONN, HOBJ, MSGDSC, GMO, BUFLN, BUFFER, DATLEN, CMPCOD, REASON)

## Kullanım notları

1. Alınan ileti olağan durumda kuyruktan silinir. Bu silme, MQGET çağrısının bir parçası olarak ya da bir uyumluluk noktasının bir parçası olarak ortaya çıkabilir. Message deletion does not occur if a GMBRWF or GMBRWN option is specified on the **GMO** parameter (see the *GMOPT* field described in “IBM üzerinde MQGMO (get-message options)” sayfa 1060 ).
2. Göz atma seçeneklerinden biri olan GMLK seçeneği belirtilirse, göz atma iletisi yalnızca bu tutamaçla görünür olacak şekilde kilitletir.

GMUNLK seçeneği belirtilirse, önceden kilitletlenmiş bir iletinin kilidi açılır. Bu durumda hiçbir ileti alınmaz; **MSGDSC**, **BUFLN**, **BUFFER** ve **DATLEN** parametreleri denetlenmez ya da değiştirilmez.

3. MQGET çağrısını yayınlayan uygulama IBM MQ MQI clientolarak çalışıyorsa, MQGET çağrısının işlenmesi sırasında IBM MQ MQI client olağandışı bir şekilde sona erdirilirse ya da istemci bağlantısı kesilirse, iletinin kaybolması mümkün olur. Bu durum, kuyruk yöneticisinin altyapısında çalışan ve istemci adına MQGET çağrısının istemci tarafından istemciyi kaybedinceye kadar istemcinin kaybını saptayamadığı için bu durum ortaya çıkar; bu işlem, iletinin kuyruktan kaldırılmasından sonradır. Bu durum hem kalıcı iletiler, hem de kalıcı olmayan iletiler için oluşabilir.

İletilerin bu şekilde kaybetme riski, her zaman iş birimi içindeki iletileri almak (yani, MQGET çağrısındaki GMSYP seçeneğini belirterek ve iletinin işlenmesi sırasında iş birimini işlemek ya da geri almak için MQCMIT ya da MQBACK çağrılarını kullanarak) ortadan kaldırılabilir. GMSYP belirtilirse ve istemci olağan dışı bir şekilde sona erdirilirse ya da bağlantı kesilirse, vekil, kuyruk yöneticilikteki çalışma birimini yedekler ve ileti kuyruktan geri gelir.

Prensipte, aynı durum kuyruk yöneticisinin altyapısında çalışan uygulamalarla da ortaya çıkabilir, ancak bu durumda, bir iletinin kaybedilebileceği pencere küçük olur. Ancak, IBM MQ MQI clients ile olduğu gibi, bir iş birimi içindeki ileti alınarak riskin ortadan kaldırılabilir.

4. Bir uygulama belirli bir ileti dizisini belirli bir yere koyarsatek bir iş birimi içinde kuyruğa girin ve bu iş birimini başarıyla kesinleştirerek, iletiler aşağıdaki şekilde alınabilmekte kullanılabilir olur:
  - Kuyruk *paylaşılmayan kuyruksa* (yani, yerel bir kuyruksa), iş birimi içindeki tüm iletiler aynı anda kullanılabilir olur.
  - Kuyruk *paylaşılan bir kuyruksa*, iş birimi içindeki iletiler, bunların yerleştirdikleri sırayla kullanılabilir olur, ancak tümü aynı anda kullanılamaz. Sistem yoğun olarak yüklü olduğunda, iş birimindeki ilk iletinin başarılı bir şekilde alınması mümkündür; ancak, iş birimindeki ikinci ya da sonraki ileti için MQGET çağrısının RC2033ile başarısız olması mümkündür. Böyle bir durumda, uygulamanın kısa bir süre beklemesi ve işlemi yeniden denemesini beklemeniz gerekir.
5. Bir uygulama, ileti grupları kullanılmadan aynı kuyruğa ileti dizisi koyarsa, Belirli koşullar karşılanırsa, bu iletilerin sırası korunur. Ayrıntılar için, MQPUT çağrısının açıklamasındaki kullanım notlarına bakın. Koşullar yerine getirildiyse, iletiler gönderildikleri sırayla, teslim alma uygulamasına sunulur:
  - Kuyruktan yalnızca bir alıcı ileti alıyor.

Kuyruktan ileti alan iki ya da daha fazla uygulama varsa, bunlar, bir sıraya ait iletileri tanımlamak için kullanılacak mekanizmayı gönderen ile kabul etmelidir. Örneğin, gönderici, iletilerde bulunan tüm MDCID alanlarını, bu ileti dizisi için benzersiz olan bir değere ayarlayabilir.
  - Alıcı, örneğin belirli bir MDMID ya da MDCIDbelirtilerek, alma sırasını kasıtlı olarak değiştirmez.

Gönderme uygulaması iletileri bir ileti grubu olarak koyarsa, alma uygulaması MQGET çağrısında GMLOGO seçeneğini belirtiyorsa, iletiler doğru sırayla teslim alma uygulamasına sunulur. İleti gruplarına ilişkin ek bilgi için aşağıdaki başlara bakın:

- MQMD ' deMDMFL alanı
- MQPMO ' da PMLOGO seçeneği
- MQGMO ' da GMLOGO seçeneği



6. **MSGDSC** parametresinin MDFB alanında FBQUIT geribildirim kodu FBQUIT için uygulamalar sınaması. Bu değer bulunursa, uygulama sona erer. Ek bilgi için "MQMD (Message descriptor) on IBM i" sayfa 1092 içinde açıklanan MDFB alanına bakın.
7. HOBJ ile tanıtilan kuyruk OOSAVA seçeneğiyle açıldıysa ve MQGET çağrısından tamamlanma kodu CCOK ya da CCWARN ise, HOBJ kuyruk işleciyle ilişkili bağlam, alınan iletinin bağlamına ayarlanır (GMBRWF ya da GMBRWN seçeneği ayarlanmadıysa, bu durumda bağlam kullanılabilir değil olarak işaretlenir). Bu bağlam, sonraki bir MQPUT ya da MQPUT1 çağrısında, PMPASI ya da PMPASA seçeneklerini belirterek kullanılabilir. Bu işlem, alınan iletinin içeriğinin tamamını ya da bir kısmı başka bir iletiye (örneğin, ileti başka bir kuyruğa iletildiğinde) aktarılmasını sağlar. İleti bağlamına ilişkin daha fazla bilgi için bkz. İleti bağlamı ve Bağlam bilgilerini denetleme.
8. DGMCONV seçeneği **GMO** parametresine eklenirse, uygulama iletisi verileri, veri **BUFFER** parametresine yerleştirilmeden önce, giriş uygulaması tarafından istenen gösteriye dönüştürülür:

- İletideki denetim bilgilerindeki MDFMT alanı, uygulama verilerinin yapısını tanımlar ve iletteki denetim bilgilerindeki MDCSI ve MDENC alanları, karakter kümesi tanıtıcısını ve kodlamasını belirtir.
- MQGET çağrısını yayınlayan uygulama, **MSGDSC** parametresindeki MDCSI ve MDENC alanlarında, uygulama iletisi verilerinin dönüştürülmesi gereken karakter kümesi tanıtıcısını ve kodlamasını belirtir.

İleti verilerinin dönüştürülmesi gerekliyse, dönüştürme işlemi kuyruk yöneticisinin kendisi tarafından ya da kullanıcı tarafından yazılan bir çıkışa göre gerçekleştirilir; bu durumda, iletteki denetim bilgilerindeki MDFMT alanının değerine bağlı olarak değişir:

- Aşağıdaki biçimler kuyruk yöneticisi tarafından otomatik olarak dönüştürülür; bu biçimler "yerleşik" biçimler olarak adlandırılır:

FMADMN	FMMDE
FMCI CS	FMPCF
FM CMD1	FMRMH
FM CMD2	FNRFH
FBIDLH	FMRFH2
FMDH	FMSTR
FMEVNT	FMTM
FMIMS	FMXQH
FMIMVS	

- FMNONE biçim adı, iletteki verilerin niteliğın tanımsız olduğunu gösteren özel bir değerdir. Sonuç olarak, kuyruk yöneticisi ileti kuyruktan alındığında dönüştürme girişiminde bulunmaz.

**Not:** MQNONE biçiminde bir FMNONE biçim adı olan bir ileti için GMCONV belirtilirse ve iletinin karakter takımı ya da kodlaması **MSGDSC** parametresinde belirtilenizden farklıysa, ileti yine **BUFFER** parametresinde döndürülür (başka hata yoktur), ancak çağrı tamamlanma kodu CCWARN ile tamamlanır ve neden kodu RC2110 ile tamamlanır.

FMNONE, ileti verilerinin niteliğı dönüştürme gerektirmediğı anlamına geldiğinde ya da gönderme ve alma uygulamaları, kendileri arasında ileti verilerinin gönderilmesi gereken form arasında anlaşılğında kullanılabilir.

- Diğer tüm biçim adları, iletinin dönüştürmeye ilişkin kullanıcı tarafından yazılmış bir çıkışa iletilmesine neden olur. Çıkış, ortama özgü eklemeler dışında, biçimle aynı adı içerir. Kullanıcı tarafından belirtilen biçim adları, "MQ" harfleriyle başlamamalıdır; bu adlar gelecekte desteklenen biçim adlarıyla çakışabilir.

İletideki kullanıcı verileri, desteklenen karakter kümeleri ve kodlamalar arasında dönüştürülebilir. Ancak, ileti bir ya da daha fazla IBM MQ üstbilgi yapısı içeriyorsa, ileti, kuyruk adlarında geçerli olan karakterler için çift byte ya da çok byte karakterler içeren bir karakter kümesinden ya da bir karakter kümesinden dönüştürülemez. Bu denenirse neden kodu RC2111 ya da RC2115 sonuçlarıyla

sonuçlanır ve ileti dönüştürülmemiş olarak döndürülür. Unicode karakter kümesi UTF-16 , bu tür bir karakter kümesine bir örnektir.

MQGET ' den dönüşte, aşağıdaki neden kodu iletinin başarıyla dönüştürülmüş olduğunu gösterir:

- YOK

The following reason code indicates that the message might have been converted successfully; the application must check the MDCSI and MDENC fields in the **MSGDSC** parameter to find out:

- RC2079

Diğer tüm neden kodları, iletinin dönüştürülmediğini gösterir.

**Not:** Bu örnekte açıklanan neden kodunun yorumlanması, yalnızca çıkışa ilişkin işleme yönergelerine uyması durumunda, kullanıcı tarafından yazılan çıkışlar tarafından gerçekleştirilen dönüştürmeler için geçerlidir.

9. Daha önce listelenen yerleşik biçimler için, kuyruk yöneticisi, GMCONV seçeneği belirtildiğinde, ileti içindeki karakter dizgilerin varsayılan dönüşümünü gerçekleştirebilirler. Varsayılan dönüştürme, kuyruk yöneticisinin dizgi verilerini dönüştürürken gerçek karakter kümesine yaklaşan, kuruluş tarafından belirlenmiş bir varsayılan karakter kümesini kullanmasına olanak sağlar. Sonuç olarak, MQGET çağrısının CCWARN ve neden kodu RC2111 ya da RC2115 ile tamamlanabilmesi yerine, tamamlanma kodu CCOK ile başarılı olabilir.

**Not:** Dizgi verilerini dönüştürmek için yaklaşık bir karakter kümesi kullanılmasının sonucu bazı karakterlerin yanlış dönüştürülebileceğini ifade eder. Bu, yalnızca hem gerçek karakter kümesi, hem de varsayılan karakter kümesi için ortak olan karakterler dizgisinde kullanılmaktan kaçınılabilir.

Varsayılan dönüştürme hem uygulama iletisi verileri hem de MQMD ve MQMDE yapılarındaki karakter alanları için geçerlidir:

- Uygulama iletisi verilerinin varsayılan dönüşümü, yalnızca aşağıdaki deyimlerin tümü doğru olduğunda gerçekleşir:
    - Uygulama, GMCONV değerini belirtir.
    - İleti, desteklenmeyen ya da desteklenmeyen bir karakter kümesinden dönüştürülmesi gereken veriler içeriyor.
    - Varsayılan dönüştürme, kuyruk yöneticisi kurulu ya da yeniden başlatıldığında etkinleştirildi.
  - Kuyruk yöneticisi için varsayılan dönüştürme etkinleştirildiyse, MQMD ve MQMDE yapılarındaki karakter alanlarının varsayılan dönüştürülmesi gerektiği gibi gerçekleşir. Dönüştürme işlemi, MQGET çağrısındaki uygulama tarafından GMCONV seçeneği belirlenirse de gerçekleştirilir.
10. RPG programlama örneğinde gösterilen **BUFFER** parametresi bir dizgi olarak bildirilir; bu parametre, parametrenin uzunluk üst sınırını 256 byte olarak sınırlandırır. Daha büyük bir arabellek gerekiyorsa, parametrenin bir yapı olarak ya da fiziksel bir dosyada bir alan olarak bildirilmesi gerekir.

Değiştirgenin bir yapı olarak bildirilmesi, mümkün olan uzunluk üst sınırını 9999 byte 'a yükseltip, değiştirgeyi fiziksel bir dosyada alan olarak bildirirken, olası uzunluk üst sınırı 32 KB ' ye kadar olabilir.

## Parametreler

MQGET çağrısında şu değiştirgeler vardır:

### HCONN (10 basamaklı işaretli tamsayı)-giriş

Bağlantı tanıtıcısı.

Bu tanıtıcı, kuyruk yöneticisiyle olan bağlantıyı gösterir. The value of HCONN was returned by a previous MQCONN or MQCONNX call.

### HOBJ (10 basamaklı işaretli tamsayı)-giriş

Nesne tanıtıcısı.

Bu tanıtıcı, bir iletinin alınacağı kuyruğu gösterir. The value of HOBJ was returned by a previous MQOPEN call. Kuyruk, aşağıdaki seçeneklerden biriyle ya da daha fazlasıyla açılmış olmalıdır (ayrıntılar için [“MQOPEN \(Open object\) on IBM i” sayfa 1299](#) konusuna bakın):

- OOINPS
- OOINPX
- OOINPQ
- OOBROW

### **MSGDSC (MQMD)-giriş/çıkış**

İleti tanımlayıcısı.

Bu yapı, gerekli iletinin özniteliklerini ve alınan iletinin özniteliklerini açıklamadır. Ayrıntılar için bkz. [“MQMD \(Message descriptor\) on IBM i” sayfa 1092](#).

If BUFLLEN is less than the message length, MSGDSC is still entered by the queue manager, whether GMATM is specified on the **GMO** parameter (see the GMOPT field described in [“IBM üzerinde MQGMO \(get-message options\)” sayfa 1060](#)).

Uygulama bir version-1 MQMD sağlıyorsa, döndürülen iletinin uygulama iletisi verilerine önek olarak bir MQMDE öneki vardır; ancak, MQMDE içindeki alanlardan biri ya da daha fazlasının varsayılan dışı bir değeri varsa, bu ileti bir MQMDE öneki içerir. MQMDE 'deki tüm alanların varsayılan değerleri varsa, MQMDE atlanır. MQMD 'deki MDFMT alanında FMMDE' nin biçim adı, bir MQMDE 'nin var olduğunu gösterir.

### **GMO (MQGMO)-giriş/çıkış**

MQGET işlemini denetleyen seçenekler.

Ayrıntılar için bkz. [“IBM üzerinde MQGMO \(get-message options\)” sayfa 1060](#).

### **BUFLLEN (10 basamaklı işaretli tamsayı)-giriş**

BUFFER alanının bayt cinsinden uzunluğu.

Veri içermeyen iletiler için sıfır ya da ileti kuyruktan kaldırılacaksa ve atılan veri (bu durumda GMATM belirtilmelidir) için de belirtilebilir.

**Not:** Kuyruktan okunmanın mümkün olduğu en uzun iletinin uzunluğu **MaxMsgLength** kuyruk özniteliği tarafından verilir; bkz. [“Kuyruklara ilişkin öznitelikler” sayfa 1345](#).

### **BUFFER (1-byte bit dizgi x BUFLLEN)-çıkış**

İleti verilerini içermek için kullanılan alan.

Arabellek, iletteki verilerin niteine uygun bir sınırla hizalanmalıdır. 4 baytlık hizalama, çoğu ileti için uygun olmalıdır ( IBM MQ üstbilgi yapılarını içeren iletiler de içinde olmak üzere), ancak bazı iletiler daha sıkı hizalamada gerektirebilir. Örneğin, 64 bitlik bir ikili tamsayı içeren bir ileti 8 byte 'lık hizalama gerektirebilir.

If BUFLLEN is less than the message length, as much of the message as possible is moved into BUFFER ; this happens whether GMATM is specified on the **GMO** parameter (see the GMOPT field described in [“IBM üzerinde MQGMO \(get-message options\)” sayfa 1060](#) for more information).

**BUFFER** içindeki verilerin karakter kümesi ve kodlaması, **MSGDSC** parametresine döndürülen MDCSI ve MDENC alanları tarafından verilir. Bu değerler, alıcının gerektirdiği değerlerden farklıysa, alıcı, uygulama iletisi verilerini karakter takımı ve kodlamaya dönüştürmelidir. İleti verilerinin dönüştürülmesini gerçekleştirmek için, kullanıcı tarafından yazılan bir çıkışta GMCONV seçeneği kullanılabilir (bu seçeneğin ayrıntıları için [“IBM üzerinde MQGMO \(get-message options\)” sayfa 1060](#) başlıklı konuya bakın).

**Not:** MQGET çağrısındaki diğer tüm parametreler, yerel kuyruk yöneticisinin karakter kümesinde ve kodlamasında ( **CodedCharSetId** kuyruk yöneticisi özniteliği ve ENNAT tarafından verilir) kodlanır.

Arama başarısız olursa, arabelleğin içeriği değişmeye devam edebilir.

## **DATLEN (10 basamaklı işaretli tamsayı)-çıkış**

İletinin uzunluğu.

Bu, iletteki uygulama verilerinin bayt cinsinden uzunluğudur. Bu ileti uzunluğu BUFLNdeğerinden büyükse, **BUFFER** parametresine yalnızca BUFLN byte 'ları döndürülür (yani, ileti kesilir). Değer sıfırda, iletinin uygulama verisi içermediği anlamına gelir.

If BUFLN is less than the message length, DATLEN is still entered by the queue manager, whether GMATM is specified on the **GMO** parameter (see the GMOPT field described in “IBM üzerinde MQGMO (get-message options)” sayfa 1060 for more information). Bu, uygulamanın ileti verilerini barındırmak için gereken arabelleğin büyüklüğünü belirlemesine olanak tanır ve daha sonra, aramayı uygun boyutta bir arabellekle yeniden yayınlayabilir.

Ancak, GMCONV seçeneği belirtilirse ve dönüştürülen ileti verilerinin BUFFERiçine sığması çok uzun olursa, DATLEN için döndürülen değer şöyledir:

- Dönüştürülemez verilerin uzunluğu, kuyruk yöneticisi tanımlı biçimler için.

In this case, if the nature of the data causes it to expand during conversion, the application must allocate a buffer bigger than the value returned by the queue manager for DATLEN.

- Uygulama tanımlı biçimler için, veri dönüştürme çıkışıyla döndürülen değer.

## **CMPCOD (10 basamaklı işaretli tamsayı)-çıkış**

Tamamlanma kodu.

Bu, aşağıdakilerden biridir:

### **CCOK**

İşlem başarıyla tamamlandı.

### **CCWARN**

Uyarı (kısmi tamamlama).

### **CCCFAIL**

Arama başarısız oldu.

## **REASON (10 basamaklı işaretli tamsayı)-çıkış**

Neden kodu ön eleme CMPCOD.

Aşağıdaki neden kodları, kuyruk yöneticisinin **REASON** parametresi için döndürebileceği kodlardır. Uygulama, GMCONV seçeneğini belirtiyorsa ve ileti verilerinin bir kısmını ya da tümünü dönüştürmek için kullanıcı tarafından yazılan bir çıkış çağırılırsa, **REASON** değıştirgesi için hangi değerin döndürüleceğine karar veren çıkışıdır. Sonuç olarak, bu bölümde daha sonra belgelenen değeler dışındaki değeler de mümkündür.

CMPCOD CCOK ise:

### **YOK**

(0, X'000 ') Raporlamak için bir neden yok.

CMPCOD CCWARN ise:

### **RC2120**

(2120, X'848 ') Dönüştürülen veriler arabellek için çok büyük.

### **RC2190**

(2190, X'88E') Dönüştürülen dize, alan için çok büyük.

### **RC2150**

(2150, X'866 ') DBCS dizgisi geçerli değil.

### **RC2110**

(2110, X'83E') İleti biçimi geçerli değil.

### **RC2243**

(2243, X'8C3') İleti kesimleri farklı CCSID' lere sahip.

**RC2244**

(2244, X'8C4') İleti kesimleri farklı kodlamalara sahiptir.

**RC2209**

(2209, X'8A1') İleti kilitlemedi.

**RC2119**

(2119, X'847 ') İleti verileri dönüştürülmedi.

**RC2272**

(2272, X'8E0') İleti verileri kısmen dönüştürüldü.

**RC2145**

(2145, X'861 ') Kaynak arabellek parametresi geçerli değil.

**RC2111**

(2111, X'83F') Kaynak kodlanmış karakter takımı tanıtıcısı geçerli değil.

**RC2113**

(2113, X'841 ') İletide paketli ondalık kodlama tanınmadı.

**RC2114**

(2114, X'842 ') İletide kayan noktalı kodlama tanınmadı.

**RC2112**

(2112, X'840 ') Kaynak tamsayı kodlaması tanınmadı.

**RC2143**

(2143, X'85F') Kaynak uzunluğu parametresi geçerli değil.

**RC2146**

(2146, X'862 ') Hedef arabellek parametresi geçerli değil.

**RC2115**

(2115, X'843 ') Hedef kodlanmış karakter takımı tanıtıcısı geçerli değil.

**RC2117**

(2117, X'845 ') Packed-receiver tarafından belirtilen paket-ondalık kodlama tanınmıyor.

**RC2118**

(2118, X'846 ') Alıcı tarafından belirtilen kayan noktalı kodlama tanınmıyor.

**RC2116**

(2116, X'844 ') Hedef tamsayı kodlaması tanınmadı.

**RC2079**

(2079, X'81F') Kesilen ileti döndürüldü (işleme tamamlandı).

**RC2080**

(2080, X'820 ') Kesilen ileti döndürüldü (işleme tamamlanmadı).

CMPCOD CCFAIL ise:

**RC2004**

(2004, X'7D4') Arabellek parametresi geçerli değil.

**RC2005**

(2005, X'7D5') Arabellek uzunluğu parametresi geçerli değil.

**RC2219**

(2219, X'8AB') MQI çağrısı, önceki arama tamamlanmadan önce yeniden girildi.

**RC2009**

(2009, X'7D9') Kuyruk yöneticisine yönelik bağlantı kaybedildi.

**RC2010**

(2010, X'7DA') Veri uzunluğu parametresi geçerli değil.

**RC2016**

(2016, X'7E0') Kuyruğun engellenmesini sağlar.

**RC2186**

(2186, X'88A') Al-ileti seçenekleri yapısı geçerli değil.

- RC2018**  
(2018, X'7E2') Bağlantı tanıtıcısı geçerli değil.
- RC2019**  
(2019, X'7E3') Nesne tanıtıcısı geçerli değil.
- RC2241**  
(2241, X'8C1') İleti grubu tamamlanmadı.
- RC2242**  
(2242, X'8C2') Mantıksal ileti tamamlanmadı.
- RC2259**  
(2259, X'8D3') Tutarsız göz atma belirtimi.
- RC2245**  
(2244, X'8C5') Tutarsız iş birliği-iş belirtimi.
- RC2246**  
(2246, X'8C6') İmleç altındaki ileti alma için geçerli değil.
- RC2247**  
(2247, X'8C7') Eşleştirme seçenekleri geçerli değil.
- RC2026**  
(2026, X'7EA') İleti tanımlayıcısı geçerli değil.
- RC2250**  
(2250, X'8CA') İleti sıra numarası geçerli değil.
- RC2033**  
(2033, X'7F1') İleti yok.
- RC2034**  
(2034, X'7F2') Browse imleci iletide konumlandırılmamış.
- RC2036**  
(2036, X'7F4') Kuyruk göz atma için açık değil.
- RC2037**  
(2037, X'7F5') Kuyruk giriş için açık değil.
- RC2041**  
(2041, X'7F9') Nesne tanımlaması açıldığından beri değiştirildi.
- RC2101**  
(2101, X'835 ') Nesne zarar gördü.
- RC2046**  
(2046, X'7FE') Seçenekler geçerli değil ya da tutarlı değil.
- RC2052**  
(2052, X'804 ') Kuyruk silinmiştir.
- RC2058**  
(2058, X'80A') Kuyruk yöneticisi adı geçerli değil ya da bilinmiyor.
- RC2059**  
(2059, X'80B') Kuyruk yöneticisi bağlantı için kullanılabilir değil.
- RC2161**  
(2161, X'871 ') Kuyruk yöneticisi susturuyor.
- RC2162**  
(2162, X'872 ') Kuyruk yöneticisi sona erdiriliyor.
- RC2102**  
(2102, X'836 ') Yetersiz sistem kaynakları var.
- RC2071**  
(2071, X'817 ') Kullanılabilir bellek yetersiz.
- RC2024**  
(2024, X'7E8') Yürürlükteki iş birimi içinde başka ileti işlenemez.

**RC2072**

(2072, X'818 ') Syncpoint desteği yok.

**RC2195**

(2195, X'893 ') Beklenmeyen bir hata oluştu.

**RC2255**

(2255, X'8CF') Kuyruk yöneticisinin kullanması için kullanılabilir iş birimi.

**RC2090**

(2090, X'82A') MQGMO' da bekleme aralığı geçerli değil.

**RC2256**

(2256, X'8D0') MQGMO' nun yanlış sürümü sağlandı.

**RC2257**

(2257, X'8D1') MQMD' nin yanlış sürümü sağlandı.

**RPG Bildirimi**

```

C*..1.....2.....3.....4.....5.....6.....7..
C          CALLP      MQGET(HCONN : HOBJ : MSGDSC : GMO :
C                               BUFLN : BUFFER : DATLEN :
C                               CMPCOD : REASON)

```

Çağrıya ilişkin prototip tanımı:

```

D*..1.....2.....3.....4.....5.....6.....7..
DMQGET          PR          EXTPROC('MQGET')
D* Connection handle
D HCONN          10I 0 VALUE
D* Object handle
D HOBJ          10I 0 VALUE
D* Message descriptor
D MSGDSC          364A
D* Options that control the action of MQGET
D GMO          112A
D* Length in bytes of the Buffer area
D BUFLN          10I 0 VALUE
D* Area to contain the message data
D BUFFER          * VALUE
D* Length of the message
D DATLEN          10I 0
D* Completion code
D CMPCOD          10I 0
D* Reason code qualifying CMPCOD
D REASON          10I 0

```

**IBM i MQINQ (Inquire about object attributes) on IBM i**

MQINQ çağrısı, bir tamsayılar dizisi ve bir nesnenin özniteliklerini içeren bir karakter dizileri kümesini döndürür.

Aşağıdaki nesne tipleri geçerlidir:

- Kuyruk
- Ad Listesi
- Süreç tanımlaması
- Kuyruk yöneticisi
- [“Sözdizimi” sayfa 1284](#)
- [“Kullanım notları” sayfa 1284](#)
- [“Parametreler” sayfa 1285](#)
- [“RPG Bildirimi” sayfa 1291](#)

## Sözdizimi

MQINQ (HCONN, HOBJ, SELCNT, SELS, IACNT, INTATR, CALEN, CHRATR, CMPCOD, REASON)

### Kullanım notları

1. Döndürülen değerler, seçilen özniteliklerin bir anlık görünüşüdür. Uygulamanın döndürülen değerler üzerinde işlem yapabilmesi için, özniteliklerin değiştirilmemesi konusunda herhangi bir garanti verilmez.
2. Bir model kuyruğunu açtığınızda, dinamik bir yerel kuyruk yaratılır. Bu, özniteliklerin sorgulamak için model kuyruğunu açsanız bile geçerlidir.  
  
Dinamik kuyruğun öznitelikleri (belirli kural dışı durumlarla), dinamik kuyruk yaratıldığı sırada, model kuyruklarıyla aynı olur. Bu kuyruktan sonra MQINQ çağrısını kullanırsanız, kuyruk yöneticisi, model kuyruğundan değil, dinamik kuyruğun özniteliklerini döndürür. Model kuyruğunun özniteliklerinin dinamik kuyruk tarafından devralındığı ayrıntılar için [Çizelge 1](#) ' e bakın.
3. Sorgulanan nesne bir diğer ad kuyruğuna, MQINQ çağrısının döndürdüğü öznitelik değerleri diğer ad kuyruğunun ve diğer adın çözümleneceği temel kuyruklardır.
4. Sorgulanmakta olan nesne bir küme kuyruğusa, sorgulanabilen öznitelikler, kuyruğun nasıl açıldığı ile ilgili olarak değişir:
  - Küme kuyruğu sorgulamak için açılırsa, bir ya da daha çok giriş, göz atma ya da küme için, kümenin başarılı olması için küme kuyruğunun yerel bir eşgörünümü olması gerekir. Bu durumda, sorgulanabilen öznitelikler, yerel kuyruklar için geçerli olan özniteliklerdir.
  - Tek başına ya da sorgulamak için küme kuyruğu açılırsa, yalnızca aşağıdaki öznitelikler sorgulanabilir; **QType** özniteliği bu durumda QTCLUS değerini içerir:
    - CAQD
    - CAQN
    - IADBND
    - IADPER
    - IADPRI
    - IAIPUT
    - IAQTYP

Küme kuyruğu değişmez bir bağ tanımı (MQOPEN çağrısında belirtilen OOBNDN ya da **DefBind** özniteliği BNDNOT değerine sahip olduğunda OOBNDQ) ile açılırsa, kuyruk için art arda gelen MQINQ çağrıları, küme kuyruğunun farklı eşgörünümlerini sorgulayabilir; ancak, tipik olarak tüm eşgörünümler aynı öznitelik değerlerine sahiptir.

Küme kuyruklarıyla ilgili ek bilgi için [Kuyruk yöneticisi kümesinin yapılandırılması](#) başlıklı konuya bakın.
5. Bir dizi öznitelik sorgulanacaksa ve bunların bazıları MQSET çağrısı kullanılarak ayarlanacaksa, seçicinin başında konumlanmak için ayarlanacak öznitelikler, bu nedenle MQSET için aynı dizilerin (azaltılmış sayılarla) kullanılabilceği şekilde ayarlanabilir.
6. Uyarı durumlarının birden fazlası ortaya çıkarsa ( **CMPCOD** parametresine bakın), döndürülen neden kodu, aşağıdaki listede yer alan *ilk* bir tanedir:
  - a. RC2068
  - b. RC2022
  - c. RC2008
7. Nesne özniteliklerine ilişkin ek bilgi için aşağıdaki başlıklara bakın:
  - [“Kuyruklara ilişkin öznitelikler” sayfa 1345](#)
  - [“Ad listelerine ilişkin öznitelikler” sayfa 1373](#)
  - [“IBM üzerindeki süreç tanımlamalarına ilişkin öznitelikler” sayfa 1374](#)



- “IBM üzerinde kuyruk yöneticisine ilişkin öznitelikler” sayfa 1376

8. Yeni bir yerel kuyruk SYSTEM.ADMIN.COMMAND.EVENT , komutlar ne zaman yayınlandığında oluşturulan kuyruğa alma iletileri için kullanılır. CMDEV kuyruk yöneticisi özniteliğinin nasıl ayarlansa bağlı olarak, iletiler çoğu komut için bu kuyruğa yerleştirilir:

- ENABLED-komut olay iletileri oluşturulur ve tüm başarılı komutlar için kuyruğa konlanır.
- NODISPLAY-komut olay iletileri oluşturulur ve DISABLE (MQSC) komutu ve Sorgula (PCF) komutu dışındaki tüm başarılı komutlar için kuyruğa konlanır.
- DISABLE-komut olay iletileri oluşturulmaz (kuyruk yöneticisinin ilk varsayılan değeri budur).

## Parametreler

MQINQ çağrısı aşağıdaki değişirgeleri içerir:

### HCONN (10 basamaklı işaretli tamsayı)-giriş

Bağlantı tanıtıcısı.

Bu tanıtıcı, kuyruk yöneticisiyle olan bağlantıyı gösterir. The value of *HCONN* was returned by a previous MQCONN or MQCONNX call.

### HOBJ (10 basamaklı işaretli tamsayı)-giriş

Nesne tanıtıcısı.

Bu tanıtıcı, gerekli özniteliklere sahip nesneyi (herhangi bir tipte) gösterir. Tanıtıcı değeri OOINQ seçeneğini belirten önceki bir MQOPER çağrısı tarafından döndürülmelidir.

### SELCNT (10 basamaklı işaretli tamsayı)-giriş

Seçicilerin sayısı.

Bu, *SELS* dizisinde sağlanan seçicilerin sayısıdır. Döndürülemekte olan özniteliklerin sayısıdır. Sıfır geçerli bir değerdir. İzin verilen sayı üst sınırı 256 'tır.

### SOLS (10 basamaklı işaretli tamsayı x SELCNT)-giriş

Öznitelik seçicilerinin dizisi.

Bu, **SELCNT** öznitelik seçicilerinden oluşan bir dizidir; her seçici, bir özniteliği (tamsayı ya da karakter) zorunlu bir değerle tanımlar.

Her seçici, *HOBJ* ' in temsil ettiği nesne tipi için geçerli olmalıdır; aksi halde çağrı, tamamlanma kodu CCFAIL ve neden kodu RC2067 ile başarısız olur.

Özel kuyruklar halinde:

- Seçici, *any* tipli kuyruklar için geçerli değilse, çağrı tamamlanma kodu CCFAIL ve neden kodu RC2067 ile başarısız olur.
- Seçici yalnızca, nesnenin tipi ya da tipi kuyruklar için geçerliyse, çağrı tamamlanma kodu CCWARN ve neden kodu RC2068 ile başarılı olur.
- Sorgulanmakta olan kuyruk bir küme kuyruğıysa, geçerli olan seçiciler, kuyruğun nasıl çözümlendiğine bağlıdır; ek ayrıntılar için bkz. kullanım notu 4.

Seçiciler herhangi bir sırada belirtilebilir. Attribute values that correspond to integer attribute selectors (IA\* selectors) are returned in *INTATR* in the same order in which these selectors occur in *SELS*. Attribute values that correspond to character attribute selectors (CA\* selectors) are returned in *CHRATR* in the same order in which those selectors occur. IA\* seçicileri, CA\* seçicileriyle birlikte ayrılabilir; yalnızca her tip içindeki görelî sipariş önemlidir.

### Not:

1. Tamsayı ve karakter özniteliği seçicileri farklı iki aralık içinde ayrılır; IA\* seçicileri, IACAST ile IACAST ile CALCAST ile CA\* seçicileri CALST ile birlikte CA\* seçicileri aralığına ayrılır.

Her bir aralık için IALSTU ve CALSTU değışmezleri, kuyruk yöneticisinin kabul ettiđi en yüksek değeri tanımlar.

2. Tüm IA\* seçicileri önce gerçekleşirse, aynı öge numaraları SELS ve INTATR dizilerinde karşılık gelen ögeleri adreslemek için kullanılabilir.

Sorgulanabilen öznitelikler aşğıdaki tablolarda listelenir. For the CA\* selectors, the constant that defines the length in bytes of the resulting string in CHRATR is given in parentheses.

Çizelge 747. Kuyruklar için MQINQ öznitelik seçicileri		
Seçici	Tanım	Not
CAALTD	En son değıştirilme tarihi (LNDATE).	1
CATTT	En son değıştirilme zamanı (LNTIME).	1
CABRQN	Aşırı yedek-istek adı (LNQN).	5
CABASQ	Diđer adın çözdüğü kuyruđun adı (LNQN).	
CACFSN	Bađlaşım-tesis yapısı adı (LNCFSN).	3
KILLER	Küme adı (LNCLUN).	1
CACLNL	Küme adılistesi (LNNLN).	1
CACRTD	Kuyruk yaratma tarihi (LNCRTD).	
CACRTT	Kuyruk yaratma süresi (LNCRTT).	
CAINIQ	Başlatma kuyruđu adı (LNQN).	
CAPRON	Süreç tanımlamasının adı (LNPRON).	
CAQD	Kuyruk tanımlaması (LNQD).	
CAQN	Kuyruk adı (LNQN).	
CARQMN	Uzak kuyruk yöneticisinin adı (LNQMN).	
CARQN	Uzak kuyruk yöneticisinde (LNQN) bilindiđi gibi uzak kuyruđun adı.	
CATRGD	Tetikleme verileri (LNTRGD).	5
CAXQN	İletim kuyruđu adı (LNQN).	
IABTHR	Geriletme eşıđi.	5
IACDEP	Kuyruktaki ileti sayısı.	
IADBND	Varsayılan bađ tanımı.	1
IADINP	Giriş için varsayılan açma seçeneđi.	5
IADPER	Varsayılan ileti kalıcılıđı.	
IADPRI	Varsayılan ileti önceliđi.	5
KALDIRIM	Kuyruk tanımlaması tipi.	
IADIST	Dađıtım listesi desteđi.	2
IAHGB	-Geri sayım mı?-Evet.	5
IAIGET	Alma işlemlerine izin verilip verilmediđi.	
IAIPUT	Koyma işlemlerine izin verilip verilmediđi.	
IAMLEN	İleti uzunluđu üst sınırı.	
IAMDEP	Kuyruktan izin verilen ileti sayısı üst sınırı.	

Çizelge 747. Kuyruklar için MQINQ öznitelik seçicileri (devamı var)		
Seçici	Tanım	Not
IAMDS	İleti önceliğinin ilgili olup olmadığı.	5
IAOIC	Kuyruğun giriş için açık olan MQOPER çağrılarının sayısı.	
IAOOC	Çıkışa ilişkin kuyruğu açık olan MQOPER çağrılarının sayısı.	
IAQDHE	Kuyruk derinliği yüksek olayları için denetim özniteliği.	4, 5
IAQDHL	Kuyruk derinliği üst sınırı.	4, 5
IAQDLE	Kuyruk derinliği düşük olaylarına ilişkin denetim özniteliği.	4, 5
IAQDLL	Kuyruk derinliği için alt sınır.	4, 5
IAQDME	Kuyruk derinliği olayları üst sınırı için denetim özniteliği.	4, 5
IAQSI	Kuyruk hizmeti aralığı sınırı.	4, 5
IAQSIE	Kuyruk hizmeti aralığı olayları için denetim özniteliği.	4, 5
IAQTYP	Kuyruk tipi.	
IAQSGD	Kuyruk paylaşımı grubu yok etme.	3
İTIRAZ	Kuyruk alıkoyma aralığı.	5
IASCOP	Kuyruk tanımlaması kapsamı.	4, 5
IASHAR	Kuyruğun giriş için paylaşılıp paylaşılmayacağı.	
IATRGC	Tetik kontrolü.	
ALRIGI	Tetik derinliği.	5
IATRGP	Tetikleyiciler için eşik iletisi önceliği.	5
İLAÇ	Tetikleyici tipi.	
IAUSAG	Kullanım.	
CLWLUSEQ	Uzak kuyrukları kullanın.	

**Not:**

1. Aşağıdaki altyapılarda desteklenir:

-  AIX
-  IBM i
-  Windows
-  z/OS

ve bu sistemlere bağlı IBM MQ MQI clients için.

2. Aşağıdaki altyapılarda desteklenir:

-  AIX
-  IBM i
-  Windows

ve bu sistemlere bağlı IBM MQ istemcileri için geçerli olur.

3. z/OS z/OS üzerinde desteklenir.
4. z/OS z/OS üzerinde desteklenmez.
5. VSE/ESA üzerinde desteklenmez.

<i>Çizelge 748. Ad listeleri için MQINQ öznelik seçicileri</i>		
<b>Seçici</b>	<b>Tanım</b>	<b>Not</b>
CAALTD	En son değiştirilme tarihi (LNDATE)	1
CATTT	En son değiştirilme zamanı (LNTIME)	1
CALSTD	Ad listesi tanımlaması (LNNLD)	1
CALSTN	Ad listesi nesnesinin adı (LNNLN)	1
CANAMS	Namelist 'teki adlar (LNQN x <i>Listedeki adların sayısı</i> )	1
IANAMC	Namelist 'teki ad sayısı	1
IAQSGD	Kuyruk paylaşımı grubu yok etme	3

<i>Çizelge 749. Süreç tanımlamaları için MQINQ öznelik seçicileri</i>		
<b>Seçici</b>	<b>Tanım</b>	<b>Not</b>
CAALTD	En son değiştirilme tarihi (LNDATE)	1
CATTT	En son değiştirilme zamanı (LNTIME)	1
HAZGEÇ	Uygulama tanıtıcısı (LNPROA)	5
CAENVD	Ortam verileri (LNPROE)	5
CAPROD	Süreç tanımlamasının tanımı (LNPROD)	5
CAPRON	Süreç tanımlamasının adı (LNPRON)	5
CAUSRD	Kullanıcı verileri (LNPROU)	5
YAKLAYICI	Uygulama tipi	5
IAQSGD	Kuyruk paylaşımı grubu yok etme	3

<i>Çizelge 750. Kuyruk yöneticisine ilişkin MQINQ öznelik seçicileri</i>		
<b>Seçici</b>	<b>Tanım</b>	<b>Not</b>
CAALTD	En son değiştirilme tarihi (LNDATE)	1
CATTT	En son değiştirilme zamanı (LNTIME)	1
CACADX	Otomatik kanal tanımlama çıkış adı (LNEXN)	1
KALICI	Küme iş yükü çıkışa aktarılan veriler (LNEXDA)	1
CLAKWX	Küme iş yükü çıkışa adı (LNEXN)	1
CACMDQ	Sistem komutu giriş kuyruğu adı (LNQN)	5
CADLQ	Ölü-mektup kuyruğunun adı (LNQN)	5
CADXQN	Varsayılan iletim kuyruğu adı (LNQN)	5
CAQMD	Kuyruk yöneticisi tanımlaması (LNQMD)	5
CAQMID	Kuyruk yöneticisi tanıtıcısı (LNQMID)	1

Çizelge 750. Kuyruk yöneticisine ilişkin MQINQ öznitelik seçicileri (devamı var)		
Seçici	Tanım	Not
CAQMN	Yerel kuyruk yöneticisinin adı (LNQMN)	5
CAQSGN	Kuyruk paylaşım grubu adı (LNQSGN)	3
CARPN	Kuyruk yöneticisinin havuz hizmetleri sağladığı kümenin adı (LNQMN)	1
CARPNL	Kuyruk yöneticisinin havuz hizmetleri sağladığı kümelerin adlarını içeren ad listesi nesnesinin adı (LNNLN)	1
CMDEV	Komutlar verildiğinde oluşturulan iletilerin bir kuyruğa konulup sunulmayacağını belirleyen denetim özniteliği	8
AAAA	Yetki olaylarına ilişkin denetim özniteliği	4, 5
IACAD	Otomatik kanal tanımlamasına ilişkin denetim özniteliği	2
BAŞLATMA	Otomatik kanal tanımlaması olaylarına ilişkin denetim özniteliği	2
IACLXQ	Varsayılan küme iletim kuyruğu tipi	4
IACLWL	Küme iş yükü uzunluğu	1
IACCSI	Kodlanmış karakter takımı tanıtıcısı	5
IACMDL	Kuyruk yöneticisi tarafından desteklenen komut düzeyi	5
IACFGE	Yapılandırma olaylarına ilişkin denetim özniteliği	3
IADIST	Dağıtım listesi desteği	2
IAINHE	Olayları engelleyici için denetim özniteliği	4, 5
IACLE	Yerel olaylara ilişkin denetim özniteliği	4, 5
IAMHND	Tanıtıcı sayısı üst sınırı	5
IAMLEN	İleti uzunluğu üst sınırı	5
IAMPRI	En yüksek öncelik	5
IAMUNC	Bir iş birimi içinde kesinleştirilmemiş ileti sayısı üst sınırı	5
IAPFME	Performans olayları için denetim özniteliği	4, 5
IAPLAT	Kuyruk yöneticisinin bulunduğu altyapı	5
IARMTE	Uzak olaylar için denetim özniteliği	4, 5
İASK	Durdurma olaylarının başlatılması için denetim özniteliği	4, 5
IASYNC	Eşitleme noktası kullanılabilirliği	5
IATRLFT	Kullanılmayan yönetimle ilgili olmayan konuların geçerlilik süresi	
IATRGI	Tetikleme aralığı	5

### IACNT (10 basamaklı işaretli tamsayı)-giriş

Tamsayı özniteliklerinin sayısı.

Bu, INTATR dizideki öğelerin sayısıdır. Sıfır geçerli bir değerdir.

Bu, **SELS** parametresindeki en az IA\* seçiciyse, istenen tüm tamsayı özniteliklerinin döndürülmesi gerekir.

## **INTATR (10 basamaklı işaretli tamsayı x IACNT)-çıkış**

Tamsayı öznitelikleri dizisi.

Bu, *IACNT* tamsayı öznitelik değerlerinden oluşan bir dizidir.

Tamsayı öznitelik değerleri, **SELS** parametresindeki IA\* seçicilikleriyle aynı sırayla döndürülür. Dizi IA\* seçicilerinin sayısından daha fazla öge içeriyorsa, fazla ögeler değişmeden kalır.

HOBJ bir kuyruğu temsil ediyorsa, ancak bir öznitelik seçici o kuyruk tipi için geçerli değilse, INTATR dizisinde karşılık gelen öge için IAVNA özel değeri döndürülür.

## **CALEN (10 basamaklı işaretli tamsayı)-giriş**

Karakter öznitelikleri arabelleğindeki uzunluk.

Bu, **CHRATR** parametresindeki bayt cinsinden uzunluğudur.

Bu, istenen karakter özniteliklerinin uzunluklarının en az toplamını olmalıdır (bkz. SELS). Sıfır geçerli bir değerdir.

## **CHRATR (1 baytlık karakter dizilimi x CALEN)-çıkış**

Karakter öznitelikleri.

Bu arabellek, karakter özniteliklerinin birlikte döndürüldüğü, birleştirilen arabelleğidir. Arabellek uzunluğu **CALEN** parametresiyle verilir.

Karakter öznitelikleri, **SELS** parametresindeki CA\* seçicilikleriyle aynı sırayla döndürülür. Her bir öznitelik için her bir öznitelik dizgisinin uzunluğu sabittir (bkz. SELS) ve gerekirse boşluktaki değer boşluklarla doldurulmuştur. Arabellek, istenen tüm karakter özniteliklerinin (doldurma da içinde olmak üzere) tümünü içermek için gerekenden büyükse, döndürülen son öznitelik değerinin ötesindeki bayt değişmez.

HOBJ bir kuyruğu temsil ediyorsa, ancak bir öznitelik seçici o kuyruk tipi için geçerli değilse, tam olarak yıldız işaretleri (\*) içeren bir karakter dizgisi, CHRATR' taki o özniteliğin değeri olarak döndürülür.

## **CMPCOD (10 basamaklı işaretli tamsayı)-çıkış**

Tamamlanma kodu.

Bu, aşağıdakilerden biridir:

### **CCOK**

İşlem başarıyla tamamlandı.

### **CCWARN**

Uyarı (kısmi tamamlama).

### **CCCFAIL**

Arama başarısız oldu.

## **REASON (10 basamaklı işaretli tamsayı)-çıkış**

Neden kodu ön eleme CMPCOD.

CMPCOD CCOK ise:

### **YOK**

(0, X'000 ') Raporlamak için bir neden yok.

CMPCOD CCWARN ise:

### **RC2008**

(2008, X'7D8') Karakter öznitelikleri için yeterli alan kullanılmasına izin verilmez.

### **RC2022**

(2022, X'7E6') Tamsayı öznitelikleri için yeterli alan kullanılmasına izin verilmez.

### **RC2068**

(2068, X'814 ') Seçici kuyruk tipi için geçerli değil.

CMPCOD CCFAIL ise:

**RC2219**

(2219, X'8AB') MQI çağrısı, önceki arama tamamlanmadan önce yeniden girildi.

**RC2006**

(2006, X'7D6') Karakter öznitelikleri uzunluğu geçerli değil.

**RC2007**

(2007, X'7D7') Karakter öznitelikleri dizgisi geçerli değil.

**RC2009**

(2009, X'7D9') Kuyruk yöneticisine yönelik bağlantı kaybedildi.

**RC2018**

(2018, X'7E2') Bağlantı tanıtıcısı geçerli değil.

**RC2019**

(2019, X'7E3') Nesne tanıtıcısı geçerli değil.

**RC2021**

(2021, X'7E5') Tamsayı öznitelikleri sayısı geçerli değil.

**RC2023**

(2023, X'7E7') Tamsayı öznitelikleri dizisi geçerli değil.

**RC2038**

(2038, X'7F6') Sorgulamak için kuyruk açık değil.

**RC2041**

(2041, X'7F9') Nesne tanımlaması açıldığından beri değiştirildi.

**RC2101**

(2101, X'835 ') Nesne zarar gördü.

**RC2052**

(2052, X'804 ') Kuyruk silinmiştir.

**RC2058**

(2058, X'80A') Kuyruk yöneticisi adı geçerli değil ya da bilinmiyor.

**RC2059**

(2059, X'80B') Kuyruk yöneticisi bağlantı için kullanılabilir değil.

**RC2162**

(2162, X'872 ') Kuyruk yöneticisi sona erdiriliyor.

**RC2102**

(2102, X'836 ') Yetersiz sistem kaynakları var.

**RC2065**

(2065, X'811 ') Seçici sayısı geçerli değil.

**RC2067**

(2067, X'813 ') Öznitelik seçicisi geçerli değil.

**RC2066**

(2066, X'812 ') Seçicilerin sayısı çok büyük.

**RC2071**

(2071, X'817 ') Kullanılabilir bellek yetersiz.

**RC2195**

(2195, X'893 ') Beklenmeyen bir hata oluştu.

## RPG Bildirimi

```
C*..1.....2.....3.....4.....5.....6.....7..
C                               CALLP    MQINQ(HCONN : HOBJ : SELCNT :
C                               SELS(1) : IACNT : INTATR(1) :
```

C  
C

CALEN : CHRATR : CMPCOD :  
REASON)

Çağrıya ilişkin prototip tanımı:

```
D* .1.....2.....3.....4.....5.....6.....7..  
DMQINQ          PR          EXTPROC('MQINQ')  
D* Connection handle  
D HCONN          10I 0 VALUE  
D* Object handle  
D HOBJ          10I 0 VALUE  
D* Count of selectors  
D SELCNT        10I 0 VALUE  
D* Array of attribute selectors  
D SELS          10I 0  
D* Count of integer attributes  
D IACNT         10I 0 VALUE  
D* Array of integer attributes  
D INTATR        10I 0  
D* Length of character attributes buffer  
D CALEN         10I 0 VALUE  
D* Character attributes  
D CHRATR        *   VALUE  
D* Completion code  
D CMPCOD        10I 0  
D* Reason code qualifying CMPCOD  
D REASON        10I 0
```

IBM i

## MQINQMP (Inquire message property) on IBM i

MQINQMP çağrısı, bir iletinin özellik değerini döndürür.

- [“Sözdizimi” sayfa 1292](#)
- [“Parametreler” sayfa 1292](#)
- [“RPG Bildirimi” sayfa 1296](#)

### Sözdizimi

MQINQMP (*Hconn*, *Hmsg*, *InqPropOpts*, *Name*, *PropDesc*, *Type*, *ValueLength*, *Value*, *DataLength*, *CompCode*, *Reason*)

### Parametreler

MQINQMP çağrısında aşağıdaki değişirgeler vardır:

#### HCONN (10 basamaklı işaretli tamsayı)-giriş

Bu tanıttıcı, kuyruk yöneticisiyle olan bağlantıyı gösterir. *Hconn* değeri, **Hmsg** değiştirgesinde belirtilen ileti tanıttıcısını yaratmak için kullanılan bağlantı tanıttıcısı ile eşleşmelidir.

İleti tanıttıcısı HCUNAS kullanılarak yaratıldıysa, ileti tutamacındaki bir özelliği geciktiren iş parçacığıda geçerli bir bağlantı oluşturulmalıdır; tersi durumda, arama RC2009ile başarısız olur.

#### HMSG (20 basamaklı işaretli tamsayı)-giriş

Bu, sorgulanacak ileti saptanıdır. Değer önceki bir **MQCRTMH** çağrısı tarafından döndürüldü.

#### INQOPT (MQIMPO)-giriş

Ayrıntılar için [MQIMPO](#) veri tipine bakın.

#### PRNAME (MQCHARV)-giriş

Bu, sorgulanacak özelliğin adını belirtir.

If no property with this name can be found, the call fails with reason RC2471.

Özellik adının sonunda yüzde işareti (%) karakterini kullanabilirsiniz. Genel arama karakteri, nokta (.) karakteri de içinde olmak üzere, sıfır ya da daha fazla karakterle eşleşir. Bu, bir uygulamanın



birçok özelliğin değerini sorgulamasına olanak sağlar. İlk eşleşen özelliği almak için, MQINQMP ' yi IPINQF seçeneğiyle çağırın ve bir sonraki eşleştirme özelliğini almak için IPINQN seçeneğiyle yeniden çağırın. Daha fazla eşleşen özellik yoksa, arama RC2471 ile başarısız olur. InqPropOpts yapısının *ReturnedName* alanı, özelliğin döndürülen adı için bir adres ya da görelî konum ile kullanıma hazırlandıysa, bu işlem, eşleştirilen özelliğin adıyla MQINQMP ' den geri dönerek tamamlanır. If the *VSBufSize* field of the *ReturnedName* in the InqPropOpts structure is less than the length of the returned property name the completion code is set CCFAIL with reason RC2465.

Bilinen eşanlamlıları olan özellikler aşağıdaki gibi döndürülür:

1. "mqps." öneğine sahip özellikler IBM MQ özellik adı ile döndürülür. Örneğin, "MQTopicString", "mqps.Top" yerine döndürülen addır.
2. "jms." öneğine sahip özellikler ya da "mcd." JMS üstbilgi alanı adı olarak döndürülür. Örneğin, "JMSEExpiration", "jms.Exp" yerine döndürülen addır.
3. "usr." öneğine sahip özellikler bu önek olmadan döndürülür. Örneğin, "usr.Color" yerine "Renk" döndürülür.

Eşanlamlıları olan özellikler yalnızca bir kez döndürülür.

RPG programlama dilinde, aşağıdaki makro değişkenleri "usr" başlayan tüm özellikleri ve tüm özellikleri sorgulamak için tanımlanır:

#### **GIRIŞ**

İletinin tüm özelliklerini sorgulayın.

#### **INQUSR**

"usr." başlayan iletinin tüm özellikleri ile ilgili olarak bilgi edinin. Döndürülen ad "usr" olmadan döndürülür. önek.

IPINQN belirtilirse, ancak önceki çağrıdan sonra ad değiştiyse ya da ilk çağrı bu, IPINQF ise örtük olarak belirtilir.

Özellik adlarının kullanımına ilişkin ek bilgi için [Özellik adları](#) ve [Özellik adı kısıtlamaları](#) başlıklı konuya bakın.

#### **PRPDSC (MQPD)-çıkış**

Bu yapı, özellik desteklenmiyorsa neler olacağı, özelliğin ait olduğu ileti bağlamı ve özelliğin hangi iletiye kopyalanması gerektiği de dahil olmak üzere, bir özelliğin özniteliklerini tanımlamak için kullanılır. Bu yapıya ilişkin ayrıntılar için [MQPD](#) başlıklı konuya bakın.

#### **TYPE (10 basamaklı işaretli tamsayı)-giriş/çıkış**

Bu değiştirgeyi MQINQMP çağrısından dönüşte, *Değer*veri tipine ayarlanır. Veri tipi aşağıdakilerden biri olabilir:

#### **TYPEYÜKLEME**

Bir boole.

#### **TYPBST**

Bayt dizilimi.

#### **TYPI8**

8 bitlik işaretli tamsayı.

#### **TYPI16**

16 bitlik bir işaretli tamsayı.

#### **TYPI32**

32 bitlik işaretli tamsayı.

#### **TYPI64**

Bir 64 bit imzalı tamsayı.

#### **TYPF32**

32 bitlik kayan noktalı sayı.

**TYPF64**

64 bitlik kayan noktalı sayı.

**TYPSTR**

Bir karakter dizgisi.

**TYPNUL**

Özellik var, ancak boş değere sahip.

Özellik değerinin veri tipi tanınmadıysa, TYPSTR döndürülür ve değer dizgi gösterimi *Değer* alanına yerleştirilir. Veri tipinin dizgi gösterimi, *IPOPT* parametresinin *IPTYP* alanında bulunabilir. Uyarı tamamlanma kodu, RC2467neden ile döndürülür.

Ek olarak, IPCTYP seçeneği belirtilirse, özellik değerinin dönüştürülmesi istenir. Özelliğin döndürülmesini istediğiniz veri tipini belirtmek için, giriş olarak *Type* (Tip) kullanın. Veri tipi dönüştürmesine ilişkin ayrıntılar için "[IBM üzerinde MQIMPO \(Sorgula ileti özelliği seçenekleri\)](#)" sayfa 1085 ' in IPCTYP seçeneğinin açıklamasına bakın.

Tip dönüştürme isteğinde bulunmuyorsanız, giriş sırasında aşağıdaki değeri kullanabilirsiniz:

**Türü**

Özelliğin değeri, veri tipini dönüştürmeden döndürülür.

**VALLEN (10 basamaklı işaretli tamsayı)-giriş**

Değer alanının bayt cinsinden uzunluğu.

Değerin döndürülmesini gerektirmediğiniz özellikler için sıfır değerini belirtin. Bu özellikler, bir uygulamanın boş değerli ya da boş bir dizgi olması için tasarlanmış özellikler olabilir. Ayrıca, IPQLEN seçeneği belirtilmişse, sıfır değerini de belirtin; bu durumda değer döndürülmez.

**VALUE (1-byte bit stringxVALLEN)-çıkış**

Bu, sorgudaki özellik değerini içermek için kullanılan alandır. Arabellek, döndürülemekte olan değer için uygun bir sınırla hizalanmalıdır. Bunun yapmaması, daha sonra değere erişildiğinde bir hataya neden olabilir.

*VALLEN* , özellik değerinin uzunluğundan küçükse, özellik değerinin büyük bölümü *VALUE* değerine taşınır ve çağrı, tamamlanma kodu CCFAIL ve neden RC2469ile başarısız olur.

The character set of the data in *DEĞER* is given by the IPRETCSE field in the INQOPT parameter. The encoding of the data in *DEĞER* is given by the IPRETENC field in the INQOPT parameter.

*VALLEN* parametresi sıfırsa, *VALUE* ' ya başvurulmaz.

**DATLEN (10 basamaklı işaretli tamsayı)-çıkış**

Bu, *Değer* alanında döndürülen gerçek özellik değerinin bayt cinsinden uzunluğudur.

*DataLength* özellik değeri uzunluğundan küçükse, MQINQMP çağrısından geri dönüş için *DataLength* değeri yine de girilir. Bu, uygulamanın özellik değerini sığdırmak için gereken arabelleğin büyüklüğünü belirlemesine olanak tanır ve daha sonra, aramayı uygun boyutta bir arabellekle yeniden yayınlayabilir.

Aşağıdaki değerler de döndürülebilirler.

*Type* değiştirgesi TYPSTR ya da TYPBST olarak ayarlandıysa:

**VLEMP**

Özellik var, ancak karakter ya da byte içermiyor.

**CMPCOD (10 basamaklı işaretli tamsayı)-çıkış**

Tamamlanma kodu; bu kod aşağıdakilerden biridir:

**CCOK**

İşlem başarıyla tamamlandı.

**CCWARN**

Uyarı (kısmi tamamlama).

**CCCFAIL**

Arama başarısız oldu.

**REASON (10 basamaklı işaretli tamsayı)-çıkış**

Neden kodu niteleyici *CompCode*.

*CMPCOD* CCOK ise:

**YOK**

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* CCWARN ise:

**RC2492**

(2492, X'09BC') Döndürülen özellik adı dönüştürülmedi.

**RC2466**

(2466, X'09A2') Özellik değeri dönüştürülmedi.

**RC2467**

(2467, X'09A3') Özellik veri tipi desteklenmiyor.

**RC2421**

(2421, X'0975 ') Özellikler içeren bir MQRFH2 klasörü ayrıştırılmadı.

*CMPCOD* CCFAIL ise:

**RC2204**

(2204, X'089C') Bağdaştırıcı kullanılmıyor.

**RC2130**

(2130, X'0852 ') Bağdaştırıcı hizmeti modülü yüklenemiyor.

**RC2157**

(2157, X'086D') Birincil ve ana ASID değerleri farklı.

**RC2004**

(2004, X'07D4') Değer parametresi geçerli değil.

**RC2005**

(2005, X'07D5') Değer uzunluğu parametresi geçerli değil.

**RC2219**

(2219, X'08AB') Önceki arama tamamlanmadan önce girilen MQI çağırısı.

**RC2009**

(2009, X'07D9') Kuyruk yöneticisine yönelik bağlantı kaybedildi.

**RC2010**

(2010, X'07DA') Veri uzunluğu parametresi geçerli değil.

**RC2464**

(2464, X'09A0') Sorgula ileti özelliği seçenekleri yapısı geçerli değil.

**RC2460**

(2460, X'099C') İleti tanıtıcısı geçerli değil.

**RC2499**

(2499, X'09C3') İleti tanıtıcısı zaten kullanılıyor.

**RC2064**

(2046, X'07F8') Seçenekler geçerli değil ya da tutarlı değil.

**RC2482**

(2482, X'09B2') Özellik tanımlayıcı yapısı geçerli değil.

**RC2470**

(2470, X'09A6') İstenilen veri tipine gerçek olarak dönüştürme desteklenmiyor.

**RC2442**

(2442, X'098A') Özellik adı geçersiz.

**RC2465**

(2465, X'09A1') Özellik adı, döndürülen ad arabelleği için çok büyük.

**RC2471**

(2471, X'09A7') Özellik yok.

**RC2469**

(2469, X'09A5') Özellik değeri, Değer alanı için çok büyük.

**RC2472**

(2472, X'09A8') Değer verilerinde sayı biçimi hatası saptandı.

**RC2473**

(2473, X'09A9') İstenen özellik tipi geçersiz.

**RC2111**

(2111, X'083F') Özellik adı kodlanmış karakter takımı tanıtıcısı geçerli değil.

**RC2071**

(2071, X'0871 ') Kullanılabilir bellek yetersiz.

**RC2195**

(2195, X'0893 ') Beklenmeyen bir hata oluştu.

Bu kodlarla ilgili ayrıntılı bilgi için aşağıdaki başlara bakın:

- [IBM MQ for z/OS iletileri, tamamlama, ve neden kodları - IBM MQ for z/OS](#)
- Diğer tüm IBM MQ platformları için [Messages and reason codes](#)

**RPG Bildirimi**

```
C*..1.....2.....3.....4.....5.....6.....7..
C          CALLP      MQINQMP(HCONN : HMSG : INQOPT :
                        PRNAME : PRPDSC : TYPE :
                        VALLEN : VALUE : DATLEN :
                        CMPCOD : REASON)
```

Çağrıya ilişkin prototip tanımı:

```
DMQINQMP          PR          EXTPROC('MQINQMP')
D* Connection handle
D HCONN           10I 0 VALUE
D* Message handle
D HMSG           20I 0 VALUE
D* Options that control the action of MQINQMP
D INQOPT         72A
D* Property name
D PRNAME         32A
D* Property descriptor
D PRPDSC         24A
D* Property data type
D TYPE           10I 0
D* Length in bytes of the Value area
D VALLEN         10I 0 VALUE
D* Property value
D VALUE          *   VALUE
D* Length of the property value
D DATLEN         10I 0
D* Completion code
D CMPCOD         10I 0
D* Reason code qualifying CompCode
D REASON         10I 0
```


**MQMHBUF (İleti tanıtıcılarını arabelleğe dönüştür) IBM üzerinde**

MQMHBUF, bir ileti tanıtıcısını arabellek içine dönüştürür ve MQBUFMH çağrısının tersi olur.

- [“Sözdizimi” sayfa 1297](#)
- [“Kullanım notları” sayfa 1297](#)

- “Parametreler” sayfa 1297
- “RPG Bildirimi” sayfa 1299

## Sözdizimi

MQMHBUF (*Hconn, Hmsg, MsgHBufOpts, Name, MsgDesc, BufferLength, Buffer, DataLength, CompCode, Reason*)

## Kullanım notları

MQMHBUF, bir ileti tanıtıcısını arabellek içine dönüştürür.

Bu özelliği, belirli özelliklere erişmek için bir MQGET API çıkışıyla, ileti özelliği API ' lerini kullanarak ve ileti tanıtıcıları yerine MQRFH2 üstbilgilerini kullanmak üzere tasarlanmış bir uygulamaya geri iletebileceğiniz bir arabellekle kullanabilirsiniz.

Bu çağrı, ileti özelliklerini bir arabellekten ileti tanıtıcısı olarak ayrıştırmak için kullanabileceğiniz MQBUFMH çağrısının tersidir.

## Parametreler

MQMHBUF çağrısının aşağıdaki deęiřtirgeleri vardır:

### HCONN (10 basamaklı iřaretli tamsayı)-giriř

Bu tanıtıcı, kuyruk yöneticisiyle olan baęlantıyı gösterir.

*HCONN* deęeri, **HMSG** deęiřtirgesinde belirtilen ileti tanıtıcısını yaratmak için kullanılan baęlantı tanıtıcısı ile eřleşmelidir.

İleti tanıtıcısı HCUNAS kullanılarak yaratıldıysa, ileti tanıtıcısını silmeye iliřkin iř parçacığıda geçerli bir baęlantı kurulmalıdır. Geçerli bir baęlantı kurulmamıřsa, arama RC2009ile başarısız olur.

### HMSG (20 basamaklı iřaretli tamsayı)-giriř

Bu tanıtıcı, bir arabelleğin doldurulması gereken ileti tanıtıcısıdır.

Deęer önceki bir MQCRTMH çağrısı tarafından döndürüldü.

### MHBOPT (MQMHBO)-giriř

MQMHBO yapısı, uygulamaların, ileti tanıtıcılarından arabelleklerin nasıl üretildiğini denetleyen seçenekleri belirlemesine olanak saęlar.

Ayrıntılar için bkz. “IBM üzerinde MQBMHO (İleti iřleme seçenekleri için arabellek)” sayfa 1002.

### PRNAME (MQCHARV)-giriř

Arabelleğe yerleřtirilecek özelliğin ya da özelliklerin adı.

Adla eřleşen bir özellik bulunamazsa, arama RC2471ile başarısız olur.

### jokerler

Arabelleğe birden çok özellik koymak için genel arama karakteri kullanabilirsiniz. Bunu yapmak için, özellik adının sonunda yüzde iřaretini (%) kullanın. Bu genel arama karakteri, nokta (.) karakteri de içinde olmak üzere, sıfır ya da daha fazla karakterle eřleşir.

Özellik adlarının kullanımına iliřkin ek bilgi için [Özellik adları](#) ve [Özellik adı kısıtlamaları](#) başlıklı konuya bakın.

### MSGDSC (MQMD)-giriř/çıkıř

*MSGDSC* yapısı, arabellek alanının içeriğini açıklar.

Çıkıřta, *Encoding*, *CodedCharSetId* ve *Format* alanları, arama tarafından yazıldıęı şekilde, arabellek alanındaki verilerin kodlamasını, karakter kümesi tanıtıcısını ve biçimini doęru şekilde tanımlamaya ayarlanır.

Bu yapıdaki veriler, uygulamanın karakter kümesinde ve kodlamasında yer alıyor.

### **BUFLEN (10 basamaklı işaretli tamsayı)-giriş**

*BUFFLEN* , arabellek alanının bayt cinsinden uzunluğudur.

### **BUFFER (1-byte bit dizgi x BUFLEN)-giriş/çıkış**

*BUFFER* , ileti arabelleğinin bulunduğu alanı tanımlar. Çoğu veri için, arabelleği 4 byte 'lık bir sınır üzerinde hizalamanız gerekir.

*BUFFER* , karakter ya da sayısal veri içeriyorsa, **MSGDSC** parametresindeki *CodedCharSetId* ve *Encoding* alanlarını, veriye uygun değerlere ayarlayın; bu, gerekirse verilerin dönüştürülmesini sağlar.

Özellikler ileti arabelleğinde bulunursa, bunlar isteğe bağlı olarak kaldırılır; daha sonra, çağrıdan dönüşte ileti tanıtıcısından kullanılabilir duruma gelir.

C programlama dilinde, parametre, parametre olarak belirtilebilecek herhangi bir veri tipinin adresi anlamına gelen, işaretçi (pointer-to-void) olarak bildirilir.

**BUFLEN** parametresi sıfırsa, *BUFFER* ifadesine başvurulmaz. Bu durumda, C ya da System/390 çevirici programlarında yazılan programlarla geçirilen parametre adresi boş değerli olabilir.

### **DATLEN (10 basamaklı işaretli tamsayı)-çıkış**

*DATLEN* , arabelleğindeki döndürülen özelliklerin uzunluğudur (byte olarak). Değer sıfır ise, hiçbir özellik *PRNAME* ' de belirtilen değerle eşleşmedi ve arama, RC2471neden koduyla başarısız olur.

*BUFLEN* , belleğindeki özellikleri saklamak için gereken uzunluktan küçükse, MQMHBUF çağrısı RC2469ile başarısız olur, ancak yine de *DATLEN* ' a bir değer girilir. Bu, uygulamanın özellikleri barındırmak için gereken arabelleğin büyüklüğünü belirlemesine olanak tanır ve daha sonra, gerekli *BUFLEN* ile çağrıyı yeniden yayınlayın.

### **CMPCOD (10 basamaklı işaretli tamsayı)-çıkış**

Tamamlanma kodu; bu kod aşağıdakilerden biridir:

#### **CCOK**

İşlem başarıyla tamamlandı.

#### **CCCFAIL**

Arama başarısız oldu.

### **REASON (10 basamaklı işaretli tamsayı)-çıkış**

Neden kodu niteleyici *CMPCOD*.

*CMPCOD* CCOK ise:

#### **YOK**

(0, X'000 ') Raporlamak için bir neden yok.

*CMPCOD* CCFAIL ise:

#### **RC2204**

(2204, X'089C') Bağdaştırıcı kullanılamıyor.

#### **RC2130**

(2130, X'852 ') Bağdaştırıcı hizmeti modülü yüklenemiyor.

#### **RC2157**

(2157, X'86D') Birincil ve ana sunucu ASID değerleri farklı.

#### **RC2501**

(2501, X'095C') Arabellek seçenekleri yapısına ilişkin ileti tanıtıcısı geçerli değil.

#### **RC2004**

(2004, X'07D4') Arabellek parametresi geçerli değil.

#### **RC2005**

(2005, X'07D5') Arabellek uzunluğu parametresi geçerli değil.

**RC2219**

(2219, X'08AB') Önceki arama tamamlanmadan önce girilen MQI çağrısı.

**RC2009**

(2009, X'07D9') Kuyruk yöneticisine yönelik bağlantı kaybedildi.

**RC2010**

(2010, X'07DA') Veri uzunluğu parametresi geçerli değil.

**RC2460**

(2460, X'099C') İleti tanıtıcısı geçerli değil.

**RC2026**

(2026, X'07EA') İleti tanımlayıcısı geçerli değil.

**RC2499**

(2499, X'09C3') İleti tanıtıcısı zaten kullanılıyor.

**RC2046**

(2046, X'07FE') Seçenekler geçerli değil ya da tutarlı değil.

**RC2442**

(2442, X'098A') Özellik adı geçerli değil.

**RC2471**

(2471, X'09A7') Özellik yok.

**RC2469**

(2469, X'09A5') BufferLength değeri, belirtilen özellikleri içermek için çok küçük.

**RC2195**

(2195, X'893 ') Beklenmeyen bir hata oluştu.

**RPG Bildirimi**

```
C*..1.....2.....3.....4.....5.....6.....7..
C          CALLP      MQMHBUF(HCONN : HMSG : MHB OPT :
                          PRNAME : MSGDSC : BUFL EN :
                          BUFFER : DATLEN :
                          CMPCOD : REASON)
```

Çağrıya ilişkin prototip tanımı:

```
DMQMHBUF          PR          EXTPROC('MQMHBUF')
D* Connection handle
D HCONN            10I 0 VALUE
D* Message handle
D HMSG             20I 0 VALUE
D* Options that control the action of MQMHBUF
D MHB OPT          12A
D* Property name
D PRNAME           32A
D* Message descriptor
D MSGDSC           364A
D* Length in bytes of the Buffer area
D BUFL EN          10I 0 VALUE
D* Area to contain the properties
D BUFFER           *   VALUE
D* Length of the properties
D DATLEN           10I 0
D* Completion code
D CMPCOD           10I 0
D* Reason code qualifying CompCode
D REASON           10I 0
```

**IBM i MQOPEN (Open object) on IBM i**

MQOPEN çağrısı bir nesneye erişim oluşturur.

Aşağıdaki nesne tipleri geçerlidir:

- Kuyruk (dağıtım listeleri de içinde olmak üzere)
- Ad Listesi
- Süreç tanımlaması
- Kuyruk yöneticisi
- Konu

## Dizin

- [“Sözdizimi” sayfa 1300](#)
- [“Kullanım notları” sayfa 1300](#)
- [“Parametreler” sayfa 1304](#)
- [“RPG Bildirimi” sayfa 1310](#)

## Sözdizimi

MQOPEN (*HCONN, OBJDSC, OPTS, HOBJ, CMPCOD, REASON*)

## Kullanım notları

1. Açılan nesne aşağıdakilerden biridir:

- Kuyruk, aşağıdaki gibi sıralanacak:
  - İletileri alma ya da göz atma (MQGET çağrısını kullanarak)
  - İletileri koy (MQPUT çağrısını kullanarak)
  - Kuyruğun özniteliklerine ilişkin olarak sorgulama (MQINQ çağrısını kullanarak)
  - Kuyruğun özniteliklerini ayarlar (MQSET çağrısını kullanarak)

Kuyruk adı bir model kuyruğsa, dinamik bir yerel kuyruk yaratılır.

Dağıtım listesi, kuyrukların listesini içeren özel bir kuyruk nesnesidir. İleti koymak için açılabilir, ancak ileti almak ya da bunlara göz atmak ya da öznitelikleri sorgulamak ya da ayarlamak için açılabilir. Ek ayrıntılar için kullanım notunun 8 'e bakın.

QSGDISP (GROUP) sahip olan bir kuyruk, MQOPEN ya da MQPUT1 çağrılarıyla kullanılamayacak özel bir kuyruk tanımlaması tipidir.

- Bir ad listesi aşağıdaki gibi sıralanacak:
  - Listedeki kuyrukların adlarına ilişkin bilgi (MQINQ çağrısını kullanarak).
- Aşağıdaki işlemleri yapmak için bir süreç tanımlaması:
  - Süreç özniteliklerine ilişkin bilgi (MQINQ çağrısını kullanarak).
- Kuyruk yöneticisi aşağıdakileri yapmak için:
  - Yerel kuyruk yöneticisinin özniteliklerine (MQINQ çağrısını kullanarak) ilişkin bilgileri girin.

2. Bir uygulamanın aynı nesneyi bir kereden fazla açma uygulaması için geçerlidir. Her bir açık için farklı bir nesne tanıtıcısı döndürülür. Döndürülen her bir tanıtıcı, ilgili açık olan işlevlerin gerçekleştirildiği işlevler için kullanılabilir.

3. Açılmakta olan nesne bir kuyruksa, ancak bir küme kuyruğu değilse, yerel kuyruk yöneticisi içindeki tüm ad çözümlemesi, MQOPER çağrısının zamanında gerçekleşir. Bu, belirli bir MQOPER çağrısına ilişkin aşağıdakilerden birini ya da birkaçını içerebilir:

- Diğer ad çözümlemesi, bir temel kuyruk adı
- Uzak bir kuyruğun yerel tanımlamasını uzak kuyruk yöneticisinin adına ve kuyruğun uzak kuyruk yöneticisinde bulunduğu ada göre çözme
- Uzak kuyruk yöneticisi adının yerel iletim kuyruğu adına çözülmesi



Ancak, tanıtıcı için sonraki MQINQ ya da MQSET çağrılarının yalnızca açılmış olan ada ve ad çözme işleminden sonra ortaya çıkan nesne için değil, bu adla ilişkilendirilmesinin farkında olun. Örneğin, açılan nesne bir diğer ad ise, MQINQ çağrısının döndürdüğü öznitelikler, diğer adın özniteliklerinin değil, diğer adın özniteliklerini değil, diğer adı çözer. Ancak, buna karşılık gelen MQOPER üzerindeki **OPTS** parametresi için belirtilenler dikkate alınmaksızın, ad çözme denetimi devam eder.

Açılmakta olan nesne bir küme kuyruğunsa, ad çözümlemesi, MQOPEN çağrısı sırasında ortaya çıkabilir ya da daha sonra ertelenebilir. Çözülme işleminin gerçekleştirildiği nokta, MQOPEN çağrısında belirlenen OOBND\* seçenekleri tarafından denetlenir:

- OOBND0
- OOBNDN
- OOBNDQ

Küme kuyruklarına ilişkin ad çözümlemesiyle ilgili ek bilgi için [Ad çözünürlüğü](#) başlıklı konuya bakın.

4. Bir nesnenin öznitelikleri, bir uygulama için açık olduğunda, nesnenin öznitelikleri değişebilir. Birçok durumda, uygulama bunu fark etmez; ancak, bazı öznitelikler için kuyruk yöneticisi tanıtıcıyı artık geçerli değil olarak işaretler. Bu bilgiler şunlardır:

- Nesnenin ad çözümlemesini etkileyen herhangi bir öznitelik. Bu, kullanılan açık seçeneklerden bağımsız olarak geçerlidir ve aşağıdakileri içerir:
  - Açık olan bir diğer ad kuyruğunun **BaseQName** özniteliğe yapılan bir değişiklik.
  - A change to the **RemoteQName** or **RemoteQMgrName** queue attributes, for any handle that is open for this queue, or for a queue which resolves through this definition as a queue manager alias.
  - Uzak bir kuyruk için açık olan açık bir tanıtıcı değeri olan herhangi bir değişiklik, farklı bir *iletim* kuyruğuna çözüyor ya da birden bire çözülemekten başarısız oldu. Örneğin, şunları içerebilir:
    - Tanımın bir kuyruk için mi, yoksa bir kuyruk yöneticisi diğer adı için mi kullanılmakta olduğunu, uzak kuyruğun yerel tanımlamasının **XmitQName** özniteliğe ilişkin bir değişiklik.Bunun bir kural dışı durumu, yani yeni bir iletim kuyruğu yaratılmasına neden olur. Bu kuyruğa çözülmüş olan bir tanıtıcı, tanıtıcı açıldığı zaman, ancak varsayılan iletim kuyruğuna çözüldüğünde, geçersiz kılınmadığında bu kuyruğa ilişkin olarak çözülmüş olabilir.
  - **DefXmitQName** kuyruk yöneticisi özniteliğe ilişkin bir değişiklik. Bu durumda, önceden adlandırılan kuyruğa (yalnızca varsayılan iletim kuyruğu olduğu için) çözülmüş olan tüm açık tutamaçlar geçersiz olarak işaretlenir. Bu kuyruğun diğer nedenleri için çözülen işler etkilenmez.
- Bu kuyruk için OOINPS erişimi sağlayan iki ya da daha fazla tanıtıcı varsa ya da bu kuyruğa çözülen bir kuyruk için, **Shareability** kuyruk özniteliği. Bu durumda, bu kuyruk için açık olan *tüm* tutamaçları ya da bu kuyruğa çözülen bir kuyruk için, açık seçeneklerden bağımsız olarak, geçersiz olarak işaretlenir.
- The **Usage** queue attribute, for all handles that are open for this queue, or for a queue that resolves to this queue, regardless of the open options.

Bir tanıtıcı geçersiz olarak imlenirse, bu tanıtıcıyı kullanarak sonraki tüm çağrılar (MQCLOSE dışında), RC2041; neden koduyla başarısız olur; uygulama bir MQCLOSE çağrısı yayınlamalıdır (özgün tanıtıcıyı kullanarak) ve kuyruğu yeniden açmalıdır. Uygulama mantığının gerektirdiği şekilde, önceki başarılı aramalardan eski tanıtıcı ile ilgili kesinleştirilmemiş güncellemeler kesinleştirilebilir ya da yedeklenebilirler.

Bir özniteliğin değiştirilmesi bunun olmasına neden olur ise, komutun özel bir "force" sürümü kullanılmalıdır.

5. Kuyruk yöneticisi, uygulamanın çalışmakta olduğu kullanıcı kimliğinin erişim izni verilmeden önce uygun yetki düzeyine sahip olduğunu doğrulamak için, bir MQOPEN çağrısı yayımlandığında güvenlik denetimleri gerçekleştirir. Yetki denetimi, bir ad çözüldükten sonra, açılmakta olan nesnenin adı ya da adlarında değil, açılmakta olan nesnenin adı üzerinde yapılır.

Açılmakta olan nesne bir model kuyruğuyorsa, kuyruk yöneticisi hem model kuyruğunun adı, hem de yaratılan dinamik kuyruğun adı için bir tam güvenlik denetimi gerçekleştirir. Sonuçta ortaya çıkan

dinamik kuyruk açık bir şekilde açılırsa, dinamik kuyruk adına göre daha fazla kaynak güvenliği denetimi gerçekleştirilir.

6. Uzak bir kuyruk, bu çağrıya ilişkin **OBJDSC** değiştirgesindeki iki yoldan biriyle belirtilebilir ( “IBM i üzerinde MQOD (Nesne tanımlayıcı)” sayfa 1140 içinde açıklanan *ODON* ve *ODMN* alanlarına bakın):

- *ODON* için, uzak kuyruğun yerel tanımlamasının adını belirtilerek. Bu durumda *ODMN* , yerel kuyruk yöneticisine başvurur ve boşluk olarak belirtilebilir.  
Yerel kuyruk yöneticisi tarafından gerçekleştirilen güvenlik doğrulaması, kullanıcının uzak kuyruğun yerel tanımlamasını açma yetkisine sahip olduğunu doğrular.
- *ODON* için, uzak kuyruk yöneticisi tarafından bilindiği şekilde uzak kuyruğun adını belirterek. Bu durumda, *ODMN* uzak kuyruk yöneticisinin adıdır.  
Yerel kuyruk yöneticisi tarafından gerçekleştirilen güvenlik doğrulaması, kullanıcının ad çözme işleminden kaynaklanan iletim kuyruğuna ileti gönderme yetkisine sahip olduğunu doğrular.

Her iki durumda da:

- Kullanıcının, kuyruktaki ileti koyma yetkisi olup olmadığını denetlemek için, yerel kuyruk yöneticisi tarafından uzak kuyruk yöneticisine hiçbir ileti gönderilmez.
- Uzak kuyruk yöneticisinde bir ileti geldiğinde, iletiyi oluşturan kullanıcının yetkisi olmadığı için uzak kuyruk yöneticisi bunu reddedebilir.

7. OOB<sub>RW</sub> seçeneği ile yapılan bir MQOPED çağrısı, nesne tanıtıcısını ve göz atma seçeneklerinden birini belirten MQGET çağrılarını kullanılmak üzere bir göz atma imleci oluşturur. Bu, kuyruğun içeriği değiştirilmeden taranmasını sağlar. Göz atma yoluyla bulunan bir ileti daha sonra GMMUC seçeneği kullanılarak kuyruktan kaldırılabilir.

Aynı kuyruk için birden çok MQOPEN isteği yayınlayarak, tek bir uygulama için birden çok imleçle etkin durumda olabilen geçici çizelge sayısı kullanılabilir.

8. Aşağıdaki notlar dağıtım listelerinin kullanımı için geçerlidir.

- Bir dağıtım listesi açılırken, MQOD yapısındaki alanların aşağıdaki gibi ayarlanması gerekir:
  - *ODVER* , *ODVER2* ya da sonraki bir değer olmalıdır.
  - *ODOT* , *OTQ* olmalıdır.
  - *ODON* boş bırakılmalı ya da boş değerli dizgi olmalıdır.
  - *ODMN* boş bırakılmalı ya da boş değerli dizgi olmalıdır.
  - *ODREC* sıfırdan büyük olmalıdır.
  - *ODORO* ve *ODORP* ' dan biri sıfır ve diğer sıfır dışında bir değer olmalıdır.
  - *ODRRO* ve *ODRRP* dışında birden çok kişi sıfır olamaz.
  - *ODORO* ya da *ODORP* tarafından adreslenen *ODREC* nesne kayıtları olmalıdır. Nesne kayıtlarının açılması için hedef kuyrukların adlarına ayarlanması gerekir.
  - If one of *ODRRO* and *ODRRP* is nonzero, there must be *ODREC* response records present. Çağrı, neden kodu RC2136 ile tamamlanırsa, kuyruk yöneticisi tarafından ayarlanır.

Bir version-2 MQOD, *ODREC* ' un sıfır olduğunu doğrulayarak, dağıtım listesinde olmayan tek bir kuyruğu açmak için de kullanılabilir.

- **OPTS** değiştirgesinde yalnızca aşağıdaki açık seçenekler geçerlidir:
  - *OOUT*
  - *OOPAS\**
  - *OOSSET\**
  - *OOALTU*
  - *OOFIQ*

- Dağıtım listesindeki hedef kuyruklar yerel, diğer ad ya da uzak kuyruklar olabilir, ancak bunlar model kuyrukları olamaz. Bir model kuyruğu belirtilirse, o kuyruk açılmaz ve neden kodu RC2057' dir. Ancak bu, listedeki diğer kuyrukların başarıyla açılmamasını engellememektedir.
- Tamamlanma kodu ve neden kodu değiştirgeleri aşağıdaki gibi ayarlanır:
  - Dağıtım listesindeki kuyruklara ilişkin açık işlemler başarılı olursa ya da aynı şekilde başarısız olursa, tamamlanma kodu ve neden kodu değiştirgeleri ortak sonucu tanımlamaya ayarlanır. Bu durumda, MQRN yanıt kayıtları (uygulama tarafından sağlandıysa) belirlenmez.  
Örneğin, her açık başarılı olursa, tamamlanma kodu CCOK olarak ayarlanır ve neden kodu RCNONE olur; her bir açma başarısız olursa, kuyrukların hiçbiri var olmadığı için, parametreler CCFAIL ve RC2085 olarak ayarlanır.
  - Dağıtım listesindeki kuyruklara ilişkin açık işlemler başarılı olamazsa ya da aynı şekilde başarısız olursa:
    - En az bir açma başarılı olursa, tamamlanma kodu parametresi CCWARN değerine ayarlanır ve tümü başarısız olursa CCFAIL ' e ayarlanır.
    - Neden kodu parametresi RC2136 olarak ayarlıdır.
    - Yanıt kayıtları (uygulama tarafından sağlandıysa), dağıtım listesindeki kuyruklara ilişkin tek tek tamamlama kodlarına ve neden kodlarına ayarlanmıştır.

- When a distribution list has been opened successfully, the handle *HOB*J returned by the call can be used on subsequent MQPUT calls to put messages to queues in the distribution list, and on an MQCLOSE call to relinquish access to the distribution list. Dağıtım listesi için geçerli olan tek kapatma seçeneği CONONE seçeneğidir.

MQPUT1 çağrısı, bir dağıtım listesine ileti koymak için de kullanılabilir; listedeki kuyrukları tanımlayan MQOD yapısı, o çağrıdaki bir parametre olarak belirtilir.

- Uygulamanın, izin verilen tanıtıcı sayısı üst sınırını aştığını ( **MaxHandles** kuyruk yöneticisi özniteliğine bakın) kontrol ederken, dağıtım listesindeki her başarıyla açılan hedef, *ayrı* bir tanıtıcı olarak sayılır. Dağıtım listesindeki hedeflerin ikisi ya da daha çoğu aynı fiziksel kuyruğa gerçekten çözüldüğünde bile bu geçerlidir. Bir dağıtım listesi için MQOPEN ya da MQPUT1 çağrısı, uygulama tarafından kullanılan tanıtıcı sayısının *MaxHandles*' u aşmasına neden olursa, çağrı neden kodu RC2017 ile başarısız olur.
- Açılan her hedef, **OpenOutputCount** özniteliğinin değerine göre bir artırılarak bir değer artırılır. Dağıtım listesindeki hedeflerin ikisi ya da daha çoğu aynı fiziksel kuyruğa çözüldürse, o kuyruk, dağıtım listesindeki hedef sayısı tarafından bu kuyruğa uygun olarak artırılan **OpenOutputCount** özniteliğine sahip olur.
- Bir tanıtıcıyı geçersiz hale getirmesine neden olacak kuyruk tanımlamalarında yapılan herhangi bir değişiklik, kuyrukların tek tek açılmasına neden oldu (örneğin, çözüm yolundaki bir değişiklik), dağıtım listesi tutamaçlarının geçersiz olmasına neden olmaz. Ancak, sonraki bir MQPUT çağrısında dağıtım listesi tanıtıcısı kullanıldığında, bu kuyruğun başarısızlığa neden olması da sonuçlanabilir.
- Bir dağıtım listesi için geçerli, yalnızca bir hedef içerilir.

#### 9. Aşağıdaki notlar küme kuyruklarının kullanımı için geçerlidir.

- Bir küme kuyruğu ilk kez açıldığında ve yerel kuyruk yöneticisi tam bir havuz kuyruğu yöneticisi değilse, yerel kuyruk yöneticisi, tam havuz kuyruk yöneticisinden küme kuyruğuna ilişkin bilgileri alır. Ağ meşgul olduğunda, yerel kuyruk yöneticisinin gerekli bilgileri havuz kuyruğu yöneticisinden alması birkaç saniye sürebilir. Sonuç olarak, MQOPER çağrısını yayınlayan uygulamanın, denetim MQOPEN çağrısından döndürmeden önce en çok 10 saniye beklemesi gerekebilir. Yerel kuyruk yöneticisi, bu süre içinde küme kuyrukla ilgili gerekli bilgileri alamazsa, arama başarısız olur ve neden kodu RC2189 ile başarısız olur.
- Bir küme kuyruğu açıldığında ve kuyrukta kuyruğun birden çok eşgörünümü varsa, gerçekte açılan eşgörünüm, MQOPEN çağrısında belirtilen seçeneklere bağlıdır:
  - Belirtilen seçenekler aşağıdakilerden birini içerirse:
    - OOB

- OOINPQ
- OOINPX
- OOINPS
- OOSSET

Açılan küme kuyruğunun yönetim ortamı yerel yönetim ortamı için gereklidir. Kuyruğun yerel bir eşgörünümü yoksa, MQOPEN çağrısı başarısız olur.

- Belirtilen seçenekler, yukarıdakilerden hiçbiri içermiyorsa, ancak aşağıdakilerden birini ya da her ikisini de içerir:

- OOINQ
- OOUT

açılan yönetim ortamı, varsa, yerel yönetim ortamı ve tersi durumda uzak bir yönetim ortamı olabilir. Ancak, kuyruk yöneticisi tarafından seçilen yönetim ortamı bir küme iş yükü çıkışıyla (varsa) değiştirilebilir.

Küme kuyruklarıyla ilgili ek bilgi için [Küme kuyrukları](#) başlıklı konuya bakın.

10. Tetikleme izleme programı tarafından başlatılan uygulamalar, uygulama başlatıldığında uygulamayla ilişkili olan kuyruğun adını iletir. Kuyruğu açmak için bu kuyruk adı **OBJDSC** değiştirilmesinde belirtilebilir. Ek ayrıntılar için MQTMC yapısının açıklamasına bakın.
11. When using the OORLOQ option, the local queue is already returned when either a local, alias, or model queue is opened, but this is not the case when, for example, a remote queue or a non-local cluster queue is opened; the ResolvedQName and ResolvedQMgrName are entered with the RemoteQName and RemoteQMgrName found in the remote queue definition, or similarly with the chosen remote cluster queue. Açılış sırasında OORLOQ belirtilirse, örneğin, uzak bir kuyruk olan ResolvedQName , artık iletilerin yerleştirilecek iletim kuyruğu olacaktır. The ResolvedQMgrName will be entered with the name of the local queue manager hosting the transmission queue. Bir kullanıcı, bir kuyruktaki göz atma, giriş ya da çıkış için yetkilendirilmişse, bu işareti MQOPEN çağrısında bu işareti belirtmek için gereken yetkiye sahip olur. Özel bir yetki gerekli değildir.

## Parametreler

MQOPER çağrısında şu değiştirgeler vardır:

### HCONN (10 basamaklı işaretli tamsayı)-giriş

Bağlantı tanıtıcısı.

Bu tanıtıcı, kuyruk yöneticisiyle olan bağlantıyı gösterir. The value of HCONN was returned by a previous MQCONN or MQCONN call.

### OBJDSC (MQOD)-giriş/çıkış

Nesne tanımlayıcısı.

Bu yapı, açılacak nesneyi tanımlayan bir yapıdır; ayrıntılar için "[IBM i üzerinde MQOD \(Nesne tanımlayıcı\)](#)" sayfa 1140 ' e bakın.

**OBJDSC** parametresindeki *ODON* alanı, bir model kuyruğunun adı ise, dinamik bir yerel kuyruk Model kuyruğunun öznitelikleriyle yaratılır; bu durum, **OPTS** parametresi tarafından belirtilen açık seçeneklerden bağımsız olarak oluşur. MQOPER çağrısı tarafından döndürülen *HOBJ* ile sonraki işlemler, model kuyruğunda değil, yeni dinamik kuyruğun üzerinde gerçekleştirilir. Bu, MQINQ ve MQSET çağrıları için bile geçerlidir. **OBJDSC** parametresindeki model kuyruğunun adı, yaratılan dinamik kuyruğun adıyla değiştirilir. Dinamik kuyruğun tipi, model kuyruğunun **DefinitionType** özniteliğinin değerine göre belirlenir (bkz. "[Kuyruklara ilişkin öznitelikler](#)" sayfa 1345 ). Dinamik kuyruklar için geçerli olan kapatma seçeneklerine ilişkin bilgi için, MQCLOSE çağrısının açıklamasına bakın.

### OPTS (10 basamaklı işaretli tamsayı)-giriş

MQOPER işlemini denetleyen seçenekler.

Aşağıdaki seçeneklerden en az birinin belirtilmesi gerekir:

- OOB<sub>RW</sub>
- OOINP\* (bunlardan yalnızca biri)
- OOINQ
- OOUT
- OOSET
- OORLQ

Diğer seçenekler, zorunlu olarak belirtilebilir. Birden fazla seçenek gerekliyse, değerler eklenebilir (aynı değışımezi bir kereden fazla eklemeyin). Geçerli olmayan birleşimler dikkat edilir; diğer tüm birleşimler geçerli olur. Yalnızca, *OBJDSC* ile belirtilen nesne tipi için geçerli olan seçenekler kullanılabilir (bkz. [Her kuyruk tipi için geçerli MQOPER seçenekleri](#)).

**Erişim seçenekleri:** Aşağıdaki seçenekler, nesne üzerinde gerçekleştirilebilecek işlemlerin tipini denetler:

#### **OOINPQ**

Kuyruk tanımlı varsayılan iletileri kullanarak iletileri almak için kuyruğu açın.

Kuyruk sonraki MQGET çağrılarıyla kullanılmak üzere açılmıştır. Erişim tipi, **DefInputOpenOption** kuyruk özniteliğinin değerine bağlı olarak paylaşılan ya da dışlayıcı bir tiptir; ayrıntılar için [“Kuyruklara ilişkin öznitelikler” sayfa 1345](#) ' e bakın.

Bu seçenek yalnızca yerel, diğer ad ve model kuyrukları için geçerlidir; uzak kuyruklar, dağıtım listeleri ve kuyruklar olmayan nesnelere için geçerli değildir.

#### **OOINPS**

Paylaşılan erişimi olan iletileri almak için kuyruğu açın.

Kuyruk sonraki MQGET çağrılarıyla kullanılmak üzere açılmıştır. The call can succeed if the queue is currently open by this or another application with OOINPS, but fails with reason code RC2042 if the queue is currently open with OOINPX.

Bu seçenek yalnızca yerel, diğer ad ve model kuyrukları için geçerlidir; uzak kuyruklar, dağıtım listeleri ve kuyruklar olmayan nesnelere için geçerli değildir.

#### **OOINPX**

Dışlayıcı erişimi olan iletileri almak için kuyruğu açın.

Kuyruk sonraki MQGET çağrılarıyla kullanılmak üzere açılmıştır. Kuyruk şu anda bu ya da herhangi bir tipte (OOINPS ya da OOINPX) giriş için başka bir uygulama tarafından açılmışsa, çağrı neden kodu RC2042 ile başarısız olur.

Bu seçenek yalnızca yerel, diğer ad ve model kuyrukları için geçerlidir; uzak kuyruklar, dağıtım listeleri ve kuyruklar olmayan nesnelere için geçerli değildir.

Aşağıdaki notlar bu seçenekler için geçerlidir:

- Bu seçeneklerden yalnızca biri belirlenebilir.
- **InhibitGet** kuyruk özniteliği QAGETI olarak ayarlansa bile, bu seçeneklerden biriyle MQOPER çağrısı başarılı olabilir (öznitelik bu değere ayarlandığında sonraki MQGET çağrıları başarısız olur).
- Kuyruk paylaşılabilir olarak tanımlandıysa (yani, **Shareability** kuyruk özniteliğinin QANSHR değeri vardır), paylaşılan erişim için kuyruğu açma girişimleri, dışlayıcı erişim ile kuyruğu açma girişimi olarak kabul edilir.
- Bu seçeneklerden biriyle bir diğer ad kuyruğu açılırsa, dışlayıcı kullanım sınaması (ya da başka bir uygulamanın dışlayıcı kullanımı olup olmadığı), diğer adın çözümleneceği temel kuyruğa karşıdır.
- These options are not valid if *ODMN* is the name of a queue manager alias; this is true even if the value of the **RemoteQMGrName** attribute in the local definition of a remote queue used for queue manager aliasing is the name of the local queue manager.

## OBRW

İletilere göz atmak için kuyruğu açın.

Kuyruk, sonraki MQGET çağrılarıyla kullanılmak üzere aşağıdaki seçeneklerden biriyle kullanılmak üzere açılır:

- GMBRWF
- GMBRWN
- GMBRWC

Kuyruğun OOINPX için açık olmasına rağmen, bu izin verilir. OBRW seçeneği ile yapılan bir MQOPED çağrısı bir göz atma imleci oluşturur ve bunu, kuyrukta ilk iletiden önce mantıksal olarak konumlandırır; daha fazla bilgi için [“IBM üzerinde MQGMO \(get-message options\)” sayfa 1060](#) ' ta açıklanan *GMOPT* alanına bakın.

Bu seçenek yalnızca yerel, diğer ad ve model kuyrukları için geçerlidir; uzak kuyruklar, dağıtım listeleri ve kuyruklar olmayan nesnelere için geçerli değildir. It is also not valid if *ODMN* is the name of a queue manager alias; this is true even if the value of the **RemoteQMGrName** attribute in the local definition of a remote queue used for queue manager aliasing is the name of the local queue manager.

## OOUT

İletileri koymak için kuyruğu açın ya da iletileri yayınlamak için bir konu ya da konu dizisi.

Kuyruk, sonraki MQPUT çağrılarıyla kullanılmak üzere açılır.

**InhibitPut** kuyruk özneliği QAPUTI olarak ayarlansa bile, bu seçenekle birlikte bir MQOP çağrısı başarılı olabilir (öznitelik bu değere ayarlandığında sonraki MQPUT çağrılarını başarısız olur).

Bu seçenek, dağıtım listeleri ve konular da içinde olmak üzere, tüm kuyruk tipleri için geçerlidir.

## OQINQ

Öznitelikleri sorgulamak için nesneyi açın.

İzleyen MQINQ çağrılarıyla kullanılmak üzere kuyruk, ad listesi, süreç tanımlaması ya da kuyruk yöneticisi açılmıştır.

Bu seçenek, dağıtım listesi dışındaki tüm nesne tipleri için geçerlidir. Bir kuyruk yöneticisi diğer adının adı *ODMN* ise, bu değer geçerli değildir; bu, kuyruk yöneticisi aliting için kullanılan uzak bir kuyruğun yerel tanımlamasındaki **RemoteQMGrName** özneliğinin değeri yerel kuyruk yöneticisinin adıdır.

## OQSET

Öznitelikleri ayarlamak için kuyruğu açın.

Kuyruk, sonraki MQSET çağrılarıyla kullanılmak üzere açılmıştır.

Bu seçenek, dağıtım listesi dışındaki tüm kuyruk tipleri için geçerlidir. It is not valid if *ODMN* is the name of a local definition of a remote queue; this is true even if the value of the **RemoteQMGrName** attribute in the local definition of a remote queue used for queue manager aliasing is the name of the local queue manager.

**Bağ tanımlama seçenekleri:** Açılmakta olan nesne bir küme kuyruğu olduğunda, aşağıdaki seçenekler geçerli olur; bu seçenekler kuyruk tanıtıcısı bağ tanımını, küme kuyruğunun bir örneğine denetler:

## OQBDO

Kuyruk açıldığında bağ tanımlama tanıtıcısı hedef değerine bağlanır.

Bu, yerel kuyruk yöneticisinin, kuyruk açıldığı sırada kuyruk tanıtıcısını hedef kuyruğun bir örneğine bağlamasına neden olur. Sonuç olarak, bu tanıtıcıyı kullanan tüm iletiler hedef kuyruğun aynı örneğine ve aynı rota tarafından gönderilir.

Bu seçenek yalnızca kuyruklar için geçerlidir ve yalnızca küme kuyruklarını etkiler. Bir küme kuyruğu olmayan bir kuyruk için belirtilirse, bu seçenek yoksayılır.

## OOBNDN

Belirli bir hedef için bağ tanımlamayın.

Bu, yerel kuyruk yöneticisinin kuyruk tanıtıcısını hedef kuyruğun bir örneğine bağlamasını durdurur. Sonuç olarak, bu tanıtıcı değeri kullanılarak yapılan MQPUT çağruları, iletilerin hedef kuyruğun *farklı* eşgörünümlerine gönderilmesine ya da aynı yönetim ortamına gönderilmesine ancak farklı rotalar tarafından gönderilmesine neden olabilir. Ayrıca, seçilen yönetim ortamının yerel kuyruk yöneticisi tarafından, uzak kuyruk yöneticisi tarafından ya da ağ koşullarına göre bir ileti kanalı aracısı (MCA) tarafından değiştirilmesini de sağlar.

**Not:** Bir işlemin tamamlanması için bir *serisi* değişik tokuş edilmesi gereken istemci ve sunucu uygulamaları OOBNDN (*DefBind* ' in BNDNOT değerine sahip olduğunda OOBNDQ ya da OOBNDQ) kullanılmaması gerekir; serideki art arda gelen iletiler sunucu uygulamasının farklı örneklerine gönderilebilir.

Bir küme kuyruğu için OOBRW ya da OOINP\* seçeneklerinden biri belirtilirse, kuyruk yöneticisi, küme kuyruğunun yerel yönetim ortamını seçmekte zorlanır. Sonuç olarak, OOBNDN belirtilse de, kuyruk tanıtıcısı bağ tanımı düzeltilir.

OOINQ, OOBNDN ile belirtilirse, bu tanıtıcıyı kullanan art arda gelen MQINQ çağruları küme kuyruğunun farklı eşgörünümlerini sorgulayabilir; ancak, tipik olarak, tüm eşgörünümler aynı öznitelik değerlerine sahip olur.

OOBNDN yalnızca kuyruklar için geçerlidir ve yalnızca küme kuyruklarını etkiler. Bir küme kuyruğu olmayan bir kuyruk için belirtilirse, bu seçenek yoksayılr.

## OOBNDQ

Kuyruk için varsayılan bağ tanımını kullan.

Bu, yerel kuyruk yöneticisinin kuyruk tanıtıcısını **DefBind** kuyruk özniteliğine göre tanımlanan şekilde bağlamasına neden olur. Bu özniteliğin değeri BNDOPN ya da BNDNOT değeridir.

OOBNDQ ve OOBNDN belirtilmediyse OOBNDQ varsayılan değerdir.

OOBNDQ, program belgelerine yardımcı olmak için tanımlıdır. Bu seçeneğin diğer iki bağ tanımlama seçeneğinden biriyle birlikte kullanılması amaçlanmaz, ancak değeri sıfır olduğu için bu tür bir kullanım saptanamaz.

**Bağlam seçenekleri:** Aşağıdaki seçenekler ileti bağlamının işlenmesini denetler:

## OOSAVA

İleti alındığında bağlam kaydedin.

Bağlam bilgileri bu kuyruk tanıtıcısı ile ilişkilendirildi. Bu bilgi, bu tanıtıcı kullanılarak alınan herhangi bir iletinin bağlamından ayarlanır. İleti bağlamına ilişkin daha fazla bilgi için bkz. [İleti bağlamı](#) ve [Bağlam bilgilerini denetleme](#).

Bu bağlam bilgileri, daha sonra MQPUT ya da MQPUT1 çağrılarını kullanarak bir kuyruğa yerleştirilecek bir iletiye geçirilebilir. "[MQPMO \(Put-message options\) on IBM i](#)" sayfa 1154 içinde açıklanan PMPASI ve PMPASA seçeneklerine bakın.

Bir ileti başarıyla alınıncaya kadar, bir kuyrukte belirtilen bir iletiye bağlam geçirilemez.

GMBRW\* göz atma seçeneklerinden biri kullanılarak alınan bir ileti, bağlam bilgilerinin saklanmasına neden olmaz ( **MSGDSC** parametresindeki bağlam alanları göz atma işleminden sonra ayarlansa da).

Bu seçenek yalnızca yerel, diğer ad ve model kuyrukları için geçerlidir; uzak kuyruklar, dağıtım listeleri ve kuyruklar olmayan nesnelere için geçerli değildir. OOINP\* seçeneklerinden biri belirtilmeli.

## OOPASI

Kimlik bağlamının iletilmesine izin verir.

This allows the PMPASI option to be specified in the **PMO** parameter when a message is put on a queue; this gives the message the identity context information from an input queue that was



opened with the OOSAVA option. İleti bağlamına ilişkin daha fazla bilgi için bkz. [İleti bağlamı ve Bağlam bilgilerini denetleme](#).

OOUT seçeneği belirtilmelidir.

Bu seçenek, dağıtım listeleri de içinde olmak üzere, tüm kuyruk tipleri için geçerlidir.

### **OOPASA**

Tüm bağlamın iletilmesine izin verir.

This allows the PMPASA option to be specified in the **PMO** parameter when a message is put on a queue; this gives the message the identity and origin context information from an input queue that was opened with the OOSAVA option. İleti bağlamına ilişkin daha fazla bilgi için bkz. [İleti bağlamı ve Bağlam bilgilerini denetleme](#).

Bu seçenek, bu nedenle belirtilmemesi gereken OOPASI ' yi belirtir. OOUT seçeneği belirtilmelidir.

Bu seçenek, dağıtım listeleri de içinde olmak üzere, tüm kuyruk tipleri için geçerlidir.

### **OOSSETI**

Kimlik bağlamının ayarlanmasına izin ver

Bu, bir kuyruğa ilişkin bir ileti konduğunda, PMSETI seçeneğinin **PMO** parametresinde belirtilmesine olanak tanır; bu ileti, MQPUT ya da MQPUT1 çağrısında belirtilen **MSGDSC** değiştirilmesinde bulunan kimlik bağlamı bilgilerini verir. İleti bağlamına ilişkin daha fazla bilgi için bkz. [İleti bağlamı ve Bağlam bilgilerini denetleme](#).

Bu seçenek, bu nedenle belirtilmemesi gereken OOPASI ' yi belirtir. OOUT seçeneği belirtilmelidir.

Bu seçenek, dağıtım listeleri de içinde olmak üzere, tüm kuyruk tipleri için geçerlidir.

### **OOSSETA**

Tüm bağlamın ayarlanmasına izin verir.

Bu, bir kuyruğa ilişkin bir ileti konduğunda, PMSETA seçeneğinin **PMO** parametresinde belirtilmesine olanak tanır; bu ileti, iletiyi MQPUT ya da MQPUT1 çağrısında belirtilen **MSGDSC** değiştirilmesinde bulunan kimlik ve başlangıç bağlamı bilgilerini verir. İleti bağlamına ilişkin daha fazla bilgi için bkz. [İleti bağlamı ve Bağlam bilgilerini denetleme](#).

Bu seçenek, bu nedenle belirtilmemesi gereken aşağıdaki seçenekleri belirtir:

- OOPASI
- OOPASA
- OOSSETI

OOUT seçeneği belirtilmelidir.

Bu seçenek, dağıtım listeleri de içinde olmak üzere, tüm kuyruk tipleri için geçerlidir.

**Diğer seçenekler:** Aşağıdaki seçenekler yetkilendirme denetlemesini denetler ve kuyruk yöneticisi susturulduğunda ne olur:

### **OOALTU**

Belirtilen kullanıcı kimliğiyle doğrulanıyor.

Bu, **OBJDSC** parametresindeki *ODAU* alanının, bu MQOPEN çağrısını doğrulamak için kullanılacak bir kullanıcı kimliği içerdiğini gösterir. Bu çağrı yalnızca, uygulamanın çalışmakta olduğu kullanıcı kimliğinin bu işlemi gerçekleştirme yetkisine sahip olup olmadığına bakılmaksızın, bu *ODAU* nesnesi belirtilen erişim seçenekleriyle nesneyi açma yetkisine sahip olduğunda başarılı olabilir. Ancak, her zaman uygulamanın çalışmakta olduğu kullanıcı kimliği ile karşılaştırılan bağlam seçenekleri için bu seçenek geçerli değildir.

Bu seçenek, tüm nesne tipleri için geçerlidir.

### **OOFIQ**

Kuyruk yöneticisi susturulmuş durumda olduğunda başarısız olun.



Bu seçenek, kuyruk yöneticisi susturma durumundaysa, MÇOPEN çağrısını başarısız olacak şekilde zorlar.

Bu seçenek, tüm nesne tipleri için geçerlidir.

### **00RLÇ**

Açılan yerel kuyruğun adını girin.

Bu seçenek, MÇOD yapısındaki ResolvedQName ' in (varsa), açılan yerel kuyruğun adıyla girileceğini belirtir. ResolvedQMgrAdı benzer şekilde, yerel kuyruk bulunduran yerel kuyruk yöneticisinin adıyla da girilir.

<i>Çizelge 751. Her kuyruk tipi için geçerli MÇOPER seçenekleri</i>						
<b>Seçenek</b>	<b>Diğer Ad (<u>"1" sayfa</u> <u>1310</u>)</b>	<b>Yerel ve Model</b>	<b>Uzak</b>	<b>Yerel Olmayan Küme</b>	<b>Dağıtım listesi</b>	<b>Konu</b>
00INPÇ	✓	✓	-	-	-	-
00INPS	✓	✓	-	-	-	-
00INPX	✓	✓	-	-	-	-
00BRW	✓	✓	-	-	-	-
00OUT	✓	✓	✓	✓	✓	✓
00INÇ	✓	✓	<u>"2" sayfa</u> <u>1310</u>	✓	-	-
00SET	✓	✓	<u>"2" sayfa</u> <u>1310</u>	-	-	-
00BNDO ( <u>"3" sayfa</u> <u>1310</u> )	✓	✓	✓	✓	✓	-
00BNDN ( <u>"3" sayfa</u> <u>1310</u> )	✓	✓	✓	✓	✓	-
00BNDÇ ( <u>"3" sayfa</u> <u>1310</u> )	✓	✓	✓	✓	✓	-
00SAVA	✓	✓	-	-	-	-
00PASI	✓	✓	✓	✓	✓	<u>"5" sayfa</u> <u>1310</u>
00PASA	✓	✓	✓	✓	✓	<u>"5" sayfa</u> <u>1310</u>
00SETI	✓	✓	✓	✓	✓	<u>"5" sayfa</u> <u>1310</u>
00SETA	✓	✓	✓	✓	✓	<u>"5" sayfa</u> <u>1310</u>
00ALTU	✓	✓	✓	✓	✓	✓
00FIÇ	✓	✓	✓	✓	✓	✓
00RLÇ	✓	✓	✓	✓	-	-

## Notlar:

1. Diğer adlara ilişkin seçeneklerin geçerliliği, diğer adın çözdüğü kuyruğa ilişkin seçeneğin geçerliliğine bağlıdır.
2. Bu seçenek yalnızca uzak kuyruğun yerel tanımlaması için geçerlidir.
3. Bu seçenek herhangi bir kuyruk tipi için belirlenebilir; ancak, kuyruk bir küme kuyruğu değilse yoksayılır.
4. Bu öznitelik bir konu için yoksayıldı.
5. Bu öznitelikler bir konu ile birlikte kullanılabilir, ancak yalnızca alıkonan ileti için bağlam kümesini etkiler, herhangi bir aboneye gönderilen bağlam alanlarını etkilemez.

## HOBJ (10 basamaklı işaretli tamsayı)-çıkış

Nesne tanıtıcısı.

Bu tanıtıcı, nesne için kurulmuş olan erişimi gösterir. Nesne üzerinde işlem yapan sonraki ileti kuyruklama çağrılarında bu değer belirtilmelidir. MQCLOSE çağrısı yayınlandığında ya da tutamaç işleminin kapsamını tanımlayan işlem birimi sonlandığında geçerli olmak üzere sona erer.

Tutamaç kapsamı, en küçük birimle sınırlıdır Uygulamanın çalıştığı altyapı tarafından desteklenen koşut işleme; tanıtıcı, MQOPEN çağrısının yayınlandığı koşut işleme birimi dışında geçerli değil:

- IBM üzerinde, tutamaç kapsamı, aramayı yayınlayan işidir.

## CMPCOD (10 basamaklı işaretli tamsayı)-çıkış

Tamamlanma kodu.

Bu, aşağıdakilerden biridir:

### CCOK

İşlem başarıyla tamamlandı.

### CCWARN

Uyarı (kısmi tamamlama).

### CCCFAIL

Arama başarısız oldu.

## RPG Bildirimi

```
C*.1.....2.....3.....4.....5.....6.....7..
C          CALLP      MQOPEN(HCONN : OBJDSC : OPTS :
C          HOBJ : CMPCOD : REASON)
```

Çağrıya ilişkin prototip tanımı:

```
D*.1.....2.....3.....4.....5.....6.....7..
DMQOPEN      PR          EXTPROC('MQOPEN')
D* Connection handle
D HCONN          10I 0 VALUE
D* Object descriptor
D OBJDSC          468A
D* Options that control the action of MQOPEN
D OPTS          10I 0 VALUE
D* Object handle
D HOBJ          10I 0
D* Completion code
D CMPCOD          10I 0
D* Reason code qualifying CMPCOD
D REASON          10I 0
```

MQPUT çağırısı bir ileti, dağıtım listesi ya da bir konuya ileti koyar. Kuyruk, dağıtım listesi ya da konu zaten açık olmalıdır.

- [“Sözdizimi” sayfa 1311](#)
- [“Kullanım notları” sayfa 1311](#)
  - [“Konular” sayfa 1311](#)
  - [“MQPUT ve MQPUT1” sayfa 1312](#)
  - [“Hedef kuyruklar” sayfa 1312](#)
  - [“Dağıtım listeleri” sayfa 1313](#)
  - [“Üst Bilgiler” sayfa 1314](#)
  - [“Arabellek” sayfa 1315](#)
- [“Parametreler” sayfa 1315](#)
- [“RPG Bildirimi” sayfa 1320](#)

## Sözdizimi

MQPUT (*HCONN, HOBJ, MSGDSC, PMO, BUFLN, BUFFER, CMPCOD, REASON*)

## Kullanım notları

### Konular

Konuların kullanımı için aşağıdaki notlar geçerlidir:

1. Bir konuya ilişkin iletileri yayınlamak için MQPUT kullanılırken, bir ya da daha çok abonenin abone kuyrukları (örneğin, dolu olduğu gibi) ile ilgili bir sorun nedeniyle yayın verilemez, MQPUT çağırısına döndürülen neden kodu ve teslim davranışı KONU üzerindeki PMSGDLV ya da NPMSGDLV özniteliklerinin ayarına bağlıdır. RODLQ belirtildiğinde ya da RODISC belirtildiğinde iletinin atılması, iletinin başarılı bir şekilde teslim edilmesi olarak kabul edildiğinde, bir yayının ölü mektup kuyruğuna teslimi olduğunu unutmayın. Yayınların hiçbiri teslim edilmezse, MQPUT RC2502 ile dönecektir. Bu durum aşağıdaki durumlarda oluşabilir:

- Bir ileti, PMSGDLV ya da NPMSGDLV (iletinin kalıcısına bağlı olarak) ALL ve herhangi bir aboneliğe (dayanıklı ya da değil) ilişkin olarak, yayını alamayacak bir kuyruğa sahip bir KONU için yayınlanır.
- Bir ileti, PMSGDLV ya da NPMSGDLV (iletinin kalıcısına bağlı olarak) değeri ALLUR ile bir TOPIC 'e yayınlanır ve kalıcı bir abonelik, yayını alamayacak bir kuyruğa sahiptir.

Yayınlara bazı abonelere aşağıdaki durumlarda teslim edilememesine rağmen, MQPUT RCT ile geri dönebiliyor:

- Bir ileti, PMSGDLV ya da NPMSGDLV (iletinin kalıcısına bağlı olarak) ALLAVAIL değerine ayarlanmış ve herhangi bir abonelik, dayanıklı ya da değil, yayını alamayacak bir kuyruğa sahip bir KONU için yayınlanır.
- Bir ileti, PMSGDLV ya da NPMSGDLV (iletinin kalıcısına bağlı olarak) değeri ALLUR ile bir TOPIC 'e yayınlanır ve kalıcı olmayan bir aboneliğe, yayını alamayan bir kuyruk vardır.

2. Bu konuya abone olmayan bir abone yoksa, yayınlanan ileti herhangi bir kuyruğa gönderilmez ve atılır. Bu iletinin kalıcı mı, kalıcı olmayan mı, yoksa sınırsız süre bitimi mi, yoksa küçük bir süre bitimi mi olduğu, abonesi yoksa yine de atılır mı, hiçbir fark etmez. Bu durumda, ileti alıkonacaksa, bu durumda, herhangi bir abonenin kuyruklarına gönderilmemiş olsa da, yeni aboneliklere ya da MQSUBRQ kullanılarak alıkonan yayınlar için tutulan yayınlara teslim edilecek herhangi bir aboneye teslim edilmeleri için bu iletinin saklanmasına ilişkin bir sorun yoktur.

## MQPUT ve MQPUT1

İletileri bir kuyruğa koymak için hem MQPUT hem de MQPUT1 çağrılarını kullanılabılır; bu çağrılar, koşullara bağlı olarak değişir.

- Birden çok ileti *aynı* kuyruğa konduğunda, MQPUT çağrısının kullanılması gerekir.

Önce OOOOUT seçeneğini belirten bir MQOPEN çağrısı yayınlanır, ardından kuyruğa ileti eklemek için bir ya da daha çok MQPUT isteği gönderilir; sonunda kuyruk bir MQCLOSE çağrısıyla kapatılır. Bu, MQPUT1 çağrısının yinelenen kullanımından daha iyi başarımlar sağlar.

- Bir kuyruğa yalnızca *bir* ileti konabildiğinde, MQPUT1 çağrısı kullanılmalıdır.

Bu çağrı, MQOPER, MQPUT ve MQCLOSE çağrılarını tek bir çağrıya sarkıyor ve verilmesi gereken çağrılarının sayısını en aza indiriyor.

## Hedef kuyruklar

Bir uygulama, ileti grupları kullanılmadan aynı kuyruğa ileti dizisi koyarsa, Aşağıdaki koşullar yerine getirilirse, bu iletilerin sırası korunur. Bazı koşullar hem yerel, hem de uzak hedef kuyruklar için geçerlidir; diğer koşullar yalnızca uzak hedef kuyruklar için geçerlidir.

### Yerel ve uzak hedef kuyruklara ilişkin koşullar

- MQPUT çağrılarının tümü aynı iş birimi içinde ya da bunların hiçbiri bir iş birimi içinde değil.

İletiler tek bir iş birimi içinde belirli bir kuyruğa konduğunda, diğer uygulamalardan gelen iletiler kuyruklardaki ileti dizisiyle birlikte gösterilebilir.

- MQPUT çağrılarının tümü, aynı nesne tanıtıcısı *HOB*Jkullanılarak yapılır.

Bazı ortamlarda, aynı uygulamadan gelen çağrılar sağlandığında, farklı nesne tanıtıcıları kullanıldığında ileti sırası da korunur. "aynı uygulama" ın anlamı, çevre tarafından belirlenir:

– IBM üzerinde, uygulama işidir.

- İletilerin hepsi aynı önceliğe sahip.

### Uzak hedef kuyruklara ilişkin ek koşullar

- Kuyruk yöneticisinden hedef kuyruk yöneticisine yalnızca bir yol var.

Sıradaki bazı iletilerin farklı bir yolda gidebileceği bir olasılık varsa (örneğin, yeniden yapılandırma, trafik dengeleme ya da ileti büyüklüğüne dayalı yol seçimi nedeniyle), hedef kuyruk yöneticisinde iletilerin sırası garanti edilemez.

- İletiler, gönderme, ara düzey ya da hedef kuyruk yöneticisinde geçici olarak ölü harf kuyruklarına yerleştirilmedi.

İletilerden biri ya da daha fazlası geçici olarak bir ölü-mektup kuyruğuna konursa (örneğin, bir iletim kuyruğu ya da hedef kuyruğu geçici olarak dolu olduğu için), iletiler hedef kuyruğa sıra dışında gelebilir.

- İletiler, kalıcı ya da kalıcı olmayan tüm iletiler.

Gönderme ve hedef kuyruk yöneticileri arasındaki rotadaki bir kanal, **CDNPM** özneliği NPFASST değerine ayarlıysa, kalıcı olmayan iletiler, kalıcı iletilerin önüne atlayabilir ve kalıcı iletilerin, kalıcı olmayan iletilere göre korunmamasına neden olur. Ancak, birbiriyle görelî kalıcı iletilerin sırası ve birbiriyle görelî kalıcı olmayan iletilerin sırası korunur.

Bu koşullar yerine getirilmezse, ileti grupları ileti sırasını korumak için kullanılabılır; ancak, bu durumun hem gönderme hem de alma uygulamalarının ileti gruplama desteğini kullanmasını gerektirdiğini unutmayın. İleti gruplarına ilişkin ek bilgi için aşağıdaki başlıklara bakın:

- MQMD ' de *MDMFL* alanı
- MQPMO ' da *PMLOGO* seçeneği
- MQGMO ' da *GMLOGO* seçeneği

## Dağıtım listeleri

Aşağıdaki notlar dağıtım listelerinin kullanımı için geçerlidir.

1. İletiler, bir version-1 ya da bir version-2 MQPMO kullanılarak dağıtım listesine konabilir. Bir version-1 MQPMO kullanılırsa (ya da *PMREC* ile sıfıra eşit bir version-2 MQPMO değeri varsa), uygulama tarafından hiçbir ileti kaydı ya da yanıt kaydı sağlanmaz. Bu, ileti dağıtım listesindeki bazı kuyruklara başarılı bir şekilde gönderildiyse ve başka bir ileti göndermezse, hatalarla karşılaşan kuyrukların saptanması anlamına gelmeyecektir.

İleti kayıtları ya da yanıt kayıtları uygulama tarafından sağlandıysa, *PMVER* alanının *PMVER2* olarak ayarlanması gerekir.

A version-2 MQPMO can also be used to send messages to a single queue that is not in a distribution list, by ensuring that *PMREC* is zero.

2. Tamamlanma kodu ve neden kodu değiştirgeleri aşağıdaki gibi ayarlanır:

- Dağıtım listesindeki kuyruklara başarılı olan ya da aynı şekilde başarısız olursa, tamamlanma kodu ve neden kodu değiştirgeleri ortak sonucu tanımlamaya ayarlanır. Bu durumda, MQRR yanıt kayıtları (uygulama tarafından sağlandıysa) belirlenmez.

Örneğin, her put başarılı olursa, tamamlanma kodu CCOK olarak ayarlanır ve neden kodu RCNONE olur; her put başarısız olursa, tüm kuyruklar engellenir; değiştirgeler CCCFAIL ve RC2051 olarak ayarlanır.

- Dağıtım listesindeki kuyruklara koyma işlemi başarılı olamazsa ya da aynı şekilde başarısız olursa:
  - En az bir put başarılı olursa, tamamlanma kodu parametresi CCWARN değerine ayarlanır ve tümü başarısız olursa CCFAIL ' e ayarlanır.
  - Neden kodu parametresi RC2136 olarak ayarlıdır.
  - Yanıt kayıtları (uygulama tarafından sağlandıysa), dağıtım listesindeki kuyruklara ilişkin tek tek tamamlama kodlarına ve neden kodlarına ayarlanmışsa.

Hedef için açık olan hedef başarısız olduğu için bir hedefe koyma başarısız olursa, yanıt kaydındaki alanlar CCFAIL ve RC2137; olarak ayarlanır ve bu hedef *PMIDC* ' ta yer alır.

3. Dağıtım listesindeki bir hedef yerel bir kuyruğa çözümlerse, ileti o kuyruğa normal biçimde yerleştirilir (yani, dağıtım listesi iletisi olarak değil). Birden çok hedef aynı yerel kuyruğa giderilirse, kuyruğa her hedef için bir ileti yerleştirilir.

Dağıtım listesindeki bir hedef uzak bir kuyruğa giderilirse, uygun iletim kuyruğuna bir ileti yerleştirilir. Birden çok hedef, aynı iletim kuyruğuna çözüldüğü yerlerde, bu hedefleri içeren tek bir dağıtım listesi iletisi, uygulama tarafından sağlanan hedefler listesinde bitişik olmasa da iletim kuyruğuna yerleştirilebilir. Ancak, bu işlem yalnızca iletim kuyruğu dağıtım listesi iletilerini destekliyorsa yapılabilir ( "Kuyruklara ilişkin öznitelikler" sayfa 1345 ' de açıklanan **DistLists** kuyruğu özneliğine bakın).

İletim kuyruğu dağıtım listelerini desteklemiyorsa, iletinin olağan biçimde bir kopyası, iletim kuyruğunu kullanan her hedef için iletim kuyruğuna yerleştirilir.

Uygulama iletisi verilerini içeren bir dağıtım listesi iletim kuyruğu için çok büyükse, dağıtım listesi iletisi daha az sayıda hedef içeren daha küçük dağıtım listesi iletilerine bölünmesi gerekir. Uygulama iletisi verileri yalnızca kuyruğa sığarsa, dağıtım listesi iletileri hiç kullanılamaz ve kuyruk yöneticisi, bu iletim kuyruğunu kullanan her hedef için iletinin olağan biçimde bir kopyasını oluşturur.

Farklı hedeflerde farklı ileti önceliği ya da ileti kalıcılığı varsa (uygulama PRQDEF ya da PEQDEF ' yi belirtiyorsa bu durum oluşabilir), iletiler aynı dağıtım listesi iletisinde tutulmaz. Bunun yerine, kuyruk yöneticisi, farklı öncelik ve kalıcılık değerlerini barındırmak için gereken sayıda dağıtım listesi iletisi oluşturur.

4. Dağıtım listesine bir kontak aşağıdakine neden olabilir:

- Tek bir dağıtım listesi iletisi ya da
- Daha küçük dağıtım listesi iletileri ya da
- Dağıtım listesi iletilerinin ve normal iletilerin bir karışımı ya da

- Yalnızca normal iletiler.

Önceki hangilerinde aşağıdakilerden hangisi söz eder:

- Listedeki hedefler yerel, uzak ya da bir karışımdır.
- Hedefler aynı ileti önceliğine ve ileti kalıcılarına sahiptir.
- İletim kuyrukları, dağıtım listesi iletilerini tutabilirler.
- İletim kuyruklarının üst sınır ileti uzunluğu, dağıtım listesi formundaki iletiyi sığdırabilmek için yeterli büyüklükte.

Ancak, yukarıdaki durumda ne olursa olsun, sonuçta ortaya çıkan her *fiziksel* ileti (yani, her normal ileti ya da dağıtım listesi iletisinin ortaya konması), aşağıdaki durumlarda yalnızca *bir* ileti olarak sayılır:

- Uygulamanın bir iş birimindeki izin verilen ileti sayısı üst sınırını aştığını kontrol etmek (**MaxUncommittedMsgs** kuyruk yöneticisi özneliğine bakın).
- Tetikleme koşullarının yerine getirilip karşılanmadığı denetleniyor.
- Kuyruk derinliklerinin artırılması ve kuyrukların kuyruk derinliği üst sınırının aşılıp aşılmayacağını kontrol etmek.

5. Bir tanıtıcıyı geçersiz hale getirmesine neden olacak kuyruk tanımlamalarında yapılan herhangi bir değişiklik, kuyrukların tek tek açılmasına neden oldu (örneğin, çözüm yolundaki bir değişiklik), dağıtım listesi tutamaçlarının geçersiz olmasına neden olmaz. Ancak, sonraki bir MQPUT çağrısında dağıtım listesi tanıtıcısı kullanıldığında, bu kuyruğun başarısızlığa neden olması da sonuçlanabilir.

## Üst Bilgiler

Bir ileti, uygulama iletisi verilerinin başına bir ya da daha çok IBM MQ üstbilgi yapısıyla konursa, kuyruk yöneticisi, geçerli olduğunu doğrulamak için üstbilgi yapılarında bazı denetimleri gerçekleştirir. Kuyruk yöneticisi bir hata saptarsa, arama uygun bir neden kodunda başarısız olur. Gerçekleştirilen denetimler, var olan yapılara göre değişiklik gösterir. Buna ek olarak, denetimlerde yalnızca, MQPUT ya da MQPUT1 çağrısında bir version-2 ya da üstü bir MQMD kullanılıyorsa, uygulama iletisi verilerinin başında bir MQMDE varsa bile, denetimler gerçekleştirilmezse, denetimler gerçekleştirilmez.

Şu IBM MQ üstbilgi yapıları, tam olarak kuyruk yöneticisi tarafından doğrulanır: MQDH, MQMDE.

Diğer IBM MQ üstbilgi yapıları için, kuyruk yöneticisi bazı geçerlilik denetimi gerçekleştirir, ancak her alanı denetmez. Yerel kuyruk yöneticisi tarafından desteklenmeyen yapılar ve iletteki ilk MQDLH ' nin ardından yapılarda geçerlilik denetimi yapılmaz.

IBM MQ yapılarındaki alanları genel denetlemelere ek olarak, aşağıdaki koşulların yerine getirilmesi gerekir:

- Bir IBM MQ yapısı, iki ya da daha fazla kesim üzerinden bölünmemelidir-yapının tamamen tek bir kesim içinde olması gerekir.
- Bir PCF iletisinde yapıların uzunluklarının toplamı, MQPUT ya da MQPUT1 çağrısında **BUFLEN** parametresiyle belirtilen uzunluğa eşit olmalıdır. Bir PCF iletisi, aşağıdaki biçim adlarından birine sahip bir iletidir:
  - FMADMN
  - FMEVNT
  - FMPCF
- Kesilmiş yapıların izin verildiği durumlar dışında,IBM MQ yapılarının kesilmemesi gerekir:
  - Rapor iletileri olan iletiler.
  - PCF iletileri.
  - MQDLH yapısı içeren iletiler. (Yapılar *takip edilen* ilk MQDLH kesilebilir; MQDLH ' den önceki yapılar olamaz.)

## Arabellek

RPG programlama örneğinde gösterilen **BUFFER** parametresi bir dizgi olarak bildirilir; bu parametre, parametrenin uzunluk üst sınırını 256 byte olarak sınırlandırır. Daha büyük bir arabellek gerekiyorsa, parametre, bir yapı olarak ya da fiziksel bir dosyada bir alan olarak bildirilmiş olmalıdır. Bu, olası uzunluk üst sınırını yaklaşık 32 KB olarak yükseltecektir.

## Parametreler

MQPUT çağrısında şu deęiřtirgeler vardır:

### HCONN (10 basamaklı iřaretli tamsayı)-giriř

Baęlantı tanıtıcısı.

Bu tanıtıcı, kuyruk yöneticisiyle olan baęlantıyı gösterir. The value of *HCONN* was returned by a previous MQCONN or MQCONNX call.

### HOBJ (10 basamaklı iřaretli tamsayı)-giriř

Nesne tanıtıcısı.

Bu tanıtıcı, iletinin eklendięi kuyruęu ya da iletinin yayınlandığı konuyu gösterir. The value of *HOBJ* was returned by a previous MQOPEN call that specified the OOOUT option.

### MSGDSC (MQMD)-giriř/çıkıř

İleti tanımlayıcısı.

Bu yapı, gönderilmekte olan iletinin özniteliklerini tanımlar ve koyma isteęi tamamlandıktan sonra iletiyle ilgili bilgileri alır. Ayrıntılar için bkz. [“MQMD \(Message descriptor\) on IBM i” sayfa 1092.](#)

If the application provides a version-1 MQMD, the message data can be prefixed with an MQMDE structure in order to specify values for the fields that exist in the version-2 MQMD but not the version-1. MQMD 'deki *MDFMT* alanının, bir MQMDE' nin var olduğunu göstermek için FMMDE olarak ayarlanması gerekir. Daha ayrıntılı bilgi için bkz. [“MQMDE \(Message descriptor extension\) on IBM i” sayfa 1134 .](#)

### PMO (MQPMO)-giriř/çıkıř

MQPUT iřleminin denetlenmesini denetleyen seçenekler.

Ayrıntılar için bkz. [“MQPMO \(Put-message options\) on IBM i” sayfa 1154.](#)

### BUFLEN (10 basamaklı iřaretli tamsayı)-giriř

*BUFFER* içindeki iletinin uzunluęu.

Sıfır geçerli ve iletinin herhangi bir uygulama verisi içermedięi anlamına gelir. *BUFLEN* için üst sınır çeřitli etmenlere baęlıdır:

- Hedef kuyruk paylaşılan bir kuyruksa, üst sınır 63 KB ' dir (64 512 bayt).
- Hedef yerel bir kuyruksa ya da yerel bir kuyruęa çözüyorsa (ancak, paylaşılan bir kuyruk deęilse), üst sınır ařağıdakilerden biri olup olmadığına baęlıdır:
  - Yerel kuyruk yöneticisi kesimlere ayırma özelliğini destekler.
  - Gönderme uygulaması, kuyruk yöneticisinin iletiyi bölümlemesine izin veren iřareti belirtir. Bu iřaret MFSEGA 'dır ve bir version-2 MQMD' de ya da version-1 MQMD ile kullanılan bir MQMDE içinde belirtilebilir.

Bu kořulların her ikisi de karřılanırsa, *BUFLEN* , MQMD ' deki *MDOFF* alanının deęeri ile 999 999 999 'u geçemez. Koyulabilecek en uzun mantıksal ileti, bu nedenle 999 999 bayttır ( *MDOFF* sıfır olduęunda). Ancak, uygulamanın çalıştırıldığı iřletim sistemi ya da ortam tarafından uygulanan kaynak kısıtlamaları daha düşük bir sınırla sonuçlanabilir.

Daha önce açıklanan koşullardan biri ya da her ikisi karşılanmazsa, *BUFLen* , kuyruğun **MaxMsgLength** özniteliğinin ve kuyruk yöneticisinin **MaxMsgLength** özniteliğinin daha küçük değerini geçemez.

- Hedef uzak bir kuyruksa ya da uzak bir kuyruğa çözülsürse, yerel kuyruklara ilişkin koşullar geçerli olur; *ancak, hedef kuyruğa ulaşmak için iletinin geçmesi gereken her bir kuyruk yöneticisinde ;* özellikle:

1. Yerel kuyruk yöneticisinde, iletiyi geçici olarak saklamak için kullanılan yerel iletim kuyruğu.
2. Ara iletim kuyrukları (varsa), yerel ve hedef kuyruk yöneticileri arasındaki rotadaki kuyruk yöneticilerindeki iletiyi saklamak için kullanılır.
3. Hedef kuyruk yöneticisinde hedef kuyruk

Bu nedenle, en uzun ileti bu kuyruklar ve kuyruk yöneticilerinin en kısıtlayıcı tarafından yönetilir.

İleti bir iletim kuyruğunda olduğunda, ek bilgiler ileti verileriyle birlikte bulunur ve bu işlem, yürütülebilecek uygulama verileri miktarını azaltır. Bu durumda, *BUFLen* için sınır belirlenirken LNMHD baytların iletim kuyruklarının *MaxMsgLength* değerlerinden çıkarılmasına neden olur.

**Not:** İleti konduğunda, yalnızca 1. koşula uymamanın zamanuyumlu olarak (neden kodu RC2030 ya da RC2031 ile) tanısı konabilir. 2 ya da 3 numaralı koşullar karşılanmazsa, ileti bir ara kuyruk yöneticisinde ya da hedef kuyruk yöneticisinde bir ölü-mektup (teslim edilemeyen ileti) kuyruğuna yeniden yönlendirilir. Bu gerçekleşirse, gönderici tarafından istendiye bir rapor iletisi oluşturulur.

### **BUFFER (1-byte bit dizgi x BUFLen)-giriş**

İleti verileri.

Bu, gönderilecek uygulama verilerini içeren bir arabelleğidir. Arabellek, iletteki verilerin niteine uygun bir sınırla hizalanmalıdır. 4 baytlık hizalama çoğu ileti için uygun olmalıdır ( MQ üstbilgi yapılarını içeren iletiler de içinde olmak üzere), ancak bazı iletiler daha sıkı hizalamada gerektirebilir. Örneğin, 64 bitlik bir ikili tamsayı içeren bir ileti 8 byte 'lık hizalama gerektirebilir.

*BUFFER* , karakter verilerini, sayısal verileri ya da her ikisini de içeriyorsa, **MSGDSC** parametresindeki *MDCSI* ve *MDENC* alanları, verilere uygun değerlere ayarlanmalıdır; bu işlem, iletinin alıcısının verileri (gerekirse) karakter takımı ve alıcı tarafından kullanılan kodlamaya dönüştürmesini sağlar.

**Not:** MQPUT çağrısındaki diğer tüm değıştirgelerin, **CodedCharSetId** kuyruk yöneticisi özniteliği tarafından verilen karakter kümesinde olması ve ENNAT tarafından verilen yerel kuyruk yöneticisinin kodlanması gerekir.

### **CMPCOD (10 basamaklı işaretili tamsayı)-çıkış**

Tamamlanma kodu.

Bu, aşağıdakilerden biridir:

#### **CCOK**

İşlem başarıyla tamamlandı.

#### **CCWARN**

Uyarı (kısmi tamamlama).

#### **CCCFAIL**

Arama başarısız oldu.

### **REASON (10 basamaklı işaretili tamsayı)-çıkış**

Neden kodu ön eleme *CMPCOD*.

*CMPCOD* CCOK ise:

#### **YOK**

(0, X'000 ') Raporlamak için bir neden yok.

*CMPCOD* CCWARN ise:

#### **RC2104**

(2104, X'838 ') İleti tanımlayıcısında rapor seçeneği tanınmadı.



**RC2136**

(2136, X'858 ') Birden çok neden kodu döndürüldü.

CMPCOD CCFAIL ise:

**RC2004**

(2004, X'7D4') Arabellek parametresi geçerli değil.

**RC2005**

(2005, X'7D5') Arabellek uzunluğu parametresi geçerli değil.

**RC2009**

(2009, X'7D9') Kuyruk yöneticisine yönelik bağlantı kaybedildi.

**RC2013**

(2013, X'7DD') Süre bitim zamanı geçerli değil.

**RC2014**

(2014, X'7DE') Geribildirim kodu geçerli değil.

**RC2018**

(2018, X'7E2') Bağlantı tanıtıcısı geçerli değil.

**RC2019**

(2019, X'7E3') Nesne tanıtıcısı geçerli değil.

**RC2024**

(2024, X'7E8') Yürürlükteki iş birimi içinde başka ileti işlenemez.

**RC2026**

(2026, X'7EA') İleti tanımlayıcısı geçerli değil.

**RC2027**

(2027, X'7EB') Yanıtın gönderileceği kuyruk.

**RC2029**

(2029, X'7ED') İleti tanımlayıcısında ileti tipi geçerli değil.

**RC2030**

(2030, X'7EE') İleti uzunluğu, kuyruk için üst sınırdan büyük.

**RC2031**

(2031, X'7EF') İleti uzunluğu kuyruk yöneticisi için üst sınırdan büyük.

**RC2039**

(2039, X'7F7') Kuyruk çıkış için açık değil.

**RC2041**

(2041, X'7F9') Nesne tanımlaması açıldığından beri değiştirildi.

**RC2046**

(2046, X'7FE') Seçenekler geçerli değil ya da tutarlı değil.

**RC2047**

(2047, X'7FF') Persistence geçerli değil.

**RC2048**

(2048, X'800 ') Kuyruk kalıcı iletileri desteklemiyor.

**RC2050**

(2050, X'802 ') İleti önceliği geçerli değil.

**RC2051**

(2051, X'803 ') Kuyruk araması engellenmiş olarak çağrılıyor.

**RC2052**

(2052, X'804 ') Kuyruk silinmiştir.

**RC2053**

(2053, X'805 ') Kuyruk, ileti sayısı üst sınırını zaten içeriyor.

**RC2056**

(2056, X'808 ') Kuyruğun diskte kullanılabilir alan yok.

**RC2058**

(2058, X'80A') Kuyruk yöneticisi adı geçerli değil ya da bilinmiyor.

**RC2059**

(2059, X'80B') Kuyruk yöneticisi bağlantı için kullanılabilir değil.

**RC2061**

(2061, X'80D') İleti tanımlayıcısındaki rapor seçenekleri geçerli değil.

**RC2071**

(2071, X'817 ') Kullanılabilir bellek yetersiz.

**RC2072**

(2072, X'818 ') Syncpoint desteği yok.

**RC2093**

(2093, X'82D') Kuyruğu, tüm bağlamın geçirilmek üzere açık değil.

**RC2094**

(2094, X'82E') Kuyruk, geçiş kimliği bağlamı için açık değil.

**RC2095**

(2095, X'82F') Tüm bağlamın ayarlanması için kuyruk açık değil.

**RC2096**

(2096, X'830 ') Kuyruk tanıtıcısı bağlamı için kuyruk açık değil.

**RC2097**

(2097, X'831 ') Bu işleme gönderme yapılan kuyruk tanıtıcısı saklama bağlamı değil.

**RC2098**

(2098, X'832 ') Kuyruk tanıtıcısı gönderme yapılan bir bağlam kullanılamaz.

**RC2101**

(2101, X'835 ') Nesne zarar gördü.

**RC2102**

(2102, X'836 ') Yetersiz sistem kaynakları var.

**RC2135**

(2135, X'857 ') Dağıtım üstbilgisi yapısı geçerli değil.

**RC2136**

(2136, X'858 ') Birden çok neden kodu döndürüldü.

**RC2137**

(2137, X'859 ') Nesne başarıyla açılmadı.

**RC2149**

(2149, X'865 ') PCF yapıları geçerli değil.

**RC2154**

(2154, X'86A') Kayıt sayısı geçerli değil.

**RC2156**

(2156, X'86C') Yanıt kayıtları geçerli değil.

**RC2158**

(2158, X'86E') İleti kaydı işaretlerinin konması geçerli değil.

**RC2159**

(2159, X'86F') İleti içeren ileti kayıtları geçerli değil.

**RC2161**

(2161, X'871 ') Kuyruk yöneticisi susturuyor.

**RC2162**

(2162, X'872 ') Kuyruk yöneticisi sona erdiriliyor.

**RC2173**

(2173, X'87D') Koyma-ileti seçenekleri yapısı geçerli değil.

**RC2185**

(2185, X'889 ') Tutarsız kalıcılık belirtimi.

**RC2188**

(2188, X'88C') Kme iŒ yk ıkıŒı tarafından aęrılan aęrı reddedildi.

**RC2189**

(2189, X'88D') Kme adının zlmesi baŒarısız oldu.

**RC2195**

(2195, X'893 ') Beklenmeyen bir hata oluŒtu.

**RC2219**

(2219, X'8AB') MQI aęrısı, nceki arama tamamlanmadan nce yeniden girildi.

**RC2241**

(2241, X'8C1') İleti grubu tamamlanmadı.

**RC2242**

(2242, X'8C2') Mantıksal ileti tamamlanmadı.

**RC2245**

(2244, X'8C5') Tutarsız iŒ birlięi-iŒ belirtimi.

**RC2248**

(2248, X'8C8') İleti tanımlayıcısı uzantısı geersiz.

**RC2249**

(2249, X'8C9') İleti iŒaretleri geerli deęil.

**RC2250**

(2250, X'8CA') İleti sıra numarası geerli deęil.

**RC2251**

(2251, X'8CB') İleti blm greli konumu geerli deęil.

**RC2252**

(2252, X'8CC') zgn uzunluk geerli deęil.

**RC2253**

(2253, X'8CD') İleti kesiminde veri uzunluęu sıfır.

**RC2255**

(2255, X'8CF') Kuyruk yneticisinin kullanması iin kullanılabilir iŒ birimi.

**RC2257**

(2257, X'8D1') MQMD' nin yanlıŒ srm saęlandı.

**RC2258**

(2258, X'8D2') Grup tanıtıcısı geerli deęil.

**RC2266**

(2266, X'8DA') Kme iŒ yk ıkıŒı baŒarısız oldu.

**RC2269**

(2269, X'8DD') Kme kaynaęı hatası.

**RC2270**

(2270, X'8DE') Kullanılabilir hedef kuyruęu yok.

**RC2420**

(2420) Bir MQPUT aęrısı yayınlandı, ancak ileti verileri geerli olmayan bir MQPH yapısı ieriyor.

**RC2479**

(2479, X'9AF') Yayın saklanamadı.

**RC2480**

(2480, X'9B0') Hedef tipi deęiŒti: Dięer ad kuyruęu bir kuyruęa gnderme yapıyor, ancak Œimdi bir konuya baŒvuruyor.

**RC2502**

(2502, X'9C6') Yayınlama baŒarısız oldu ve yayın herhangi bir aboneye teslim edilmedi

**RC2551**

(2551, X'9F7') Belirtilen seim dizgisi kullanılmıyor.

## RC2554

(2554, X'9FA') İleti içeriği, iletinin genişletilmiş ileti seçiciyle bir aboneye teslim edilip edilmeyeceğini belirlemek için ayrıştırılmadı.

## RPG Bildirimi

```
C*..1.....2.....3.....4.....5.....6.....7..
C                               CALLP      MQPUT(HCONN : HOBJ : MSGDSC : PMO :
C                               BUFLLEN : BUFFER : CMPCOD :
C                               REASON)
```

Çağrıya ilişkin prototip tanımı:

```
D*..1.....2.....3.....4.....5.....6.....7..
DMQPUT      PR                EXTPROC('MQPUT')
D* Connection handle
D HCONN          10I 0 VALUE
D* Object handle
D HOBJ          10I 0 VALUE
D* Message descriptor
D MSGDSC          364A
D* Options that control the action of MQPUT
D PMO            200A
D* Length of the message in Buffer
D BUFLLEN        10I 0 VALUE
D* Message data
D BUFFER          *   VALUE
D* Completion code
D CMPCOD          10I 0
D* Reason code qualifying CMPCOD
D REASON          10I 0
```

## IBM i IBM üzerindeMQPUT1 (Bir ileti koy)

MQPUT1 çağrısı, bir kuyruğa ya da dağıtım listesine ya da bir konuya ilişkin bir ileti koyar. Kuyruk, dağıtım listesi ya da konunun açık olması gerekmez.

- “Sözdizimi” sayfa 1320
- “Kullanım notları” sayfa 1320
- “Parametreler” sayfa 1321
- “RPG Bildirimi” sayfa 1326

## Sözdizimi

MQPUT1 (HCONN, OBJDSC, MSGDSC, PMO, BUFLLEN, BUFFER, CMPCOD, REASON)

## Kullanım notları

1. İletileri bir kuyruğa koymak için hem MQPUT hem de MQPUT1 çağrıları kullanılabilir; bu çağrılar, koşullara bağlı olarak değişir:

- Birden çok ileti aynı kuyruğa konduğunda, MQPUT çağrısının kullanılması gerekir.

Önce OOOOUT seçeneğini belirten bir MQOPEN çağrısı yayınlanır, ardından kuyruğa ileti eklemek için bir ya da daha çok MQPUT isteği gönderilir; sonunda kuyruk bir MQCLOSE çağrısıyla kapatılır. Bu, MQPUT1 çağrısının yinelenen kullanımından daha iyi başarımlar sağlar.

- Bir kuyruğa yalnızca bir ileti konabildiğinde, MQPUT1 çağrısı kullanılmalıdır.

Bu çağrı, MQOPER, MQPUT ve MQCLOSE çağrılarını tek bir çağrıya sarkıyor ve verilmesi gereken çağrıların sayısını en aza indiriyor.

2. Bir uygulama, ileti grupları kullanılmadan aynı kuyruğa ileti dizisi koyarsa, Belirli koşullar karşılanırsa, bu iletilerin sırası korunur. Ancak, çoğu ortamda MQPUT1 çağrısı bu koşulları karşılamıyor ve ileti düzenini korumuyor. MQPUT çağrısı bu ortamlarda kullanılmalı. Ayrıntılar için, MQPUT çağrısının açıklamasındaki kullanım notlarına bakın.
3. İletileri dağıtım listelerine yerleştirmek için MQPUT1 çağrısı kullanılabilir. Bu konuda genel bilgi edinmek için, MQOPER ve MQPUT çağrılarına ilişkin kullanım notlarına bakın.  
MQPUT1 çağrısı kullanılırken aşağıdaki farklar geçerlidir:
  - a. Uygulama tarafından MQRR yanıt kayıtları sağlandıysa, bunlar MQOD yapısı kullanılarak sağlanmalıdır; bunlar MQPMO yapısı kullanılarak sağlanamaz.
  - b. Yanıt kayıtlarında RC2137 neden kodu hiçbir zaman MQPUT1 tarafından döndürülmez; bir kuyruk açılmazsa, o kuyruğa ilişkin yanıt kaydı, açık işlemde kaynaklanan gerçek neden kodunu içerir.  
Bir kuyruğa ilişkin açma işlemi, CCWARN tamamlanma kodu ile başarılı olursa, bu kuyruğa ilişkin yanıt kaydındaki tamamlanma kodu ve neden kodu, koyma işleminden kaynaklanan tamamlanma ve neden kodlarıyla değiştirilir.  
As with the MQOPEN and MQPUT calls, the queue manager sets the response records (if provided) only when the outcome of the call is not the same for all queues in the distribution list; this is indicated by the call completing with reason code RC2136.
4. MQPUT1 çağrısı bir iletiyi bir küme kuyruğuna yerleştirmek için kullanılırsa, arama MQOPEN çağrısında OOBNDN belirlendiği gibi davranır.
5. Bir ileti, uygulama iletileri verilerinin başına bir ya da daha çok IBM MQ üstbilgi yapısıyla konursa, kuyruk yöneticisi, geçerli olduğunu doğrulamak için üstbilgi yapılarında bazı denetimleri gerçekleştirir. Bu konuya ilişkin ek bilgi için, MQPUT çağrısına ilişkin kullanım notlarına bakın.
6. Uyarı durumlarının birden fazlası ortaya çıkarsa ( **CMPCOD** parametresine bakın), döndürülen neden kodu, aşağıdaki listede yer alan *ilk* bir tanedir:
  - a. RC2136
  - b. RC2242
  - c. RC2241
  - d. RC2049 ya da RC2104
7. RPG programlama örneğinde gösterilen **BUFFER** parametresi bir dizgi olarak bildirilir; bu parametre, parametrenin uzunluk üst sınırını 256 byte olarak sınırlandırır. Daha büyük bir arabellek gerekiyorsa, parametre, bir yapı olarak ya da fiziksel bir dosyada bir alan olarak bildirilmiş olmalıdır. Bu, olası uzunluk üst sınırını yaklaşık 32 KB olarak yükseltecektir.

## Parametreler

MQPUT1 çağrısı aşağıdaki deęiřtirgeleri içerir:

### **HCONN (10 basamaklı iřaretli tamsayı)-giriř**

Baęlantı tanıtıcısı.

Bu tanıtıcı, kuyruk yöneticisiyle olan baęlantıyı gösterir. The value of *HCONN* was returned by a previous MQCONN or MQCONNX call.

### **OBJDSC (MQOD)-giriř/çıkıř**

Nesne tanımlayıcısı.

Bu yapı, iletinin eklendięi kuyruęu tanımlayan bir yapıdır. Ayrıntılar için bkz. "IBM i üzerinde MQOD (Nesne tanımlayıcı)" sayfa 1140.

Kullanıcı, çıkıř için kuyruęu açma yetkisine sahip olmalıdır. Kuyruk, bir model kuyruęu **olmamalıdır** .

### **MSGDSC (MQMD)-giriř/çıkıř**

İleti tanımlayıcısı.

Bu yapı, gönderilmekte olan iletinin özniteliklerini tanımlar ve koyma isteği tamamlandıktan sonra geri bildirim bilgilerini alır. Ayrıntılar için bkz. [“MQMD \(Message descriptor\) on IBM i” sayfa 1092.](#)

If the application provides a version-1 MQMD, the message data can be prefixed with an MQMDE structure in order to specify values for the fields that exist in the version-2 MQMD but not the version-1. MQMD 'deki *MDFMT* alanının, bir MQMDE' nin var olduğunu göstermek için FMMDE olarak ayarlanması gerekir. Daha ayrıntılı bilgi için bkz. [“MQMDE \(Message descriptor extension\) on IBM i” sayfa 1134 .](#)

### **PMO (MQPMO)-giriş/çıkış**

MQPUT1işleminin denetlenmesini denetleyen seçenekler.

Ayrıntılar için bkz. [“MQPMO \(Put-message options\) on IBM i” sayfa 1154.](#)

### **BUFLEN (10 basamaklı işaretli tamsayı)-giriş**

*BUFFER*içindeki iletinin uzunluğu.

Sıfır geçerli ve iletinin herhangi bir uygulama verisi içermediği anlamına gelir. Üst sınır çeşitli etkenlere bağlıdır; ek ayrıntılar için MQPUT çağrısının **BUFLEN** değıştirgesine ilişkin açıklamalara bakın.

### **BUFFER (1-byte bit dizgi x BUFLEN)-giriş**

İleti verileri.

Bu, gönderilecek uygulama iletisi verilerini içeren bir arabelleğidir. Arabellek, iletteki verilerin niteine uygun bir sınırla hizalanmalıdır. 4 baytlık hizalama çoğu ileti için uygun olmalıdır ( IBM MQ üstbilgi yapılarını içeren iletler de içinde olmak üzere), ancak bazı iletler daha sıkı hizalamada gerektirebilir. Örneğin, 64 bitlik bir ikili tamsayı içeren bir ileti 8 byte 'lık hizalama gerektirebilir.

*BUFFER* , karakter verilerini, sayısal verileri ya da her ikisini de içeriyorsa, **MSGDSC** parametresindeki *MDCSI* ve *MDENC* alanları, verilere uygun değerlere ayarlanmalıdır; bu işlem, iletinin alıcısının verileri (gerekirse) karakter takımı ve alıcı tarafından kullanılan kodlamaya dönüştürmesini sağlar.

**Not:** MQPUT1 çağrısındaki diğere parametrelerin tümü, ENNAT tarafından verilen yerel kuyruk yöneticisinin **CodedCharSetId** kuyruk yöneticisi özniteliği ve kodlaması tarafından verilen karakter kümesinde olmalıdır.

### **CMPCOD (10 basamaklı işaretli tamsayı)-çıkış**

Tamamlanma kodu.

Bu, aşağıdakilerden biridir:

#### **CCOK**

İşlem başarıyla tamamlandı.

#### **CCWARN**

Uyarı (kısmi tamamlama).

#### **CCCFAIL**

Arama başarısız oldu.

### **REASON (10 basamaklı işaretli tamsayı)-çıkış**

Neden kodu ön eleme *CMPCOD*.

*CMPCOD* CCOK ise:

#### **YOK**

(0, X'000 ') Raporlamak için bir neden yok.

*CMPCOD* CCWARN ise:

#### **RC2104**

(2104, X'838 ') İleti tanımlayıcısında rapor seçeneği tanınmadı.

#### **RC2136**

(2136, X'858 ') Birden çok neden kodu döndürüldü.

**RC2049**

(2049, X'801 ') İleti Önceliği desteklenen üst sınır değerini aşıyor.

**RC2241**

(2241, X'8C1') İleti grubu tamamlanmadı.

**RC2242**

(2242, X'8C2') Mantıksal ileti tamamlanmadı.

CMPCOD CCFAIL ise:

**RC2001**

(2001, X'7D1') Diğer ad taban kuyruğu geçerli bir tip değil.

**RC2004**

(2004, X'7D4') Arabellek parametresi geçerli değil.

**RC2005**

(2005, X'7D5') Arabellek uzunluğu parametresi geçerli değil.

**RC2009**

(2009, X'7D9') Kuyruk yöneticisine yönelik bağlantı kaybedildi.

**RC2013**

(2013, X'7DD') Süre bitim zamanı geçerli değil.

**RC2014**

(2014, X'7DE') Geribildirim kodu geçerli değil.

**RC2017**

(2017, X'7E1') Kullanılabilir başka tanıtıcı yok.

**RC2018**

(2018, X'7E2') Bağlantı tanıtıcısı geçerli değil.

**RC2024**

(2024, X'7E8') Yürürlükteki iş birimi içinde başka ileti işlenemez.

**RC2026**

(2026, X'7EA') İleti tanımlayıcısı geçerli değil.

**RC2027**

(2027, X'7EB') Yanıtın gönderileceği kuyruk.

**RC2029**

(2029, X'7ED') İleti tanımlayıcısında ileti tipi geçerli değil.

**RC2030**

(2030, X'7EE') İleti uzunluğu, kuyruk için üst sınırdan büyük.

**RC2031**

(2031, X'7EF') İleti uzunluğu kuyruk yöneticisi için üst sınırdan büyük.

**RC2035**

(2035, X'7F3') Erişim için yetkili değil.

**RC2042**

(2042, X'7FA') Nesnesi çakışan seçeneklerle zaten açık.

**RC2043**

(2043, X'7FB') Nesne tipi geçerli değil.

**RC2044**

(2044, X'7FC') Nesne tanımlayıcı yapısı geçerli değil.

**RC2046**

(2046, X'7FE') Seçenekler geçerli değil ya da tutarlı değil.

**RC2047**

(2047, X'7FF') Persistence geçerli değil.

**RC2048**

(2048, X'800 ') Kuyruk kalıcı iletileri desteklemiyor.

**RC2050**

(2050, X'802 ') İleti önceliği geçerli değil.

**RC2051**

(2051, X'803 ') Kuyruk araması engellenmiş olarak çağrılıyor.

**RC2052**

(2052, X'804 ') Kuyruk silinmiştir.

**RC2053**

(2053, X'805 ') Kuyruk, ileti sayısı üst sınırını zaten içeriyor.

**RC2056**

(2056, X'808 ') Kuyruğun diskte kullanılabilir alan yok.

**RC2057**

(2057, X'809 ') Kuyruk tipi geçerli değil.

**RC2058**

(2058, X'80A') Kuyruk yöneticisi adı geçerli değil ya da bilinmiyor.

**RC2059**

(2059, X'80B') Kuyruk yöneticisi bağlantı için kullanılabilir değil.

**RC2061**

(2061, X'80D') İleti tanımlayıcısındaki rapor seçenekleri geçerli değil.

**RC2063**

(2063, X'80F') Güvenlik hatası oluştu.

**RC2071**

(2071, X'817 ') Kullanılabilir bellek yetersiz.

**RC2072**

(2072, X'818 ') Syncpoint desteği yok.

**RC2082**

(2082, X'822 ') Bilinmeyen diğer ad temel kuyruğu.

**RC2085**

(2085, X'825 ') Bilinmeyen nesne adı.

**RC2086**

(2086, X'826 ') Bilinmeyen nesne kuyruk yöneticisi.

**RC2087**

(2087, X'827 ') Bilinmeyen uzak kuyruk yöneticisi.

**RC2091**

(2091, X'82B') İletim kuyruğu yerel değil.

**RC2092**

(2092, X'82C') İletim kuyruğu yanlış kullanıma sahip.

**RC2097**

(2097, X'831 ') Bu işleme gönderme yapılan kuyruk tanıtıcısı saklama bağlamı değil.

**RC2098**

(2098, X'832 ') Kuyruk tanıtıcısı gönderme yapılan bir bağlam kullanılamaz.

**RC2101**

(2101, X'835 ') Nesne zarar gördü.

**RC2102**

(2102, X'836 ') Yetersiz sistem kaynakları var.

**RC2135**

(2135, X'857 ') Dağıtım üstbilgisi yapısı geçerli değil.

**RC2136**

(2136, X'858 ') Birden çok neden kodu döndürüldü.

**RC2149**

(2149, X'865 ') PCF yapıları geçerli değil.



**RC2154**

(2154, X'86A') Kayıt sayısı geçerli değil.

**RC2155**

(2155, X'86B') Nesne kayıtları geçerli değil.

**RC2156**

(2156, X'86C') Yanıt kayıtları geçerli değil.

**RC2158**

(2158, X'86E') İleti kaydı işaretlerinin konması geçerli değil.

**RC2159**

(2159, X'86F') İleti içeren ileti kayıtları geçerli değil.

**RC2161**

(2161, X'871 ') Kuyruk yöneticisi susturuyor.

**RC2162**

(2162, X'872 ') Kuyruk yöneticisi sona erdiriliyor.

**RC2173**

(2173, X'87D') Koyma-ileti seçenekleri yapısı geçerli değil.

**RC2184**

(2184, X'888 ') Uzak kuyruk adı geçerli değil.

**RC2188**

(2188, X'88C') Küme iş yükü çıkışı tarafından çağrılan çağrı reddedildi.

**RC2189**

(2189, X'88D') Küme adının çözülmesi başarısız oldu.

**RC2195**

(2195, X'893 ') Beklenmeyen bir hata oluştu.

**RC2196**

(2196, X'894 ') Bilinmeyen iletim kuyruğu.

**RC2197**

(2197, X'895 ') Bilinmeyen varsayılan iletim kuyruğu.

**RC2198**

(2198, X'896 ') Varsayılan iletim kuyruğu yerel değil.

**RC2199**

(2199, X'897 ') Varsayılan iletim kuyruğu kullanım hatası.

**RC2258**

(2258, X'8D2') Grup tanıtıcısı geçerli değil.

**RC2248**

(2248, X'8C8') İleti tanımlayıcısı uzantısı geçersiz.

**RC2219**

(2219, X'8AB') MQI çağrısı, önceki arama tamamlanmadan önce yeniden girildi.

**RC2249**

(2249, X'8C9') İleti işaretleri geçerli değil.

**RC2250**

(2250, X'8CA') İleti sıra numarası geçerli değil.

**RC2251**

(2251, X'8CB') İleti bölümü görelî konumu geçerli değil.

**RC2252**

(2252, X'8CC') Özgün uzunluk geçerli değil.

**RC2253**

(2253, X'8CD') İleti kesiminde veri uzunluğu sıfır.

**RC2255**

(2255, X'8CF') Kuyruk yöneticisinin kullanması için kullanılabilecek iş birimi.

**RC2257**

(2257, X'8D1') MQMD' nin yanlış sürümü sağlandı.

**RC2266**

(2266, X'8DA') Küme iş yükü çıkışı başarısız oldu.

**RC2269**

(2269, X'8DD') Küme kaynağı hatası.

**RC2270**

(2270, X'8DE') Kullanılabilir hedef kuyruğu yok.

**RC2420**

(2420) Bir MQPUT1 çağrısı yayınlandı, ancak ileti verileri geçerli olmayan bir MQEPH yapısı içeriyor.

**RC2551**

(2551, X'9F7') Belirtilen seçim dizgisi kullanılamıyor.

**RC2554**

(2554, X'9FA') İleti içeriği, iletinin genişletilmiş ileti seçiciyle bir aboneye teslim edilip edilmeyeceğini belirlemek için ayrıştırılmadı.

**RPG Bildirimi**

```

C*.1.....2.....3.....4.....5.....6.....7..
C          CALLP      MQPUT1(HCONN : OBJDSC : MSGDSC :
C                               PMO : BUFLN : BUFFER :
C                               CMPCOD : REASON)

```

Çağrıya ilişkin prototip tanımı:

```

D*.1.....2.....3.....4.....5.....6.....7..
DMQPUT1      PR          EXTPROC('MQPUT1')
D* Connection handle
D HCONN          10I 0 VALUE
D* Object descriptor
D OBJDSC          468A
D* Message descriptor
D MSGDSC          364A
D* Options that control the action of MQPUT1
D PMO          200A
D* Length of the message in BUFFER
D BUFLN          10I 0 VALUE
D* Message data
D BUFFER          *   VALUE
D* Completion code
D CMPCOD          10I 0
D* Reason code qualifying CMPCOD
D REASON          10I 0

```

**IBM üzerinde MQSET (nesne özniteliklerini ayarla)**

MQSET çağrısı, bir tanıtıcı tarafından temsil edilen bir nesnenin özniteliklerini değiştirmek için kullanılır. Nesne bir kuyruk olmalıdır.

- [“Sözdizimi” sayfa 1326](#)
- [“Kullanım notları” sayfa 1327](#)
- [“Parametreler” sayfa 1327](#)
- [“RPG Bildirimi” sayfa 1330](#)

**Sözdizimi**

MQSET (HCONN, HOBJ, SELCNT, SELS, IACNT, INTATR, CALEN, CHRATR, CMPCOD, REASON)

## Kullanım notları

1. Bu çağrıyı kullanarak, uygulama bir tamsayı öznitelikleri dizisi ya da bir karakter özniteliği dizileri derlemi ya da her ikisini birden belirleyebilir. Herhangi bir hata ortaya çıkmazsa, belirtilen öznitelikler aynı anda ayarlanır. Bir hata oluşursa (örneğin, bir seçici geçerli değilse ya da bir özniteliği geçerli olmayan bir değere ayarlama girişiminde bulunulduysa), çağrı başarısız olur ve öznitelikler ayarlanmaz.
2. Özniteliklerin değerleri MQINQ çağrısı kullanılarak saptlanabilir; Ayrıntılar için bkz. [“MQINQ \(Inquire about object attributes\) on IBM i” sayfa 1283](#) .

**Not:** MQINQ çağrısının kullanılmasıyla ilgili olarak sorgulanabilen değerlere sahip tüm öznitelikler, MQSET çağrısını kullanarak değerlerini değiştiremez. Örneğin, bu çağrıyla hiçbir süreç nesnesi ya da kuyruk yöneticisi özniteliği ayarlanamaz.

3. Öznitelik değişiklikleri, kuyruk yöneticisinin yeniden başlatma işlemlerinde (kuyruk yöneticisinin yeniden başlatılmalarından hayatta kalmayan geçici dinamik kuyruklara ilişkin değişiklikler dışında) korunur.
4. MQSET çağrısını kullanarak, bir model kuyruğunun özniteliklerini değiştiremezsiniz. Ancak, MQOO\_SET seçeneğiyle MQOP çağrısını kullanarak bir model kuyruğu açsanız, MQSET çağrısıyla yaratılan dinamik yerel kuyruğun özniteliklerini ayarlamak için MQSET çağrısını kullanabilirsiniz.
5. Ayarlanmakta olan nesne bir küme kuyruğuna, açılacak açık için küme kuyruğunun yerel bir yönetim ortamı olmalıdır.

Nesne özniteliklerine ilişkin ek bilgi için aşağıdaki başlıklara bakın:

- [“Kuyruklara ilişkin öznitelikler” sayfa 1345](#)
- [“Ad listelerine ilişkin öznitelikler” sayfa 1373](#)
- [“IBM üzerindeki süreç tanımlamalarına ilişkin öznitelikler” sayfa 1374](#)
- [“IBM üzerinde kuyruk yöneticisine ilişkin öznitelikler” sayfa 1376](#)

## Parametreler

MQSET çağrısı aşağıdaki deęiřtirgeleri ierir:

### HCONN (10 basamaklı iřaretli tamsayı)-giriř

Baęlantı tanıtıcısı.

Bu tanıtıcı, kuyruk yöneticisiyle olan baęlantıyı gösterir. The value of HCONN was returned by a previous MQCONN or MQCONNX call.

### HOBJ (10 basamaklı iřaretli tamsayı)-giriř

Nesne tanıtıcısı.

Bu tanıtıcı, ayarlanacak özniteliklere sahip kuyruk nesnesini gösterir. Tanıtıcı değeri OOSSET (OOSSET) seçeneğini belirten önceki bir MQOPER çağrısı tarafından döndürüldü.

### SELCNT (10 basamaklı iřaretli tamsayı)-giriř

Seicilerin sayısı.

Bu, SELS dizisinde saęlanan seicilerin sayısıdır. Bu, ayarlanacak özniteliklerin sayısıdır. Sıfır geçerli bir deęerdir. İzin verilen sayı üst sınırı 256 'tır.

### SOLS (10 basamaklı iřaretli tamsayı x SELCNT)-giriř

Öznitelik seicilerinin dizisi.

Bu, **SELCNT** öznitelik seicilerinden oluşan bir dizidir; her seici, bir öznitelięi (tamsayı ya da karakter) ayarlanacak bir deęerle tanımlar.

Her seici, HOBJ ' in temsil ettięi kuyruk türü için geçerli olmalıdır. Yalnızca belirli IA\* ve CA\* deęerlerine izin verilir; bu deęerler daha sonra bu bölümde listelenir.

Seçiciler herhangi bir sırada belirtilebilir. Attribute values that correspond to integer attribute selectors (IA\* selectors) must be specified in INTATR in the same order in which these selectors occur in SELS. Karakter özniteliği seçicilere karşılık gelen öznitelik değerleri (CA\* seçiciler), bu seçicilerin olduğu sırayla CHRATR içinde belirtilmelidir. IA\* seçicileri, CA\* seçicileriyle birlikte ayrılabilir; yalnızca her tip içindeki görelî sipariş önemlidir.

Aynı seçiciyi bir kereden fazla belirtmek bir hata değildir; bu işlem yapılırsa, belirli bir seçici için belirtilen son değer, geçerli olan bir seçicidir.

**Not:**

1. Tamsayı ve karakter özniteliği seçicileri farklı iki aralık içinde ayrılır; IA\* seçicileri, IACAST ile IACAST ile CALCAST ile CA\* seçicileri CALST ile birlikte CA\* seçicileri aralığına ayrılır.

Her bir aralık için IALSTU ve CALSTU değişmezleri, kuyruk yöneticisinin kabul edeceği en yüksek değeri tanımlar.

2. Tüm IA\* seçicileri önce gerçekleşirse, aynı öge numaraları SELS ve INTATR dizilerinde karşılık gelen öğeleri adreslemek için kullanılabilir.

Ayarlanabilen öznitelikler aşağıdaki çizelgede listelenir. Bu çağrıyı kullanarak başka hiçbir öznitelik ayarlanmaz. CA\* özniteliği seçicileri için, CHRATR içinde gerekli olan dizginin bayt cinsinden uzunluğunu tanımlayan sabit parantez içinde sağlanır.

<i>Çizelge 752. Kuyruklar için MQSET öznitelik seçicileri</i>		
<b>Seçici</b>	<b>Tanım</b>	<b>Not</b>
CATRGD	Tetikleme verileri (LNTRGD).	<a href="#">“2” sayfa 1329</a>
IADIST	Dağıtım listesi desteği.	<a href="#">“1” sayfa 1328</a>
IAIGET	Alma işlemlerine izin verilip verilmediği.	
IAIPUT	Koyma işlemlerine izin verilip verilmediği.	
IATRGC	Tetik kontrolü.	<a href="#">“2” sayfa 1329</a>
ALRIGI	Tetik derinliği.	<a href="#">“2” sayfa 1329</a>
IATRGP	Tetikleyiciler için eşik iletisi önceliği.	<a href="#">“2” sayfa 1329</a>
İLAÇ	Tetikleyici tipi.	<a href="#">“2” sayfa 1329</a>

**Notlar:**

1. Yalnızca şu altyapılarda desteklenir:

-  AIX
-  IBM i
-  Windows

ve bu sistemlere bağlı IBM MQ istemcileri için geçerli olur.

2. VSE/ESA üzerinde desteklenmez.

### **IACNT (10 basamaklı işaretli tamsayı)-giriş**

Tamsayı özniteliklerinin sayısı.

Bu, INTATR dizideki öğelerin sayısıdır ve **SELS** parametresindeki IA\* seçicilerinin en az sayıda olması gerekir. Sıfır (sıfır) değeri yoksa, geçerli bir değerdir.

### **INTATR (10 basamaklı işaretli integ x rxIACNT)-giriş**

Tamsayı öznitelikleri dizisi.

Bu, IACNT tamsayı öznitelik değerlerinden oluşan bir dizidir. Bu öznitelik değerleri, SELS dizisinde IA\* seçicileriyle aynı sırada olmalıdır.

### **CALEN (10 basamaklı işaretli tamsayı)-giriş**

Karakter öznitelikleri arabelleğindeki uzunluk.

Bu, **CHRATR** parametresinin bayt cinsinden uzunluğudur ve en az SELS dizisinde belirtilen karakter özniteliklerinin uzunluklarının toplamını olmalıdır. SELS içinde CA\* seçicileri yoksa, sıfır geçerli bir değerdir.

### **CHRATR (1 byte 'lık karakter dizilimi x CALEN)-giriş**

Karakter öznitelikleri.

Bu, karakter öznitelik değerlerini içeren arabelleğidir ve birlikte bitişirilir. Arabellek uzunluğu **CALEN** parametresiyle verilir.

Karakter öznitelikleri, SELS dizisinde CA\* seçiciyle aynı sırada belirtilmelidir. Her bir karakter özniteliğinin uzunluğu sabittir (bkz. SELS). Bir öznitelik için ayarlanacak değer, özniteliğin tanımlı uzunluğundan daha az boşluk içermeyen karakter içeriyorsa, öznitelik değerinin özniteliğin tanımlı uzunlukla eşleşmesi için CHRATR içindeki değer boşlukla doldurulmalıdır.

### **CMPCOD (10 basamaklı işaretli tamsayı)-çıkış**

Tamamlanma kodu.

Bu, aşağıdakilerden biridir:

#### **CCOK**

İşlem başarıyla tamamlandı.

#### **CCCFAIL**

Arama başarısız oldu.

### **REASON (10 basamaklı işaretli tamsayı)-çıkış**

Neden kodu ön eleme CMPCOD.

CMPCOD CCOK ise:

#### **YOK**

(0, X'000 ') Raporlamak için bir neden yok.

CMPCOD CCFAIL ise:

#### **RC2219**

(2219, X'8AB') MQI çağrısı, önceki arama tamamlanmadan önce yeniden girildi.

#### **RC2006**

(2006, X'7D6') Karakter öznitelikleri uzunluğu geçerli değil.

#### **RC2007**

(2007, X'7D7') Karakter öznitelikleri dizisi geçerli değil.

#### **RC2009**

(2009, X'7D9') Kuyruk yöneticisine yönelik bağlantı kaybedildi.

#### **RC2018**

(2018, X'7E2') Bağlantı tanıtıcısı geçerli değil.

**RC2019**

(2019, X'7E3') Nesne tanıtıcısı geçerli değil.

**RC2020**

(2020, X'7E4') Inhibit-get ya da inhibit-put queue özniteliğinin değeri geçerli değil.

**RC2021**

(2021, X'7E5') Tamsayı öznitelikleri sayısı geçerli değil.

**RC2023**

(2023, X'7E7') Tamsayı öznitelikleri dizisi geçerli değil.

**RC2040**

(2040, X'7F8') Kuyruk, küme için açık değil.

**RC2041**

(2041, X'7F9') Nesne tanımlaması açıldığından beri değiştirildi.

**RC2101**

(2101, X'835 ') Nesne zarar gördü.

**RC2052**

(2052, X'804 ') Kuyruk silinmiştir.

**RC2058**

(2058, X'80A') Kuyruk yöneticisi adı geçerli değil ya da bilinmiyor.

**RC2059**

(2059, X'80B') Kuyruk yöneticisi bağlantı için kullanılabilir değil.

**RC2162**

(2162, X'872 ') Kuyruk yöneticisi sona erdiriliyor.

**RC2102**

(2102, X'836 ') Yetersiz sistem kaynakları var.

**RC2065**

(2065, X'811 ') Seçici sayısı geçerli değil.

**RC2067**

(2067, X'813 ') Öznitelik seçicisi geçerli değil.

**RC2066**

(2066, X'812 ') Seçicilerin sayısı çok büyük.

**RC2071**

(2071, X'817 ') Kullanılabilir bellek yetersiz.

**RC2075**

(2075, X'81B') Tetikleme denetimi özniteliği için değer geçerli değil.

**RC2076**

(2076, X'81C') Tetikleme derinliği özniteliği için değer geçerli değil.

**RC2077**

(2077, X'81D') Tetikleme-Message-priority özniteliği için değer geçerli değil.

**RC2078**

(2078, X'81E') Tetikleme tipi özniteliği için değer geçerli değil.

**RC2195**

(2195, X'893 ') Beklenmeyen bir hata oluştu.

**RPG Bildirimi**

```
C*..1.....2.....3.....4.....5.....6.....7..
C          CALLP      MQSET(HCONN : HOBJ : SELCNT :
C          SELS(1) : IACNT : INTATR(1) :
C          CALEN : CHRATR : CMPCOD :
C          REASON)
```

Çağrıya ilişkin prototip tanımı:

```
D*..1.....2.....:.....3.....4.....:.....5.....6.....7..
MQSET      PR      EXTPROC('MQSET')
D* Connection handle
D HCONN      10I 0 VALUE
D* Object handle
D HOBJ      10I 0 VALUE
D* Count of selectors
D SELCNT      10I 0 VALUE
D* Array of attribute selectors
D SELS      10I 0
D* Count of integer attributes
D IACNT      10I 0 VALUE
D* Array of integer attributes
D INTATR      10I 0
D* Length of character attributes buffer
D CALEN      10I 0 VALUE
D* Character attributes
D CHRATR      * VALUE
D* Completion code
D CMPCOD      10I 0
D* Reason code qualifying CMPCOD
D REASON      10I 0
```

## IBM i MQSETMP (Set message handle property) on IBM i

MQSETMP çağrı kümeleri ya da bir ileti tanıtıcısının bir özelliğini değiştirir.

- [“Sözdizimi” sayfa 1331](#)
- [“Kullanım notları” sayfa 1331](#)
- [“Parametreler” sayfa 1333](#)
- [“RPG Bildirimi” sayfa 1335](#)

### Sözdizimi

MQSETMP (*Hconn, Hmsg, SetPropOpts, Name, PropDesc, Type, ValueLength, Value, CompCode, Reason*)

### Kullanım notları

- Bu çağrıyı, yalnızca kuyruk yöneticisinin iş birimini koordine ettiğinde kullanabilirsiniz. Bu durumda şunlar olabilir:
  - Değişikliklerin yalnızca IBM MQ kaynaklarını etkilediği yerel bir iş birimi.
  - Değişikliklerin, diğer kaynak yöneticilerine ait kaynakları etkileyebileceği gibi, IBM MQ kaynaklarını etkilediği genel bir iş birimi.
- Yerel ve genel çalışma birimleriyle ilgili daha fazla ayrıntı için bkz. [“IBM üzerinde MQBEGIN \(İş birimini başlat\)” sayfa 1233](#).
- Kuyruk yöneticisinin iş birimini koordine etmediği ortamlarda, MQBACK yerine uygun geri çağırma çağrısını kullanın. Ortam, uygulamanın olağan dışı bir şekilde sonlandırılmasına neden olan örtük bir geri dönüş özelliğini de destekleyebilir.
  - z/OS' ta aşağıdaki çağrıları kullanın:
    - İş birimi yalnızca IBM MQ kaynaklarını etkilediğinde, toplu iş programları ( IMS toplu DL/I programları da içinde olmak üzere) MQBACK çağrısını kullanabilir. However, if the unit of work affects both IBM MQ resources and resources belonging to other resource managers (for example, Db2), use the SRRBACK call provided by the z/OS Recoverable Resource Service (RRS). SRRBACK çağrısı, RRS eşgüdümü için etkinleştirilen kaynak yöneticilerine ait kaynaklarda yapılan değişiklikleri destekliyor.

- CICS applications must use the EXEC CICS SYNCPOINT ROLLBACK command to back out the unit of work. Do not use the MQBACK call for CICS applications.
- IMS uygulamaları (toplu DL/I programlarından farklı), iş birimini yedeklemek için ROLB gibi IMS çağrıları kullanılmalıdır. Do not use the MQBACK call for IMS applications (other than batch DL/I programs).
- IBM i' ta, kuyruk yöneticisi tarafından koordine edilen yerel iş birimleri için bu çağrıyı kullanın. This means that a commitment definition must not exist at job level, that is, the STRCMTCTL command with the **CMTSCOPE(\*JOB)** parameter must not have been issued for the job.
- Bir uygulama, bir iş biriminde kesinleştirilmemiş değişikliklerle sona ererse, bu değişikliklerin yok edilmesi, uygulamanın olağan ya da olağan dışı bir şekilde sona ermesine bağlıdır. Ek ayrıntılar için "IBM üzerinde MQDISC (Bağlantı kesme kuyruğu yöneticisi)" sayfa 1268 içindeki kullanım notlarına bakın.
- Bir uygulama, gruplara ya da mantıksal ileti bölümlerine ileti yerleştirdiğinde ya da ileti aldığıda, kuyruk yöneticisi, son başarılı MQPUT ve MQGET çağrılarına ilişkin ileti grubuyla ve mantıksal iletiyle ilgili bilgileri saklar. Bu bilgiler kuyruk tanıtıcısı ile ilişkilendirilir ve aşağıdaki gibi şeyler içerir:

- MQMD ' de *GroupId, MsgSeqNumber, Offset* ve *MsgFlags* alanlarının değerleri.
- İletinin bir iş biriminin parçası olup olmadığı.
- MQPUT çağrısına ilişkin: İletinin kalıcı mı, yoksa kalıcı değil mi olduğu.

Kuyruk yöneticisi, her biri için bir küme olmak üzere üç grup grup ve bölüm bilgisi kümesi tutar:

- Son başarılı MQPUT çağrısı (bu, bir iş biriminin bir parçası olabilir).
- Kuyruktan ileti kaldıran son başarılı MQGET çağrısı (bu, bir iş biriminin parçası olabilir).
- Kuyruktan ileti alan son başarılı MQGET çağrısı (bu, bir iş biriminin parçası olamaz).

Uygulama, iletileri bir iş biriminin bir parçası olarak koyduysa ya da alıyorsa ve uygulama iş birimini yedeklemeye karar verirse, grup ve bölüm bilgileri daha önce sahip olduğu değere geri yüklenir:

- MQPUT çağrısıyla ilişkili bilgiler, yürürlükteki iş biriminde o kuyruk tanıtıcısı için ilk başarılı MQPUT çağrısından önce sahip olduğu değere geri yüklendi.
- MQGET çağrısıyla ilişkili bilgiler, yürürlükteki iş biriminde o kuyruk tanıtıcısı için ilk başarılı MQGET çağrısından önce sahip olduğu değere geri yüklenir.

Çalışma birimi başladıktan sonra uygulama tarafından güncellenen, ancak iş biriminin kapsamı dışında olan kuyruklar, iş birimi geriletilirse, grup ve bölüm bilgilerini geri yükleyemez.

Bir iş birimi yedeklendiğinde grup ve kesim bilgilerinin önceki değerine geri yüklenmesi, uygulamanın çok sayıda iş biriminden oluşan büyük bir ileti grubu ya da büyük mantıksal ileti yayınlamasını ve iş birimlerinden biri başarısız olursa, ileti grubunun ya da mantıksal iletinin doğru noktasında yeniden başlatılmasına olanak tanır.

Yerel kuyruk yöneticisinin kuyruk depolaması sınırlı olması durumunda, birkaç iş birimi kullanılması avantajlı olabilir. Ancak, bir sistem hatası ortaya çıkarsa, uygulama, iletileri yeniden başlatabilmek ya da iletileri doğru noktaya getirmek için yeterli bilgileri sağlamalıdır.

Sistem hatasından sonra doğru noktada yeniden başlatılabilmeye ilişkin ayrıntılar için, PMOPT (10 basamaklı imzalı tamsayı) içinde açıklanan PMLOGO seçeneğine ve GMOPT (10 basamaklı işaretli tamsayı) içinde açıklanan GMLOGO seçeneğini kullanın.

Kalan kullanım notları, yalnızca kuyruk yöneticisi iş birimlerini koordine ettiğinde geçerlidir:

- Bir iş birimi, bağlantı tanıtıcısı ile aynı kapsama sahiptir. Belirli bir iş birimini etkileyen tüm IBM MQ çağrıları, aynı bağlantı tanıtıcısı kullanılarak gerçekleştirilmelidir. Farklı bir bağlantı tanıtıcısı kullanılarak verilen çağrılar (örneğin, başka bir uygulama tarafından yayınlanan çağrılar) farklı bir iş birimini etkiler. Bağlantı tutamaçlarının kapsamına ilişkin bilgi için HCONN (10 basamaklı imzalı tamsayı)-çıkış başlıklı konuya bakın.
- Bu çağrıdan etkilenecek, yalnızca yürürlükteki iş biriminin bir parçası olarak alınan ya da alınan iletiler etkilendir.



- Bir iş birimi içinde MQGET, MQPUT ya da MQPUT1 tarafından çağrılan, ancak bir kesinleştirme ya da geriletme çağrısını hiçbir zaman yayınlamayan, uzun süredir çalışan bir uygulama, diğer uygulamaların kullanımına sunulmayan iletilerle kuyrukları doldurabilir. Bu olasılığa karşı koruma sağlamak için, denetimcinin **MaxUncommittedMsgs** kuyruk yöneticisi özniteliğini, kuyrukları dolduran kaçak uygulamaları önleyecek kadar düşük bir değere ayarlaması gerekir, ancak beklenen ileti sistemi uygulamalarının doğru şekilde çalışmasına izin verecek kadar yüksek olmalıdır.

## Parametreler

MQSETMP çağrısında şu deęiřtirgeler vardır:

### HCONN (10 basamaklı iřaretli tamsayı)-giriř

Bu tanıtıcı, kuyruk yöneticisiyle olan baęlantıyı gösterir.

Deęerin, **HMSG** parametresinde belirtilen ileti tanıtıcısını yaratmak için kullanılan baęlantı tanıtıcısı ile eřleşmesi gerekir.

İleti tanıtıcısı HCUNAS kullanılarak yaratıldıysa, ileti tutamacının bir özellięi ayarlanırken iş parçacısında geçerli bir baęlantı oluşturulmalıdır; tersi durumda, çağrı neden kodu RC2009 ile başarısız olur.

### HMSG (20 basamaklı iřaretli tamsayı)-giriř

Bu, deęiřtirilecek ileti iřidir. Deęer önceki bir MQCRTMH çağrısı tarafından döndürüldü.

### SETOPT (MQSMPO)-giriř

İleti özelliklerinin nasıl ayarlanacaęını denetler.

Bu yapı, uygulamaların ileti özelliklerinin nasıl ayarlanmasını denetleyen seçenekleri belirlemesine olanak sağlar. Yapı, MQSETMP çağrısında bir giriř deęiřtirgiřidir. Ek bilgi için [MQSMPO](#) bařlıklı konuya bakın.

### PRNAME (MQCHARV)-giriř

Ayarlanacak özellięin adı.

Özellik adlarının kullanımına iliřkin ek bilgi için [Özellik adları](#) ve [Özellik adı kısıtlamaları](#) bařlıklı konuya bakın.

### PRPDSC (MQPD)-giriř/çıkıř

Bu yapı, ařaęıdakiler de içinde olmak üzere bir özellięe iliřkin öznitelikleri tanımlamak için kullanılır:

- özellik desteklenmiyorsa ne olur
- özellięin ait olduęu ileti baęlamı
- malın akındıęı gibi kopyalandıęı mesajlar

Bu yapıyla ilgili ek bilgi için [MQPD](#) bařlıklı konuya bakın.

### TYPE (10 basamaklı iřaretli tamsayı)-giriř

Ayarlanmakta olan özellięin veri tipi. Ařaęıdakilerden biri olabilir:

#### TYPEYÜKLEME

Bir boole. *ValueLength* deęeri 4 olmalıdır.

#### TYPBST

Bayt dizilimi. *ValueLength* deęeri sıfır ya da daha büyük olmalıdır.

#### TYPI8

8 bitlik iřaretli tamsayı. *ValueLength* deęeri 1 olmalıdır.

#### TYPI16

16 bitlik bir iřaretli tamsayı. *ValueLength* deęeri 2 olmalıdır.

#### TYPI32

32 bitlik iřaretli tamsayı. *ValueLength* deęeri 4 olmalıdır.

**TYP164**

Bir 64 bit işaretli tamsayı. *ValueLength* değeri 8 olmalıdır.

**TYPF32**

32 bitlik kayan noktalı sayı. *ValueLength* değeri 4 olmalıdır.

**TYPF64**

Bir 64 bitlik kayan noktalı sayı. *ValueLength* değeri 8 olmalıdır.

**TYPSTR**

Bir karakter dizgisi. *ValueLength* değeri sıfır ya da daha büyük olmalı ya da VLNULL özel değeri olmalıdır.

**TYPNUL**

Özellik var, ancak boş değere sahip. *ValueLength* değeri sıfır olmalıdır.

**VALLEN (10 basamaklı işaretli tamsayı)-giriş**

*Value* (Değer) parametresindeki özellik değerinin bayt cinsinden uzunluğu.

Sıfır değeri yalnızca boş değerler için ya da dizgiler ya da byte dizgileri için geçerlidir. Sıfır değeri, özelliğin var olduğunu, ancak değer karakter ya da byte içermediği anlamına gelir.

*Type* değiştirgesi TYPSTR ayarına sahipse, değer sıfırdan büyük ya da bu değere eşit ya da şu özel değere eşit olmalıdır:

**VLNULL**

Değer, dizgide karşılaşılan ilk boş değerle sınırlanır. Boş değer, dizginin bir parçası olarak içerilmedi. Bu değer, TYPSTR da ayarlanmadıysa geçersizdir.

Not: VLNULL ayarlıysa, bir dizgiyi sonlandırmak için kullanılan boş değer, değer karakter takımından boş değerde olur.

**VALUE (1-byte bit dizgisi x VALLEN)-giriş**

Ayarlanacak özelliğin değeri. Arabellek, değerdeki verilerin niteine uygun bir sınırdan hizalanmalıdır.

C programlama dilinde, parametre bir işaretçi olarak bildirilir; parametrenin herhangi bir tipinin adresi, parametre olarak belirtilebilir.

*ValueLength* sıfırsa, *Value* (Değer) ifadesine başvurulmaz. Bu durumda, C ya da System/390 çevirici programlarında yazılan programlarla geçirilen parametre adresi boş değerli olabilir.

**CMPCOD (10 basamaklı işaretli tamsayı)-çıkış**

Tamamlanma kodu; bu kod aşağıdakilerden biridir:

**CCOK**

İşlem başarıyla tamamlandı.

**CCCFAIL**

Arama başarısız oldu.

**REASON (10 basamaklı işaretli tamsayı)-çıkış**

Neden kodu niteleyici *CMPCOD*.

*CMPCOD* CCOK ise:

**YOK**

(0, X'000 ') Raporlamak için bir neden yok.

*CMPCOD* , CCWARN ise:

**RC2421**

(2421, X'0975 ') Özellikler içeren bir MQRFH2 klasörü ayrıştırılmadı.

*CMPCOD* değeri CCFAIL ise:

**RC2204**

(2204, X'089C') Bağdaştırıcı kullanılmıyor.

**RC2130**

(2130, X'852 ') Baędařtırıcı hizmeti modülü yüklenemiyor.

**RC2157**

(2157, X'86D') Birincil ve ana sunucu ASID deęerleri farklı.

**RC2004**

(2004, X'07D4') Deęer parametresi geęerli deęil.

**RC2005**

(2005, X'07D5') Deęer uzunluęu parametresi geęerli deęil.

**RC2219**

(2219, X'08AB') Önceki arama tamamlanmadan önce girilen MQI çağırısı.

**RC2460**

(2460, X'099C') İleti tanıtıcısı göstergesi geęerli deęil.

**RC2499**

(2499, X'09C3') İleti tanıtıcısı zaten kullanılıyor.

**RC2046**

(2046, X'07FE') Seęenekler geęerli deęil ya da tutarlı deęil.

**RC2482**

(2482, X'09B2') Özellik tanımlayıcı yapısı geęerli deęil.

**RC2442**

(2442, X'098A') Özellik adı geęersiz.

**RC2473**

(2473, X'09A9') Özellik veri tipi geęersiz.

**RC2472**

(2472, X'09A8') Deęer verilerinde sayı biçimi hatası saptandı.

**RC2463**

(2463, X'099F') İleti özellięi seęenekleri yapısını ayarlama geęerli deęil.

**RC2111**

(2111, X'083F') Özellik adı kodlanmış karakter takımı tanıtıcısı geęerli deęil.

**RC2071**

(2071, X'817 ') Kullanılabilir bellek yetersiz.

**RC2195**

(2195, X'893 ') Beklenmeyen bir hata oluştu.

Daha ayrıntılı bilgi için bkz. ["IBM i için dönüş kodları \(ILE RPG\)" sayfa 1403](#) .

**RPG Bildirimi**

```
C*.1.....2.....3.....4.....5.....6.....7..
C          CALLP      MQSETMP(HCONN : HMSG : SETOPT :
                          PRNAME : PRPDSC :
                          TYPE : VALLEN : VALUE :
                          CMPCOD : REASON)
```

Çaęrıya ilişkin prototip tanımı:

```
DMQSETMP          PR          EXTPROC('MQSETMP')
D* Connection handle
D HCONN           10I 0 VALUE
D* Message handle
D HMSG           10I 0 VALUE
D* Options that control the action of MQSETMP
D SETOPT         20A
D* Property name
D PRNAME         32A
D* Property descriptor
D PRPDSC         24A
```

D* Property data type	
D TYPE	10I 0 VALUE
D* Length of the Value area	
D VALLEN	10I 0 VALUE
D* Property value	
D VALUE	* VALUE
D* Completion code	
D CMPCOD	10I 0
D* Reason code qualifying CompCode	
D REASON	10I 0

## IBM i IBM üzerinde MQSTAT (Durum bilgilerini al)

Durum bilgilerini almak için MQSTAT çağrısını kullanın. Döndürülen durum bilgisi tipi, çağrıda belirtilen STYPE değeri tarafından belirlenir.

- “Sözdizimi” sayfa 1336
- “Kullanım notları” sayfa 1336
- “Parametreler” sayfa 1336
- “RPG Bildirimi” sayfa 1337

### Sözdizimi

MQSTAT (*HCONN*, *STYPE*, *STAT*, *CMPCOD*, *REASON*)

### Kullanım notları

1. Bir deyim tipi (STATAPT) belirten bir MQSTAT çağrısı, önceki zamanuyumsuz MQPUT ve MQPUT1 işlemleriyle ilgili bilgileri döndürür. Çağrıya geçirilen MQSTAT yapısı, o bağlantıya ilişkin ilk kaydedilen zamanuyumsuz uyarı ya da hata bilgisiyle tamamlandı. Daha fazla hata ya da uyarı ilk olarak izlenirse, bunlar olağan durumda bu değerleri değiştirmez. Ancak, bir hata CCWARN tamamlanma koduyla oluşursa, bunun yerine, tamamlanma kodu CCFAIL olan bir hata döndürülür.
2. Bağlantı kurulduğundan ya da MQSTAT ' a son çağrıdan bu yana hata oluştuysa, CCCCOD CCOK ve RTMYT NEMASY döndürülmesi döndürüldü.
3. Bağlantı tanıtıcısı altında işlenen zamanuyumsuz çağrılarının sayısı, üç sayaç kullanılarak, STSPSC, STSPWC ve STSPFC kullanılarak döndürülür. Bu sayaçlar, kuyruk yöneticisi tarafından her zamanuyumsuz işlemin başarıyla işlendiği, bir uyarıya sahip ya da başarısız olduğunda (dağıtım listesi başına bir dağıtım listesi için bir kez dağıtım listesi başına bir kez konduğunda) bir uyarı ya da başarısız olduğunda, bu sayaçlar artırılır (not, dağıtım listesi başına bir kez dağıtım listesi sayım işlemi için bir kez konduğunda).
4. MQSTAT ' a başarılı bir çağrı, önceki hata bilgilerinde ya da sıfırlanmakta olan sayılarla sonuçlanıyor.

### Parametreler

MQSTAT çağrısında aşağıdaki deęiřtirgeler vardır:

#### Hconn (MQHCONN)-giriř

Bu tanıtıcı, kuyruk yöneticisiyle olan bağlantıyı gösterir. The value of *Hconn* was returned by a previous MQCONN or MQCONNX call.

#### STYPE (10 basamaklı işaretli tamsayı)-giriř

İstenilen durum bilgilerinin tipi. Geçerli tek deęer řunlardır:

##### STATAPT

Önceki zamanuyumsuz put işlemleriyle ilgili bilgi döndürür.

#### STS (MQSTS)-giriř/çıkıř

Durum bilgisi yapısı. Ayrıntılar için bkz. “MQSTS (Status reporting structure) on IBM i” sayfa 1210.

## CMPCOD (10 basamaklı işaretli tamsayı)-çıkış

Tamamlanma kodu; bu kod aşağıdakilerden biridir:

### CCOK

İşlem başarıyla tamamlandı.

### CCCFAIL

Arama başarısız oldu.

## REASON (10 basamaklı işaretli tamsayı)-çıkış

Neden kodu niteleyici *CMPCOD*.

*CMPCOD* CCOK ise:

### YOK

(0, X'000 ') Raporlamak için bir neden yok.

*CMPCOD* CCFAIL ise:

### RC2374

(2374, X' 946 ') API çıkışı başarısız oldu

### RC2183

(2183, X'887 ') API çıkışı yüklenemiyor.

### RC2219

(2219, X'8AB') Önceki arama tamamlanmadan önce girilen MQI çağrısı.

### RC2009

(2009, X'7D9') Kuyruk yöneticisine yönelik bağlantı kaybedildi.

### RC2203

(2203, X'89B') Bağlantı sona erdiriliyor.

### RC2018

(2018, X'7E2') Bağlantı tanıtıcısı geçerli değil.

### RC2162

(2162, X'872 ') Kuyruk yöneticisi durduruluyor

### RC2102

(2102, X'836 ') Yetersiz sistem kaynakları var.

### RC2430

(2430, X'97E') MQSTAT tipi ile ilgili hata.

### RC2071

(2071, X'817 ') Kullanılabilir bellek yetersiz.

### RC2424

(2424, X' 978 ') MQSTS yapısıyla ilgili hata

### RC2195

(2195, X'893 ') Beklenmeyen bir hata oluştu.

### RC2298

(2298, X'8FA') İstenen işlev geçerli ortamda yok.

Bu kodlarla ilgili ayrıntılı bilgi için aşağıdaki başlara bakın:

- [Messages and reason codes](#)

## RPG Bildirimi

```
C*.. 1 ..+... 2 ..+... 3 ..+... 4 ..+... 5 ..+... 6 ..+... 7
C          CALLP      MQSTAT(HCONN : ETYPE : ERR :
C                               CMPCOD : REASON)
```

Çağrıya ilişkin prototip tanımı:

```

D.. 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7
DMQSTAT          PR          EXTPROC('MQSTAT')
D* Connection handle
D HCONN          10I 0 VALUE
D* Status information type
D STYPE          10I 0 VALUE
D* Status information
D STATUS          296A
D* Completion code
D CMPCOD          10I 0
D* Reason code qualifying CompCode
D REASON          10I 0

```

## IBM i IBM i üzerinde MQSUB (Aboneliği Kaydet)

MQSUB çağrısı, belirli bir konuya ilişkin uygulama aboneliğini kaydeder.

- “Sözdizimi” sayfa 1338
- “Kullanım notları” sayfa 1338
- “Parametreler” sayfa 1339
- “RPG Bildirimi” sayfa 1342

### Sözdizimi

MQSUB (*HCONN*, *SUBDSC*, *HOBJ*, *HSUB*, *CMPCOD*, *REASON*)

### Kullanım notları

- Abonelik, önceden tanımlanmış bir konu nesnesinin kısa adı, konu dizgisinin tam adı kullanılarak ya da Konu dizgilerinin birleştirilmesibaşlıklı konuda açıklandığı gibi, iki kısmın birleştirilmesiyle oluşturulur.
  - Kuyruk yöneticisi, bir MQSUB çağrısı yayınlandığında, uygulamanın çalıştırıldığı kullanıcı kimliğinin erişime izin verilmeden önce uygun yetki düzeyine sahip olduğunu doğrulamak için güvenlik denetimleri gerçekleştirir. Uygun konu nesnesi, çağrıda sağlanan kısa bir adla ya da uzun bir ad sağlanırsa, konu sıradüzeninde bulunan en yakın kısa ad nesnesiyle bulunur. Bu konu nesnesi üzerinde, abone olma yetkisinin ayarlandığından emin olmak için bir yetki denetimi yapılır ve çıkışa ilişkin yetkinin ayarlandığından emin olmak için hedef kuyrukta bir yetki denetimi yapılır. SDMAN seçeneği kullanılırsa, bu, bu konu nesnesiyle ilişkili yönetilen kuyruk adında bir yetki denetimi yapıldığı ve yönetilmeyen bir kuyruk sağlandığı anlamına gelir; bu, **HOBJ** parametresiyle gösterilen kuyrukta bir yetki denetimi yapıldığı anlamına gelir.
  - SOMAN seçeneği kullanıldığında MQSUB çağrısında döndürülen *HOBJ*, Gerileme eşiği ve Aşırı geriletme yeniden kuyruğa alma adı gibi öznitelikleri bulmak için sorulabilir. Yönetilen kuyruğun adını da sorabilirsiniz, ancak bu kuyruğu doğrudan açmayı denememelisiniz.
  - Abonelikler, birden fazla grubun yayınıla eşleştiği abonelikler grubuna yalnızca tek bir yayının teslim edilmesine izin vererek gruplandırılabilir. Abonelikler SOGRP seçeneği kullanılarak gruplandırılır ve abonelikleri gruplamak için şunları yapmanız gerekir:
    - aynı adlandırılmış kuyruğu (SOMAN seçeneğini kullanmayan) aynı kuyruk yöneticisinde kullanın- MQSUB çağrısında **HOBJ** değiştirilmesiyle gösterilir
    - aynı *SDCID* paylaşımını paylaşma
    - aynı *SDSL*
- Bu öznitelikler, grupta olduğu kabul edilen abonelik kümesini tanımlar ve bir abonelik gruplandığında değiştirilemeyen özniteliklerdir. *SDSL* değiştirmesi RC2512 ile sonuçlanır ve diğerlerinden herhangi birinin değiştirilmesi (abonelik gruplanmamışsa değiştirilebilir) RC2515 ile sonuçlanır.
- MQSD ' deki alanlar, SORES seçeneğini kullanan bir MQSUB çağrısından dönüşte tamamlanır. Döndürülen MQSD, MQSD ' ye uygulanan abonelikte yapmanız gereken değişikliklerle SOALT seçeneğini kullanan bir MQSUB çağrısına doğrudan geçirilebilir. Bazı alanlarda, çizelgede belirtildiği gibi, dikkat edilmesi gereken özel noktalar vardır.

Çizelge 753. MQSUB ' den MQSD çıkışı	
MQSD ' de alan adı	Dikkat edilmesi gereken özel noktalar
Erişim ya da yaratma seçenekleri	MQSUB çağrısından dönüşte bu seçeneklerin hiçbiri ayarlanmadı. MQSD ' yi daha sonra bir MQSUB çağrısında yeniden kullanırsanız, belirtik olarak ayarlanması gerekir.
Dayanıklılık seçenekleri, Hedef seçenekleri, Kayıt Seçenekleri ve Genel Arama Kartı seçenekleri	Bu seçenekler uygun şekilde ayarlanacak
Yayınlama seçenekleri	Bu seçenekler, yalnızca SOCRE için geçerli olan SONEWP dışında uygun şekilde ayarlanacaktır.
Diğer seçenekler	Bu seçenekler, bir MQSUB çağrısından dönüşte değiştirilmez. API çağrılarının nasıl yayınlandığını ve abonelikte birlikte nasıl saklanmadığını denetler. Bunlar, MQSD ' yi yeniden kullanan sonraki MQSUB çağrılarında gerektiği şekilde ayarlanmalıdır.
ObjectName	Bir MQSUB çağrısından dönüşte yalnızca bu giriş alanı değiştirilmez.
ObjectString	Bir MQSUB çağrısından dönüşte yalnızca bu giriş alanı değiştirilmez. Bir arabellek sağlanırsa, SDR0 alanında kullanılan Tam konu adı döndürülür.
AlternateUserTanıtıcısı ve AlternateSecurityTanıtıcısı	Bir MQSUB çağrısından dönüşte yalnızca bu giriş alanları değişmez. API çağrılarının nasıl yayınlandığını ve abonelikte birlikte nasıl saklanmadığını denetler. Bunlar, MQSD ' yi yeniden kullanan sonraki MQSUB çağrılarında gerektiği gibi ayarlanmalıdır.
SubExpiry	SORES seçeneği kullanılarak bir MQSUB çağrısından döndüğünüzde bu alan, kalan süre bitimine değil, aboneliğin özgün süre bitimine ayarlanır. SOALT seçeneğini kullanarak MQSD ' yi bir MQSUB çağrısında yeniden kullanırsanız, aboneliğin süre bitimini yeniden saymaya başlayacak şekilde sıfırlarsınız.
SubName	Bu alan, MQSUB çağrısındaki bir giriş alanıdır ve çıkışta değiştirilmez.
SubUserVeri ve SelectionString	Bu değişken uzunluklu alanlar, SORES seçeneği kullanılarak bir MQSUB çağrısının çıkışında döndürülür (bir arabellek sağlandıysa) ve VCHRPiçinde artı bir arabellek uzunluğu da döndürülür. Herhangi bir arabellek sağlanmazsa, MQCHARV.If ' un VCHRL alanında sağlanan arabellek, alanı döndürmek için gereken alandan daha küçük bir değer döndürülür; sağlanan arabellekte yalnızca VCHRP byte döndürülür. Daha sonra SOALT seçeneğini kullanarak MQSD ' yi bir MQSUB çağrısında yeniden kullanırsanız ve bir arabellek sağlanmazsa, ancak sıfır dışında bir VCHRL sağlanırsa, bu uzunluk alanının var olan uzunluğuyla eşleşirse, alanda herhangi bir değişiklik yapılmaz.
SubCorrelTanıtıcısı ve PubAccountingSimgesi	SOSCID kullanmazsanız, SDCID kuyruk yöneticisi tarafından oluşturulur. SOSETI kullanmazsanız, SDACC kuyruk yöneticisi tarafından oluşturulur. Bu alanlar, SORES seçeneği kullanılarak MQSUB çağrısından MQSD ' de döndürülür. Bunlar kuyruk yöneticisi tarafından oluşturulursa, oluşturulan değer SOCRE ya da SOALT seçeneği kullanılarak bir MQSUB çağrısında döndürülür.
PubPriority, SubLevel & PubApplIdentityData	Bu alanlar MQSD ' de döndürülür.
ResObjectDizesi	Bir arabellek sağlanırsa, MQSD ' de yalnızca bu çıkış alanı döndürülür.

## Parametreler

MQSUB çağrısındaki değiştirgeler şunlardır:

### HCONN (10 basamaklı işaretli tamsayı)-giriş

Bu tanıtıcı, kuyruk yöneticisine yönelik bağlantıyı gösterir. HCONN değeri önceki bir MQCONN ya da MQCONNX çağrısı tarafından döndürüldü.

## SUBDSC (MQSD)-giriş/çıkış

Bu, uygulama tarafından kaydedilmekte olan, kullanımla nesneyi tanımlayan bir yapıdır. Ek bilgi için bkz. [“IBM i üzerinde MQSD \(Abonelik tanımlayıcısı\)” sayfa 1193](#).

## HOBJ (10 basamaklı işaretli tamsayı)-giriş/çıkış

Bu tanıtıcı, bu aboneliğe gönderilen iletileri almak için oluşturulan erişimi temsil eder. Bu iletiler belirli bir kuyrukta saklanabilir ya da kuyruk yöneticisinden belirli bir kuyruğa gerek kalmadan saklama alanını yönetmesi istenebilir.

Nesne tanıtıcısı.

Belirli bir kuyruk kullanılacaksa, yaratma sırasında abonelik ile ilişkilendirilmelidir. Bu iki şekilde yapılabilir:

- MQSUB 'yi SDCRT seçeneğiyle çağırırken bu tanıtıcıyı sağlayarak. Bu tanıtıcı çağrı için bir giriş değiştirgesi olarak sağlandıysa, bu tanıtıcı en az bir OOINP\*, OOOOUT (örneğin, bir uzak kuyruk) ya da OOBROW seçeneği kullanılarak bir kuyruğun önceki MQOPEN çağrısından döndürülen geçerli bir nesne tanıtıcısı olmalıdır. Bu durumda arama RC2019 ile başarısız olur. Bir konu nesnesine çözülen diğer ad kuyruğuna ilişkin bir nesne tanıtıcısı olamaz. Bu durumda, arama RC2019 ile başarısız olur.
- DEFINE SUB MQSC komutunu kullanarak ve o komutu bir kuyruk nesnesi adıyla belirterek.

Kuyruk yöneticisi bu aboneliğe gönderilen iletilerin depolanmasını yönetecekse, SOMAN seçeneğini kullanarak ve parametre değerini HONONE olarak ayarlayarak abonelik oluşturulduğunda bunu belirtmeniz gerekir. Kuyruk yöneticisi, tanıtıcıyı çağrıda çıkış parametresi olarak döndürür ve döndürülen tanıtıcı yönetilen tanıtıcı olarak bilinir. HONONE belirtilirse ve SOMAN da belirtilmezse, çağrı RC2019 ile başarısız olur.

Kuyruk yöneticisi tarafından döndürülen yönetilen bir tanıtıcı, MQGET ya da MQCB çağrısında, göz atma seçenekleriyle ya da göz atma seçenekleri olmadan, MQINQ çağrısında ya da MQCLOSE üzerinde kullanılabilir. MQPUT, MQSET ya da sonraki bir MQSUB üzerinde kullanılamaz; bunu yapma girişimi MQPUT için RC2039, MQSET için RC2040 ya da MQSUB için RC2038 ile başarısız olur.

Bu aboneliği sürdürmek için MQSD yapısındaki OPTS alanındaki SORES seçeneği kullanılırsa, HONONE belirtildiyse, tanıtıcı bu değiştirgede uygulamaya döndürülebilir. Abonelik yönetilen bir tanıtıcı kullanıyor ya da kullanmıyor olsa da bunu kullanabilirsiniz. DEFINE SUB komutunda tanımlanan abonelik kuyruğuna ilişkin tanıtıcıyı istiyorsanız, bu işlem DEFINE SUB kullanılarak yaratılan abonelikler için yararlı olabilir. Yönetimsel olarak oluşturulan bir aboneliğin sürdürülmesi durumunda, kuyruk OOINPQ ve OOBROW ile açılır. Başka seçenekler gerekiyorsa, uygulamanın abonelik kuyruğunu açık olarak açması ve çağrıdaki nesne tanıtıcısını sağlaması gerekir. Kuyruk açılırken bir sorun ortaya çıkarsa, arama RC2522 ile başarısız olur. HOBJ sağlanırsa, özgün MQSUB çağrısındaki HOBJ ile eşdeğer olmalıdır. Bu, bir MQOPEN çağrısından döndürülen bir nesne tanıtıcısı sağlandığında, tanıtıcı değeri daha önce kullanılan kuyrukla aynı olmalıdır ya da çağrı RC2019 ile başarısız olur.

Bu abonelik değiştiriliyorsa, MQSD yapısındaki OPTS alanındaki SOALT seçeneği kullanılarak farklı bir HOBJ sağlanabilir. Bu parametre aracılığıyla daha önce kuyruğa gönderilen yayınlar bu kuyrukta kalır ve HOBJ parametresi artık farklı bir kuyruğu gösteriyorsa, bu iletileri almak uygulamanın sorumluluğundadır.

Bu parametrenin çeşitli abonelik seçenekleriyle kullanımı aşağıdaki tabloda özetlenmiştir:

Seçenekler	Hobj.	Açıklama
SOCRT + SOMAN	Girişte yoksayıldı	Kuyruk yöneticisi tarafından yönetilen ileti depolanmasıyla bir abonelik oluşturur.
SOCRT	Geçerli nesne tanıtıcısı	İletilerin hedefi olarak belirli bir kuyruğu sağlayan bir abonelik oluşturur.
SIRALAR	TONON	Önceden yaratılmış bir aboneliği sürdürür (yönetilen ya da değil) ve kuyruk yöneticisinin nesne tanıtıcısını uygulama tarafından kullanılmak üzere döndürmesini sağlar.



Çizelge 754. Hobj 'u çeşitli abonelik seçenekleriyle kullanma (devamı var)		
Seçenekler	Hobj.	Açıklama
SIRALAR	Geçerli, eşleşen, nesne tanıtıcısı	İletiler için hedef olarak belirli bir kuyruğu kullanan önceden oluşturulmuş bir aboneliği sürdürür ve belirli açık seçeneklerle bir nesne tanıtıcısı kullanır.
SOALT + SOMAN	TONON	Önceden belirli bir kuyruğu kullanan var olan bir aboneliği şimdi yönetilecek şekilde değiştirir.
SOALT	Geçerli nesne tanıtıcısı	Var olan bir aboneliği, belirli bir kuyruğu (yönetilen ya da farklı bir belirli kuyruktan) kullanacak şekilde değiştirir.

İster sağlanmış, ister döndürülmüş olsun, yayınları almak için gereksinim duyduğunuz sonraki MQGET çağrılarında **HOBJ** belirtilmelidir.

**HOBJ** tanıtıcısı, MQCLOSE çağrısı yayınlandığında ya da tutamacın kapsamını tanımlayan işlem birimi sona erdiğinde geçerli olmaktan çıkar. Döndürülen nesne tanıtıcısının kapsamı, çağrıda belirlenen bağlantı tanıtıcısıyla aynı. Tanıtıcı kapsamına ilişkin bilgi için bkz. [HCONN](#) . **HOBJ** tanıtıcısının MQCLOSE ' nin **HSUB** tanıtıcısı üzerinde bir etkisi yoktur.

#### **HSUB (10 basamaklı işaretli tamsayı)-çıkış**

Bu tanıtıcı, yapılan aboneliği gösterir. İki işlem daha için kullanılabilir:

- Abonelik gerçekleştirilirken SOPUBR seçeneği kullanıldığında yayınların gönderilmesini istemek için sonraki bir MQSUBRQ çağrısında kullanılabilir.
- Yapılan aboneliği kaldırmak için sonraki bir MQCLOSE çağrısında kullanılabilir. **HSUB** tanıtıcısı, MQCLOSE çağrısı yayınlandığında ya da tanıtıcı kapsamını tanımlayan işlem birimi sona erdiğinde geçerli olmaktan çıkar. Döndürülen nesne tanıtıcısının kapsamı, çağrıda belirlenen bağlantı tanıtıcısıyla aynı. **HSUB** tanıtıcısının MQCLOSE ' nin **HOBJ** tanıtıcısı üzerinde bir etkisi yoktur.

Bu tanıtıcı bir MQGET ya da MQCB çağrısına geçirilemez. **HOBJ** parametresini kullanmalısınız. Bu tanıtıcı, RC2019 ile sonuçlanan başka bir IBM MQ çağrısına geçirilmiştir.

#### **CMPCOD (10 basamaklı işaretli tamsayı)-çıkış**

Tamamlanma kodu; aşağıdakilerden biridir:

##### **CCOK**

Başarıyla tamamlandı

##### **CCWARN**

Uyarı (kısmi tamamlama)

##### **CCFAIL**

Arama başarısız oldu

#### **REASON (10 basamaklı işaretli tamsayı)-çıkış**

Neden kodu niteleyici **CMPCOD**.

**CMPCOD** CCOK ise:

##### **RCNONE**

(0, X'000 ') Raporlamak için bir neden yok.

**CMPCOD** CCFAIL ise:

##### **RC2019**

(2019 X'07E3') Nesne tanıtıcısı geçerli değil

##### **RC2046**

(2046 X'07FE') Seçenekler geçerli değil ya da tutarlı değil

##### **RC2085**

(2085 X'0825 ') Tanımlanmış nesne bulunamıyor

**RC2161**

(2161 X'0871 ') Kuyruk yöneticisi susturuyor

**RC2298**

(2298 X'08FA') İşlev desteklenmiyor.

**RC2424**

(2424 X'0978 ') Abonelik tanımlayıcısı (MQSD) geçerli değil

**RC2425**

(2441 X' 979 ') Konu dizgisi geçerli değil

**RC2428**

(2428 X'097C') Belirtilen abonelik adı var olan aboneliklerle eşleşmiyor

**RC2429**

(2429 X'097D') Abonelik adı var ve başka bir uygulama tarafından kullanılıyor

**RC2431**

(2431 X'097F') SubUserVeri alanı geçerli değil

**RC2432**

(2432 X'0980 ') Abonelik var

**RC2434**

(2434 X'0982 ') Abonelik adı var olan abonelikle eşleşiyor

**RC2440**

(2440 X'0988 ') SubName alanı geçerli değil

**RC2441**

(2441 X'0989 ') Nesne dizgisi alanı geçersiz

**RC2435**

(2435 X'0983 ') Öznitelik SDALT kullanılarak değiştirilemez ya da abonelik SDIMM ile oluşturulmuştur.

**RC2436**

(2436 X'0984 ') SODUR seçeneği geçerli değil

**RC2459**

(2459, X'99B') Seçim dizgisi sözdizimi hatası.

**RC2503**

(2503 X'09C7') MQSUB çağrılarları şu anda abone olunan konular için engelleniyor.

**RC2519**

(2519, X'9D7') Seçim dizgisi, MQCHARV yapısının nasıl kullanılacağına ilişkin tanımda belirtildiği gibi değil.

**RC2551**

(2551, X'9F7') Belirtilen seçim dizgisi kullanılmıyor.

**RPG Bildirimi**

```

C*..1.....2.....3.....4.....5.....6.....7..
C                               CALLP      MQSUB(HCONN : SUBDSC : HOBJ :
C                               HSUB : CMPCOD : REASON)

```

Çağrıya ilişkin prototip tanımlaması:

```

D*..1.....2.....3.....4.....5.....6.....7..
DMQSUB          PR              EXTPROC('MQSUB')
D* Connection handle
D HCONN          10I 0 VALUE
D* Subscription descriptor
D SUBDSC          400A
D* Object handle for queue
D HOBJ          10I 0
D* Subscription object handle

```

D HSUB	10I 0
D* Completion code	
D CMPCOD	10I 0
D* Reason code qualifying CompCode	
D REASON	10I 0

## IBM i IBM üzerinde MQSUBRQ (Abonelik İsteği)

MQSUBRQ çağrısı bir abonelik ile ilgili istekte bulundur.

- [“Sözdizimi” sayfa 1343](#)
- [“Kullanım notları” sayfa 1343](#)
- [“Parametreler” sayfa 1343](#)
- [“RPG Bildirimi” sayfa 1344](#)

### Sözdizimi

MQSUBRQ (*HCONN*, *HSUB*, *ACTION*, *SUBROPT*, *CMPCOD*, *REASON*)

### Kullanım notları

Aşağıdaki kullanım notları, SRAPUB ' un kullanımı için geçerlidir:

1. Bu komut başarıyla tamamlanırsa, belirtilen abonelik ile eşleşen alıkonan yayınlar aboneliğe gönderilir ve MQGET ya da MQCB kullanılarak, aboneliği yaratan özgün MQSUB yükünün döndürdüğü HOBJS kullanılarak alınabilir.
2. Aboneliği yaratan özgün MQSUB yüklemi tarafından abone olunan konu genel arama karakteri içeriyorsa, saklanacak birden çok yayın gönderilebilir. Bu çağrı sonucu olarak gönderilen yayınların sayısı, SUBROPT yapısındaki *SRNMP* alanına kaydedilir.
3. Bu fiil, RC2437 neden koduyla tamamlanırsa, belirtilen konu için yürürlükte tutulan yayınların olmamasıdır.
4. Bu komut bir RC2525 ya da RC2526 neden koduyla tamamlanırsa, belirtilen konuya ilişkin güncel yayınlar vardır; ancak, bunların teslim edilememesi anlamına gelen bir hata oluştu.
5. Uygulamanın, bu çağrıyı gerçekleştirebilmesi için önce konuya ilişkin geçerli bir aboneliği olmalıdır. Abonelik, uygulamanın önceki bir örneğinde yapıldıysa ve abonelik için geçerli bir tanıtıcı kullanılamıyorsa, uygulamanın bu çağrıya kullanılmak üzere bir tanıtıcı elde etmek için önce SORES seçeneğiyle MQSUB seçeneğini çağırmalıdır.
6. Bu yayınlar, bu uygulamanın yürürlükteki abonelikleriyle kullanılmak üzere kayıtlı olan hedefe gönderilir. Yayınların başka bir yerde gönderilmesi gerekiyorsa, önce SOALT seçeneğiyle MQSUB çağrısıyla aboneliğin değiştirilmelidir.

### Parametreler

MQSUBRQ çağrısına aşağıdaki deęiřtirgeler vardır:

#### HCONN (10 basamaklı iřaretli tamsayı)-giriř

Bu tanıtıcı, kuyruk yöneticisiyle olan baęlantıyı gösterir. The value of *HCONN* was returned by a previous MQCONN or MQCONNX call.

On z/OS for CICS applications the MQCONN call can be omitted, and the following value specified for *HCONN* :

#### HCDEFH

Varsayılan baęlantı tanıtıcısı.

#### HSUB (10 basamaklı iřaretli tamsayı)-giriř

Bu tanıtıcı, bir güncellemenin isteneceęi abonelięi temsil eder. *HSUB* deęeri önceki bir MQSUB çağrısından döndürüldü.

## İŞLEM (10 basamaklı işaretli tamsayı)-giriş

Bu deęiştirge, abonelikte istenmekte olan işlemi denetler. Aşağıdakilerden biri (ve yalnızca bir tanesi) belirtilmelidir:

### SSRAPUB

Bu işlem, belirtilen konu için bir güncelleme yayınının gönderilmesini ister. Bu durum, abonenin, abonelięi gerçekleştirdiğinde MQSUB çağrısında SOPUBR seçeneğini belirlemesi durumunda kullanılır. Kuyruk yöneticisinin konu için saklanacak bir yayını varsa, bu, aboneye gönderilir. Yoksa, arama başarısız olur. Bir uygulama tutulan bir yayını gönderdiyse, bu, yayının MQIsRetained ileti özellięi ile gösterilir.

Since the topic in the existing subscription represented by the **HSUB** parameter can contain wildcards, the subscriber might receive multiple retained publications.

## SBROPT (MQSRO)-giriş/çıkış

Bu seçenekler MQSUBRQ 'nun işlemini denetler, ayrıntılar için [“MQSRO-Abonelik isteęi seçenekleri” sayfa 578](#) ' e bakın.

## CMPCOD (10 basamaklı işaretli tamsayı)-çıkış

Tamamlanma kodu; bu kod aşağıdakilerden biridir:

### CCOK

Başarıyla tamamlandı

### CCWARN

Uyarı (kısmi tamamlama)

### CCCFAIL

Arama başarısız oldu

## Neden (10 basamaklı işaretli tamsayı)-çıkış

Neden kodu niteleyici *CMPCOD*.

*CPMPCOD* CCOK ise:

### YOK

(0, X'000 ') Raporlamak için bir neden yok.

*CPMPCOD* CCFAIL ise:

### RC2298

2298 (X'08FA') İstenen işlev, yürürlükteki ortamda yok.

### RC2437

2437 (X'0985 ') Bu konu için saklanmış yayınların saklanmasına neden yok.

### RC2046

2046 (X'07FE') Seçenekleri parametresi ya da alanında geçerli olmayan seçenekler ya da geçerli olmayan bir seçenek birleşimi var.

### RC2161

2161 (X'0871 ') Kuyruk yöneticisi susturuyor

### RC2438

2438 (X'0986 ') MQSUBRQ çağrısında, Subscription Request Options MQSRO geçerli deęil.

## RPG Bildirimi

```
C*..1.....2.....3.....4.....5.....6.....7..
C                               CALLP    MQSUBRQ(HCONN : HSUB : ACTION :
C                               SBROPT : CMPCOD : REASON)
```

Çaęrıya ilişkin prototip tanımı:

```
D*..1.....2.....3.....4.....5.....6.....7..
MQSUBRQ      PR      EXTPROC('MQSUBRQ')
D* Connection handle
D HCONN      10I 0 VALUE
D* Subscription handle
D HSUB      10I 0 VALUE
D* Action requested on the subscription
D ACTION     10I 0 VALUE
D* Subscription Request Options
D SBROPT     16A
D* Completion code
D CMPCOD     10I 0
D* Reason code qualifying CompCode
D REASON     10I 0
```

IBM i

## IBM üzerindeki nesnelerin öznitelikleri

Bu konu derlemi, yalnızca bir MQINQ işlev çağrısının konusu olabilecek IBM MQ nesnelerini listeler ve sorgulanabilen özniteliklerin ve kullanılacak seçicilerin ayrıntılarını verir.

### Kuyruklara ilişkin öznitelikler

Bu bilgileri, kuyruk tanımlamalarının farklı tipleri ve her biri tarafından desteklenen öznitelikler hakkında bilgi edinmek için kullanın.

**Kuyruk tipleri:** Kuyruk yöneticisi, aşağıdaki kuyruk tanımlaması tiplerini destekler:

#### Yerel kuyruk

Bu, iletileri saklayan fiziksel bir kuyruğdur. Kuyruk yerel kuyruk yöneticisinde var.

Yerel kuyruk yöneticisine bağlı uygulamalar, bu tipteki kuyruklara ileti yerleştirebilir ya da bu tipteki kuyruklara ileti kaldırılabilir. **QType** kuyruk özniteliğinin değeri QTLOC ' dir.

#### Paylaşılan kuyruk

Bu, iletileri saklayan fiziksel bir kuyruğdur. Kuyruk, paylaşılan havuzun sahibi olan kuyruk paylaşım grubuna ait olan tüm kuyruk yöneticilerinin erişebildiği paylaşılan bir havuzda var olur.

Kuyruk paylaşım grubundaki herhangi bir kuyruk yöneticisine bağlı uygulamalar, bu tipteki kuyruklara ileti yerleştirebilir ya da bu tipteki kuyruklara ileti kaldırılabilir. Bu tür kuyruklar, yerel kuyruklar ile aynı şekilde etkili olur. **QType** kuyruk özniteliğinin değeri QTLOC ' dir.

- Paylaşılan kuyruklar yalnızca z/OSüzerinde desteklenir.

#### Küme kuyruğu

Bu, iletileri saklayan fiziksel bir kuyruğdur. Kuyruk yerel kuyruk yöneticisinde ya da yerel kuyruk yöneticisiyle aynı kümeye ait olan bir ya da daha çok kuyruk yöneticisinde var.

Yerel kuyruk yöneticisine bağlı uygulamalar, kuyruğun yeri ne olursa olsun, bu tipteki kuyruklara ileti yerleştirebilir. Yerel kuyruk yöneticisinde bir kuyruk örneği varsa, kuyruk yerel bir kuyrukta aynı şekilde hareket eder ve yerel kuyruk yöneticisine bağlı uygulamalar kuyruktan iletileri kaldırabilir. **QType** kuyruk özniteliğinin değeri QTCLUS ' dur.

#### Diğer ad kuyruğu

Bu bir fiziksel kuyruk değil-yerel bir kuyruk için alternatif bir addır. Diğer adın çözdüğü yerel kuyruğun adı, diğer ad kuyruğunun tanımının bir parçasıdır.

Yerel kuyruk yöneticisine bağlı uygulamalar, diğer ad kuyruklarına ileti yerleştirebilir ve diğer ad kuyruklarından iletileri yerleştirebilir; iletiler yerleştirilir ve diğer adın çözdüğü yerel kuyruktan kaldırılır. **QType** kuyruk özniteliğinin değeri QTALS 'dir.

#### Uzak kuyruk

Bu bir fiziksel kuyruk değil; uzak kuyruk yöneticisinde bulunan bir kuyruğun yerel tanığıdır. Uzak kuyruğun yerel tanımlaması, yerel kuyruk yöneticisine iletilerin uzak kuyruk yöneticisine nasıl yönelteceğini bildiren bilgiler içerir.

Yerel kuyruk yöneticisine bağlı uygulamalar, uzak kuyruklara ileti yerleştirebilir; iletiler, iletileri uzak kuyruk yöneticisine yöneltmek için kullanılan yerel iletim kuyruğuna yerleştirilir. Uygulamalar uzak kuyruklardan iletileri kaldıramaz. **QType** kuyruk özniteliğinin değeri QTREM ' dir.

Uzak kuyruk tanımlaması aşağıdaki şekilde de kullanılabilir:

- Yanıt kuyruğu yöneltmesi

Bu durumda, tanımlamanın adı bir yanıtlama kuyruğu adıdır. Ek bilgi için bkz. [Reply-to queue alias definitions](#).

- Kuyruk yöneticisi yumuşatması

Bu durumda, tanımın adı bir kuyruk yöneticisinin diğer adı ve kuyruğun adı değildir. Ek bilgi için [Kuyruk yöneticisi diğer ad tanımlamaları](#) başlıklı konuya bakın.

### Model kuyruğu

Bu bir fiziksel kuyruk değil-yerel bir kuyruğun yaratılabileceği bir kuyruk öznitelikleri kümesidir.

İletiler bu tipteki kuyruklarda saklanamaz.

Bazı kuyruk öznitelikleri tüm kuyruk tipleri için geçerlidir; diğer kuyruk öznitelikleri yalnızca belirli kuyruk tipleri için geçerlidir. Bir özniteliğin geçerli olduğu kuyruk türleri, [Çizelge 755 sayfa 1346](#) ve sonraki tablolarda bir "X" ile gösterilir.

[Çizelge 755 sayfa 1346](#) , kuyruklara özgü öznitelikleri özetler. Öznitelikler alfabetik sırayla açıklanır.

Çizelgede gösterilen özniteliklerin adları, MQINQ ve MQSET çağrılarıyla kullanılan adlardır. MQSC komutları, öznitelikleri tanımlamak, değiştirmek ya da görüntülemek için kullanıldığında, diğer kısa adlar kullanılır; ayrıntılar için [MQSC komutları](#) konusuna bakın.

Aşağıdaki tabloda, sütunlar aşağıdaki gibi geçerlidir:

- Yerel kuyruklar için kolon da paylaşılan kuyruklara uygulanır.
- Model kuyruklarına ilişkin kolon, hangi özniteliklerin, model kuyruğundan yaratılan yerel kuyruk tarafından edinilmiş olduğunu gösterir.
- Küme kuyruklarına ilişkin kolon, küme kuyruğu tek başına sorgu için açıldığında ya da sorgu ve çıkış için açıldığında sorgulanabilen öznitelikleri gösterir. Bir ya da daha fazla giriş, göz atma ya da küme sorgulamak üzere küme kuyruğu açılırsa, bunun yerine yerel kuyruklar için kolon uygulanır.

Çizelge 755. Kuyruklara ilişkin öznitelikler						
Öznitelik	Tanım	Yerel	Model	Diğer Ad	Uzak	Küme
<a href="#">AlterationDate</a>	Tanımlamanın son olarak değiştirildiği tarih	X		X	X	
<a href="#">AlterationTime</a>	Tanımlamanın son olarak değiştirildiği saat	X		X	X	
<a href="#">BackoutRequeueQName</a>	Aşırı geriletme yeniden kuyruğa alma kuyruğu adı	X	X			
<a href="#">BackoutThreshold</a>	Geriletme eşiği	X	X			
<a href="#">BaseQName</a>	Diğer adın çözümleyicileceği kuyruk adı			X		
<a href="#">ClusterChannelAd</a>	Küme-gönderen kanal adı	✓	✓			

Çizelge 755. Kuyruklara ilişkin öznitelikler (devamı var)

Öznitelik	Tanım	Yerel	Model	Diğer Ad	Uzak	Küme
<u>ClusterName</u>	Kuyruğun ait olduğu kümenin adı	X		X	X	
<u>ClusterNameList</u>	Kuyruğun ait olduğu kümelerin adlarını içeren ad listesi nesnesinin adı	X		X	X	
<u>CreationDate</u>	Kuyruğun yaratıldığı tarih	X				
<u>CreationTime</u>	Kuyruğun yaratıldığı saat	X				
<u>CurrentQDepth</u>	Yürürlükteki kuyruk derinliği	X				
<u>DefBind</u>	Varsayılan bağ tanımı	X		X	X	X
<u>DefinitionType</u>	Kuyruk tanımlaması tipi	X	X			
<u>DefInputOpenOption</u>	Varsayılan giriş açma seçeneği	X	X			
<u>DefPersistence</u>	Varsayılan ileti kalıcılığı	X	X	X	X	X
<u>DefPriority</u>	Varsayılan ileti önceliği	X	X	X	X	X
<u>DistLists</u>	Dağıtım listesi desteği	X	X			
<u>HardenGetGeri çekişi</u>	Doğru bir geriletme sayısının sağlanıp korunmayacağı	X	X			
<u>InhibitGet</u>	Kuyruğa ilişkin alma işlemlerinin izin verilip verilmediğini denetler	X	X	X		
<u>InhibitPut</u>	Kuyruğa ilişkin koyma işlemlerine izin verilip verilmediğini denetler	X	X	X	X	X
<u>InitiationQName</u>	Başlatma kuyruğunun adı	X	X			
<u>MaxMsgUzunluğu</u>	İleti uzunluğu üst sınırı (byte)	X	X			
<u>MaxQDepth</u>	Kuyruk derinliği üst sınırı	X	X			

Çizelge 755. Kuyruklara ilişkin öznitelikler (devamı var)

Öznitelik	Tanım	Yerel	Model	Diğer Ad	Uzak	Küme
MediaLog	En eski günlük kapsamının tanıtıcısı (ya da IBM i üzerindeki en eski günlük nesnesi) belirtilen bir kuyruğun ortam kurtarma işlemi için gerekli	✓	✓			
MsgDeliverySırası	İleti teslim sırası	X	X			
OpenInputSayı	Giriş için açma sayısı	X				
OpenOutputSayısı	Çıkış için açma sayısı	X				
ProcessName	İşlem adı	X	X			
QDepthHighOlayı	Kuyruk Derinliği Yüksek olaylarının oluşturulup oluşturulmayacağını denetler	X	X			
QDepthHighSınırı	Kuyruk derinliği üst sınırı	X	X			
QDepthLowOlayı	Kuyruk Derinliği Düşük olaylarının oluşturulup oluşturulmayacağını denetler	X	X			
QDepthLowSınırı	Kuyruk derinliği için alt sınır	X	X			
QDepthMaxOlayı	Kuyruk Dolu olaylarının oluşturulup oluşturulmayacağını denetler	X	X			
QDesc	Kuyruk tanımlaması	X	X	X	X	X
QName	Kuyruk adı	X		X	X	X
QServiceInterval	Kuyruk hizmet aralığı hedefi	X	X			
QServiceIntervalOlayı	Hizmet Aralığı Yüksek ya da Hizmet Aralığı Tamam olaylarının oluşturulup oluşturulmayacağını denetler	X	X			
QType	Kuyruk tipi	X		X	X	X
RemoteQMgrAd	Uzak kuyruk yöneticisinin adı				X	



Çizelge 755. Kuyruklara ilişkin öznitelikler (devamı var)

Öznitelik	Tanım	Yerel	Model	Diğer Ad	Uzak	Küme
<u>RemoteQName</u>	Uzak kuyruğun adı				X	
<u>RetentionInterval</u>	Alıkoyma aralığı	X	X			
<u>Kapsam</u>	Kuyruğun bir girişinin bir hücre dizininde de olup olmadığını denetler	X		X	X	
<u>Paylaşılabilirlik</u>	Kuyruk paylaşılabilirliği	X	X			
<u>TriggerControl</u>	Tetikleyici denetimi	X	X			
<u>TriggerData</u>	Verileri tetikle	X	X			
<u>TriggerDepth</u>	Tetikleyici derinliği	X	X			
<u>TriggerMsgÖnceliği</u>	Tetikleyiciler için eşik iletisi önceliği	X	X			
<u>TriggerType</u>	Tetikleme Tipi	X	X			
<u>Kullanım</u>	Kuyruk kullanımı	X	X			
<u>XmitQName</u>	İletim kuyruğu adı				X	

IBM i

### IBM üzerinde AlterationDate (12 baytlık karakter dizgisi)

Tanımlamanın en son değiştirildiği tarih.

Çizelge 756. Bu özneliğin geçerli olduğu kuyruk tipleri

Yerel	Model	Diğer Ad	Uzak	Küme
X		X	X	

Bu, tanımın en son değiştirildiği tarihtir. The format of the date is YYYY-MM-DD, padded with two trailing blanks to make the length 12 bytes (for example, 1992-09-23--), where - represents a single blank character).

Kuyruk yöneticisi olarak, belirli özneliklerin değerleri (örneğin, *CurrentQDepth*) değişir. Bu özneliklerde yapılan değişiklikler *AlterationDate* işlemini etkilemez.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla CAALTD seçiciyi kullanın. Bu özneliğin uzunluğu LNDATE tarafından verilir.

IBM i

### IBM üzerinde AlterationTime (8 baytlık karakter dizesi)

Tanımlamanın en son değiştirildiği saat.

Çizelge 757. Bu özneliğin geçerli olduğu kuyruk tipleri

Yerel	Model	Diğer Ad	Uzak	Küme
X		X	X	

Bu, tanımın en son değiştirildiği dönemdir. Saatin biçimi, saatin 10 'dan küçük olması (örneğin, 09.10.20gibi), 24 saatlik zaman biçimini kullanarak HH.MM.SS ' dir. Zaman yerel saattir.

Kuyruk yöneticisi olarak, belirli özniteliklerin değerleri (örneğin, *CurrentQDepth*) değişir. Bu özniteliklerde yapılan değişiklikler *AlterationTime* işlemini etkilemez.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla CAALTT seçiciyi kullanın. Bu özniteliğin uzunluğu LNTIME tarafından verilir.

### **IBM i IBM üzerindeBackoutRequeueQName (48 baytlık karakter dizgisi)**

Gereğinden fazla geriletme yeniden kuyruğa alma kuyruğu adı.

Çizelge 758. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

WebSphere Application Server içinde çalışan ve IBM MQ Application Server Faciliti'lerini kullanan uygulamalar, yedeklenen iletilerin nereye gidileceğini belirlemek için bu özniteliği kullanır. Diğer tüm uygulamalar için, değerinin sorgulanmasına izin verilmesinin dışında, kuyruk yöneticisi özniteliğin değerini temel alarak bir işlem gerçekleştirmez.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla CABRQN seçicisini kullanın. Bu özniteliğin uzunluğu LNQN tarafından verilir.

### **IBM i IBM üzerindeBackoutThreshold (10 basamaklı işaretli tamsayı)**

Geriletme eşiği.

Çizelge 759. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

WebSphere Application Server içinde çalışan ve IBM MQ Application Server Faciliti'lerini kullanan uygulamalar, bir iletinin yedeklenip yedeklenmemesi gerektiğini saptamak için bu özniteliği kullanır. Diğer tüm uygulamalar için, değerinin sorgulanmasına izin verilmesinin dışında, kuyruk yöneticisi özniteliğin değerini temel alarak bir işlem gerçekleştirmez.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IABTHR seçiciyi kullanın.

### **IBM i IBM üzerindeBaseQName (48 baytlık karakter dizgisi)**

Diğer adın çözümleyicilerin bulunduğu kuyruk adı.

Çizelge 760. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
		X		

Bu ad, yerel kuyruk yöneticisinde tanımlı olan bir kuyruğun adıdır. (Kuyruk adlarıyla ilgili daha fazla bilgi için, MQOD 'daki *ODON* alanının tanımına bakın. Kuyruk, aşağıdaki tiplerden biridir:

#### **QTLOC**

Yerel kuyruk.

**QTREM**

Uzak kuyruğun yerel tanımlaması.

**QTCLUS**

Küme kuyruğu.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla CABASQ seçiciyi kullanın. Bu özniteliğin uzunluğu LNQN tarafından verilir.

**IBM i IBM üzerinde BaseType (tamsayı parametresi yapısı)**

Diğer adın çözdüğü nesne tipi.

Çizelge 761. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
		X		

Bu öznitelik aşağıdaki değerlerden birine sahip olabilir:

**OTQ**

Temel nesne tipi bir kuyruktur

**OTTOP**

Temel nesne tipi bir konudur

**IBM i IBM üzerinde CFStrucName (12 baytlık karakter dizgisi)**

Bağlaşım-tesis yapısı adı.

Çizelge 762. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Bu ad, kuyruklardaki iletilerin saklandığı bağlantı olanağı yapısının adıdır. Adın ilk karakteri A-Z aralığında, kalan karakterler A-Z, 0-9 aralığında ya da boşluk karakterleridir.

The full name of the structure in the coupling facility is obtained by suffixing the value of the **QSGName** queue manager attribute with the value of the **CFStrucName** queue attribute.

Bu öznitelik yalnızca paylaşılan kuyruklar için geçerlidir; *QSGDisp*, *QSGDSH* değerine sahip değilse yoksayılr.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla CACFSN seçiciyi kullanın. Bu özniteliğin uzunluğu LNCFSN tarafından verilir.

**z/OS** Bu öznitelik yalnızca z/OS üzerinde desteklenir.

**ClusterChannelAd (20 baytlık karakter dizgisi)**

ClusterChannelName, bu kuyruğu bir iletim kuyruğu olarak kullanan küme gönderen kanallarının sosyal adıdır. Öznitelik, hangi küme-gönderen kanallarının bu küme iletim kuyruğundan bir küme-alıcı kanalına ileti gönderdiğini belirtir.

Çizelge 763. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Varsayılan kuyruk yöneticisi yapılandırması, tek bir iletim kuyruğundan ( SYSTEM . CLUSTER . TRANSMIT . QUEUE ) ileti göndermek için tüm küme gönderen kanalları içindir. Varsayılan yapılanış, **DefClusterXmitQueueType** kuyruk yöneticisi özniteliği değiştirilerek değiştirilebilir. Özniteliğin varsayılan değeri SCTQ' dur. Değeri CHANNEL olarak değiştirebilirsiniz. **DefClusterXmitQueueType** özniteliğini CHANNEL olarak ayarlarsanız, her küme gönderen kanalı varsayılan olarak belirli bir küme iletim kuyruğunu ( SYSTEM . CLUSTER . TRANSMIT . *ChannelName* ) kullanır.

İletim kuyruğu özniteliğini ClusterChannelName özniteliğini kümeyi gönderen bir kanala el ile ayarlayabilirsiniz. Kümeyi gönderen kanal tarafından bağlanan kuyruk yöneticisine yönlendirilen iletiler, küme gönderen kanalı tanımlayan iletim kuyruğunda saklanır. Bunlar varsayılan küme iletim kuyruğunda saklanmaz. ClusterChannelName özniteliğini boş olarak ayarlarsanız, kanal yeniden başlatıldığında kanal varsayılan küme iletim kuyruğuna geçer. Varsayılan kuyruk, kuyruk yöneticisinin değerine DefClusterXmitQueueType (Tip) özniteliğine bağlı olarak SYSTEM . CLUSTER . TRANSMIT . *ChannelName* ya da SYSTEM . CLUSTER . TRANSMIT . QUEUE şeklinde olur.

**ClusterChannelName** içinde yıldız işaretlerini ( "\*" ) belirterek, bir iletim kuyruğunu bir küme gönderen kanalları kümesiyle ilişkilendirebilirsiniz. Yıldız işaretleri, kanal adı dizgisinin başında, sonunda ya da ortasında herhangi bir sayıda yer olabilir. **ClusterChannelName** 20 karakterlik bir uzunlukla sınırlıdır: MQ\_CHANNEL\_NAME\_LENGTH.

### **IBM i** **IBM üzerinde ClusterName (48 baytlık karakter dizgisi)**

Kuyruğun ait olduğu kümenin adı.

Çizelge 764. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X		X	X	

Bu, kuyruğun ait olduğu kümenin adıdır. Kuyruk birden çok kümeye aitse, *ClusterNameList* kümeleri tanımlayan bir ad listesi nesnesinin adını belirtir ve *ClusterName* boş olur. *ClusterName* ve *ClusterNameList* ' dan en az biri boş bırakılmalı.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla CACLN seçiciyi kullanın. Bu özniteliğin uzunluğu LNCLUN tarafından verilir.

### **IBM i** **IBM üzerinde ClusterNameList (48 baytlık karakter dizgisi)**

Kuyruğun ait olduğu kümelerin adlarını içeren ad listesi nesnesinin adı.

Çizelge 765. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X		X	X	

Bu, bu kuyruğun ait olduğu kümelerin adlarını içeren bir ad listesi nesnesinin adıdır. Kuyruk yalnızca bir kümeye aitse, ad listesi nesnesi yalnızca bir ad içerir. Alternatively, *ClusterName* can be used to specify the name of the cluster, in which case *ClusterNameList* is blank. *ClusterName* ve *ClusterNameList* ' dan en az biri boş bırakılmalı.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla CACLNL seçiciyi kullanın. Bu özniteliğin uzunluğu LNCLN tarafından verilir.

### **IBM i** **IBM üzerinde CreationDate (12 baytlık karakter dizgisi)**

Kuyruk yaratıldığı tarih.

Çizelge 766. Bu özniteliğin geçerli olduğu kuyruk tipleri

Yerel	Model	Diğer Ad	Uzak	Küme
X				

Bu, kuyruğun oluşturulduğu tarihtir. Tarihin biçimi, uzunluğu 12 baytı (örneğin, 1992-09-23-- , - , tek bir boş karakteri temsil eder) izleyen iki sondaki boşlukla doldurulmuş olarak YYYY-MM-DD biçimindedir.

- IBM i' ta, bir kuyruğun oluşturulma tarihi, kuyruğu temsil eden temel işletim sistemi varlığından (dosya ya da kullanıcı alanı) farklı olabilir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla CACRTD seçiciyi kullanın. Bu özniteliğin uzunluğu LNCRTD tarafından verilir.

### IBM i **IBM üzerindeCreationTime (8 baytlık karakter dizgisi)**

Kuyruğun oluşturulduğu saat.

Çizelge 767. Bu özniteliğin geçerli olduğu kuyruk tipleri

Yerel	Model	Diğer Ad	Uzak	Küme
X				

Bu, kuyruğun yaratıldığı zamandır. Saatin biçimi, saatin 10 'dan küçük olması (örneğin, 09 . 10 . 20 gibi), 24 saatlik zaman biçimini kullanarak HH . MM . SS ' dir. Zaman yerel saattir.

- IBM i' ta, bir kuyruğun oluşturulma saati, kuyruğu temsil eden temel işletim sistemi varlığından (dosya ya da kullanıcı alanı) farklı olabilir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla CACRTT seçiciyi kullanın. Bu özniteliğin uzunluğu LNCRTT tarafından verilir.

### IBM i **IBM üzerindeCurrentQDepth (10 basamaklı işaretli tamsayı)**

Yürürlükteki kuyruk derinliği.

Çizelge 768. Bu özniteliğin geçerli olduğu kuyruk tipleri

Yerel	Model	Diğer Ad	Uzak	Küme
X				

Bu, şu anda kuyruktaki iletilerin sayısıdır. Bir MQPUT çağrısı sırasında ve bir MQGET çağrısının geri tepesi sırasında artırılır. Göz atma olmayan bir MQGET çağrısı sırasında ve bir MQPUT çağrısının geri aşımı sırasında bu değer azaltılır. Bunun etkisi, sayımın, bir iş birimi içinde kuyruğa yerleştirilmiş olan, ancak MQGET çağrısıyla alınıp alınmalarına rağmen henüz kesinleştirilmemiş iletileri içermesi olabilir. Benzer şekilde, MQGET çağrısını kullanarak bir iş birimi içinde alınan ancak henüz kesinleştirilmemiş iletileri dışlar.

Bu iletiler, son kullanma saatlerini geçmiş, ancak henüz atılmamış olan iletileri de içerir, ancak bu iletiler alınmak için uygun değildir. [“MQMD \(Message descriptor\) on IBM i” sayfa 1092](#) içinde açıklanan MDEXP alanına bakın.

Unit-of-work processing and the segmentation of messages can both cause *CurrentQDepth* to exceed *MaxQDepth*. However, this does not affect the retrievability of the messages - *Tümü* messages on the queue can be retrieved using the MQGET call in the normal way.

Bu özniteliğin değeri kuyruk yöneticisi olarak işlev görmektedir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IACDEP seçicisini kullanın.

## IBM i **IBM üzerinde DefBind (10 basamaklı işaretli tamsayı)**

Varsayılan bağ tanımı.

Çizelge 769. Bu özneliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X		X	X	X

Bu öznelik, MQOPEN çağrısında OOBNDQ belirtildiğinde ve kuyruk bir küme kuyruğu olduğunda kullanılan varsayılan bağlayıcıdır. DefBind , aşağıdaki değerlerden birine sahip olabilir:

### **BNDOPN**

Bağ tanımı, MQOPER çağrısı tarafından düzeltilen

### **BNDNOT**

Bağ tanımı düzeltilmedi.

### **BNDGRP**

Bağ tanımı, MQOPEN çağrısı tarafından düzeltilmez, ancak bir mantıksal gruptaki tüm iletiler için MQPUT ' ta düzeltilmez.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla IADBND seçicisini kullanın.

## IBM i **IBM üzerinde DefinitionType (10 basamaklı işaretli tamsayı)**

Kuyruk tanımlaması tipi.

Çizelge 770. Bu özneliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Bu, kuyruğun nasıl tanımlandığını gösterir. Değer aşağıdakilerden biridir:

### **QDPRE**

Önceden tanımlanmış kalıcı kuyruk.

Kuyruk, sistem yöneticisi tarafından oluşturulan kalıcı bir kuyruktır; yalnızca sistem yöneticisi bunu silebilir.

Önceden tanımlanmış kuyruklar, DEFINE MQSC komutu kullanılarak yaratılır ve yalnızca DELETE MQSC komutu kullanılarak silinebilir. Önceden tanımlanmış kuyruklar, model kuyruklarından yaratılamaz.

Commands can be issued either by an operator, or by an authorized user sending a command message to the command input queue (see the **CommandInputQName** attribute described in “IBM üzerinde kuyruk yöneticisine ilişkin öznelikler” sayfa 1376 ).

### **QDPERM**

Dinamik olarak tanımlı kalıcı kuyruk.

Kuyruk, bir uygulama tarafından yaratılan kalıcı bir kuyruksa, MQOD nesne tanımlayıcısında belirtilen bir model kuyruğu adıyla MQOPER çağrısı yayınlayan bir çağrıdır. Model kuyruğu tanımlaması, **DefinitionType** özneliği için QDPERM değerine sahiptir.

Bu kuyruk tipi, MQCLOSE çağrısını kullanarak silinebilir. Daha ayrıntılı bilgi için bkz. “MQCLOSE (Close object) on IBM i” sayfa 1247 .

Kalıcı dinamik kuyruğa ilişkin **QSGDisp** özneliğinin değeri QSGDQM 'dir.

## QDTEMP

Dinamik olarak tanımlanan geçici kuyruk.

Kuyruk, bir uygulama tarafından yaratılan ve MQOD nesne tanımlayıcısı MQOD ' da belirtilen bir model kuyruğu adıyla MQOPER çağrısı yayınlayan geçici bir kuyruktır. Model kuyruğu tanımı, **DefinitionType** özniteliği için QDTEMP değerini aldı.

Bu kuyruk tipi, yaratılan uygulama tarafından kapatıldığında, MQCLOSE çağrısıyla otomatik olarak silinir.

Geçici dinamik kuyruğa ilişkin **QSGDisp** özniteliğinin değeri QSGDQM 'dir.

## QDSHAR

Dinamik olarak tanımlanan paylaşılan kuyruk.

Kuyruk, bir uygulama tarafından yaratılan paylaşılan kalıcı bir kuyrukla, MQOD nesne tanımlayıcısında belirtilen bir model kuyruğu adıyla bir MQOPER çağrısı yayınlayan bir uygulama tarafından yaratılmış. Model kuyruğu tanımlaması, **DefinitionType** özniteliği için QDSHAR değerine sahiptir.

Bu kuyruk tipi, MQCLOSE çağrısını kullanarak silinebilir. Daha ayrıntılı bilgi için bkz. "MQCLOSE (Close object) on IBM i" sayfa 1247 .

Paylaşılan dinamik kuyruk için **QSGDisp** özniteliğinin değeri QSGDSH ' dir.

Bir model kuyruğu tanımlamasındaki bu öznitelik, model kuyruklarının her zaman önceden tanımlanmış olduğu için, model kuyruğunun nasıl tanımlandığını göstermez. Bunun yerine, model kuyruğundaki bu özniteliğin değeri, MQOPEN çağrısını kullanarak model kuyruğu tanımlamasından yaratılan devingen kuyrukların her birinin *DefinitionType* değerini saptamak için kullanılır.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IADEFTE seçiciyi kullanın.

## IBM i

### **IBM üzerindeDefInputOpenOption (10 basamaklı işaretli tamsayı)**

Varsayılan giriş açma seçeneği.

Çizelge 771. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Bu, kuyruğun giriş için açılacağı varsayılan yoldur. Bu, kuyruk açıldığında MQOPEN çağrısında OOINPQ seçeneği belirtilirse geçerlidir. Bu, aşağıdaki değerlerden birine sahip olabilir:

## OOINPX

Dışlayıcı erişimi olan iletileri almak için kuyruğu açın.

Kuyruk sonraki MQGET çağrılarıyla kullanılmak üzere açılmıştır. Kuyruk şu anda bu ya da herhangi bir tipte (OOINPS ya da OOINPX) giriş için başka bir uygulama tarafından açılmışsa, çağrı neden kodu RC2042 ile başarısız olur.

## OOINPS

Paylaşılan erişimi olan iletileri almak için kuyruğu açın.

Kuyruk sonraki MQGET çağrılarıyla kullanılmak üzere açılmıştır. The call can succeed if the queue is currently open by this or another application with OOINPS, but fails with reason code RC2042 if the queue is currently open with OOINPX.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IADINP seçicisini kullanın.

## IBM i

### **IBM üzerindeDefPersistence (10 basamaklı işaretli tamsayı)**

Varsayılan ileti kalıcılığı.

Çizelge 772. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X	X	X	X

Bu, kuyruktaki iletilerin varsayılan kalıcısıdır. İleti konulduğunda ileti tanımlayıcısında PEQDEF belirtilirse geçerlidir.

Kuyruk adı çözümleme yolunda birden çok tanımlama varsa, MQPUT ya da MQPUT1 çağrısı sırasında yol içindeki *ilk* tanımlamasında varsayılan kalıcılık varsayılan kalıcılık değeri alınır. Bu durum şöyle olabilir:

- Diğer ad kuyruğu
- Yerel kuyruk
- Uzak kuyruğun yerel tanımlaması
- Kuyruk yöneticisi diğer adı
- Bir iletim kuyruğu (örneğin, *DefXmitQName* kuyruğu)

Bu, aşağıdaki değerlerden birine sahip olabilir:

#### PEPER

İleti kalıcı.

Bu, ileti, kuyruk yöneticisinin sistem hatalarından ve yeniden başlatılabildiğinden emin olmak anlamına gelir. Kalıcı iletiler üzerine yerleştirilemiyor:

- Geçici dinamik kuyruklar
- Paylaşılan kuyruklar

Kalıcı iletiler kalıcı dinamik kuyruklara ve önceden tanımlanmış kuyruklara yerleştirilebilir.

#### PENPER

İleti kalıcı değil.

Diğer bir deyişle, ileti, kuyruk yöneticisinin sistem hatalarından ya da yeniden başlatılmalarından sağ olarak kurtulmaz. Bu durum, kuyruk yöneticisinin yeniden başlatılması sırasında yardımcı bellekte iletinin el değmemiş bir kopyası bulursa bile geçerlidir.

Paylaşılan kuyrukların özel durumunda, kalıcı olmayan iletiler *do* kuyruk paylaşım grubundaki kuyruk yöneticilerinin yeniden başlatılmasını sağlar, ancak paylaşılan kuyruklarda iletileri depolamak için kullanılan bağlaşım tesisinin başarısızlıklarını atlatmaz.

Hem kalıcı hem de kalıcı olmayan iletiler aynı kuyruğun üzerinde var olabilir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IADPER seçiciyi kullanın.

#### IBM i **IBM üzerindeDefPriority (10 basamaklı işaretli tamsayı)**

Varsayılan ileti önceliği.

Çizelge 773. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X	X	X	X

Kuyruktaki iletiler için varsayılan öncelik budur. Bu, ileti kuyruğa konduğunda ileti tanımlayıcısında PRQDEF değeri belirtilirse geçerlidir.

Kuyruk adı çözümleme yolunda birden çok tanımlama varsa, ileti için varsayılan öncelik, bu özniteliğin değerinden sonra, put işlemi sırasında yol içindeki *ilk* tanımlamasından alınır. Bu durum şöyle olabilir:

- Diğer ad kuyruğu



- Yerel kuyruk
- Uzak kuyruğun yerel tanımlaması
- Kuyruk yöneticisi diğer adı
- Bir iletim kuyruğu (örneğin, *DefXmitQName* kuyruğu)

Bir iletinin kuyruğa konacağı yol, kuyruğun **MsgDeliverySequence** özniteliğinin değerine bağlıdır:

- **MsgDeliverySequence** özniteliği MSPRIO ise, bir iletinin kuyruğa konacağı mantıksal konum, ileti tanımlayıcısındaki *MDPRI* alanının değerine bağlıdır.
- **MsgDeliverySequence** özniteliği MSFIFO ise, iletiler kuyruksa, ileti tanımlayıcısındaki *MDPRI* alanının değerinden bağımsız olarak, çözülmüş kuyruğun *DefPriority* değerine eşit bir önceliğe sahip oldukları gibi yerleştirilir. Ancak, *MDPRI* alanı, iletiyi koyan uygulama tarafından belirtilen değeri korur. Ek bilgi için “[Kuyruklara ilişkin öznitelikler](#)” sayfa 1345 içinde açıklanan **MsgDeliverySequence** özniteliğine bakın.

Priorities are in the range zero (lowest) through *MaxPriority* (highest); see the **MaxPriority** attribute described in “[IBM üzerinde kuyruk yöneticisine ilişkin öznitelikler](#)” sayfa 1376.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IADPRI seçiciyi kullanın.

### **IBM i IBM İleri (10 basamaklı işaretli tamsayı)DefReadtarihinde**

İstemciye teslim edilen kalıcı olmayan ileteler için varsayılan okuma tamamlama davranışını belirtir.

Çizelge 774. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X	X		

DefReadİleriye göre aşağıdaki değerlerden birine ayarlanabilir:

#### **RAHNO.**

Kalıcı olmayan ileteler, bir uygulama tarafından istekte bulunmadan önce istemciye gönderilmez. İstemci olağandışı şekilde sona ererse, en çok bir kalıcı olmayan ileti kaybedilebilir.

#### **RAHYES**

Kalıcı olmayan ileteler, bir uygulama tarafından istekte bulunmadan önce istemciye gönderilmektedir. İstemcinin olağandışı bir şekilde sona ermesi ya da istemcinin gönderdiği tüm ileteleri tüketmemesi durumunda, kalıcı olmayan ileteler kaybedilebilir.

#### **RAHDIS**

Bu kuyruk için kalıcı olmayan iletelerin etkinleştirilmesini oku. İleriye ilişkin okuma, istemci uygulaması tarafından istenip istenmemesinden bağımsız olarak istemciye gönderilmez.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IADRAH seçiciyi kullanın.

### **IBM i IBM üzerindeDefPResp (10 basamaklı işaretli tamsayı)**

Varsayılan koyma yanıt tipi (DEFPRESP) özniteliği, MQPMO içindeki PutResponsetipi PMRASQ değerine ayarlandığında, uygulamaların kullandığı değeri tanımlar. Bu öznitelik, tüm kuyruk tipleri için geçerlidir.

Çizelge 775. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X	X	X	X

Bu, aşağıdaki değerlerden birine sahip olabilir:

#### **Sync**

Koyma işlemi, zamanuyumlu olarak bir yanıt döndürerek yayınlanır.

#### **Async**

Koyma işlemi zamanuyumsuz olarak yayınlanır ve MQMD alanlarının bir alt kümesini döndürür.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IADPRT seçicisini kullanın.

### **IBM i IBM üzerinde DistLists (10 basamaklı işaretli tamsayı)**

Dağıtım listesi desteği.

Çizelge 776. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Bu, dağıtım listesi iletilerinin kuyruğa yerleştirilip yerleştirilemeyeceğini gösterir. Bu öznitelik, yerel kuyruk yöneticisine, kanalın diğer ucundaki kuyruk yöneticisinin dağıtım listelerini destekleyip desteklemediğini bildirmek için bir ileti kanalı aracısı (MCA) tarafından ayarlanır. Bu ikinci kuyruk yöneticisi ("iş ortaklığı kuyruk yöneticisi" olarak adlandırılır), bir gönderen MCA tarafından yerel iletim kuyruğundan kaldırıldıktan sonra, iletiyi alan bir sonraki kuyruk yöneticisidir.

Bu öznitelik, iş ortaklığı kuyruk yöneticisinde alınan MCA 'ya bir bağlantı kurduğunda, gönderme MCA tarafından ayarlanır. Bu şekilde, gönderilen MCA 'lar yerel kuyruk yöneticisinin iletim kuyruğu üzerinde yer alabilmesi için yalnızca iş ortaklığı kuyruk yöneticisinin doğru bir şekilde işleyebileceği iletiler olabilir.

Bu öznitelik öncelikle iletim kuyruklarıyla birlikte kullanılır, ancak tanımlanan işleme, kuyruk için tanımlanan kullanıdan bağımsız olarak gerçekleştirilir (**Usage** özniteliğe bakın).

Bu, aşağıdaki değerlerden birine sahip olabilir:

#### **DLTÜL**

Dağıtım listeleri desteklenir.

Bu, dağıtım listesi iletilerinin kuyrukta saklanabileceğini ve bu formdaki iş ortaklığı kuyruk yöneticisine iletebileceğini gösterir. Bu işlem, iletiyi birden çok hedefe göndermek için gereken işlem miktarını azaltır.

#### **DLNSUP**

Dağıtım listeleri desteklenmiyor.

Bu, ortak kuyruk yöneticisi dağıtım listelerini desteklemediğinden, dağıtım listesi iletilerinin kuyruğun üzerinde saklanamadığını gösterir. Bir uygulama dağıtım listesi ileti koyarsa ve bu ileti bu kuyruğa konacaksa, kuyruk yöneticisi dağıtım listesi iletilerini böler ve her bir iletiyi kuyruğa yerleştirir. Bu, iletiyi birden çok hedefe göndermek için gereken işleme miktarını artırır, ancak iletilerin iş ortaklığı kuyruk yöneticisi tarafından doğru bir şekilde işlenmesini sağlar.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IADIST seçiciyi kullanın. Bu özniteliğin değerini değiştirmek için MQSET çağrısını kullanın.

### **IBM i HardenGetYedekle (10 basamaklı işaretli tamsayı) IBM üzerinde**

Doğru bir geriletme sayımının sağlanıp korunmayacağı.

Çizelge 777. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Her ileti için, bir iş birimi içindeki bir MQGET çağrısıyla iletinin alındığı zaman sayısı ve daha sonra bu iş birimi geriletildiğinden bir sayı tutulur. Bu sayı, MQGET çağrısının tamamlanmasından sonra ileti tanımlayıcısında bulunan *MDBOC* alanında bulunur.

Kuyruk yöneticisi yeniden başlatıldığında, ileti geri alma sayısı kurtarılır. Ancak, sayımın doğru olduğundan emin olmak için, bu kuyruk için bir iş birimi içinde bir MQGET çağrısıyla her ileti alındığında bilgilerin

"sertleştirilmiş" (disk ya da başka bir kalıcı depolama aygıtı üzerinde kaydedilmiştir) olması gerekir. Bu işlem yapılmazsa ve kuyruk yöneticisinin başarısızlığı, MQGET çağrısının geri çıkılması ile birlikte gerçekleşirse, sayı artırılmayabilir.

Bir iş birimi içindeki her bir MQGET çağrısına ilişkin sertleşme bilgileri, bir performans maliyeti oluşturur ve **HardenGetBackout** özniteliği yalnızca sayımın doğru olması durumunda QABH olarak ayarlanmalıdır.

- IBM i' ta, bu özniteliğin ayarından bağımsız olarak, ileti geri alma sayısı her zaman sertleştirilir.

Aşağıdaki değerler olanaklı:

#### QABH

Geri sayım hatırladı.

Bu kuyruktaki iletilere ilişkin geriletme sayısının doğru olduğundan emin olmak için sertleştirme kullanılır.

#### QABNH

Geri alma sayısı anımsanmamış olabilir.

Bu kuyruktaki iletilere ilişkin geriletme sayısının doğru olduğundan emin olmak için sertleştirme kullanılmaz. Bu nedenle, sayı olması gerekenden daha düşük olabilir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IAHGB seçiciyi kullanın.

#### IBM i **IBM üzerindeInhibitGet (10 basamaklı işaretli tamsayı)**

Bu kuyruğa ilişkin alma işlemlerine izin verilip verilmediğini denetler.

Çizelge 778. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X	X		

Kuyruk bir diğer ad kuyruksa, MQGET çağrısının başarılı olması için, alma işlemi sırasında hem diğer ad, hem de temel kuyruk için alma işlemlerine izin verilmelidir. Değer aşağıdakilerden biridir:

#### QAGETI

Operasyonların engellenmesini engelliyorsunuz.

MQGET çağrıları, RC2016neden koduyla başarısız olur. Bu, GMBRWF ya da GMBRWN belirten MQGET çağrıları içerir.

**Not:** Bir iş birimi içinde çalışan bir MQGET çağrısı başarılı bir şekilde tamamlanırsa, **InhibitGet** özniteliğinin değerini QAGETI ' dan sonra değiştirerek, iş biriminin kesinleştirilmesini önlemez.

#### QAGETA

Alma işlemlerine izin verilir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IAIGET seçiciyi kullanın. Bu özniteliğin değerini değiştirmek için MQSET çağrısını kullanın.

#### IBM i **IBM üzerindeInhibitPut (10 basamaklı işaretli tamsayı)**

Bu kuyruğa ilişkin koyma işlemlerine izin verilip verilmediğini denetler.

Çizelge 779. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X	X	X	X

Kuyruk-adi çözünürlük yolunda birden fazla tanımlama varsa, MQPUT ya da MQPUT1 çağrısı başarılı olmak için, put işlemi sırasında yoldaki *her* tanımlaması (herhangi bir kuyruk yöneticisi diğer adı tanımlamaları da içinde olmak üzere) için izin verilmelidir. Bu, aşağıdaki değerlerden birine sahip olabilir:

#### QAPUTI

Put operasyonları engellenir.

MQPUT ve MQPUT1 çağrıları başarısız olur; neden kodu: RC2051.

**Not:** Bir iş birimi içinde çalışan bir MQPUT çağrısı başarıyla tamamlanırsa, **InhibitPut** özniteliğinin daha sonra QAPUTI değerine değiştirilmesi, iş biriminin kesinleştirilmesini engellemektedir.

#### QAPUTA

Put işlemlerine izin verilir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IAIPUT seçiciyi kullanın. Bu özniteliğin değerini değiştirmek için MQSET çağrısını kullanın.

#### IBM i **IBM üzerindeInitiationQName (48 byte 'lık karakter dizgisi)**

Başlatma kuyruğunun adı.

Çizelge 780. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Bu, yerel kuyruk yöneticisinde tanımlı bir kuyruğun adıdır; kuyruğun QTLOC tipinde olması gerekir. Kuyruk yöneticisi, bu özniteliğin ait olduğu kuyruğa gelen iletinin bir sonucu olarak, uygulama başlatıldığında başlatma kuyruğuna bir tetikleyici iletisi göndermektedir. Başlatma kuyruğu, tetikleme iletisinin alınmasından sonra uygun uygulamayı başlatacak bir tetikleyici izleme uygulaması tarafından izlenmelidir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla CAINIQ seçiciyi kullanın. Bu özniteliğin uzunluğu LNQN tarafından verilir.

#### IBM i **IBM üzerindeMaxMsgUzunluğu (10 basamaklı işaretli tamsayı)**

Byte cinsinden ileti uzunluğu üst sınırı.

Çizelge 781. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Bu, kuyruğa yerleştirilebilecek en uzun *fiziksel* iletinin uzunluğuna ilişkin bir üst sınırdır. Ancak, **MaxMsgLength** kuyruk özniteliği **MaxMsgLength** kuyruk yöneticisi özniteinden bağımsız olarak ayarlanabildiği için, kuyruğa yerleştirilebilecek en uzun fiziksel iletinin uzunluğuna ilişkin gerçek üst sınır, bu iki değerden küçük olan uzunluğun en küçük olan sınırlarından biri olabilir.

Kuyruk yöneticisi kesimlere ayırma özelliğini destekliyorsa, bir uygulamanın, iki **MaxMsgLength** öznitelikten daha uzun olan bir *mantıksal* iletisi koyması mümkündür; ancak uygulama MQMD ' de MFSEGA işaretini belirtirse. Bu işaret belirlenirse, mantıksal iletinin uzunluğuna ilişkin üst sınır 999 999 999 bayttır; ancak genellikle, işletim sisteminin ya da uygulamanın çalıştırıldığı ortam tarafından uygulanan kaynak kısıtlamaları, daha düşük bir sınırla sonuçlanır.

Kuyruk üzerinde çok uzun bir ileti kuyruğa alma girişimi, neden kodunda başarısız oldu:

- İleti, kuyruk için çok büyük olacak şekilde gönderiliyorsa,RC2030
- RC2031 , kuyruk yöneticisi için çok büyükse, ancak kuyruk için çok büyük değilse,

**MaxMsgLength** özniteliği için alt sınır sıfır. Üst sınır, ortam tarafından belirlenir:

- IBM i'ta ileti uzunluğu üst sınırı 100 MB' dir (104 857 600 bayt).

Daha fazla bilgi için, "[MQPUT \(Put message\) on IBM i](#)" sayfa 1311'inde açıklanan **BUFLEN** parametresine bakın.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IAMLEN seçiciyi kullanın.

### **IBM i** **IBM üzerindeMaxQDepth (10 basamaklı işaretli tamsayı)**

Kuyruk derinliği üst sınırı.

Çizelge 782. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Bu, herhangi bir zamanda kuyrukta bulunabilecek fiziksel ileti sayısı için tanımlanan üst sınırdır. *MaxQDepth* iletilerini içeren bir kuyruğa ileti koyma girişimi, neden kodu RC2053 ile başarısız olur.

İş biriminin işlenmesi ve iletilerin bölümlenmesinin her ikisi de kuyruktaki gerçek fiziksel ileti sayısının *MaxQDepth*' u aşmasına neden olabilir. However, this does not affect the retrievability of the messages - *Tümü* messages on the queue can be retrieved using the MQGET call in the normal way.

Bu özniteliğin değeri sıfır ya da daha büyük. Üst sınır, ortam tarafından belirlenir.

**Not:** Kuyruğun kuyruğunda *MaxQDepth* ' den az ileti olsa da, kuyruğun tükenmesi için kullanılabilir olan saklama alanı için bu olanak kullanılabilir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IAMDEP seçicisini kullanın.

### **IBM i** **IBM üzerindeMediaLog (10 basamaklı işaretli tamsayı)**

Günlük kapsamının tanıtıcısı (ya da IBM i üzerindeki günlük nesnesi) Belirli bir kuyruğun ortam kurtarma işlemi için gereklidir.

Çizelge 783. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Dairesel günlük kaydının kullanımda olduğu kuyruk yöneticilerinde, değer boş bir dizgi olarak döndürülür.

### **IBM i** **IBM üzerindeMsgDeliverySequence (10 basamaklı imzalı tamsayı)**

Mesaj teslim sırası.

Çizelge 784. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Bu, iletilerin MQGET çağrısıyla uygulamaya döndürüleceği sırayı belirler:

#### **MSFIFO**

İletiler FIFO sırasına geri döndürülür (ilk olarak ilk giren ilk çıkar).

Bu, bir MQGET çağrısının, iletinin önceliği ne olursa olsun, çağrıda belirtilen seçim ölçütlerine uyan ilk iletisini döndüreceği anlamına gelir.

### MSPRIO

İletiler öncelik sırasına geri döndürülür.

Bu, bir MQGET çağrısının, çağrıda belirtilen seçim ölçütlerine uygun *en yüksek öncelikli* iletiyi döndüreceği anlamına gelir. Her bir öncelik düzeyi içinde, iletiler FIFO sırasına geri döndürülür (ilk olarak ilk giren ilk çıkar).

Kuyruğun üzerinde iletiler varken ilgili öznitelikler değiştirilirse, teslim sırası aşağıdaki gibidir:

- The order in which messages are returned by the MQGET call is determined by the values of the **MsgDeliverySequence** and **DefPriority** attributes in force for the queue at the time the message arrives on the queue:
  - If *MsgDeliverySequence* is MSFIFO when the message arrives, the message is placed on the queue as though its priority were *DefPriority*. Bu, iletinin ileti tanımlayıcısındaki *MDPRI* alanının değerini etkilemez; alan, ileti ilk kez konulduğunda sahip olduğu değeri korur.
  - İleti geldiğinde *MsgDeliverySequence* MSPRIO ise, ileti, ileti tanımlayıcısındaki *MDPRI* alanı tarafından verilen önceliğe uygun olan sıraya yerleştirilir.

Kuyruktaki iletiler varken **MsgDeliverySequence** özniteliğinin değeri değiştirilirse, kuyruklardaki iletilerin sırası değiştirilmez.

Kuyrukta iletiler varken **DefPriority** özniteliğinin değeri değiştirilirse, **MsgDeliverySequence** özniteliği MSFIFO olarak ayarlansa da iletiler, FIFO sırasına göre teslim edilmeyecektir; daha yüksek öncelikli sıraya yerleştirilen kuyruğa ilk olarak yerleştirilen iletiler teslim edilir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IAMDS seçiciyi kullanın.

### IBM i **IBM üzerindeOpenInputSayı (10 basamaklı işaretli tamsayı)**

Giriş için açılan açma sayısı.

Çizelge 785. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X				

Bu, MQGET çağrısıyla kuyruktan iletileri kaldırmak için geçerli olan tanıtıcıların sayısıdır. Bu, *yerel* kuyruk yöneticisi tarafından bilinen tanıtıcı sayıların toplam sayısıdır. Kuyruk paylaşılan bir kuyruksa, yerel kuyruk yöneticisinin ait olduğu kuyruk paylaşım grubundaki diğer kuyruk yöneticilerindeki kuyruk için gerçekleştirilen giriş için sayım açılmaz.

Sayı, bu kuyruğa çözülen bir diğer ad kuyruğunun giriş için açıldığı tanıtıcıyı içerir. Sayı, giriş içermeyen işlemler için kuyruğun açıldığı noktaları içermez (örneğin, kuyruk yalnızca göz atma için açıldı).

Bu özniteliğin değeri kuyruk yöneticisi olarak işlev görmektedir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IAOIC seçicisini kullanın.

### IBM i **IBM üzerindeOpenOutputCount (10 basamaklı işaretli tamsayı)**

Çıkış için açılan açma sayısı.

Çizelge 786. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X				

Bu, MQPUT çağrısıyla kuyruğa ileti eklenmesi için geçerli olan tanıtıcıların sayısıdır. Bu, yerel kuyruk yöneticisi tarafından bilinen bu tür tanıtıcıların toplam sayısıdır; uzak kuyruk yöneticilerindeki bu kuyruk için gerçekleştirilen çıkışa ilişkin açma işlemlerini içermez. Kuyruk paylaşılan bir kuyruksa, yerel kuyruk yöneticisinin ait olduğu kuyruk paylaşım grubundaki diğer kuyruk yöneticilerindeki kuyruk için gerçekleştirilen çıkışa ilişkin sayı açılmaz.

Bu sayı, bu kuyruğa çözülen bir diğer ad kuyruğunun çıkış için açıldığı tanıtıcıyı içerir. Sayı, çıkış içermeyen işlemler için kuyruğun açıldığı noktaları içermez (örneğin, kuyruk yalnızca sorgulamak için açılır).

Bu özniteliğin değeri kuyruk yöneticisi olarak işlev görmektedir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IAOC seçicisini kullanın.

### **IBM i IBM üzerindeProcessName (48 baytlık karakter dizgisi)**

Süreç adı.

Çizelge 787. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Yerel kuyruk yöneticisinde tanımlı olan bir süreç nesnesinin adıdır. Süreç nesnesi, kuyruğa hizmet verebilen bir programı tanımlar.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla CAPRON seçiciyi kullanın. Bu özniteliğin uzunluğu LNPRON tarafından verilir.

### **IBM i IBM üzerindeQDepthHighOlayı (10 basamaklı işaretli tamsayı)**

Kuyruk Derinliği Yüksek olaylarının oluşturulup oluşturulmayacağını denetler.

Çizelge 788. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Kuyruk Derinliği Yüksek olayı, bir uygulamanın, kuyruktaki ileti sayısının kuyruk derinliği üst eşliğinden büyük ya da bu değere eşit olmasına neden olan bir kuyruğa ileti koyduğunu gösterir (**QDepthHighLimit** özniteliğine bakın).

**Not:** Bu özniteliğin değeri dinamik olarak değişebilir.

QDepthHighOlayı iki değerden birine sahip olabilir:

#### **EVRDIS**

Olay raporlama devre dışı bırakıldı.

#### **EVRENA**

Olay raporlaması etkinleştirildi.

Olaylarla ilgili daha fazla bilgi için [Olay izleme](#) başlıklı konuya bakın.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IAQDHE seçicisini kullanın.

### **IBM i IBM üzerindeQDepthHighLimiti (10 basamaklı işaretli tamsayı)**

Kuyruk derinliği üst sınırı.

Çizelge 789. Bu özneliğin geçerli olduğu kuyruk tipleri

Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Kuyruk Derinliği Yüksek bir olay oluşturmak için kuyruk derinliğinin karşılaştırıldığı eşik değeri budur. Bu olay, bir uygulamanın bir kuyruğa ileti koyduğunu ve kuyruğun ileti sayısının kuyruk derinliği üst eşikinden büyük ya da ona eşit olmasına neden olduğunu gösterir. See the **QDepthHighEvent** attribute.

Değer, kuyruk derinliği üst sınırının bir yüzdesi olarak ifade edilir (**MaxQDepth** özneliği) ve 0 ile 100 aralığında yer alıyor. Varsayılan değer 80 'dir.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla IAQDHL seçicisini kullanın.

### IBM i **IBM üzerindeQDepthLowOlayı (10 basamaklı işaretli tamsayı)**

Kuyruk Derinliği Düşük olaylarının oluşturulup oluşturulmayacağını denetler.

Çizelge 790. Bu özneliğin geçerli olduğu kuyruk tipleri

Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Kuyruk Derinliği Düşük olayı, bir uygulamanın kuyruktaki ileti sayısının, kuyruk derinliği alt eşikinden küçük ya da ona eşit olmasına neden olan bir kuyruktan ileti alındığını belirtir ( **QDepthLowLimit** özneliğine bakın).

**Not:** Bu özneliğin değeri dinamik olarak değişebilir.

QDepthLowOlayı aşağıdaki değerlerden birine sahip olabilir:

#### **EVRDIS**

Olay raporlama devre dışı bırakıldı.

#### **EVRENA**

Olay raporlaması etkinleştirildi.

Olaylarla ilgili daha fazla bilgi için [Olay izleme](#) başlıklı konuya bakın.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla IAQDLE seçiciyi kullanın.

### IBM i **IBM üzerindeQDepthLowLimiti (10 basamaklı işaretli tamsayı)**

Kuyruk derinliği için alt sınır.

Çizelge 791. Bu özneliğin geçerli olduğu kuyruk tipleri

Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Kuyruk Derinliği Düşük bir olay oluşturmak için kuyruk derinliğinin karşılaştırıldığı eşik budur. Bu olay, bir uygulamanın kuyruktan ileti aldığına ve kuyruğun kuyruk derinliği düşük eşikinden küçük ya da bu değere eşit olmasına neden olduğunu gösterir. See the **QDepthLowEvent** attribute.

Değer, kuyruk derinliği üst sınırının bir yüzdesi olarak ifade edilir (**MaxQDepth** özneliği) ve 0 ile 100 aralığında yer alıyor. Varsayılan değer 20'dir.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla IAQDLL seçicisini kullanın.



**IBM üzerinde QDepthMaxEvent (10 basamaklı işaretli tamsayı)**

Kuyruk Dolu olaylarının oluşturulup oluşturulmayacağını denetler.

Çizelge 792. Bu özneliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Kuyruk Dolu olayı, kuyruğun dolu olduğu için, kuyruğun bir kuyruğa koyma değerinin reddedildiğini, yani kuyruk derinliğinin zaten üst sınır değerine ulaştığını gösterir.

**Not:** Bu özneliğin değeri dinamik olarak değişebilir.

Bu, aşağıdaki değerlerden birine sahip olabilir:

**EVRDIS**

Olay raporlama devre dışı bırakıldı.

**EVRENA**

Olay raporlaması etkinleştirildi.

Olaylarla ilgili daha fazla bilgi için [Olay izleme](#) başlıklı konuya bakın.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla IAQDME seçicisini kullanın.

**IBM üzerinde QDesc (64 baytlık karakter dizgisi)**

Kuyruk tanımlaması.

Çizelge 793. Bu özneliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X	X	X	X

Bu alan, tanımlayıcı açıklamalar için kullanılacak bir alandır. Alanın içeriği kuyruk yöneticisi için önemli değildir; ancak kuyruk yöneticisi, alanın yalnızca görüntülenebilecek karakterleri içermesini gerektirebilir. Boş değerli karakterler içeremez; gerekirse, boşlukla doldurulabilir. Bir DBCS kuruluşunda, alan DBCS karakterleri içerebilir (en çok 64 baytlık alan uzunluğuna tabi).

**Not:** Bu alan, kuyruk yöneticisinin karakter kümesinde olmayan karakterler içeriyorsa ( **CodedCharSetId** kuyruk yöneticisi öznelmesiyle tanımlandığı gibi), bu alan başka bir kuyruk yöneticisine gönderilirse, bu karakterler yanlış çevrilebilir.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla CAQD seçicisini kullanın. Bu özneliğin uzunluğu LNQD tarafından verilir.

**IBM üzerinde QName (48 baytlık karakter dizgisi)**

Kuyruk adı.

Çizelge 794. Bu özneliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X		X	X	X

Bu ad, yerel kuyruk yöneticisinde tanımlı olan bir kuyruğun adıdır. Kuyruk adlarıyla ilgili ek bilgi için [IBM MQ nesnelilerinin adlandırılmasına ilişkin kurallar](#) başlıklı konuya bakın. Bir kuyruk yöneticisinde tanımlı olan

tüm kuyruklar, aynı kuyruk ad alanını paylaşır. Bu nedenle, QTLOC kuyruğu ve QTALS kuyruğu aynı ada sahip olamaz.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla CAQN seçiciyi kullanın. Bu özniteliğin uzunluğu LNQN tarafından verilir.

### IBM i **IBM üzerinde QServiceInterval (10 basamaklı işaretli tamsayı)**

Kuyruk hizmet aralığı hedefi.

Çizelge 795. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Bu, Hizmet Aralığı Yüksek ve Hizmet Aralığı Tamam olayları oluşturmak için karşılaştırma için kullanılan hizmet aralığından bir değer. See the **QServiceIntervalEvent** attribute.

Değer milisaniye cinsinden ve 0-999 999 999 aralığında yer alıyor.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IAQSI seçicisini kullanın.

### IBM i **IBM üzerinde QServiceIntervalOlayı (10 basamaklı işaretli tamsayı)**

Hizmet Aralığı Yüksek ya da Hizmet Aralığı Tamam olaylarının oluşturulup oluşturulmayacağını denetler.

Çizelge 796. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

- Bir Hizmet Aralığı Yüksek olayı, **QServiceInterval** özniteliğinden en az belirtilen zaman için kuyruktan herhangi bir ileti alınmadığına işaret edildiğinde oluşturulur.
- Bir Hizmet Aralığı Tamam olayı, **QServiceInterval** özniteliğinden belirtilen süre içinde kuyruktan iletilerin alındığını gösterdiğinde oluşturulur.

**Not:** Bu özniteliğin değeri dinamik olarak değişebilir.

Bu öznitelik aşağıdaki değerlerden birine sahip olabilir:

#### **QSIEHI**

Kuyruk Hizmet Aralığı Yüksek olayları etkinleştirildi.

- Kuyruk Hizmet Aralığı Yüksek olayları **etkin** ve
- Kuyruk Hizmet Aralığı Tamam olayları **devre dışı**.

#### **QSIEOK**

Kuyruk Hizmet Aralığı Tamam olayları etkinleştirildi.

- Kuyruk Hizmet Aralığı Yüksek olayları **geçersiz kılındı** ve
- Kuyruk Hizmet Aralığı Tamam olayları **etkindir**.

#### **QSIENO**

Kuyruk hizmeti aralık olayı etkinleştirilmedi.

- Kuyruk Hizmet Aralığı Yüksek olayları **geçersiz kılındı** ve
- Kuyruk Hizmet Aralığı Tamam olayları da **devre dışı bırakıldı**.

Paylaşılan kuyruklar için, bu özniteliğin değeri yoksayılr; QSIENO değeri kabul edilir.

Olaylarla ilgili daha fazla bilgi için [Olay izleme](#) başlıklı konuya bakın.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IAQSIE seçicisini kullanın.

**IBM i IBM üzerinde QSGDisp (10 basamaklı işaretli tamsayı)**

Kuyruk paylaşımı grubu yok etme.

Çizelge 797. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X		X	X	

Bu, kuyruğun yok edilmesini belirtir. Değer aşağıdakilerden biridir:

**QSGDQM**

Kuyruk yöneticisi yok etme.

Nesnenin kuyruk yöneticisi yok etme işlemi var. Bu, nesne tanımlamasının yalnızca yerel kuyruk yöneticisinde bulunduğu anlamına gelir; tanımlama, kuyruk paylaşım grubundaki diğer kuyruk yöneticilerine bilinmez.

Kuyruk paylaşım grubundaki her bir kuyruk yöneticisinin aynı adı taşıyan ve yürürlükteki nesneyle aynı olan bir nesneye sahip olması mümkündür, ancak bunlar ayrı nesnelere ve bunlar arasında ilinti yoktur. Onların özniteliklerinin birbiriyle aynı olması engellenir.

**QSGDCP**

Kopyalanan nesne yok etme.

Nesne, paylaşılan havuzda var olan bir ana nesne tanımlamasının yerel bir kopyasıdır. Kuyruk paylaşım grubundaki her kuyruk yöneticisi nesnenin kendi kopyasına sahip olabilir. Başlangıçta, tüm kopyalar aynı özniteliklere sahiptir, ancak MQSC komutlarını kullanarak her kopya değiştirilebilir, böylece öznitelikleri diğer kopyalara göre farklılık gösterebilir. Paylaşılan havuzdaki ana tanım değiştirildiğinde, kopyaların öznitelikleri yeniden uyumlulaştırılır.

**QSGDSH**

Paylaşılan yok etme.

Nesne, paylaşılmış yok etme içeriyor. Bu, paylaşılan havuzda, kuyruğun paylaşım grubundaki tüm kuyruk yöneticileri tarafından bilinen tek bir nesnenin tek bir eşgörünümü var demektir. Gruptaki bir kuyruk yöneticisi nesneye eriştiğinde, nesneye ilişkin tek paylaşılan eşgörünüme erişir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IAQSGD seçicisini kullanın.

**z/OS** Bu öznitelik yalnızca z/OS üzerinde desteklenir.

**IBM i IBM üzerinde QType (10 basamaklı işaretli tamsayı)**

Kuyruk tipi.

Çizelge 798. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X		X	X	X

Bu öznitelik aşağıdaki değerlerden birine sahiptir:

**QALS**

Diğer ad kuyruğu tanımlaması.

**QTCLUS**

Küme kuyruğu.

## QTLOC

Yerel kuyruk.

## QTREM

Uzak kuyruğun yerel tanımlaması.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IAQTYP seçicisini kullanın.

## IBM i **IBM üzerindeRemoteQMGrAdı (48 byte 'lık karakter dizgisi)**

Uzak kuyruk yöneticisinin adı.

Çizelge 799. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
			X	

This is the name of the remote queue manager on which the queue *RemoteQName* is defined. *RemoteQName* kuyruğu, QSGDCP ya da QSGDSH 'nin *QSGDisp* değerine sahipse, *RemoteQMGrName* , *RemoteQName*' un sahibi olan kuyruk paylaşım grubunun adı olabilir.

Bir uygulama uzak kuyruğun yerel tanımlamasını açarsa, *RemoteQMGrName* boş bırakılmamalı ve yerel kuyruk yöneticisinin adı olmamalıdır. *XmitQName* boşsa, iletim kuyruğu olarak *RemoteQMGrName* ile aynı adı taşıyan yerel kuyruk kullanılır. *RemoteQMGrName* adlı bir kuyruk yoksa, **DefXmitQName** kuyruk yöneticisi özniteliği tarafından tanımlanan kuyruk kullanılır.

Bu tanımlama bir kuyruk yöneticisi diğer adı için kullanılırsa, *RemoteQMGrName* , alışan olan kuyruk yöneticisi adıdır. Yerel kuyruk yöneticisinin adı olabilir. Ters durumda, açık olduğunda *XmitQName* boşsa, *RemoteQMGrName* ile aynı adı taşıyan bir yerel kuyruk olması gerekir. Bu kuyruk iletim kuyruğu olarak kullanılır.

Bu tanımlama bir yanıt-yanıt diğer adı için kullanılırsa, bu ad, *MDRM* olacak kuyruk yöneticisinin adıdır.

**Not:** Kuyruk tanımlaması yaratıldığında ya da değiştirildiğinde, bu öznitelik için belirtilen değerde geçerlilik denetimi gerçekleştirilmez.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla CARQMN seçiciyi kullanın. Bu özniteliğin uzunluğu LNQMNM tarafından verilir.

## IBM i **IBM üzerindeRemoteQName (48 baytlık karakter dizgisi)**

Uzak kuyruğun adı.

Çizelge 800. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
			X	

Bu, uzak kuyruk yöneticisinde bilindiği gibi, kuyruğun adıdır. *RemoteQMGrName*.

Bir uygulama uzak bir kuyruğun yerel tanımlamasını açarsa, açma işlemi *RemoteQName* boş bırakılmamalı.

Bu tanımlama bir kuyruk yöneticisi diğer adı tanımlaması için kullanılırsa, açma işlemi *RemoteQName* boş bırakılmalıdır.

Tanımlama bir yanıt-yanıt diğer adı için kullanıldıysa, bu ad, *MDRQ* olacak olan kuyruğun adıdır.

**Not:** Kuyruk tanımlaması yaratıldığında ya da değiştirildiğinde, bu öznitelik için belirtilen değerde geçerlilik denetimi gerçekleştirilmez.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla CARQN seçiciyi kullanın. Bu özniteliğin uzunluğu LNQN tarafından verilir.

### IBM i IBM üzerinde RetentionInterval (10 basamaklı işaretli tamsayı)

Alıkoyma aralığı.

Çizelge 801. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Bu, kuyruğun alıkonacağı süredir. Bu süre geçtikten sonra kuyruk silinmeye uygun olur.

Saat, kuyruğun oluşturulduğu tarih ve saatten başlayarak saat olarak ölçülür. Kuyruğun oluşturulma tarihi *CreationDate* 'de kaydedilir ve kuyruğun oluşturma zamanı **CreationTime** özniteliğe kaydedilir.

Bu bilgiler, artık gerekli olmayan kuyrukları tanımlamak ve silmek için bir ev bakımı uygulamasının ya da işletmenin etkinleştirilmesini sağlamak amacıyla sağlanmıştır.

**Not:** Kuyruk yöneticisi, bu özniteliğe dayalı olarak kuyrukları silmeyi ya da süresi dolmayan bir alıkoyma aralığıyla kuyrukların silinmesini önlemeye çalışmamaktadır; gereken herhangi bir işlemin yapılmasına neden olan kullanıcının sorumluluğundadır.

Kalıcı dinamik kuyrukların birikmesini önlemek için gerçekçi bir alıkoyma aralığı kullanılmalıdır (bkz. *DefinitionType*). Ancak, bu öznitelik önceden tanımlanmış kuyruklarla da kullanılabilir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IARINT seçiciyi kullanın.

### IBM i IBM üzerinde kapsam (10 basamaklı işaretli tamsayı)

Bu kuyruğa ilişkin bir girişin bir hücre dizininde de bulunup bulunmadığını denetler.

Çizelge 802. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X		X	X	

Kurulabilir bir Ad hizmeti tarafından bir hücre dizini sağlanır. Bu, aşağıdaki değerlerden birine sahip olabilir:

#### SCOM

Kuyruk yöneticisi kapsamı.

Kuyruk tanımlamasının kuyruk yöneticisi kapsamı var. Bu, kuyruğun tanımlamasının, sahip olduğu kuyruk yöneticisinin ötesine geçmediği anlamına gelir. Kuyruğu başka bir kuyruk yöneticisinden çıkış için açmak için, sahip olan kuyruk yöneticisinin adı belirtilmelidir ya da diğer kuyruk yöneticisinin kuyruğun yerel tanımlaması olmalıdır.

#### SCOMSK

Hücre kapsamı.

Kuyruk tanımlamasının hücre kapsamı var. Bu, kuyruk tanımlamasının aynı zamanda hücredeki tüm kuyruk yöneticilerinin kullanabileceği bir hücre dizinine de yerleştirileceği anlamına gelir. Kuyruk, yalnızca kuyruğun adını belirterek, hücredeki kuyruk yöneticilerinden herhangi birinden çıkış için açılabilir; kuyruk sahibi olan kuyruk yöneticisinin adı belirtilemez. Ancak, kuyruk tanımlaması, yerel tanımlama önceliğe sahip olduğu için, o adı taşıyan bir kuyruğun yerel tanımlaması olan hücredeki herhangi bir kuyruk yöneticisi tarafından kullanılamaz.

LDAP (Lightweight Directory Access Protocol; Temel Dizin Erişimi Protokolü) gibi kurulabilir bir ad hizmeti tarafından bir hücre dizini sağlanır. IBM MQ artık DCE (Distributed Computing

Environment) ad hizmetini desteklemediğini ve kuyruk tanımlamalarını bir DCE dizinine (ayrıca artık desteklenmemesine) neden olarak kullanıldığını unutmayın.

Model ve dinamik kuyrukların hücre kapsamı olamaz.

Bu değer yalnızca, bir hücre dizinini destekleyen bir ad hizmeti yapılandırıldıysa geçerlidir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IASCOP seçicisini kullanın.

Bu özniteliğe ilişkin destek aşağıdaki kısıtlamalara tabidir:

- IBM üzerinde öznitelik desteklenir, ancak yalnızca SCOQM geçerli olur.

### **IBM i IBM üzerinde paylaşılabilirlik (10 basamaklı işaretli tamsayı)**

Kuyruğun giriş için paylaşılıp paylaşılmayacağı.

Çizelge 803. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Bu, kuyruğun koşut zamanlı olarak birden çok kez giriş için açılıp açılmayacağını gösterir. Bu, aşağıdaki değerlerden birine sahip olabilir:

#### **QASHR**

Kuyruk paylaşılabilir.

OOINPS seçeneği ile birden çok açma işlemine izin verilir.

#### **QANSHR**

Kuyruk paylaşılabilir değil.

OOINPS seçeneğiyle bir MQOP çağrısı OOINPX olarak işlem görür.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IASHAR seçiciyi kullanın.

### **IBM i IBM üzerinde TriggerControl (10 basamaklı işaretli tamsayı)**

Tetik kontrolü.

Çizelge 804. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Bu, bir uygulamanın kuyruğa hizmet etmeye başlamasına neden olmak için, tetikleme iletilerinin bir başlatma kuyruğuna yazılıp yazılmayacağını denetler. Bu, aşağıdakilerden biridir:

#### **TCOFF**

Tetikleme iletileri gerekli değil.

Bu kuyruk için herhangi bir tetikleme iletileri yazılmasına izin verilmeyecek. *TriggerType* değeri, bu durumda ilgisiz bir değerdir.

#### **SIMGE**

Tetikleme iletileri gerekiyor.

Bu kuyruk için, uygun tetikleme olayları oluştuğunda, bu kuyruğa ilişkin tetikleme iletileri yazılır.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IATRGC seçiciyi kullanın. Bu özniteliğin değerini değiştirmek için MQSET çağrısını kullanın.

Verileri tetikler.

Çizelge 805. Bu özneliğin geçerli olduđu kuyruk tipleri				
Yerel	Model	Diđer Ad	Uzak	Küme
X	X			

Bu, kuyruk yöneticisinin, bu kuyruğa gelen bir ileti bir tetikleme iletilisinin başlatma kuyruğuna yazılmasına neden olduğunda, tetikleme iletilisine eklediği serbest biçimli verilerdir.

Bu verilerin içeriği kuyruk yöneticisi için bir önem göstermiyor. Bu, başlatma kuyruğunu işleyen tetikleme izleme uygulaması ya da tetikleyici izleme programı tarafından başlatılan uygulama için anlamlıdır.

Karakter dizgisi boş boş (nulls) içeremez. Gerekirse boşluklarla doldurulabilir.

Bu özneliğin değerini saptamak için, MQINQ çağrısına sahip CATRGD seçicisini kullanın. Bu özneliğin değerini değiştirmek için MQSET çağrısını kullanın. Bu özneliğin uzunluğu INTRGD tarafından verilir.

Tetik derinliği.

Çizelge 806. Bu özneliğin geçerli olduđu kuyruk tipleri				
Yerel	Model	Diđer Ad	Uzak	Küme
X	X			

Bu, bir tetikleme iletilisi yazılmadan önce kuyruğun üzerinde olması gereken, öncelikli *TriggerMsgPriority* ya da daha büyük iletilerin sayısıdır. Bu, *TriggerType* TTDPTH olarak ayarlandığında geçerlidir. *TriggerDepth* değeri bir ya da daha fazlasıdır. Bu öznelik, başka bir şekilde kullanılmamaktadır.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla IAATGD seçiciyi kullanın. Bu özneliğin değerini değiştirmek için MQSET çağrısını kullanın.

IBM MQ for IBM üzerindeki tetikleyiciler için eşik iletilisi önceliği.

Çizelge 807. Bu özneliğin geçerli olduđu kuyruk tipleri				
Yerel	Model	Diđer Ad	Uzak	Küme
X	X			

Bu, iletilerin tetikleme iletilerinin oluşturulmasına katkıda bulunmadığı (yani, kuyruk yöneticisi, tetikleme iletilisinin oluşturulup oluşturulmayacağını belirlerken bu iletileri yoksaydığı) ileti önceliğidir. *TriggerMsgPriority*, sıfır (en düşük) ile *MaxPriority* (en düşük) arasında olabilir (highest; see “IBM üzerinde kuyruk yöneticisine ilişkin öznelikler” sayfa 1376); sıfır değeri tüm iletilerin, tetikleyici iletileri oluşturulmasına katkıda bulunmasına neden olur.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla IAATGP seçiciyi kullanın. Bu özneliğin değerini değiştirmek için MQSET çağrısını kullanın.

Tetikleyici tipi.

Çizelge 808. Bu öncelikliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Bu, kuyruğa gelen iletilerin bir sonucu olarak tetikleme iletilerinin yazıldığı koşulları denetler. Değer aşağıdakilerden biridir:

#### TTNONE

Tetikleyici ileti yok.

Bu kuyruktaki iletilerin sonucu olarak hiçbir tetikleme iletisi yazılmadı. Bu, *TriggerControl* ile TCOFF ayarına aynı etkiyi sağlar.

#### TTFRST

Kuyruk derinliği 0 ile 1 arasında olduğunda ileti tetiklenir.

Kuyruk üzerindeki önceliğin *TriggerMsgPriority* ya da daha büyük ileti sayısı 0 'dan 1 'e kadar olduğunda, bir tetikleme iletisi yazılır.

#### TTEVRY

Her ileti için ileti tetiklenir.

Kuyruğa bir öncelik iletisi *TriggerMsgPriority* ya da daha büyük bir ileti geldiğinde bir tetikleme iletisi yazılır.

#### TTDPTH

Derinlik eşiği aşıldığında ileti tetiklenir.

A trigger message is written whenever the number of messages of priority *TriggerMsgPriority* or greater on the queue equals or exceeds *TriggerDepth*. Tetikleme iletisi yazıldıktan sonra, *TriggerControl* yeniden açık bir şekilde açılıncaya kadar tetiklemeyi önlemek için TCOFFTCOFF olarak ayarlanır.

Bu öncelikliğin değerini saptamak için, MQINQ çağrısıyla IAATRGT seçiciyi kullanın. Bu öncelikliğin değerini değiştirmek için MQSET çağrısını kullanın.

### IBM üzerinde kullanım (10 basamaklı işaretli tamsayı)

Kuyruk kullanımı.

Çizelge 809. Bu öncelikliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Bu, kuyruğun ne için kullanıldığını gösterir. Değer aşağıdakilerden biridir:

#### KULLANICIM

Normal kullanım.

Bu, olağan uygulamaların iletileri yerleştirirken ve alırken kullandığı bir kuyruğdur; kuyruk bir iletim kuyruğu değildir.

#### USTRAN

İletim kuyruğu.

Bu, uzak kuyruk yöneticilerine gönderilen iletileri tutmak için kullanılan bir kuyruktur. Olağan bir uygulama uzak bir kuyruğa ileti gönderdiğinde, yerel kuyruk yöneticisi iletiyi özel bir biçimde uygun iletim kuyruğunda geçici olarak saklar. Daha sonra bir ileti kanalı aracısında ileti, iletim



kuyruğundan okur ve iletiyi uzak kuyruk yöneticisine aktarır. İletim kuyruklarıyla ilgili ek bilgi için [İletim kuyrukları](#) başlıklı konuya bakın.

Yalnızca ayrıcalıklı uygulamalar, iletileri doğrudan doğruya yerleştirmek için bir iletim kuyruğunu açabilir. Normalde yalnızca yardımcı program uygulamalarının bunu yapması beklenir. İleti veri biçiminin doğru olduğu (bkz. “IBM üzerinde MQXQH (Transmission-queue header)” sayfa 1225 ) dikkatli olun, tersi durumda iletim işlemi sırasında hatalar oluşabilir. Koruyucu Bakım (PM\*) bağlam seçeneklerinden biri belirlenmedikçe, bağlam geçirilmez ya da ayarlanmaz.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IAUSAG seçiciyi kullanın.

### **IBM i** **IBM üzerinde XmitQName (48 baytlık karakter dizgisi)**

İletim kuyruğu adı.

Çizelge 810. Bu özniteliğin geçerli olduğu kuyruk tipleri				
Yerel	Model	Diğer Ad	Uzak	Küme
			X	

Bir uzak kuyruk ya da kuyruk yöneticisi diğer adı tanımlaması için açık bir durum olduğunda bu öznitelik boş değilse, iletiyi iletmek için kullanılacak yerel iletim kuyruğunun adını belirler.

*XmitQName* boşsa, iletim kuyruğu olarak *RemoteQMGrName* ile aynı adı taşıyan yerel kuyruk kullanılır. *RemoteQMGrName* adlı bir kuyruk yoksa, **DefXmitQName** kuyruk yöneticisi özniteliği tarafından tanımlanan kuyruk kullanılır.

Tanımlama, bir kuyruk yöneticisi diğer adı olarak kullanılıyorsa ve *RemoteQMGrName* yerel kuyruk yöneticisinin adı ise bu öznitelik dikkate alınmaz. Tanımlama, bir yanıt kuyruğu diğer adı tanımı olarak kullanılırsa da yoksayılır.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla CAXQN seçiciyi kullanın. Bu özniteliğin uzunluğu LNQN tarafından verilir.

## Ad listelerine ilişkin öznitelikler

Bu konuda, ad listelerine özgü öznitelikler özetlenir. Öznitelikler alfabetik sırayla açıklanır.

**Not:** Gösterilen özniteliklerin adları, MQINQ ve MQSET çağrıları ile kullanılan adlardır.

## Öznitelik açıklamaları

Bir ad listesi nesnesi aşağıdaki özniteliklere sahiptir:

### **AlterationDate (12 baytlık karakter dizgisi)**

Tanımlamanın en son değiştirildiği tarih.

Bu, tanımın en son değiştirildiği tarihtir. Tarih biçimi şöyledir: YYYY-MM-DD; uzunluğun 12 byte 'ı olması için sondaki iki boşlukla doldurulması.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla CAALTD seçiciyi kullanın. Bu özniteliğin uzunluğu LNDATE tarafından verilir.

### **AlterationTime (8 baytlık karakter dizgisi)**

Tanımlamanın en son değiştirildiği saat.

Bu, tanımın en son değiştirildiği dönemdir. Saatin biçimi şöyledir: HH.MM.SS.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla CAALTT seçiciyi kullanın. Bu özniteliğin uzunluğu LNTIME tarafından verilir.

### **NameCount (10 basamaklı işaretli tamsayı)**

Ad listesindeki ad sayısı.

Bu, sıfıra eşit ya da daha büyük bir değer. Aşağıdaki değer tanımlıdır:

#### **NCMXNL**

Bir ad listesindeki ad sayısı üst sınırı.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IANAMC seçicisini kullanın.

### **NamelistDesc (64 byte 'lık karakter dizgisi)**

Ad listesi açıklaması.

Bu alan, tanımlayıcı açıklamalar için kullanılacak bir alandır; değeri, tanımlama işlemi tarafından oluşturulur. Alanın içeriği kuyruk yöneticisi için önemli değildir; ancak kuyruk yöneticisi, alanın yalnızca görüntülenebilecek karakterleri içermesini gerektirebilir. Boş değerli karakterler içeremez; gerekirse, boşlukla doldurulabilir. Bir DBCS kuruluşunda, bu alan DBCS karakterleri içerebilir (en çok 64 bayt alan uzunluğuna bağlı).

**Not:** Bu alan, kuyruk yöneticisinin karakter kümesinde olmayan karakterler içeriyorsa (**CodedCharSetId** kuyruk yöneticisi öznitelemesiyle tanımlandığı gibi), bu alan başka bir kuyruk yöneticisine gönderilirse, bu karakterler yanlış çevrilebilir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla CALSTD seçicisini kullanın.

Bu özniteliğin uzunluğu LNNLD tarafından verilir.

### **NamelistName (48 baytlık karakter dizgisi)**

Ad listesi adı.

Bu ad, yerel kuyruk yöneticisinde tanımlı bir ad listesinin adıdır.

Her bir ad listesi, kuyruk yöneticisine ait diğer ad listelerinin adlarından farklı bir ada sahiptir, ancak farklı tiplerdeki diğer kuyruk yöneticisi nesnelere adlarını çoğaltabilir (örneğin, kuyruklar).

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla CALSTN seçicisini kullanın.

Bu özniteliğin uzunluğu LNNLN tarafından verilir.

### **Adlar (48 byte 'lık karakter dizgisi x NameCount)**

*NameCount* adlarından oluşan bir liste.

Her ad, yerel kuyruk yöneticisinde tanımlı olan bir nesnenin adıdır. Nesne adlarıyla ilgili daha fazla bilgi için [IBM MQ nesnelere adlandırma başlıklı konuya](#) bakın.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla CANAMS seçiciyi kullanın.

Listedeki her bir adın uzunluğu LNOBJN tarafından verilir.

**IBM**

## **IBM üzerindeki süreç tanımlamalarına ilişkin öznitelikler**

Bu konuda, süreç tanımlamalarına özgü öznitelikler özetlenmektedir. Öznitelikler alfabetik sırayla açıklanır.

**Not:** Gösterilen özniteliklerin adları, MQINQ ve MQSET çağrıları ile kullanılan adlardır. MQSC komutları, öznitelikleri tanımlamak, değiştirmek ya da görüntülemek için kullanıldığında, diğer kısa adlar kullanılır; ayrıntılar için [MQSC komutları](#) konusuna bakın.

## **Öznitelik açıklamaları**

Bir süreç tanımlaması nesnesi aşağıdaki özniteliklere sahiptir:

### **AlterationDate (12 baytlık karakter dizgisi)**

Tanımlamanın en son değiştirildiği tarih.

Bu, tanımın en son değiştirildiği tarihtir. Tarih biçimi şöyledir: YYYY-MM-DD; uzunluğun 12 byte 'ı olması için sondaki iki boşlukla doldurulması.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla CAALTD seçiciyi kullanın. Bu özniteliğin uzunluğu LNDATE tarafından verilir.

### **AlterationTime (8 baytlık karakter dizgisi)**

Tanımlamanın en son değiştirildiği saat.

Bu, tanımın en son değiştirildiği dönemdir. Saatin biçimi şöyledir: HH.MM.SS.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla CAALTT seçiciyi kullanın. Bu özniteliğin uzunluğu LNTIME tarafından verilir.

### **AppId (256 baytlık karakter dizgisi)**

Uygulama tanıtıcısı.

Bu, başlatılacak uygulamayı tanımlayan bir karakter dizilimidir. Bu bilgiler, başlatma kuyruğunda iletileri işleyen bir tetikleme izleme programı uygulaması tarafından kullanılır; bilgiler, tetikleme iletilerinin bir parçası olarak başlatıcı kuyruğuna gönderilir.

The meaning of *AppId* is determined by the trigger-monitor application. The trigger monitor provided by IBM MQ requires *AppId* to be the name of an executable program.

Karakter dizgisi boş boş (nulls) içeremez. Gerekirse boşluklarla doldurulabilir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla CAAPPI seçiciyi kullanın. Bu özniteliğin uzunluğu LNPROA tarafından verilir.

### **AppType (10 basamaklı işaretli tamsayı)**

Uygulama tipi.

Bu, bir tetikleme iletilerinin alınına yanıt olarak başlatılacak programın niteliyi tanımlar. Bu bilgiler, başlatma kuyruğunda iletileri işleyen bir tetikleme izleme programı uygulaması tarafından kullanılır; bilgiler, tetikleme iletilerinin bir parçası olarak başlatıcı kuyruğuna gönderilir.

*AppType* herhangi bir değere sahip olabilir. Standart tipler için aşağıdaki değerleri kullanabilirsiniz; kullanıcı tanımlı uygulama tipleri ATUFST ile ATULST arasındaki değerlerle sınırlandırılmıştır:

#### **SAATCICS**

CICS işlemi.

#### **AT400**

IBM i uygulaması.

#### **ATUFST**

Kullanıcı tanımlı uygulama tipi için en düşük değer.

#### **ATULST**

Kullanıcı tanımlı uygulama tipi için en yüksek değer.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IAAPPT seçiciyi kullanın.

### **EnvData (128 baytlık karakter dizgisi)**

Ortam verileri.

Bu dizgi, başlatılacak uygulamaya ilişkin ortamla ilgili bilgileri içeren bir karakter dizilimidir. Bu bilgiler, başlatma kuyruğunda iletileri işleyen bir tetikleme izleme programı uygulaması tarafından kullanılır; bilgiler, tetikleme iletilerinin bir parçası olarak başlatıcı kuyruğuna gönderilir.

The meaning of *EnvData* is determined by the trigger-monitor application. The trigger monitor provided by IBM MQ appends *EnvData* to the parameter list passed to the started application. Parametre listesi, MQTMC2 yapısından oluşur, ardından bir boşluk, ardından *EnvData* sonunda boşluklar kaldırılmış olur.

Karakter dizgisi boş boş (nulls) içeremez. Gerekirse boşluklarla doldurulabilir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla CAENVD seçicisini kullanın. Bu özniteliğin uzunluğu LNPROE tarafından verilir.

### **ProcessDesc (64 byte 'lık karakter dizgisi)**

Süreç açıklaması.

Bu alan, tanımlayıcı açıklamalar için kullanılabilir bir alandır. Alanın içeriği kuyruk yöneticisi için önemli değildir; ancak kuyruk yöneticisi, alanın yalnızca görüntülenebilecek karakterleri içermesini gerektirebilir. Boş değerli karakterler içeremez; gerekirse, boşlukla doldurulabilir. Bir DBCS kuruluşunda, alan DBCS karakterleri içerebilir (en çok 64 baytlık alan uzunluğuna tabi).

**Not:** Bu alan, kuyruk yöneticisinin karakter kümesinde olmayan karakterler içeriyorsa (**CodedCharSetId** kuyruk yöneticisi öznitelemesiyle tanımlandığı gibi), bu alan başka bir kuyruk yöneticisine gönderilirse, bu karakterler yanlış çevrilebilir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla CAPROD seçiciyi kullanın.

Bu özniteliğin uzunluğu INPROD tarafından verilir.

### **ProcessName (48 baytlık karakter dizgisi)**

Süreç adı.

Yerel kuyruk yöneticisinde tanımlı olan bir süreç tanımlamasının adı.

Her süreç tanımlaması, kuyruk yöneticisine ait diğer süreç tanımlamalarının adlarından farklı bir ada sahiptir. Ancak, süreç tanımlamasının adı, farklı tiplerde bulunan diğer kuyruk yöneticisi nesnelere adları (örneğin, kuyruklar) ile aynı olabilir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla CAPRON seçiciyi kullanın.

Bu özniteliğin uzunluğu LNPRON tarafından verilir.

### **UserData (128 baytlık karakter dizgisi)**

Kullanıcı verileri.

Bu dizgi, başlatılacak uygulamayla ilgili kullanıcı bilgilerini içeren bir karakter dizilimidir. Bu bilgiler, başlatma kuyruğunda iletileri işleyen bir tetikleyici izleme uygulaması ya da tetikleyici izleme programı tarafından başlatılan uygulama tarafından kullanılmak içindir. Bilgiler, tetikleme iletilerinin bir parçası olarak başlatıcı kuyruğuna gönderilir.

The meaning of *UserData* is determined by the trigger-monitor application. The trigger monitor provided by IBM MQ passes *UserData* to the started application as part of the parameter list. Parametre listesi, MQTMC2 yapısından (*UserData* içerir) oluşur, ardından bir boşluk, ardından *EnvData* sonunda boşluklar kaldırılmış olarak bulunur.

Karakter dizgisi boş boş (nulls) içeremez. Gerekirse boşluklarla doldurulabilir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla CAUSRD seçicisini kullanın. Bu özniteliğin uzunluğu LNPROU tarafından verilir.

## **IBM i IBM üzerinde kuyruk yöneticisine ilişkin öznitelikler**

Kuyruk yöneticisi özniteliklerinin bir özeti.

Bazı kuyruk yöneticisi öznitelikleri belirli uygulamalar için sabitlenirken, diğer kullanıcılar da ALTER QMGRMQSC komutu kullanılarak değiştirilebilir. Öznitelikler, DISPLAY QMGR komutu kullanılarak da görüntülenebilir. Çoğu kuyruk yöneticisi öznitelikleri özel bir OTQM nesnesi açarak ve döndürülen tanıtıcı ile MQINQ çağrısını kullanarak sorgulanabilir.

Aşağıdaki çizelge, kuyruk yöneticisine özgü öznitelikleri özetlemektedir. Öznitelikler alfabetik sırayla açıklanır.

**Not:** Bu kısımda gösterilen özniteliklerin adları, MQINQ ve MQSET çağrıları ile kullanılan adlardır. MQSC komutları, öznitelikleri tanımlamak, değiştirmek ya da görüntülemek için kullanıldığında, diğer kısa adlar kullanılır; ek bilgi için [MQSC komutları](#) konusuna bakın.

Çizelge 811. Kuyruk yöneticisine ilişkin öznitelikler

Öznitelik	Tanım
<u>AlterationDate</u>	Tanımlamanın son olarak değiştirildiği tarih
<u>AlterationTime</u>	Tanımlamanın son olarak değiştirildiği saat
<u>AuthorityEvent</u>	Yetkilendirme (Yetkili Değil) olaylarının oluşturulup oluşturulmayacağını denetler
<u>BridgeEvent</u>	IMS köprüsü olaylarının oluşturulup oluşturulmayacağını denetler
<u>ChannelAutoDef</u>	Otomatik kanal tanımlamasının kullanılmasına izin verilip verilmediğini denetler
<u>ChannelAutoDefEvent</u>	Kanal otomatik tanımlama olaylarının oluşturulup oluşturulmayacağını denetler
<u>ChannelAutoDefExit</u>	Otomatik kanal tanımlaması için kullanıcı çıkışının adı
<u>ChannelEvent</u>	Kanal olaylarının oluşturulup oluşturulmayacağını denetler
<u>ClusterCacheTipi</u>	Küme önbelleğinin boyutlarda mı, yoksa dinamik olarak boyutlandırılıp düzeltilmediğini denetler
<u>ClusterWorkloadVerileri</u>	Küme iş yükü çıkışa ilişkin kullanıcı verileri
<u>ClusterWorkloadÇıkış</u>	Küme iş yükü yönetimi için kullanıcı çıkışının adı
<u>ClusterWorkloadUzunluğu</u>	Küme iş yükü çıkışa aktarılan ileti verisi uzunluğu üst sınırı
<u>CodedCharSetId</u>	Kodlanmış karakter takımı tanıtıcısı
<u>CommandEvent</u>	Komut olay iletilerinin kuyruğa gönderilip gönderilmediğini denetler
<u>CommandInputQName</u>	Komut giriş kuyruğu adı
<u>CommandLevel</u>	Komut düzeyi
<u>ConfigurationEvent</u>	Yapılandırma olayı
<u>DeadLetterQName</u>	Ölü-mektup kuyruğunun adı
<u>DefClusterXmitQueueTipi</u>	Varsayılan küme iletim kuyruğu tipi
<u>DefXmitQName</u>	Varsayılan iletim kuyruğu adı
<u>DistLists</u>	Dağıtım listesi desteği
<u>InhibitEvent</u>	Engelleyici (Inhibit Get ve Inhibit Sput) olaylarının oluşturulup oluşturulmayacağını denetler
<u>LocalEvent</u>	Yerel hata olaylarının oluşturulup oluşturulmadığını denetler
<u>LoggerEvent</u>	Kurtarma günlüğü olaylarının oluşturulup oluşturulmayacağını denetler
<u>MaxHandles</u>	Tanıtıcı sayısı üst sınırı
<u>MaxMsgUzunluğu</u>	İleti uzunluğu üst sınırı (byte)
<u>MaxPriority</u>	En yüksek öncelik
<u>MaxUncommittedMsgs</u>	Bir iş birimi içinde kesinleştirilmemiş ileti sayısı üst sınırı
<u>PerformanceEvent</u>	Başarımla ilgili olayların oluşturulup oluşturulmayacağını denetler
<u>Altyapı</u>	Kuyruk yöneticisinin çalışmakta olduğu altyapı

Çizelge 811. Kuyruk yöneticisine ilişkin öznitelikler (devamı var)	
Öznitelik	Tanım
<u>PubSubKipi</u>	Yayınlama/abone olma motoru ve kuyruğa alınmış yayınlama/abone olma arabiriminin çalıştırılıp çalıştırılmayacağı
<u>QMgrDesc</u>	Kuyruk yöneticisi tanımı
<u>QMgrIdentifier</u>	Kuyruk yöneticisinin iç sistem tarafından oluşturulan benzersiz tanıtıcısı
<u>QMgrName</u>	Kuyruk yöneticisi adı
<u>RemoteEvent</u>	Uzak hata olaylarının oluşturulup oluşturulmadığını denetler
<u>RepositoryName</u>	Bu kuyruk yöneticisinin havuz hizmetleri sağladığı kümenin adı
<u>RepositoryNamelist</u>	Bu kuyruk yöneticisinin havuz hizmetleri sağladığı kümelerin adlarını içeren ad listesi nesnesinin adı
<u>SSLCRLNamelist</u>	Kimlik doğrulama bilgileri nesnelere adlarını içeren ad listesi nesnesinin adı (Bkz. Not 1)
<u>SSLEC</u>	TLS olaylarının oluşturulup oluşturulmayacağını denetler
<u>SSLKeyRepository</u>	TLS anahtarı havuzunun yeri (Bkz. Not 1)
<u>SSLKeyResetSayı</u>	Şifreleme anahtarı yeniden anlaşılmadan önce bir TLS iletişimde gönderilen ve alınan şifrelenmemiş baytların sayısını belirler.
<u>StartStopOlayı</u>	Başlatma ve durdurma olaylarının oluşturulup oluşturulmayacağını denetler
<u>SyncPoint</u>	Eşitleme noktası kullanılabilirliği
<u>TraceRouteKaydı</u>	İletilere ilişkin izleme rotası bilgilerinin kaydedilmesini denetler
<u>TreeLifeSaati</u>	Yönetimle ilgili olmayan konuların ömrü (saniye)
<u>TriggerInterval</u>	Tetikleme-ileti aralığı
<b>Notlar:</b>	
1. Bu öznitelik, MQINQ çağrısını kullanarak sorgulanamaz ve bu bölümde anlatılmaz. Bu özneliğe ilişkin ek bilgi için <a href="#">Kuyruk Yöneticisini Değiştirme</a> başlıklı konuya bakın.	

### **IBM i IBM üzerinde AlterationDate (12 baytlık karakter dizgisi)**

Tanımlamanın en son değiştirildiği tarih.

Bu, tanımın en son değiştirildiği tarihtir. Tarih biçimi şöyledir: YYYY-MM-DD; uzunluğun 12 byte 'ı olması için sondaki iki boşlukla doldurulması.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla CAALTD seçiciyi kullanın. Bu özneliğin uzunluğu LNDATE tarafından verilir.

### **IBM i IBM üzerinde AlterationTime (8 baytlık karakter dizesi)**

Tanımlamanın en son değiştirildiği saat.

Bu, tanımın en son değiştirildiği dönemdir. Saatin biçimi şöyledir: HH.MM.SS.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla CAALTT seçiciyi kullanın. Bu özneliğin uzunluğu LNTIME tarafından verilir.

IBM i

### **IBM üzerinde AuthorityEvent (10 basamaklı işaretli tamsayı)**

Yetkilendirme (Yetkili Değil) olaylarının oluşturulup oluşturulmayacağını denetler.

AuthorityEvent özniteliğinin şu değerlerden birine ayarlanması gerekir:

#### **EVRDIS**

Olay raporlama devre dışı bırakıldı.

#### **EVRENA**

Olay raporlaması etkinleştirildi.

Olaylarla ilgili daha fazla bilgi için [Olay izleme](#) başlıklı konuya bakın.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IAAUTE seçiciyi kullanın.

IBM i

### **IBM üzerinde BridgeEvent (karakter dizgisi)**

Bu öznitelik, IMS köprüsü olay iletilerinin SYSTEM.ADMIN.CHANNEL.EVENT kuyruğu. Yalnızca z/OS üzerinde desteklenir.

IBM i

### **Üzerinde ChannelAutoDef (10 basamaklı işaretli tamsayı) IBM i**

Otomatik kanal tanımlamasına izin verilip verilmediğini denetler.

Bu öznitelik, CTCVR ve CTSVCN tipli kanalların otomatik tanımını denetler. CTCLSD kanallarının otomatik tanımlarının her zaman etkinleştirildiğini unutmayın. Bu, aşağıdaki değerlerden birine sahip olabilir:

#### **CHADDI**

Kanal otomatik tanımlama devre dışı bırakıldı.

#### **CAYDEN**

Kanal otomatik tanımlama etkinleştirildi.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IACAD seçiciyi kullanın.

IBM i

### **IBM üzerinde ChannelAutoDefEvent (10 basamaklı işaretli tamsayı)**

Kanal otomatik tanımlama olaylarının oluşturulup oluşturulmayacağını denetler.

Bu, CTCVR, CTSVCN ve CTCLSD tipindeki kanallar için geçerlidir. Bu, aşağıdaki değerlerden birine sahip olabilir:

#### **EVRDIS**

Olay raporlama devre dışı bırakıldı.

#### **EVRENA**

Olay raporlaması etkinleştirildi.

Olaylar hakkında daha fazla bilgi için bkz. [İzleme ve performans](#).

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IACADE seçiciyi kullanın.

IBM i

### **IBM üzerinde ChannelAutoDefExit (20 baytlık karakter dizgisi)**

Otomatik kanal tanımlaması için kullanıcı çıkışının adı.

Bu ad boş değilse ve *ChannelAutoDef* değeri CHADEN değerine sahipse, kuyruk yöneticisinin kanal tanımlaması yaratmak üzere olduğu her saat çıkış olarak çağrılır. Bu, CTCVR, CTSVCN ve CTCLSD tipindeki kanallar için geçerlidir. Bundan sonra, çıkış aşağıdakilerden birini yapabilir:

- Kanal tanımlamasının yaratılmasına, değişiklik yapılmadan devam etmek için izin verin.
- Oluşturulan kanal tanımlamasının özniteliklerini değiştirin.
- Kanalın oluşturulmasını tamamen engelle.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla CACADX seçiciyi kullanın. Bu özniteliğin uzunluğu LNEXT tarafından verilir.

**IBM üzerindeChannelEvent (karakter dizgisi)**

Kanal olay iletilerinin oluşturulup oluşturulmayacağını belirler.

Bu öznitelik, kanal olay iletilerinin SYSTEM.ADMIN.CHANNEL.EVENT kuyruğu, varsa, kuyruğa yollanan ileti tipi (örneğin 'kanal başlatıldı', 'kanal durduruldu', 'kanal etkinleştirilmedi'). Bu özneliğin uygulanmasından önce, kanal olay iletilerinin kuyruğa alınmasından önlemenin tek yolu, hedef kuyruğu silmesiydi.

Bu öznitelik, yalnızca IMS köprüsü olaylarını toplamanıza da olanak tanır (artık kanal olaylarını kapatabileceğiniz için, aynı kuyruğa konmaz). Aynı durum, kanal olaylarını toplamak zorunda kalmadan da toplanabilen TLS olayları için de geçerlidir.

Bu öznitelik, yalnızca önemli olayları (örneğin, kanalların olağan bir şekilde başlatılıp durdurulmadığında değil, hataları olduğu durumlarda) toplamanıza da olanak sağlar.

ChannelEvent özneliğine ilişkin değer aşağıdakilerden biri olabilir:

- EVREXP (yalnızca şu kanal olayları oluşturulur: RC2279, RC2283, RC2284, RC2295, RC2296).
- EVRENA (tüm kanal olayları oluşturulur; bu, EVREXP, RC2282ve RC2283 olayları tarafından oluşturulan olaylara ek olarak üretilir).
- EVRDIS (hiçbir kanal olayı oluşturulmadı; bu, kuyruk yöneticisi ilk varsayılan değeridir).

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla IACHNE seçiciyi kullanın.

**IBM üzerindeClusterCacheType (32 baytlık karakter dizgisi)**

Küme önbelleğinin sabit boyutta olup olmadığını ya da dinamik olarak boyutlandırılıp boyutlandırılmayacağını denetler.

Bu, çağrıldığında küme iş yükü çıkışa geçilen, kullanıcı tanımlı 32 baytlık karakter dizilimidir. Çıkışa geçilecek veri yoksa, dizgi boş olur.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla CALWD seçiciyi kullanın.

**IBM üzerindeClusterWorkloadData (32 baytlık karakter dizgisi)**

Küme iş yükü çıkışa ilişkin kullanıcı verileri.

Bu, çağrıldığında küme iş yükü çıkışa geçilen, kullanıcı tanımlı 32 baytlık karakter dizilimidir. Çıkışa geçilecek veri yoksa, dizgi boş olur.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla CALWD seçiciyi kullanın.

**IBM üzerindeClusterWorkloadExit (20 baytlık karakter dizesi)**

Küme iş yükü yönetimine ilişkin kullanıcı çıkışının adı.

Bu ad boş değilse, çıkış, bir küme kuyruğuna her ileti konursa ya da bir kümeden diğerine taşındığında çağrılır. Çıkış, ileti için hedef olarak kuyruk yöneticisi tarafından seçilen kuyruk örneğini kabul edebilir ya da başka bir kuyruk eşgörünümü seçebilirler.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla CACLWX seçiciyi kullanın. Bu özneliğin uzunluğu LNEXTN tarafından verilir.

**IBM üzerindeClusterWorkloadUzunluğu (10 basamaklı işaretli tamsayı)**

Küme iş yükü çıkışa aktarılan ileti verisi uzunluğu üst sınırı.

Bu, küme iş yükü çıkışa geçirilen ileti verilerinin uzunluk üst sınısıdır. Çıkışa aktarılan verilerin gerçek uzunluğu, aşağıdakilerin alt sınırlarından biri olur:

- İletinin uzunluğu.
- Kuyruk yöneticisinin **MaxMsgLength** özneliği.
- **ClusterWorkloadLength** özneliği.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla IACLWL seçiciyi kullanın.



**IBM üzerinde CodedCharSetId (10 basamaklı işaretli tamsayı)**

Kodlanmış karakter takımı tanıtıcısı.

Bu, kuyruk yöneticisi tarafından, nesnelerin adları ve kuyruk yaratma tarihi ve saati gibi, MQI ' de tanımlı olan tüm karakter dizgisi alanları için kuyruk yöneticisi tarafından kullanılan karakter kümesini tanımlar. Karakter takımı, nesne adlarında geçerli olan karakterler için tek byte 'lık karakterler içeren bir karakter takımı olmalıdır. İletide taşınan uygulama verileri için geçerli değildir. Değer, ortama bağlıdır:

- IBM üzerinde, kuyruk yöneticisi ilk yaratıldığında, bu değer, ortamdaki ayarlandır.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IACCSI seçicisini kullanın.

**IBM üzerinde CommandEvent (tamsayı)**

Komutlar verildiğinde iletilerin yerel bir kuyruğa konulup konulmayacağını denetler.

Bu, iletilerin yeni bir olay kuyruğuna yazılıp yazılmayacağını denetler SYSTEM.ADMIN.COMMAND.EVENT, komutlar ne zaman verilirse. Bu özellik, komut izleme bildirim ve sorun tanılama için kullanışlıdır. CommandEvent kuyruk yöneticisi özniteliğini sorgulamak için, aşağıdaki değerlerden biriyle yeni öznitelik seçici iacev özniteliğini kullanın:

- EVRENA-komut olay iletileri oluşturulur ve tüm başarılı komutlar için kuyruğa konlanır.
- EVND-komut olay iletileri oluşturulur ve DISPLAY (MQSC) komutu dışındaki tüm başarılı komutlar için ve Sorgula (PCF) komutu için kuyruğa konlanır.
- EVRDIS-komut olay iletileri oluşturulmaz ya da kuyruğa konmaz (bu, kuyruk yöneticisinin başlangıçtaki varsayılan değeridir).

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla CMDEV seçiciyi kullanın.

**IBM üzerinde CommandInputQName (48 baytlık karakter dizgisi)**

Komut giriş kuyruğu adı.

CommandInputQName, yerel kuyruk yöneticisinde tanımlı olan komut giriş kuyruğunun adıdır. Bu işlem, kullanıcıların komut gönderebileceği bir kuyruğdur (eğer yetkiniz varsa). Kuyruğun adı ortama bağlıdır:

- IBM i' ta, kuyruğun adı SYSTEM.ADMIN.COMMAND.QUEUE(Kuyruk) ve yalnızca PCF komutları gönderilebilir. Ancak, MQSC komutu CMESC tipinde bir PCF komutu içine alındıysa, bu kuyruğa MQSC komutu gönderilebilir. Escape komutuna ilişkin ek bilgi için [Escape\(Çıkış\)](#) konusuna bakın.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla CACMDQ seçiciyi kullanın. Bu özniteliğin uzunluğu LNQN tarafından verilir.

**IBM üzerinde CommandLevel (10 basamaklı işaretli tamsayı)**

Komut Düzeyi. Bu, kuyruk yöneticisi tarafından desteklenen sistem denetimi komutlarının düzeyini gösterir.

Düzye, aşağıdaki değerlerden biridir:

**CML800**

Sistem denetimi komutlarının 800. düzeyi.

Bu değer aşağıdaki uygulamalar tarafından döndürülür:

- IBM MQ for IBM i
  - Sürüm 8.0

**CML900**

Sistem denetimi komutlarının 900. düzeyi.

Bu değer aşağıdaki uygulamalar tarafından döndürülür:

- IBM MQ for IBM i
  - Sürüm 9.0

## CML910

Sistem denetimi komutlarının düzey 910 'unu.

Bu değer aşağıdaki uygulamalar tarafından döndürülür:

- IBM MQ for IBM i
  - Sürüm 9.1

## CML920

Sistem denetimi komutlarının 920. düzeyi.

Bu değer aşağıdaki uygulamalar tarafından döndürülür:

- IBM MQ for IBM i
  - Sürüm 9.2

**CommandLevel** özniteliğinin belirli bir değerine karşılık gelen sistem denetimi komutları kümesi, **Platform** özniteliğinin değerine göre değişiklik gösterir; her ikisi de hangi sistem denetimi komutlarının destekleneceğine karar vermek için kullanılmalıdır.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IACMDL seçiciyi kullanın.

## IBM i **IBM üzerindeConfigurationEvent**

Yapılandırma olaylarının oluşturulup oluşturulmadığını ve SYSTEM.ADMIN.CONFIG.EVENT kuyruğu varsayılan nesnesi.

ConfigurationEvent özniteliği şu değerlerden biri olabilir:

- EVRENA
- EVRDIS

ConfigurationEvent özniteliği EVRENA değerine ayarlıysa ve bazı komutlar runmqsc ya da PCF tarafından başarıyla yayınlandıysa, yapılandırma olayları oluşturulur ve SYSTEM.ADMIN.CONFIG.EVENT kuyruğu. Aşağıdaki komutlara ilişkin olaylar, bir alter komutu bu işe ilişkin nesneyi değiştirmese de yayınlanır. Konfigürasyon olaylarının oluşturulduğu ve gönderildiği komutlar şunlardır:

- DEFINE/ALTER YAZAR
- TANIT/ALTER KANAL
- TANIT/ALTER ADLARI
- DEFINE/ALTER PROCESS
- DEFINE/ALTER QLOCAL (geçici bir dinamik kuyruk değilse)
- DEFINE/ALTER QMODEL/QALIAS/QREMOTE
- YAZAR BILGILERINI SIL
- KANAL SILME
- ADI SIL
- Süreci Sil
- QLOCAL 'ı DELETE (geçici bir dinamik kuyruk olmadığı sürece)
- QMODELI/QALIAS/QREMOETE SIL
- ALTER QMGR (CONFIG.EVV özniteliği geçersiz kılınmadıysa ve etkinleştirilmemişse)
- QMGR ' YI YENILE
- Geçici bir dinamik kuyruk için başka bir MQSET çağrısı.

Aşağıdaki durumlarda olaylar oluşturulmaz (etkinleştirilmişse):

- Komut ya da MQSET çağrısı başarısız olur.
- Kuyruk yöneticisi olay iletisini olay kuyruğuna koyamıyor. Komut başarıyla tamamlanmalıdır.
- Geçici dinamik kuyruklar.

- İç öznitelik değişiklikleri doğrudan ya da örtük olarak (MQSET ya da komutuna göre değil) değiştirilir; bu, TRIGGER, CURDEPTH, IPPROCS, OPPROCS, QDPHIEV, QDPLOEV, QDPMAXEV, QSVCIIEV gibi etkiler.
- Yapılanış olayı kuyruğu değiştirildiğinde, yenileme istendiğinde bu değişiklik için bir olay iletisi oluşturulur.
- Kümeleme değişiklikleri REFRESH/RESET CLUSTER ve RESUME/ASPEND QMGR komutlarıyla değişir.
- Kuyruk yöneticisi yaratılıyor ya da siliniyor.

## **IBM i** **IBM üzerinde DeadLetterQName (48 baytlık karakter dizgisi)**

Ölü harfin adı (teslim edilemeyen ileti).

Bu ad, yerel kuyruk yöneticisinde tanımlı olan bir kuyruğun adıdır. İletiler doğru hedefe yönlendirilemiyorsa, bu kuyruğa ileti gönderilir.

Örneğin, aşağıdaki durumlarda iletiler şu kuyruğa konabiliyor:

- Kuyruk yöneticisinde henüz tanımlı olmayan bir kuyruğa yollayan bir ileti kuyruk yöneticisinde gönderilir.
- Kuyruk yöneticisinde bir ileti gönderilir, ancak gönderileceği kuyruk bunu alamıyor; nedeni şunlar olabilir:
  - Kuyruk dolu
  - Put istekleri engellenir
  - Gönderme düğümünün kuyruğa ileti koyma yetkisi yok

Uygulamalar, ileti kuyruğunda da ileti yerleştirebilir.

Rapor iletileri sıradan iletilerde aynı şekilde ele alınır; rapor iletisi hedef kuyruğuna teslim edilemiyorsa (genellikle özgün iletinin ileti tanımlayıcısında MDRQ alanı tarafından belirtilen kuyruk), rapor iletisi ölü-mektup (teslim edilemeyen ileti) kuyruğuna yerleştirilir.

**Not:** Süre bitim sürelerini geçen iletiler ( "MQMD (Message descriptor) on IBM i" sayfa 1092 içinde açıklanan MDEXP alanına bakın) **değil**, atıldığında bu kuyruğa aktarılır. Ancak, gönderme uygulaması tarafından istenirse, bir süre bitimi rapor iletisi (ROEXP) oluşturulur ve MDRQ kuyruğuna gönderilir.

Put isteğini yayınlayan uygulama, MQPUT ya da MQPUT1 çağrısının döndürdüğü neden kodlarıyla ilgili olarak sorunun zamanuyumlu olarak bildirildiğini (örneğin, koyma isteklerinin engellendiğini belirten bir ileti) konulduğunda, iletiler ölü-mektup (teslim edilemeyen ileti) kuyruğuna konmaz.

Ölü-mektup (teslim edilemeyen ileti) kuyruğunda iletiler bazen bir MQDLH yapısıyla önekli olarak uygulama iletisi verilerine sahip olur. Bu yapı, iletinin ölü harf (teslim edilemeyen ileti) kuyruğuna neden yerleştirildiğini gösteren ek bilgiler içerir. Bu yapıyla ilgili daha fazla bilgi için bkz. "IBM üzerinde MQDLH (Dead-letter üstbilgisi)" sayfa 1048 .

Bu kuyruk, USNORM ' un **Usage** özneliği olan bir yerel kuyruk olmalıdır.

Bir kuyruk yöneticisi tarafından ya da bir kuyruk yöneticisi tarafından desteklenmeyen bir kuyruk (teslim edilemeyen ileti) desteklenmiyorsa, ad boyluk Yeklilariya yanYlanmYtir. Tüm IBM MQ kuyruk yöneticileri bir dead-letter (teslim edilemeyen ileti) kuyruğunu destekler, ancak varsayılan olarak tanımlı değildir.

Ölü-harf (teslim edilemeyen ileti) kuyruğu tanımlı değilse ya da başka bir nedenle dolu değilse ya da başka bir nedenden dolayı kullanılamaz durumdaysa, ileti kanalı aracısı tarafından iletim kuyruğunda yerine bir ileti kanalı aracısı tarafından aktarılmış olan bir ileti saklanır.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla CADLQ seçiciyi kullanın. Bu özneliğin uzunluğu LNQN tarafından verilir.

## **DefClusterXmitQueueType (10 basamaklı işaretli tamsayı)**

DefClusterXmitQueueType özneliği, iletileri küme alıcı kanallarına göndermek için küme gönderen kanalları tarafından varsayılan olarak hangi iletim kuyruğunun seçileceğini denetler.

DefClusterXmitQueueType değerleri MQCLXQ\_SCTQ ya da MQCLXQ\_CHANNEL' dır.

## **MQCLXQ\_SCTQ**

Tüm küme gönderen kanallar SYSTEM . CLUSTER . TRANSMIT . QUEUE' den ileti gönderir. İletim kuyruğuna yerleştirilen iletilerin correlID , iletinin gönderileceği küme gönderen kanalı belirler.

SCTQ , bir kuyruk yöneticisi tanımlandığında ayarlanır. Bu davranış, IBM WebSphere MQ 7.5' dan önceki IBM WebSphere MQ sürümlerinde örtük olarak bulunur. Önceki sürümlerde, kuyruk yöneticisi özneliği DefClusterXmitQueueType yoktu.

## **MQCLXQ\_CHANNEL**

Her küme gönderen kanal farklı bir iletim kuyruğundan ileti gönderir. Her iletim kuyruğu, SYSTEM . CLUSTER . TRANSMIT . MODEL . QUEUEmodel kuyruğundan kalıcı dinamik bir kuyruk olarak yaratılır.

If the queue manager attribute, DefClusterXmitQueueTürü, is set to KANAL, varsayılan yapılandırma, tek tek küme iletim kuyruklarıyla ilişkilendirilmiş küme gönderen kanalları olarak değiştirilir. İletim kuyrukları, SYSTEM . CLUSTER . TRANSMIT . MODEL . QUEUEmodel kuyruğundan yaratılan kalıcı dinamik kuyruklardır. Her iletim kuyruğu bir küme-gönderen kanalla ilişkilendirilir. Bir küme gönderen kanal bir küme iletim kuyruğuna hizmet sunarken, iletim kuyruğu tek bir kümede yalnızca bir kuyruk yöneticisine ilişkin iletiler içerir. Kümeleri, bir kümedeki her kuyruk yöneticisinin tek bir küme kuyruğu içereceği şekilde yapılandırabilirsiniz. Bu durumda, bir kuyruk yöneticisinden her küme kuyruğuna ileti trafiği iletilerden diğer kuyruklara ayrı olarak aktarılır.

Değeri sorgulamak için MQINQ' u arayın ya da MQIA\_DEF\_CLUSTER\_XMIT\_Q\_TYPE seçicisini ayarlamak için bir Query queue Manager ( MQCMD\_INQUIRE\_Q\_MGR) PCF komutunu gönderin. Değeri değiştirmek için, MQIA\_DEF\_CLUSTER\_XMIT\_Q\_TYPE seçicisini ayarlamak için bir Change Queue Manager ( MQCMD\_CHANGE\_Q\_MGR) PCF komutu gönderin.

### **İlgili başvurular**

[Kuyruk Yöneticisini Değiştir](#)

[Sorgu Kuyruğu Yöneticisi](#)

“MQINQ (Inquire about object attributes) on IBM i” sayfa 1283

MQINQ çağrısı, bir tamsayılar dizisi ve bir nesnenin özneliklerini içeren bir karakter dizileri kümesini döndürür.

**IBM i**

### **IBM üzerindeDefXmitQName (48 byte 'lık karakter dizisi)**

Varsayılan iletim kuyruğu adı.

Bu ad, hangi iletim kuyruğunun kullanılacağına ilişkin başka bir gösterge yoksa, iletilerin uzak kuyruk yöneticilerine iletilmesi için kullanılan iletim kuyruğunun adıdır.

Varsayılan iletim kuyruğu yoksa, ad tümüyle boştur. Bu özneliğin ilk değeri boştur.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla CADXQN seçiciyi kullanın. Bu özneliğin uzunluğu LNQN tarafından verilir.

**IBM i**

### **IBM üzerindeDistLists (10 basamaklı işaretli tamsayı)**

Dağıtım listesi desteği.

Bu, yerel kuyruk yöneticisinin MQPUT ve MQPUT1 çağrılarında dağıtım listelerini destekleyip desteklemediğini gösterir. Bu, aşağıdaki değerlerden birine sahip olabilir:

#### **DLTül**

Dağıtım listeleri desteklenir.

#### **DLNSUP**

Dağıtım listeleri desteklenmiyor.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla IADIST seçiciyi kullanın.

**IBM i**

### **IBM üzerindeInhibitEvent (10 basamaklı işaretli tamsayı)**

İnhibit (Inhibit Get ve Inhibit Put) olaylarının oluşturulmasını önler.

Bu, aşağıdaki değerlerden birine sahip olabilir:

#### **EVRDIS**

Olay raporlama devre dışı bırakıldı.

#### **EVRENA**

Olay raporlaması etkinleştirildi.

Olaylar hakkında daha fazla bilgi için bkz. [İzleme ve performans](#).

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IAINHE seçiciyi kullanın.

#### **IBM i IBM üzerinde LocalEvent (10 basamaklı işaretli tamsayı)**

Yerel hata olaylarının oluşturulup oluşturulmadığını denetler.

Değer aşağıdakilerden biridir:

#### **EVRDIS**

Olay raporlama devre dışı bırakıldı.

#### **EVRENA**

Olay raporlaması etkinleştirildi.

Olaylarla ilgili daha fazla bilgi için [Olay izleme](#) başlıklı konuya bakın.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IALCLE seçiciyi kullanın.

#### **IBM i IBM üzerinde LoggerEvent (10 basamaklı işaretli tamsayı)**

Kurtarma kaydedici olaylarının oluşturulup oluşturulmayacağını denetler.

Bu, aşağıdaki değerlerden birine sahip olabilir:

#### **Etkinleştirildi**

Günlüğe kaydedici olayları oluşturulur.

#### **DEVRE DIŞI**

Günlüğe kaydedici olayları oluşturulmaz. Kuyruk yöneticilerinin ilk varsayılan değeri budur.

Olaylar hakkında daha fazla bilgi için bkz. [İzleme ve performans](#).

#### **IBM i IBM üzerinde MaxHandles (10 basamaklı işaretli tamsayı)**

Tanıtıcı sayısı üst sınırı.

Bu, herhangi bir görevin eşzamanlı olarak kullanabileceği açık tutamaç sayısı üst sınısıdır. Tek bir kuyruk (ya da kuyruk olmayan bir nesne için) için her başarılı MQOPEN çağrısı tek bir tanıtıcı kullanır. Bu tanıtıcı, nesne kapatıldığında yeniden kullanım için kullanılabilir olur. Ancak, bir dağıtım listesi açıldığında, dağıtım listesindeki her bir kuyruk ayrı bir tanıtıcı olarak ayrılır; böylece, MQOPER çağrısı dağıtım listesindeki kuyruklar olduğu için, bu tanıtıcıyı bir çok tanıtıcı olarak kullanır. Bu, *MaxHandles* için uygun bir değer üzerinde karar verilirken dikkate alınmalıdır.

MQPUT1 çağrısı, işlenmesinin bir parçası olarak bir MQOPEN çağrısı gerçekleştirir; sonuç olarak, MQPUT1, MQXX\_ENCODE\_CASE\_ONE open işlevi kadar tanıtıcı kullanır; ancak, tutamaçlar yalnızca MQPUT1 çağrısının süresi boyunca kullanılır.

Değer, 1-999 999 999 aralığında olmalıdır. IBM i' ta varsayılan değer 256 'tir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IAMHND seçicisini kullanın.

#### **IBM i IBM üzerinde MaxMsgUzunluğu (10 basamaklı işaretli tamsayı)**

Byte cinsinden ileti uzunluğu üst sınırı.

Bu, kuyruk yöneticisi tarafından işlenebilecek en uzun *fiziksel* iletinin uzunluğudur. Ancak, **MaxMsgLength** kuyruk yöneticisi özniteliği **MaxMsgLength** kuyruk özniteliden bağımsız olarak ayarlanabildiği için, bir kuyruğa yerleştirilebilecek en uzun fiziksel ileti, bu iki değerden küçük olan en küçük fiziksel iletidir.

Kuyruk yöneticisi kesimlere ayırma özelliğini destekliyse, bir uygulamanın, iki **MaxMsgLength** özniteliğinden daha uzun olan bir *mantıksal* iletisi koyması mümkündür; ancak uygulama MQMD ' de MFSEGA işaretini belirtirse. Bu işaret belirlenirse, mantıksal iletinin uzunluğuna ilişkin üst sınır 999 999 999 bayttır; ancak genellikle, işletim sisteminin ya da uygulamanın çalıştırıldığı ortam tarafından uygulanan kaynak kısıtlamaları daha düşük bir sınıra neden olur.

**MaxMsgLength** özniteliği için alt sınır 32 KB ' dir (32 768 bayt). IBM i'ta ileti uzunluğu üst sınırı 100 MB' dir (104 857 600 bayt).

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IAMLEN seçiciyi kullanın.

### **IBM i IBM üzerindeMaxPriority (10 basamaklı işaretli tamsayı)**

Maksimum öncelik.

Bu, kuyruk yöneticisi tarafından desteklenen ileti önceliği üst sınığıdır. Öncelikler, sıfır (en düşük) ile *MaxPriority* (en yüksek) arasında değişiyor.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IAMPRI seçiciyi kullanın.

### **IBM i IBM üzerindeMaxUncommittedMsgs (10 basamaklı işaretli tamsayı)**

Bir iş birimi içinde kesinleştirilmemiş ileti sayısı üst sınırı.

Bu, bir iş birimi içinde bulunabilecek kesinleştirilmemiş ileti sayısı üst sınısıdır. Kesinleştirilmemiş ileti sayısı, yürürlükteki iş biriminin başlangıcından bu yana aşağıdaki toplamın toplamını içerir:

- Uygulama tarafından PMSYP seçeneği ile gönderilen ileteler
- Uygulama tarafından GMSYP seçeneği ile alınan ileteler
- PMSYP seçeneği içeren ileteler için kuyruk yöneticisi tarafından oluşturulan ileteleri ve COA rapor iletelerini tetikler
- GMSYP seçeneğiyle alınan ileteler için kuyruk yöneticisi tarafından oluşturulan COD raporu ileteleri

Şu ileteler kesinleştirilmemiş olarak sayılmaz:

- Uygulama tarafından bir iş biriminin dışında konarak ya da alınan ileteler
- Kuyruk yöneticisi tarafından oluşturulan ya da bir iş birimi dışında alınan iletelerin sonucu olarak, kuyruk yöneticisi tarafından oluşturulan ileteleri ya da COA/COD rapor iletelerini tetikler.
- Kuyruk yöneticisi tarafından oluşturulan süre bitim raporu ileteleri (sona erme raporu iletisinin GMSYP ' yi belirtmesine neden olsa bile)
- Kuyruk yöneticisi tarafından oluşturulan olay ileteleri (olay iletisine neden olarak PMSYP ya da GMSYP olmasına neden olsa bile)

#### **Not:**

1. Kural dışı durum rapor ileteleri Message Channel Agent (MCA) ya da uygulama tarafından oluşturulur ve bu nedenle, uygulama tarafından konulan ya da alınan sıradan ileteler ile aynı şekilde işlem görür.
2. Bir ileti ya da bölüm, PMSYP seçeneğiyle birlikte konulduğunda, kesinleştirilmemiş iletelerin sayısı, ne kadar fiziksel ileti konmasına neden olursa olsun, bu ileti bir artırılır. (Kuyruk yöneticisinin iletiyi ya da bölümü alt bölümlere ayırabilir olması gerekiyorsa, birden çok fiziksel ileti oluşabilir.)
3. Bir dağıtım listesi, PMSYP seçeneğiyle birlikte konulduğunda, kesinleştirilmemiş iletelerin sayısı, oluşturulan her fiziksel ileti için bir *tarafından artırılır*. Bu, en az biri kadar küçük ya da dağıtım listesindeki hedef sayısı kadar büyük olabilir.

Bu öznitelik için alt sınır 1; üst sınır 999 999 999 'tır.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IAMUNC seçicisini kullanın.

### **IBM i IBM üzerindePerformanceEvent (10 basamaklı işaretli tamsayı)**

Başarımla ilgili olayların oluşturulup oluşturulmayacağını denetler.

PerformanceEvent , aşağıdaki değerlerden birine sahip olabilir:

## **EVRDIS**

Olay raporlama devre dışı bırakıldı.

## **EVRENA**

Olay raporlaması etkinleştirildi.

Olaylarla ilgili daha fazla bilgi için [Olay izleme](#) başlıklı konuya bakın.

Bu özneteliğin değerini saptamak için, MQINQ çağrısıyla IAPFME seçiciyi kullanın.

## **IBM i Platform (10-digit signed integer) on IBM i**

Kuyruk yöneticisinin çalışmakta olduğu altyapı.

Kuyruk yöneticisinin üzerinde çalışmakta olduğu işletim sistemini gösterir. Değer:

### **PL400**

IBM i.

## **IBM i IBM üzerinde PubSubKipi (10 basamaklı işaretli tamsayı)**

Yayınlama/abone olma motorunun ve kuyruğa alınan yayınlama/abone olma arabiriminin çalışıp çalışmadığı, bu nedenle uygulama programlama arabirimini ve kuyruğa alınan yayınlama/abone olma arabirimi tarafından izlenmekte olan kuyrukları kullanarak uygulamaların yayınlanmasına/abone olmasına izin verilmesine olanak sağlar.

Bu, aşağıdaki değerlerden birine sahip olabilir:

### **PSMCP**

publish/subscreen motoru çalışıyor. Bu nedenle, uygulama programlama arabirimi kullanılarak yayınlama/abone olma olanaklıdır. Kuyruğa alınan yayınlama/abone olma arabirimi çalışmıyor; bu nedenle, kuyruğa alınan yayınlama/abone olma arabirimi tarafından izlenen kuyruklara yapılan tüm iletiler üzerinde işlem yapmamaktadır. Bu ayar, bu kuyruk yöneticisini kullanan WebSphere Message Broker V6 ya da önceki sürümleriyle uyumluluk için kullanılır; bunun nedeni, kuyruğa yollanmış yayınlama/abone olma arabiriminin normalde okuduğu kuyrukları okuması gerekir.

### **PSMDS**

publish/subscreen motoru ve kuyruğa alınan yayınlama/abone olma arabirimi çalışmıyor. Bu nedenle, uygulama programlama arabirimi kullanılarak yayınlama/abone olma olanaklı değildir. Kuyruğa alınan yayınlama/abone olma arabirimiyle izlenen kuyruklara yapılan yayınlama/abone olma iletileri üzerinde işlem yapmamış olur.

### **PSMEN**

publish/subscreen motoru ve kuyruğa alınan yayınlama/abone olma arabirimi çalışıyor. Bu nedenle, uygulama programlama arabirimi ve kuyruklanan yayınlama/abone olma arabirimi tarafından izlenmekte olan kuyruklar kullanılarak yayınlama/abone olma olanaklıdır. Kuyruk yöneticisinin ilk varsayılan değeri budur.

Bu özneteliğin değerini saptamak için, MQINQ çağrısıyla PSMODE seçicisini kullanın.

## **IBM i IBM üzerinde QMgrDesc (64 baytlık karakter dizgisi)**

Kuyruk yöneticisi tanımlaması.

Bu alan, tanımlayıcı açıklamalar için kullanılacak bir alandır. Alanın içeriği kuyruk yöneticisi için önemli değildir; ancak kuyruk yöneticisi, alanın yalnızca görüntülenebilecek karakterleri içermesini gerektirebilir. Boş değerli karakterler içeremez; gerekirse, boşlukla doldurulabilir. Bir DBCS kuruluşunda, bu alan DBCS karakterleri içerebilir (en çok 64 bayt alan uzunluğuna bağlı).

**Not:** Bu alan, kuyruk yöneticisinin karakter kümesinde olmayan karakterler içeriyorsa (**CodedCharSetId** kuyruk yöneticisi öznetelemesiyle tanımlandığı gibi), bu alan başka bir kuyruk yöneticisine gönderilirse, bu karakterler yanlış çevrilebilir.

IBM i' ta varsayılan değer boşluktur.

Bu özneteliğin değerini saptamak için, MQINQ çağrısıyla CAQMD seçiciyi kullanın. Bu özneteliğin uzunluğu LNQMD tarafından verilir.



IBM i

### **IBM üzerindeQMGrIdentifier (48 baytlık karakter dizgisi)**

Kuyruk yöneticisinin dahili olarak oluşturulan benzersiz tanıtıcısı.

Bu, kuyruk yöneticisi için dahili olarak oluşturulmuş benzersiz bir addır.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla CAQMID seçiciyi kullanın. Bu özniteliğin uzunluğu LNQMID tarafından verilir.

IBM i

### **IBM üzerindeQMGrName (48 baytlık karakter dizgisi)**

Kuyruk yöneticisi adı.

Bu ad, uygulamanın bağlı olduğu kuyruk yöneticisinin adı olan yerel kuyruk yöneticisinin adıdır.

Adın ilk 12 karakteri, benzersiz bir ileti tanıtıcısı oluşturmak için kullanılır ( "MQMD (Message descriptor) on IBM i" sayfa 1092 ' ta açıklanan *MDMID* alanına bakın). İletişim kurabilen kuyruk yöneticilerinin, ileti tanıtıcılarının kuyruk yöneticisi ağında benzersiz olması için ilk 12 karakterde farklı olan adları olması gerekir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla CAQMN seçiciyi kullanın. Bu özniteliğin uzunluğu LNQMN tarafından verilir.

IBM i

### **IBM üzerindeRemoteEvent (10 basamaklı işaretli tamsayı)**

Uzak hata olaylarının oluşturulup oluşturulmadığını denetler.

Değer aşağıdakilerden biridir:

#### **EVRDIS**

Olay raporlama devre dışı bırakıldı.

#### **EVRENA**

Olay raporlaması etkinleştirildi.

Olaylarla ilgili daha fazla bilgi için [Olay izleme](#) başlıklı konuya bakın.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IARMTE seçiciyi kullanın.

IBM i

### **IBM üzerindekiRepositoryName (48 byte 'lık karakter dizgisi)**

Bu kuyruk yöneticisinin havuz hizmetleri sağladığı kümenin adı.

Bu, kuyruk yöneticisinin bir havuz yöneticisi hizmeti sağladığı bir kümenin adıdır. Kuyruk yöneticisi bu hizmeti birden çok küme için sağlıyorsa, *RepositoryNameList* kümeleri tanımlayan bir ad listesi nesnesinin adını belirtir ve *RepositoryName* boş olur. *RepositoryName* ve *RepositoryNameList* ' dan en az biri boş bırakılmalı.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla CARPN seçiciyi kullanın. Bu özniteliğin uzunluğu LNQMN tarafından verilir.

IBM i

### **IBM üzerindekiRepositoryNameList (48 byte 'lık karakter dizgisi)**

Bu kuyruk yöneticisinin havuz hizmetleri sağladığı kümelerin adlarını içeren ad listesi nesnesinin adı.

Bu ad, bu kuyruk yöneticisinin bir havuz yöneticisi hizmeti sağladığı kümelerin adlarını içeren bir ad listesi nesnesinin adıdır. Kuyruk yöneticisi bu hizmeti yalnızca bir küme için sağlıyorsa, ad listesi nesnesi yalnızca bir ad içerir. Alternatively, *RepositoryName* can be used to specify the name of the cluster, in which case *RepositoryNameList* is blank. *RepositoryName* ve *RepositoryNameList* ' dan en az biri boş bırakılmalı.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla CARPNL seçiciyi kullanın. Bu özniteliğin uzunluğu LNQLN tarafından verilir.

IBM i

### **IBM üzerindeSSLEVat (karakter dizgisi)**

TLS olaylarının oluşturulup oluşturulmayacağını belirler.

Değer aşağıdakilerden biridir:



- EVRENA (MQINQ/PCF/config olayı) ENABLED (MQSC): TLS olayları oluşturulur (yani, RC2371 olayı oluşturulur).
- EVRDIS (MQINQ/PCF/config olayı) DISABLE (MQSC): TLS olayları oluşturulmaz. Kuyruk yöneticisinin ilk varsayılan değeri budur.

Bu özneliğın değeri saptamak için, MQINQ çağrısıyla IASSLE seçiciyi kullanın.

### **IBM i IBM üzerindeSSLKeyResetSayı (tamsayı)**

Bir TLS iletişiminde gönderilen ve alınan, gizli anahtar yeniden anlaşılmeden önce, gönderilen ve alınan toplam şifrenmemiş bayt sayısını belirler. Byte sayısı, ileti kanalı aracısı (MCA) tarafından gönderilen denetim bilgilerini içerir.

Bu değer yalnızca, bu kuyruk yöneticisinden iletişimi başlatan (yani, gönderici ve alıcı kanalı eşleme içindeki gönderen kanalı MCA) TLS kanalı MCA tarafından kullanılır.

Bu özneliğın değeri 0 'dan büyükse ve kanal kalp atışları bir kanal için etkinleştirilmişse, bir kanal sağlıklı işletim bildirim gönderilmeden önce gizli anahtar, veriler gönderilmeden ya da alınmadan önce yeniden görüşülür. Başarılı olan her yeniden anlaşma gerçekleştikten sonra, bir sonraki gizli anahtar yeniden anlaşma sıfırlanmaya kadar bayt sayısı sıfırlanır.

Değer, 0 ile 999 999 999 aralığında olabilir. Bu öznelik için 0 değeri, gizli anahtarın hiçbir zaman yeniden anlaşma vermediğini gösterir. 1 bayt-32 KB aralığında bir TLS gizli anahtar sıfırlama sayısı belirtirseniz, TLS kanalları 32 KB ' lik gizli anahtar sıfırlama sayısını kullanır. Bu, küçük TLS gizli anahtar ilk duruma getirme değerleri için oluşacağı aşırı anahtar sıfırlamalarının işlem maliyetinden kaçınmak içindir.

SSL sunucusu bir IBM MQ kuyruk yöneticisiyse ve hem gizli anahtar ilk duruma getirme hem de kanal kalp atışları etkin olduğunda, her kanal sağlıklı işletim bildirim hemen sonra yeniden ilişki gerçekleşir.

Bu özneliğın değeri saptamak için, MQINQ çağrısıyla IASSRC seçiciyi kullanın.

### **IBM i IBM üzerindeStartStopEvent (10 basamaklı işaretli tamsayı)**

Başlatma ve durdurma olaylarının oluşturulup oluşturulmayacağını denetler.

Bu öznelik aşağıdaki değerlerden birine sahip olabilir:

#### **EVRDIS**

Olay raporlama devre dışı bırakıldı.

#### **EVRENA**

Olay raporlaması etkinleştirildi.

Olaylarla ilgili daha fazla bilgi için [Olay izleme](#) başlıklı konuya bakın.

Bu özneliğın değeri saptamak için, MQINQ çağrısıyla IASSE seçicisini kullanın.

### **IBM i IBM üzerindeSyncPoint (10 basamaklı işaretli tamsayı)**

Uyumluluk noktası kullanılabilirliği.

Bu durum, yerel kuyruk yöneticisinin, MQGET, MQPUT ve MQPUT1 çağrılarında sahip iş birimlerini destekleyip desteklemediğini gösterir.

#### **SPAVL**

İş birimleri ve syncpoint'leme kullanılabilir.

#### **SPNAVL**

İş birimi ve uyumluluk noktası kullanılmıyor.

Bu özneliğın değeri saptamak için, MQINQ çağrısıyla IASync seçiciyi kullanın.

### **IBM i IBM üzerindeTraceRouteKaydı (10 basamaklı imzalı tamsayı)**

Bu, iletilere ilişkin bilgilerin bir kuyruk yöneticisiyle akış olarak kaydedilip kaydedilmeyeceğini denetler.

Değer aşağıdakilerden biridir:

- RECDD: İzleme rotası iletilerinin sonuna eklenmesine izin verilmez

- RECDQ: İletiler, değişmez adlandırılmış bir kuyruğa konabiliyor
- RECDM: iletiyi kullanarak belirleyin (bu, ilk varsayılan ayardır)

İzleme rotası iletilerinin sistemde kalmamasını önlemek için, sıfırdan büyük bir süre bitimi değeri ayarlayın ve RODISC rapor seçeneğini belirtin. Sistemde kalan rapor ya da yanıt iletilerini önlemek için ROPDAE rapor seçeneğini ayarlayın. Daha fazla bilgi için [“IBM i' ta rapor seçenekleri ve ileti işaretleri” sayfa 1410](#) başlıklı konuya bakın.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IATRGI seçiciyi kullanın.

### **IBM i IBM üzerindeTreeLifeSaati (10 basamaklı işaretli tamsayı)**

Yönetimle ilgili olmayan konuların başına geçecek süre (saniye).

Yönetimle ilgili olmayan konular, bir uygulama yönetim düğümü olarak var olmayan bir konu dizesine yayınlandığında ya da abone olarak abone olduğunda oluşturulanlardır. Bu yönetimle görevli olmayan düğümün artık etkin aboneliği olmadığında, bu değıştirge kuyruk yöneticisinin o düğümü kaldırmadan önce ne kadar bekleyeceğini belirler. Yalnızca, kalıcı abonelik tarafından kullanılan yönetimsel olmayan konular, kuyruk yöneticisinin geri dönüştürülmesinden sonra kalır.

0 ile 604 000 aralığında bir değeri belirleyin. 0 değeri, yönetici olmayan konuların kuyruk yöneticisi tarafından kaldırılmayacağı anlamına gelir. Kuyruk yöneticisinin ilk varsayılan değeri 1800 'dür.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IATRLFT seçiciyi kullanın.

### **IBM i IBM üzerindeTriggerInterval (10 basamaklı işaretli tamsayı)**

Tetikleme iletileri aralığı.

Tetikleme iletilerinin sayısını sınırlamak için kullanılan bir zaman aralığı (milisaniye cinsinden). Bu, yalnızca *TriggerType* TFRST olduğunda anlamlıdır. Bu durumda, tetikleme iletileri yalnızca, kuyruğa uygun bir ileti geldiğinde ve kuyruk daha önce boş olduğunda üretilir. Ancak belirli koşullar altında, kuyruk boş olmasa da TFRST tetikleyicisiyle ek bir tetikleyici iletileri oluşturulabilir. Bu ek tetikleyici iletileri her *TriggerInterval* milisaniyeden daha sık oluşturulmaz.

Tetikleme ile ilgili daha fazla bilgi için [Tetikleme kanalları](#) başlıklı konuya bakın.

Değer, 0 ile 999 999 999 arasında bir değeri aralığında yer alıyor. Varsayılan değeri 999 999 999 'tır.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IATRGI seçiciyi kullanın.

## Uygulamalar

Bu bilgiler, RPG için IBM MQ for IBM i ile birlikte gönderilen örnek programları açıklar. Ayrıca, yazdığınız programlardan yürütülebilir uygulamaların nasıl oluşturulacağı hakkında bilgi edinin.

### Uygulamanızı oluşturma

IBM i yayınları, yazdığınız programlardan yürütülebilir uygulamaların nasıl oluşturulacağını açıklar. This topic describes the additional tasks, and the changes to the standard tasks, you must perform when building IBM MQ for IBM i applications to run under IBM i.

Kaynak kodunuzda MQI çağrılarını kodlamaya ek olarak, RPG diline ilişkin IBM MQ for IBM i kopya dosyalarını eklemek için uygun dil deyimlerini eklemelisiniz. Kendinizi bu dosyaların içeriğine aşına olmalısınız; isimleri ve içerikleri hakkında kısa bir açıklama aşağıdaki metinde yer almalıydınız.

### **IBM i IBM MQ copy files on IBM i**

IBM MQ for IBM i , uygulamalarınızı RPG programlama diline yazmanıza yardımcı olacak kopya dosyaları sağlar. Bunlar, WebSphere Development toolset (5722 WDS) ILE RPG 4 Compiler ile birlikte kullanıma uygundur.

The copy files that IBM MQ for IBM i provides to assist with the writing of channel exits are described in [İleti alışverişi kanallarına ilişkin kanal çıkışı programları](#).

RPG ' ye ilişkin IBM MQ for IBM i kopyası dosyalarının adları CMQ öneki ne sahiptir. G ya da H soneki var. Adlandırılmış değişmezleri içeren ayrı kopya dosyaları ve her bir yapı için bir dosya vardır. Kopyalama dosyaları [“Dil bilgileri” sayfa 994](#) içinde listelenir.

**Not:** ILE RPG/400 için, dosya üyeleri olarak sağlanır. QMQM kitaplığındaki QRPGLSRC.

Yapı bildirimlerinde DS deyimi yoktur. Bu, uygulamanın DS deyimini kodlayarak ve bildirim geri kalan kısmında kopyalanacak /KOPYALA deyimini kullanarak bir veri yapısı (ya da birden çok oluşumlu veri yapısı) bildirmesini sağlar:

ILE RPG/400 için deyim aşağıdaki gibi olur:

```
D*..1.....2.....3.....4.....5.....6.....7
D* Declare an MQMD data structure
D MQMD          DS
D/COPY CMQMDG
```

### ***Programlarınızı çalıştırmak üzere hazırlama***

Yürütülebilir bir IBM MQ for IBM i uygulaması oluşturmak için, yazdığınız kaynak kodu derlemeniz gerekir.

ILE RPG/400 için bu işlemi yapmak için tipik IBM i komutlarını, CRTRPGMOD ve CRTPGM ' yi kullanabilirsiniz.

\*MOBILY değerini yarattıktan sonra, CRTPGM komutunda BNDSRVPGM (QMQM/LIBMQM) değerini belirtmeniz gerekir. Bu, programınızdaki çeşitli IBM MQ yordamlarını içerir.

Derleme işlemi gerçekleştirdiğinizde, kopya dosyalarını (QMQM) içeren kitaplığın kitaplık listesinde yer aldığından emin olun.

İstemci kipleri de içinde olmak üzere programlama konuları hakkında daha fazla bilgi için bkz. [“Dil bilgileri” sayfa 994](#).

### ***IBM i dış eşitleme noktası yöneticisine arabirimler***

IBM MQ for IBM i , dış eşitleme noktası eşgüdücüsü olarak yerel IBM i kesinleştirme denetimini kullanır.

IBM i ' in kesinleştirme denetimi yetenekleriyle ilgili daha fazla bilgi için *IBM i Programming: Backup and Recovery Guide* adlı yayına bakın.

IBM i kesinleştirme denetimi olanaklarını başlatmak için STRCMTCTL sistem komutunu kullanın. Kesinleştirme denetimini sona erdirmek için ENDCMTCTL sistem komutunu kullanın.

**Not:** Varsayılan değer olan *Kesinleştirme tanımlaması kapsamı* \*ACTGRP değeridir. This must be defined as \*JOB for IBM MQ for IBM i. Örneğin:

```
STRCMTCTL LCKLVL(*ALL) CMTSCOPE(*JOB)
```

Kesinleştirme denetimini başlattıktan sonra, PMSYP ya da GMSYP belirtilerek MQPUT ya da MQGET ögesini çağırırsanız, IBM MQ for IBM i kendisini kesinleştirme tanımına bir API kesinleştirme kaynağı olarak ekler. Bu, genellikle bir işteki ilk çağrıdır. Belirli bir kesinleştirme tanımı altında kayıtlı herhangi bir API kesinleştirme kaynağı varsa, bu tanım için kesinleştirme denetimini sona erdiremezsiniz.

IBM MQ for IBM i , yürürlükteki iş biriminde bekleyen bir MQI işlemi olmaması koşuluyla, kuyruk yöneticisiyle bağlantınızı kestiğinizde API kesinleştirme kaynağı olarak kaydını kaldırır.

Yürürlükteki iş biriminde bekleyen MQPUT, MQPUT1 ya da MQGET işlemleri varken kuyruk yöneticisinden bağlantıyı kesiyorsanız, IBM MQ for IBM i , bir sonraki kesinleştirme ya da geriye işleme işleminin bildirmesi için bir API kesinleştirme kaynağı olarak kayıtlı kalır. Bir sonraki eşitleme noktasına ulaşıldığında IBM MQ , değişiklikleri gerektiği şekilde kesinleştirir veya geri alır. Bir uygulamanın etkin

bir iş birimi sırasında bir kuyruk yöneticisine bağlanmasını ve yeniden bağlanmasını ve aynı iş birimi içinde daha fazla MQGET ve MQPUT işlemleri gerçekleştirmesi mümkündür (bu, beklemedeki bir kopumadır).

Söz konusu kesinleştirme tanımı için bir ENDCMTCTL sistem komutu yayınlamaya çalışırsanız, bekleyen değişikliklerin etkin olduğunu belirten CPF8355 iletisi yayınlanır. Bu ileti, iş sona erdiğinde iş günlüğünde de görüntülenir. Bunu önlemek için, beklemedeki tüm IBM MQ işlemlerini kesinleştirdiğinizden ya da geri yüklediğinizden ve kuyruk yöneticisinden bağlantınızı sağladığınızdan emin olun. Bu nedenle, ENDCMTCTL öncesinde COMMIT ya da ROLLBACK komutlarının kullanılması, son kesinleştirme denetiminin başarıyla tamamlanmasını sağlar.

IBM i kesinleştirme denetimi bir dış eşitleme noktası eşgüdümçüsü olarak kullanıldığında, MQCMIT, MQBACK ve MQBEGIN çağruları yayınlanmayabilir. Bu işlemlere yapılan aramalar neden kodu RC2012 ile başarısız olur.

Geri almak ya da geri almak (yani, geri dönmek) için, iş biriminiz, kesinleştirme denetimini destekleyen programlama dillerinden birini kullanın. Örneğin:

- CL komutları: COMMIT ve ROLLBAC
- ILE C Programming İşlevleri: \_Rcommit ve \_Rrollback
- RPG/400: COMMIT ve ROLBK
- COBOL/400: COMMIT ve ROLLBAC

### ***Syncpoints in CICS for IBM i applications***

IBM MQ for IBM i participates in units of work with CICS. Yürürlükteki iş biriminin içine ileti koymak ve iletileri almak için bir CICS uygulaması içindeki MQI ' yı kullanabilirsiniz.

You can use the EXEC CICS SYNCPOINT command to establish a syncpoint that includes the IBM MQ for IBM i operations. Önceki eşitleme noktasına kadar olan tüm değişiklikleri geri almak için, EXEC CICS SYNCPOINT ROLLBACK komutunu kullanabilirsiniz.

If you use MQPUT, MQPUT1, or MQGET with the PMSYP, or GMSYP , option set in a CICS application, you cannot log off CICS until IBM MQ for IBM i has removed its registration as an API commitment resource. Bu nedenle, kuyruk yöneticisinden bağlantıyı kesmeden önce, beklemedeki bir put ya da alma işlemini kesinleştirmeniz ya da geri almamanız gerekir. Bu, CICS oturumunu kapatmanıza olanak tanır.

## **IBM üzerinde örnek programlar**

Bu konuda, RPG için IBM MQ for IBM i ile verilen örnek programlar açıklanmaktadır. Örnekler, Message Queue Interface (MQI) ' in tipik kullanımları gösterir.

Örnekler, genel programlama tekniklerini göstermek üzere tasarlanmadığından, bir üretim programına dahil etmek isteyebileceğiniz bazı hata denetimi atlandı. Ancak, bu örnekler kendi ileti kuyruklama programlarınız için temel olarak kullanıma uygundur.

Tüm örneklere ilişkin kaynak kodu ürünle birlikte sağlanır; bu kaynak, programlarda gösterilen ileti kuyruklama tekniklerini açıklayan yorumları içerir.

Bir ILE örnek programı kümesi vardır:

### **1. MQI ' ye prototip çağrılar kullanan programlar (statik bağlı çağrılar)**

Kaynak QMQMSAMP/QRPGLESRC içinde var. Üyeler AMQ3xxx4 adını taşır; burada xxx , örnek işlevi belirtir. Kopyalama üyeleri QMQM/QRPGLESRC içinde var olabilir. Her üye adı, G ya da Hsonekine sahiptir.

Çizelge 812 sayfa 1393 , IBM MQ for IBM i ile birlikte teslim edilen örnek programların tam bir listesini verir ve desteklenen programlama dillerinin her birindeki programların adlarını gösterir. Adlarının tümünün AMQ öneki ile başladığına dikkat edin. Addaki dördüncü karakter, programlama dilini gösterir.

<i>Çizelge 812. Örnek programların adları</i>	
	<b>RPG (ILE)</b>
Örnekleri koy	AMQ3PUT4
Örnekleri göz at	AMQ3GBR4
Örnekleri al	AMQ3GET4
İstek örnekleri	AMQ3REQ4
Yankı örnekleri	AMQ3ECH4
Sorgu örnekleri	AMQ3INQ4
Örnekleri ayarla	AMQ3SET4
Tetikleyici İzleyicisi örneği	AMQ3TRG4
Tetikleyici Sunucusu örneği	AMQ3SRV4

Bunlara ek olarak, IBM MQ for IBM i örnek seçeneği, bazı örnek programlar ve yönetim görevlerini gösteren örnek CL programlarına giriş olarak kullanılacak bir örnek veri dosyası olan AMQSDATA ' yı da içerir. CL örnekleri, IBM uygulamasını yönetme içinde açıklanmıştır. Bu konuda açıklanan örnek programlarla kullanmak üzere kuyruklar yaratmak için örnek CL programını kullanabilirsiniz.

Örnek programların nasıl çalıştırılabileceği hakkında bilgi için bkz. [“Preparing and running the sample programs on IBM i”](#) sayfa 1394.

### **IBM üzerinde örnek programlar içinde gösterilen özellikler**

IBM MQ for IBM i örnek programları tarafından gösterilen teknikleri gösteren bir çizelge.

Bazı teknikler birden çok örnek programda ortaya çıkar, ancak çizelgede yalnızca bir program listelenir. MQOPER ve MQCLOSE çağrılarını kullanan tüm örnekler açık ve kapatılır, bu nedenle bu teknikler tabloda ayrı olarak listelenmez.

<i>Çizelge 813. MQI ' nin (MQI) kullanımını gösteren örnek programlar</i>	
<b>Teknik</b>	<b>RPG (ILE)</b>
MQCONN ve MQDISC çağrılarının kullanılması	AMQ3ECH4 ya da AMQ3INQ4
Örtük olarak bağlanma ve bağlantı kesme	AMQ3PUT4
MQPUT çağrısını kullanarak ileti koyma	AMQ3PUT4
MQPUT1 çağrısını kullanarak tek bir ileti koyma	AMQ3ECH4 ya da AMQ3INQ4
İstek iletileri yanıtlanma	AMQ3INQ4
İleti alınıyor (bekleme yok)	AMQ3GBR4
İletileri alma (bir süre sınırlaması ile bekleme)	AMQ3GET4
İletileri alma (veri dönüştürme ile)	AMQ3ECH4
Kuyruğa Göz Atma	AMQ3GBR4
Paylaşılan giriş kuyruğunun kullanılması	AMQ3INQ4
Dışlayıcı bir giriş kuyruğunu kullanma	AMQ3REQ4
MQINQ çağrısının kullanılması	AMQ3INQ4
MQSET çağrısının kullanılması	AMQ3SET4

Çizelge 813. MQI ' nin (MQI) kullanımını gösteren örnek programlar (devamı var)	
Teknik	RPG (ILE)
Yanıtlama Kuyruğu Kullanılması	AMQ3REQ4
Kural dışı durum iletileri isteniyor	AMQ3REQ4
Kesilmiş bir iletinin kabul edilmesi	AMQ3GBR4
Çözülmüş bir kuyruk adının kullanılması	AMQ3GBR4
Tetikleme işlemi	AMQ3SRV4 ya da AMQ3TRG4

**Not:** Tüm örnek programlar, işlenmenin sonuçlarını içeren bir kuyruğa yollanmış dosya üretir.

### ***Preparing and running the sample programs on IBM i***

IBM MQ for IBM i örnek programlarını çalıştırabilmeniz için, bunları diğer IBM MQ for IBM i uygulamalarıyla derlemeniz gerekir. To do so, you can use the IBM i commands CRTRPGMOD and CRTPGM.

AMQ3xxx4 programlarını oluşturduğunuzda, CRTPGM komutuna BNDSRVPGM (QMOM/LIBMOM) belirtmeniz gerekir. Bu işlem, programınızdaki çeşitli IBM MQ yordamlarını içerir.

Örnek programlar QMQMSAMP kitaplığında QRPGLSRC üyeleri olarak sağlanır. Bunlar, QMOM kitaplığında bulunan kopya dosyalarını kullanırlar, bu nedenle bunları derlerken bu kitaplığın kitaplık listesinde yer aldığından emin olun. RPG derleyicisi, bilgi iletileri verir; örnekler, kopya dosyalarında bildirilmiş olan değişkenlerin çoğunu kullanmaz.

### **Örnek programların çalıştırılması**

Örnekleri çalıştırırken kendi kuyruklarınızı kullanabilir ya da bazı örnek kuyruklar oluşturmak için AMQSAMP4 derleyip çalıştırabilirsiniz. Bu program için kaynak, QMQMSAMP kitaplığındaki QCLSRC dosyasına gönderilir. Bu komut, CRTCLPGM komutu kullanılarak derlenebilir.

Örnek programlardan birini çağırmak için aşağıdaki gibi bir komut kullanın:

```
CALL PGM(QMQMSAMP/AMQ3PUT4) PARM('Queue_Name','Queue_Manager_Name')
```

Burada Queue\_Name ve Queue\_Manager\_Name , 48 karakter uzunluğunda olmalı ve Queue\_Name ve Queue\_Manager\_Name ' yi gereken sayıda boşlukla doldurarak elde etmeniz gerekir.

Sorgulamak ve Ayarlamak için örnek programları, AMQSAMP4 ile yaratılan örnek tanımlamaları, bu örneklerin C sürümlerinin tetiklenmesine neden olur. RPG sürümlerini tetiklemek istiyorsanız, SYSTEM.SAMPLE.ECHOPROCESS ve SYSTEM.SAMPLE.INQPROCESS ve SYSTEM.SAMPLE.SETPROCESS işlem tanımlarını değiştirmeniz gerekir. CHGMQMPCRC komutunu ( [Change MQ Process \(CHGMQMPCRC\)](#) içinde anlatılan) kullanabilirsiniz. Bunu yapmak ya da alternatif tanımlamayla AMQSAMP4 dosyasını düzenlemek ya da çalıştırmak için.

### ***The Put sample program on IBM i***

Put örnek programı ( AMQ3PUT4), MQPUT çağrısını kullanan bir kuyruğa ileti yerleştirir.

Programı başlatmak için, programı çağırın ve hedef kuyruğunuzun adını program parametresi olarak verin. Program, kuyrukta bir dizi sabit ileti yerleştiriyor; bu iletiler, program kaynak kodunun sonundaki veri bloğundan alınır. QMQMSAMP kitaplığındaki örnek bir örnek program AMQ3PUT4 ' dir.

Bu örnek programı kullanarak komut şöyle olur:

```
CALL PGM(QMQMSAMP/AMQ3PUT4) PARM('Queue_Name','Queue_Manager_Name')
```

Burada Queue\_Name ve Queue\_Manager\_Name , 48 karakter uzunluğunda olmalı ve Queue\_Name ve Queue\_Manager\_Name ' yi gereken sayıda boşlukla doldurarak elde etmeniz gerekir.

## Put Sample programının tasarımı

Program, iletileri koymak üzere hedef kuyruğu açmak için OOOOUT seçeneğiyle MQOP çağrısını kullanır. Sonuçlar, kuyruğa yollanmış bir dosyaya çıkışıdır. Kuyruk açılmazsa, program, MQOPED çağrısının döndürdüğü neden kodunu içeren bir hata iletisi yazar. Programı basit tutmak için, bu konuda ve sonraki MQI çağrılarında, program seçeneklerin çoğu için varsayılan değerleri kullanır.

Kaynak kodda bulunan her veri satırı için, program metni bir arabelleğe okur ve MQPUT çağrısını kullanarak, o satırın metnini içeren bir veri paketi iletisi yaratır. Program, girişin sonuna ulaşıncaya kadar ya da MQPUT çağrısının başarısız olduğu zamana kadar devam eder. Program girişin sonuna ulaşırsa, MQCLOSE çağrısını kullanarak kuyruğu kapatır.

## The Browse sample program on IBM i

Göz At örnek programı ( AMQ3GBR4), MQGET çağrısını kullanarak kuyruklardaki iletileri göz attır.

Program, programı çağırdığınızda belirttiğiniz kuyruklardaki tüm iletilerin kopyalarını alır; iletiler kuyrukta kalır. You could use the supplied queue SYSTEM.SAMPLE.LOCAL; run the Put sample program first to put some messages on the queue. Aynı yerel kuyruk için bir diğer ad olan SYSTEM.SAMPLE.ALIASKuyruğunu kullanabilirsiniz. Program, kuyruğun sonuna ulaşıncaya ya da bir MQI çağrısı başarısız oluncaya kadar devam eder.

RPG (RPG) programını çağırma ilişkin bir komut örneği:

```
CALL PGM(QMQMSAMP/AMQ3GBR4) PARM('Queue_Name','Queue_Manager_Name')
```

Burada Queue\_Name ve Queue\_Manager\_Name , 48 karakter uzunluğunda olmalı ve Queue\_Name ve Queue\_Manager\_Name ' yi gereken sayıda boşlukla doldurarak elde etmeniz gerekir. Bu nedenle, SYSTEM.SAMPLE.LOCAL hedef kuyruğunuz olarak, 29 boş karaktere sahip olacaktır.

## Göz At örnek programının tasarımı

Program, OOBROW seçeneğiyle MQOP çağrısını kullanarak hedef kuyruğu açar. Kuyruğu açamazsa, program, MQOPED çağrısının döndürdüğü neden kodunu içeren, kuyruğa yollanmış dosyasına bir hata iletisi yazar.

Kuyrukta yer alan her ileti için, program iletiyi kuyruktan kopyalamak için MQGET çağrısını kullanır, ardından iletide yer alan verileri görüntüler. MQGET çağrısı aşağıdaki seçenekleri kullanır:

### GMBRWN

MQOPEN çağrısının ardından, göz atma imleci kuyrukta ilk iletiden önce mantıksal olarak konumlandırılır, bu nedenle bu seçenek, arama ilk kez yapıldığında ilk iletisinin döndürülmesine neden olur.

### GDNWT

Kuyruğun üzerinde ileti yoksa program beklemez.

### GMATM

MQGET çağrısı, sabit büyüklükten arabelleğinden birini belirtir. Bu arabellekten daha uzun bir ileti varsa, program kısaltılmış iletiyi görüntüler; bu ileti, iletinin kesildiğini bildiren bir uyarıyla birlikte görüntülenir.

Bu program, bu alanları, aldığı iletide yer alan değerlere ayarlaması nedeniyle, her MQGET çağrısından sonra MQMD yapısının MDMID ve MDCID alanlarını nasıl temizlemeniz gerektiğini gösterir. Bu alanların temizlenmesi, art arda gelen MQGET çağrılarının iletilerin kuyrukta tutulmakta olduğu sırayla alma çağrılarını anlamına gelir.

Program kuyruğun sonuna kadar devam eder; burada, MQGET çağrısı RC2033 (kullanılabilir ileti yok) neden kodunu döndürür ve program bir uyarı iletisi görüntüler. MQGET çağrısının başarısız olması durumunda, program, kuyruk dosyasındaki neden kodunu içeren bir hata iletisi yazar.

Daha sonra, program MQCLOSE çağrısını kullanarak kuyruğu kapatır.

### **IBM i' daki örnek alma programı**

Alma örnek programı ( AMQ3GET4), MQGET çağrısını kullanarak kuyruktan ileti alır.

Program çağrıldığında, belirtilen kuyruktan iletileri kaldırır. You could use the supplied queue SYSTEM.SAMPLE.LOCAL; run the Put sample program first to put some messages on the queue. SYSTEM.SAMPLE.ALIAS kuyruğu, aynı yerel kuyruk için bir diğer ad. Kuyruk boş oluncaya ya da bir MQI çağrısı başarısız oluncaya kadar program devam eder.

RPG (RPG) programını çağırma ilişkin bir komut örneği:

```
CALL PGM(QMQMSAMP/AMQ3GET4) PARM('Queue_Name', 'Queue_Manager_Name')
```

Burada Queue\_Name ve Queue\_Manager\_Name , 48 karakter uzunluğunda olmalı ve Queue\_Name ve Queue\_Manager\_Name ' yi gereken sayıda boşlukla doldurarak elde etmeniz gerekir. Bu nedenle, SYSTEM.SAMPLE.LOCAL hedef kuyruğunuz olarak, 29 boş karaktere sahip olacaktır.

### **Get Sample programının tasarımı**

Program, iletileri almak için hedef kuyruğu açar; OOINPQ seçeneğiyle MQOP çağrısını kullanır. Kuyruğu açamazsa, program, kuyruğa yollanmış dosyasında MQOPED çağrısının döndürdüğü neden kodunu içeren bir hata ileti yazar.

Kuyruқта yer alan her ileti için, program iletiyi kuyruktan kaldırmak için MQGET çağrısını kullanır; daha sonra, iletide bulunan verileri görüntüler. The MQGET call uses the GMWT option, specifying a wait interval (*GMWT*) of 15 seconds, so that the program waits for this period if there is no message on the queue. Bu aralığın süresi dolmadan bir ileti gelmezse, arama başarısız olur ve RC2033 (kullanılabilir ileti yok) neden kodu döndürülür.

Bu program, bu alanları, aldığı iletide yer alan değerlere ayarlaması nedeniyle, her MQGET çağrısından sonra MQMD yapısının *MDMID* ve *MDCID* alanlarını nasıl temizlemeniz gerektiğini gösterir. Bu alanların temizlenmesi, art arda gelen MQGET çağrılarının iletilerin kuyruқта tutulmakta olduğu sırayla alma çağrıları anlamına gelir.

MQGET çağrısı, sabit büyüklerin arabelleğinden birini belirtir. Bu arabellekten daha uzun bir ileti varsa, arama başarısız olur ve program durur.

Bu program, MQGET çağrısının RC2033 (kullanılabilir ileti yok) neden kodunu döndürmesini ya da MQGET çağrısının başarısız oluncaya kadar devam eder. Arama başarısız olursa, program neden kodunu içeren bir hata ileti görüntüler.

Daha sonra, program MQCLOSE çağrısını kullanarak kuyruğu kapatır.

### **The Request sample program on IBM i**

İstek örneği programı ( AMQ3REQ4), istemci/sunucu işlemlerini gösterir. Örnek, istek iletilerini bir sunucu programı tarafından işlenen bir kuyruğa yerleştiren istemcidir. Sunucu programının bir yanıtlama kuyruğuna yanıt ileti göndermesini bekler.

İstek örneği, MQPUT çağrısını kullanan bir kuyruğa ilişkin istek iletilerini bir kuyruğa koyar. Bu iletiler, yanıtlama kuyruğu olarak SYSTEM.SAMPLE.REPLY değerini belirtir. Program yanıt iletilerini bekler, sonra bunları görüntüler. Yanıtlar yalnızca hedef kuyruk ( *sunucu kuyruğu* adını veriyorsa) gönderilir. bir sunucu uygulaması tarafından işlenmekte ya da bir uygulama bu amaçla tetiklenmişse (Sorgula ve Ayarla örnek programları tetiklenecek şekilde tasarlanmıştır). Örnek ilk yanıt için 5 dakika bekler (bir sunucu uygulamasının tetiklenmesine izin vermek için) ve sonraki yanıtlar için 15 saniye bekler, ancak yanıt almadan sona erebilir.

Programı başlatmak için, programı çağırın ve hedef kuyruğunuzun adını program parametresi olarak verin. Program, kuyruқта bir dizi sabit ileti yerleştiriyor; bu iletiler, program kaynak kodunun sonundaki veri bloğundan alınır.



## İstek örnek programının tasarımı

Program, iletileri koyabilmesi için sunucu kuyruğunu açar. OOOUT seçeneğiyle MQOP çağrısını kullanır. Kuyruğu açamazsa, program, MQOPED çağrısının döndürdüğü neden kodunu içeren bir hata ileti görüntüler.

Daha sonra program, SYSTEM.SAMPLE.REPLY , böylece yanıt iletileri alabilirler. Bunun için, program OOINPX seçeneği ile MQOPER çağrısını kullanır. Kuyruğu açamazsa, program, MQOPED çağrısının döndürdüğü neden kodunu içeren bir hata ileti görüntüler.

Her giriş satırı için, program metni bir arabelleğe okur ve MQPUT çağrısını kullanarak, o satırın metnini içeren bir istek ileti yaratır. Bu çağrıda program, istek iletişine ilişkin gönderilen herhangi bir rapor iletişinin ileti verilerinin ilk 100 baytı içermesini istemek için ROEXCD rapor seçeneğini kullanır. Program, girişin sonuna ulaşıncaya kadar ya da MQPUT çağrısının başarısız olduğu zamana kadar devam eder.

Daha sonra, program yanıtlama iletilerini kuyruktan kaldırmak için MQGET çağrısını kullanır ve yanıtlarda yer alan verileri görüntüler. The MQGET call uses the GMWT option, specifying a wait interval (*GMWT*) of 5 minutes for the first reply (to allow time for a server application to be triggered) and 15 seconds for subsequent replies. Bu süre, kuyruğun üzerinde herhangi bir ileti yoksa, bu dönemleri bekler. Bu aralığın süresi dolmadan bir ileti gelmezse, arama başarısız olur ve RC2033 (kullanılabilir ileti yok) neden kodu döndürülür. Çağrı, GMATM seçeneğini de kullanır; bu nedenle, bildirilmiş arabellek büyüklüğünden daha uzun iletiler kesilir.

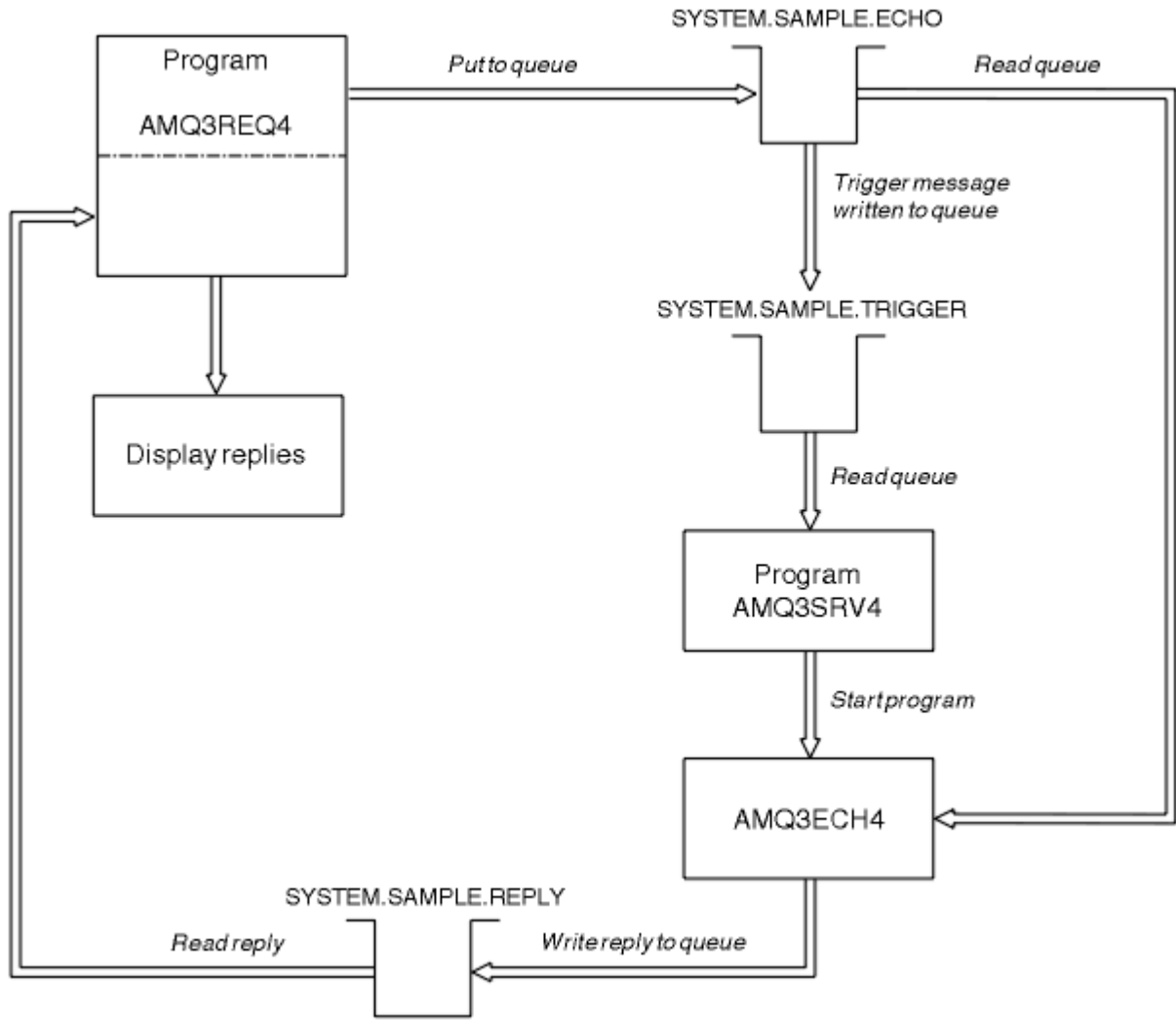
Bu program, bu alanları, aldığı iletide yer alan değerlere ayarlaması nedeniyle, her MQGET çağrısından sonra MQMD yapısının *MDMID* ve *MDCOD* alanlarını nasıl temizlemeniz gerektiğini gösterir. Bu alanların temizlenmesi, art arda gelen MQGET çağrılarının iletilerin kuyrukta tutulmakta olduğu sırayla alma çağrıları anlamına gelir.

Bu program, MQGET çağrısının RC2033 (kullanılabilir ileti yok) neden kodunu döndürmesini ya da MQGET çağrısının başarısız oluncaya kadar devam eder. Arama başarısız olursa, program neden kodunu içeren bir hata ileti görüntüler.

Daha sonra, program, MQCLOSE çağrısını kullanarak hem sunucu kuyruğunu, hem de yanıtlamayı kuyruğa kapatır. Çizelge 814 sayfa 1397 , Sorgula ve Ayarla örnek programlarını çalıştırmak için gerekli olan Echo örnek programında yapılan değişiklikleri gösterir.

**Not:** Echo örnek programına ilişkin ayrıntılar başvuru amaçlı olarak eklenmiştir.

Çizelge 814. İstemci/Sunucu örnek programı ayrıntıları		
Program adı	SYSTEM/örnek kuyruğu	Program başlatıldı
Yankı	YANKI	AMQ3ECH4
Sor	INQ	AMQ3INQ4
Belirle	Belirle	AMQ3SET4



Şekil 9. Örnek İstemci/Sunucu (Echo) programı akış şeması

#### IBM i Using triggering with the Request sample on IBM i

To run the sample using triggering, start the trigger server program, AMQ3SRV4, against the required initiation queue in one job, then start AMQ3REQ4 in another job.

Bu, İstek örnek programı bir ileti gönderdiğinde tetikleme sunucusunun hazır olduğu anlamına gelir.

#### Not:

1. Örnekler, SYSTEM SAMPLE TRIGGER kuyruğunu SYSTEM.SAMPLE.ECHO, SYSTEM.SAMPLE.INQya da SYSTEM.SAMPLE.SET yerel kuyrukları. Diğer bir seçenek olarak, kendi başlatma kuyruğunuzu tanımlayabilirsiniz.
2. AMQSAMP4 ile yaratılan örnek tanımlamaları, örnekteki C sürümünün tetiklenmesine neden olur. RPG sürümünü tetiklemek istiyorsanız, SYSTEM.SAMPLE.ECHOPROCESS ve SYSTEM.SAMPLE.INQPROCESS ve SYSTEM.SAMPLE.SETPROCESS işlem tanımlarını değiştirmeniz gerekir. You can use the CHGMQMPRC command (see [Change MQ Process \(CHGMQMPRC\)](#) for more details) to do this, or edit and run your own version of AMQSAMP4.
3. Tetikleyici sunucu programını, QMQMSAMP/QRPGLESRC ' de sağlanan kaynaktan derlemeniz gerekir.

Çalıştırmak istediğiniz tetikleme işlemine bağlı olarak, AMQ3REQ4 , bu örnek sunucu kuyruklarından birine yerleştirilecek istek iletilerini belirten değiştirgeyle çağrılmalıdır:

- SYSTEM.SAMPLE.ECHO (Yankı örnek programları için)
- SYSTEM.SAMPLE.INQ (Sorgula ilgili örnek programlar için)

- SYSTEM.SAMPLE.SET (örnek programları ayarlamak için)

SYSTEM.SAMPLE.ECHO programı Şekil 9 sayfa 1398’inde gösterilir. Bu sunucuya yönelik RPG programı isteğini yayınlamak için bu komutu kullanma komutu aşağıdaki gibi olur:

```
CALL PGM(QMQMSAMP/AMQ3REQ4) PARM('SYSTEM.SAMPLE.ECHO  
+ 30 blank characters','Queue_Manager_Name')
```

Kuyruk adı ve kuyruk yöneticisi adının 48 karakter uzunluğunda olması gerekir.

**Not:** Bu örnek kuyruğun bir FIRST tetikleyicisi tipi vardır; bu nedenle, İstek örneğini çalıştırmadan önce kuyruğunda önceden iletiler varsa, sunucu uygulamaları gönderdiğiniz iletiler tarafından tetiklenmez.

Daha fazla örnek vermek istiyorsanız, aşağıdaki çeşitlemeleri deneyebilirsiniz:

- İş sunmak için AMQ3SRV4 yerine AMQ3TRG4 ' ü kullanın, ancak olası iş gönderimi gecikmeleri, gerçekleşenleri takip etmeyi daha az kolaylaştırabilirdi.
- SYSTEM.SAMPLE.INQ ve SYSTEM.SAMPLE.SET örnek kuyrukları. Örnek veri dosyasını kullanarak, bu sunuculara yönelik RPG programı isteklerini yayınlamaya ilişkin komutlar şunlardır:

```
CALL PGM(QMQMSAMP/AMQ3INQ4) PARM('SYSTEM.SAMPLE.INQ  
+ 31 blank characters')  
CALL PGM(QMQMSAMP/AMQ3SET4) PARM('SYSTEM.SAMPLE.SET  
+ 31 blank characters')
```

çünkü kuyruk adı 48 karakter uzunluğunda olmalıdır.

Bu örnek kuyruklarda, FIRST tetikleme tipi de vardır.

### **IBM üzerindeki Echo örnek programı**

Echo örnek programları, ileti gönderisini bir yanıt kuyruğuna geri döndürür. Program AMQ3ECH4adını taşır.

Tetikleme işleminin çalışması için, kullanmak istediğiniz Echo örnek programının SYSTEM.SAMPLE.ECHO kuyruğuna gelen iletiler tarafından tetiklendiğinden emin olmanız gerekir. To do this, specify the name of the Echo sample program you want to use in the *AppId* field of the process definition SYSTEM.SAMPLE.ECHOPROCESS. (Bunun için, [IBM uygulamasını yönetme](#) içinde açıklanan CHGMQMPRC komutunu kullanabilirsiniz.) Örnek kuyruğun bir FIRST tetikleyicisi tipi vardır; İstek örneğini çalıştırmadan önce kuyruğunda önceden iletiler varsa, gönderdiğiniz iletiler Echo örneği tetiklenmez.

When you have set the definition correctly, first start AMQ3SRV4 in one job, then start AMQ3REQ4 in another. AMQ3SRV4 yerine AMQ3TRG4 ' yi kullanabilirsiniz, ancak olası iş gönderimi gecikmeleri, gerçekleşenleri takip etmeyi daha az kolaylaştırabilirdi.

Use the Request sample programs to send messages to queue SYSTEM.SAMPLE.ECHO. Echo örnek programları, istek iletilerinde, istek iletilerinde belirtilen yanıt kuyruğuna veri içeren bir yanıt iletileri gönderir.

### **Echo örnek programının tasarımı**

Program tetiklendiğinde, MQCONN çağrısını kullanarak varsayılan kuyruk yöneticisine belirttik olarak bağlanır. IBM için bu gerekli olmasa da, bu, kaynak kodu değiştirmeden diğer platformlarda da aynı programı kullanabildiğinizi gösterir.

Program daha sonra, başlatıldığı sırada tetikleme iletileri yapısında adı belirtilen kuyruğu açar. (For clarity, we will call this the *istek kuyruğu*.) Program, bu kuyruğu paylaşılan giriş için açmak için MQOPEN çağrısını kullanır.

Program, bu kuyruktan iletileri kaldırmak için MQGET çağrısını kullanır. Bu çağrı, 5 saniye bekleme süresi ile GMATM ve GMWT seçeneklerini kullanır. Program, bir istek iletileri olup olmadığını görmek için her iletilerin tanımlayıcısını sınar; değilse, program iletileri atar ve bir uyarı iletileri görüntüler.

İstek kuyruğundan kaldırılan her istek iletileri için, program, yanıtlama kuyruğuna yanıt iletileri koymak için MQPUT çağrısını kullanır. Bu ileti, istek iletilerinin içeriğini içerir.

İstek kuyruğunda bir ileti kalmadığında, program o kuyruğu kapatır ve kuyruk yöneticisinden bağlantıyı keser.

This program can also respond to messages sent to the queue from platforms other than IBM i, although no sample is supplied for this situation. ECHO program çalışmasını yapmak için aşağıdaki işlemi gerçekleştirdiniz:

- Metin isteği iletileri göndermek için *Format, Encoding ve CCSID* alanlarını doğru biçimde belirten bir program yazın.

ECHO programı, gerekiyorsa, kuyruk yöneticisinin ileti verisi dönüştürme işlemi gerçekleştirmesini ister.

- Specify CONVERT(\*YES) on the IBM MQ for IBM i sending channel, if the program you have written does not provide similar conversion for the reply.

### **IBM üzerindeki Sorgula örnek programı**

Sorgu örnek programı ( AMQ3INQ4), MQINQ çağrısını kullanan bir kuyruğun bazı özniteliklerine ilişkin bilgi içerir.

Program, tetiklenen bir program olarak çalıştırılmak üzere tasarlanmıştır, bu nedenle tek girişi bir MQTMC (tetikleme iletisi) yapısıdır. Bu yapı, sorgulanacak özniteliklere sahip bir hedef kuyruğun adını içerir.

Tetikleme işleminin çalışması için, SYSTEM.SAMPLE.INQ. To do so, specify the name of the Inquire sample program in the *AppLId* field of the SYSTEM.SAMPLE.INQPROCESS process definition. (Bunun için, [Change MQ Process \(CHGMQMPRC\)](#) içinde açıklanan CHGMQMPRC komutunu kullanabilirsiniz). Örnek kuyruğun tetikleme tipi FIRST, bu nedenle, İstek örneğini çalıştırmadan önce kuyruğunda önceden iletiler varsa, gönderdiğiniz iletiler ile sorgulamak örneği tetiklenmez.

When you have set the definition correctly, first start AMQ3SRV4 in one job, then start AMQ3REQ4 in another. AMQ3SRV4 yerine AMQ3TRG4 ' yi kullanabilirsiniz, ancak olası iş gönderimi gecikmeleri, gerçekleşenleri takip etmeyi daha az kolaylaştırabilir.

İstek örnek programını kullanarak, her biri yalnızca bir kuyruk adı içeren istek iletilerini SYSTEM.SAMPLE.INQ. Her istek iletisi için, Sorgula örnek programı, istek iletisinde belirtilen kuyruğa ilişkin bilgileri içeren bir yanıt iletisi gönderir. Yanıtlar, istek iletisinde belirtilen yanıtlama kuyruğuna gönderilir.

### **Sorgulamak için örnek program tasarımı**

Program tetiklendiğinde, MQCONN çağrısını kullanarak varsayılan kuyruk yöneticisine belirtik olarak bağlanır. IBM üzerinde gerekli olmasa da, bu tasarım özelliği, kaynak kodu değiştirmeden diğer platformlarda aynı programı kullanabilmenize yardımcı olur.

Program daha sonra, başlatıldığı sırada tetikleme iletisi yapısında adı belirtilen kuyruğu açar. (For clarity, we will call this the *istek kuyruğu*.) Program, bu kuyruğu paylaşılan giriş için açmak için MQOPEN çağrısını kullanır.

Program, bu kuyruktan iletileri kaldırmak için MQGET çağrısını kullanır. Bu çağrı, 5 saniye bekleme süresi ile GMATM ve GMWT seçeneklerini kullanır. Program, bir istek iletisi olup olmadığını görmek için her iletinin tanımlayıcısını sınar; değilse, program iletiyi atar ve bir uyarı iletisi görüntüler.

İstek kuyruğundan kaldırılan her istek iletisi için, program kuyruğun adını okur (*hedef kuyruk* adını arayacağız). verilerde bulunur ve OOINQ seçeneği ile MQOPEN çağrısını kullanarak kuyruğu açar. Daha sonra, program, hedef kuyruğun **InhibitGet**, **CurrentQDepth** ve **OpenInputCount** özniteliklerinin değerlerini sorgulamak için MQINQ çağrısını kullanır.

MQINQ çağrısı başarılı olursa, program, yanıtlama kuyruğuna yanıt iletisi koymak için MQPUT çağrısını kullanır. Bu ileti, üç özniteliğin değerlerini içerir.

MQAUT ya da MQINQ çağrısı başarısız olursa, program, yanıtlama kuyruğuna bir *rapor* iletisi koymak için MQPUT çağrısını kullanır. Bu rapor iletisinin ileti tanımlayıcısının *MDFB* alanında, başarısız olan buna bağlı olarak, MQOPED ya da MQINQ çağrısının döndürdüğü neden kodudur.

MQINQ çağrısından sonra program, MQCLOSE çağrısını kullanarak hedef kuyruğu kapatır.

İstek kuyruğunda bir ileti kalmadığında, program o kuyruğu kapatır ve kuyruk yöneticisinden bağlantıyı keser.

### **IBM üzerinde Set Sample programı**

Set örnek programı ( AMQ3SET4), kuyruğun **InhibitPut** özniteliğini değiştirmek için MQSET çağrısını kullanarak, bir kuyruğa ilişkin işlemleri engeller.

Program, tetiklenen bir program olarak çalıştırılmak üzere tasarlanmıştır, bu nedenle tek girişi, sorgulanacak özniteliklere sahip bir hedef kuyruğun adını içeren bir MQTMC (tetikleme iletisi) yapısıdır.

Tetikleme işleminin çalışması için, SYSTEM.SAMPLE.SETkuyruğuna gelen ileteler için Set Sample programının tetiklendiğinden emin olmanız gerekir. To do this, specify the name of the Set sample program in the *AppId* field of the process definition SYSTEM.SAMPLE.SETPROCESS. (Bunun için, IBM uygulamasını yönetme ' da açıklanan CHGMQMPRC komutunu kullanabilirsiniz.) Örnek kuyruğun tetikleme tipi FIRST, bu nedenle, İstek örneğini çalıştırmadan önce kuyruğunda önceden ileteler varsa, Küme örneği gönderdiğiniz ileteler tarafından tetiklenmez.

When you have set the definition correctly, first start AMQ3SRV4 in one job, then start AMQ3REQ4 in another. AMQ3SRV4yerine AMQ3TRG4 ' yi kullanabilirsiniz, ancak olası iş gönderimi gecikmeleri, gerçekleşenleri takip etmeyi daha az kolaylaştırabilirdi.

İstek örnek programını kullanarak, her biri yalnızca bir kuyruk adı içeren istek iletelerini SYSTEM.SAMPLE.SETkuyruğuna yollamak için kullanın. Her istek iletisi için, Set Sample programı, belirtilen kuyruk üzerinde işlem engellenmiş olan bir doğrulama içeren bir yanıt iletisi gönderir. Yanıtlar, istek iletisinde belirtilen yanıtlama kuyruğuna gönderilir.

### **Set Sample programının tasarımı**

Program tetiklendiğinde, MQCONN çağrısını kullanarak varsayılan kuyruk yöneticisine belirtik olarak bağlanır. IBM üzerinde gerekli olmasa da, bu, kaynak kodu değiştirmeden diğer platformlarda da aynı programı kullanabildiğinizi gösterir.

Program daha sonra, başlatıldığı sırada tetikleme iletisi yapısında adı belirtilen kuyruğu açar. (For clarity, we will call this the *istek kuyruğu*.) Program, bu kuyruğu paylaşılan giriş için açmak için MQOPEN çağrısını kullanır.

Program, bu kuyruktan ileteleri kaldırmak için MQGET çağrısını kullanır. Bu çağrı, 5 saniye bekleme süresi ile GMATM ve GMWT seçeneklerini kullanır. Program, bir istek iletisi olup olmadığını görmek için her iletinin tanımlayıcısını sınar; değilse, program iletiyi atar ve bir uyarı iletisi görüntüler.

İstek kuyruğundan kaldırılan her istek iletisi için, program kuyruğun adını okur ( *hedef kuyruk* adını arayacağız). verilerde yer alır ve OOSSET seçeneğiyle MQOPEN çağrısını kullanarak kuyruğu açar. Daha sonra, program, hedef kuyruğun **InhibitPut** özniteliğinin değerini QAPUTI olarak ayarlamak için MQSET çağrısını kullanır.

MQSET çağrısı başarılı olursa, program, yanıtlama kuyruğuna yanıt iletisi koymak için MQPUT çağrısını kullanır. Bu ileti, PUT *inhibited* dizgisini içerir.

MQPUT ya da MQSET çağrısı başarısız olursa, program, yanıtlama kuyruğuna bir *rapor* iletisi koymak için MQPUT çağrısını kullanır. Bu rapor iletisinin ileti tanımlayıcısının *MDFB* alanında, başarısız olan buna bağlı olarak, MQOPEN ya da MQSET çağrısının döndürdüğü neden kodudur.

MQSET çağrısından sonra program, MQCLOSE çağrısını kullanarak hedef kuyruğu kapatır.

İstek kuyruğunda bir ileti kalmadığında, program o kuyruğu kapatır ve kuyruk yöneticisinden bağlantıyı keser.

### **IBM üzerinde Tetikleme örneği programları**

IBM MQ for IBM i , ILE/RPG içinde yazılmış iki tetikleyici örnek programı sağlar.

Programlar şunlardır:

## AMQ3TRG4

Bu, IBM i ortamı için bir tetikleyici izleyicidir. Uygulamanın başlatılması için bir IBM i işi teslim edilir, ancak bu, her bir tetikleyici iletilisiyle ilişkili ek işlem maliyeti olduğu anlamına gelir.

## AMQ3SRV4

Bu, IBM i ortamı için bir tetikleme sunucusudur. Her bir tetikleme iletilisi için, bu sunucu belirlenen uygulamayı başlatmak için kendi işinde başlatma komutunu çalıştırır. Tetikleyici sunucusu CICS işlemlerini çağırabilir.

C language versions of these samples are also available as executable programs in library QMQM, called AMQSTRG4 and AMQSERV4.

### *IBM üzerinde AMQ3TRG4 örnek tetikleme izleyicisi*

AMQ3TRG4 , bir tetikleme izleyicidir. Bir değiştirgeyi alır: Hizmet vermek için kullanılan başlangıç kuyruğunun adı. AMQSAMP4 , bir örnek başlatma kuyruğu ( SYSTEM.SAMPLE.TRIGGER, örnek programları denediğinizde kullanabileceğiniz bir tetikleyiciye sahip olabilir.

AMQ3TRG4 , başlatma kuyruğundan aldığı her geçerli tetikleyici iletilisine ilişkin bir IBM i işi gönderir.

## Tetikleme izleyicisinin tasarımı

Tetikleme izleme programı başlatma kuyruğunu açar ve kuyruktan iletiler alır; sınırsız bekleme aralığı da sağlar.

Tetikleyici izleyicisi, tetikleme iletilisinde belirtilen uygulamayı başlatmak için bir IBM i işi gönderir ve bir MQTMC (tetikleme iletilisi karakter sürümü) yapısını geçirir. Tetikleme iletilisinde yer alan ortam verileri, iş gönderme değiştirgelemleri olarak kullanılır.

Son olarak, program başlatma kuyruğunu kapatır.

### *AMQ3SRV4 örnek tetikleyici sunucusu*

AMQ3SRV4 bir tetikleyici sunucusudur. Bir değiştirgeyi alır: Hizmet vermek için kullanılan başlangıç kuyruğunun adı. AMQSAMP4 , bir örnek başlatma kuyruğu ( SYSTEM.SAMPLE.TRIGGER, örnek programları denediğinizde kullanabileceğiniz bir tetikleyiciye sahip olabilir.

Her bir tetikleyici iletilisi için, AMQ3SRV4 , belirlenen uygulamayı başlatmak için kendi işinde bir başlatma komutu çalıştırır.

Örnek tetikleme kuyruğunun kullanılması, komutun yayınına aşağıdaki komutu verir:

```
CALL PGM(QMQM/AMQ3SRV4) PARM('Queue Name')
```

Burada Queue Name , 48 karakter uzunluğunda olmalı ve kuyruk adını gereken sayıda boşlukla doldurarak elde etmeniz gerekir. Bu nedenle, SYSTEM.SAMPLE.TRIGGER hedef kuyruğunuz olarak, 28 boşluk karakterine gereksinim duyarsınız.

## Tetikleme sunucusunun tasarımı

Tetikleme sunucusunun tasarımı, tetikleme sunucusu dışında, tetikleme izleyicisinin tipi gibidir:

- IBM i uygulamalarının yanı sıra CICS uygulamalarını da sağlar
- Tetikleyici iletilisinden ortam verilerini kullanmaz
- Calls IBM i applications in its own job (or uses STRCICSUSR to start CICS applications) rather than submitting an IBM i job
- Paylaşılan girişe ilişkin başlatma kuyruğunu açar; bu nedenle, birçok tetikleme sunucusu aynı anda çalışabilir

**Not:** AMQ3SRV4 tarafından başlatılan programlar, tetikleme sunucusunu durduracağı için MQDISC çağrısını kullanmamalıdır. AMQ3SRV4 tarafından başlatılan programlar MQCONN çağrısını kullanırsa, RC2002 neden kodunu alır.

### Ending the Triggering sample programs on IBM i

Bir tetikleme izleme programı, sysrequest seçeneği 2 (ENDRQS) tarafından ya da tetikleyici kuyruğundan alıkonabilecek bir program tarafından sona erdirilebilir.

Örnek tetikleyici kuyruğu kullanıldıysa, komut:

```
CHGMQM QNAME('SYSTEM.SAMPLE.TRIGGER') GETENBL(*NO)
```

**Not:** Bu kuyrukta yeniden tetikleme işlemini başlatmak için şu komutu girmeniz gerekir:

```
CHGMQM QNAME('SYSTEM.SAMPLE.TRIGGER') GETENBL(*YES)
```

### Running the samples using remote queues on IBM i

Bağlantılı ileti kuyruğu yöneticilerindeki örnekleri çalıştırarak uzak kuyruklama gösterebilirsiniz.

AMQSAMP4 programı, OTHERadlı uzak kuyruk yöneticisini kullanan bir uzak kuyruğun (SYSTEM.SAMPLE.REMOTE) yerel tanımlamasını sağlar. Bu örnek tanımlamasını kullanmak için, kullanmak istediğiniz ikinci ileti kuyruğu yöneticisinin adını OTHER olarak değiştirin. Ayrıca, iki ileti kuyruğu yöneticiniz arasında bir ileti kanalı da ayarlamanız gerekir; böylece, [ileti alışverişi kanallarına ilişkin kanal çıkışı programları](#) başlıklı konuya bakın.

İstek örneği programı, gönderdiği iletilerin MDRM alanına kendi yerel kuyruk yöneticisi adını koyar. Sorgula ve Set örnekleri, yanıt iletilerini, işlendikleri istek iletilerinin MDRQ ve MDRM alanlarında belirtilen kuyruğa ve ileti kuyruğu yöneticisine gönderir.

## IBM i için dönüş kodları (ILE RPG)

Bu bilgilerde, MQI ve MQAI ile ilişkili dönüş kodları açıklanır.

Aşağıdakilerle ilişkili dönüş kodları:

- Programlanabilir Komut Biçimi (PCF) komutları [Programlanabilir komut biçimleri başvurusulistesinde](#) yer alıyor.
- C++ çağrılarını [Using C++](#) içinde listelenir.

Her çağrı için, bir tamamlanma kodu ve bir neden kodu, arama işleminin başarılı olduğunu ya da başarısız olduğunu göstermek için kuyruk yöneticisi ya da bir çıkış yordamı tarafından döndürülür.

Uygulamaların, özel olarak belirtildiği durumlar dışında, belirli bir sırada denetlenmekte olan hatalara bağlı olmamaları gerekir. Bir çağrıdan birden fazla tamamlanma kodu ya da neden kodu ortaya çıkarsa, bildirilen belirli bir hata uygulamaya bağlıdır.

## IBM i için tamamlanma kodları (ILE RPG)

Tamamlanma kodu parametresi (CMPCOD), çağıranın, çağrılarının başarıyla tamamlanıp tamamlanmadığını, kısmen mi, yoksa başarısız mı olduğunu hızlı bir şekilde görmelerini sağlar.

### CCOK

(Diğer altyapılarda MQCC\_OK)

İşlem başarıyla tamamlandı.

Çağrı tam olarak tamamlandı; tüm çıkış parametreleri ayarlandı. **REASON** parametresi her zaman bu durumda değer RCNONE değerine sahiptir.

### CCWARN

(Diğer altyapılarda MQCC\_WARN)

Uyarı (kısmi tamamlama).

Arama kısmen tamamlandı. *CMPCOD* ve *REASON* çıkış parametrelerinin yanı sıra bazı çıkış parametreleri de ayarlanmış olabilir. **REASON** parametresi, kısmi tamamlama hakkında ek bilgi verir.

### **CCCFAIL**

(Diğer altyapılarda MQCC\_FAIL)

Arama başarısız oldu.

Çağrıyı işleme tamamlanmadı ve kuyruk yöneticisinin durumu normal olarak değiştirilmez; özel durumlar dikkate alındı. *CMPCOD* ve *REASON* çıkış parametreleri ayarlandı; burada belirtilenler dışında, diğer parametreler değiştirilmez.

Neden, uygulama programında bir hata olabilir ya da programın dışındaki bazı durumların bir sonucu olabilir; örneğin, kullanıcının yetkisi iptal edilmiş olabilir. **REASON** parametresi hatayla ilgili ek bilgi verir.

## **IBM i için neden kodları (ILE RPG)**

Neden kodu parametresi (*REASON*), tamamlanma kodu parametresine (*CMPCOD*) bir niteliğe sahip.

Raporlamak için özel bir neden yoksa, RCF geri döndürülür. Başarılı bir çağrı, CCOK ve RCNONE değerini döndürür.

Tamamlanma kodu CCWARN ya da CCCFAIL ise, kuyruk yöneticisi her zaman uygun bir neden bildirir; ayrıntılar her çağrı açıklaması altında verilir.

Kullanıcı çıkış yordamlarının tamamlanma kodlarını ve nedenlerini belirledikleri durumlarda, bu kuralların bu kurallara uyması gerekir. Ayrıca, kuyruk yöneticisi tarafından tanımlanan değerlerle çakışmamasını sağlamak için kullanıcı çıkışlarıyla tanımlanan özel neden değerlerinin sıfırdan küçük olması gerekir. Çıkışlar, kuyruk yöneticisi tarafından önceden tanımlanmış olan nedenleri ve bunların uygun olduğu nedenleri ayarlayabilir.

Neden kodları da aşağıdaki yerde oluşur:

- MQDLH yapısının *DLREA* alanı
- MQMD yapısındaki *MDFB* alanı

Neden kodlarının tam listesi için bkz. [API tamamlama ve neden kodları](#).

Bu listede IBM i neden kodunuzu bulmak için öndeki "RC" yi kaldırın; örneğin, RC2002 , 2002olur. Ayrıca, diğer platformlarda olduğu gibi gösterilen tamamlama kodları da gösterilir:

<i>Çizelge 815. IBM i ve diğer platformlardaki neden kodu adları</i>	
<b>IBM i</b>	<b>Diğer platformlar</b>
CCOK	MQCC_OK
CCWARN	MQCC_WARN
CCCFAIL	MQCC_FAIL

## **IBM i için MQI seçeneklerinin geçerliliğini denetleme kuralları (ILE RPG)**

Bu konuda, bir MQPUT, MQPUT, MQPUT1, MQGET ya da MQCLOSE çağrısından bir RC2046 neden kodu üreten durumlar hakkında bilgi verilir.

### **IBM üzerinde MQOPEN çağrısı**

MQOPER çağrısına ilişkin seçenekler için:

- *En az bir of the following must be specified:*
  - OOBROW
  - OOINPQ



- OOINPX
- OOINPS
- OOINQ
- OOUT
- OOSET
- Aşağıdakilerin yalnızca *bir* ' e izin verilir:
  - OOINPQ
  - OOINPX
  - OOINPS
- Aşağıdakilerin yalnızca *bir* ' e izin verilir:
  - OOBNDO
  - OOBNDN
  - OOBNDQ

**Not:** Daha önce listelenen seçenekler karşılıklı olarak birbirini dışlar. Ancak, OOBNDQ değeri sıfır olduğu için, diğer iki bağlama seçeneğinden biriyle birlikte belirtilmesi neden kodu RC2046 ile sonuçlanmaz. OOBNDQ, program belgelerine yardımcı olmak için sağlanmıştır.

- OOSAVA belirtilirse, OOINP\* seçeneğinden biri de belirtilmelidir.
- OOSET\* ya da OOPAS\* seçeneklerinden biri belirtilirse, OOUT değeri de belirlenmelidir.

## **MQPUT call on IBM i**

Put-message seçenekleri için:

- PMSYP ve PMNSYP birleşimine izin verilmiyor.
- Aşağıdakilerin yalnızca *bir* ' e izin verilir:
  - PMDEFC
  - PMSNOK
  - PMPASA
  - PMPASI
  - PMSETA
  - PMSSETI
- PMALTU ' ya izin verilmiyor (yalnızca MQPUT1 çağrısında geçerlidir).

## **IBM üzerindeMQPUT1 çağrısı**

Put-message seçenekleri için kurallar, aşağıdaki seçenekler dışında, MQPUT çağrısına göre aynıdır:

- PMALTU ' ya izin verilir.
- PMLOGO ' ya izin verilmiyor.

## **MQGET call on IBM i**

Get-message seçenekleri için:

- Aşağıdaki seçeneklerden yalnızca *bir* seçeneğe izin verilir:
  - GMNSYP
  - GGMSYP
  - GMPSYP

- Aşağıdaki seçeneklerden yalnızca *bir* seçeneğe izin verilir:
  - GMBRWF
  - GMBRWC
  - GMBRWN
  - GMMUC
- GMSYP ' ye aşağıdaki seçeneklerin hiçbiriyle izin verilmez:
  - GMBRWF
  - GMBRWC
  - GMBRWN
  - GMLK
  - GGMUNLK
- GMPSP, aşağıdaki seçeneklerin hiçbiriyle kullanılamaz:
  - GMBRWF
  - GMBRWC
  - GMBRWN
  - GMCMM
  - GGMUNLK
- GMLK belirtilirse, aşağıdaki seçeneklerden birinin de belirtilmesi gerekir:
  - GMBRWF
  - GMBRWC
  - GMBRWN
- GMUNLK belirtilirse, yalnızca aşağıdaki seçeneklere izin verilir:
  - GMNSYP
  - GDNWT

## **MQCLOSE call on IBM i**

- MQCLOSE çağrısına ilişkin seçenekler için. CODEL ve COPURG birleşimine izin verilmiyor.
- Aşağıdakilerin yalnızca birine izin verilir:
  - COKPSB
  - CORMSB

## **MQSUB call on IBM i**

MQSUB çağrısına ilişkin seçenekler için:

- Aşağıdakilerden en az birinin belirtilmesi gerekir:
- Aşağıdakilerden en az birinin belirtilmesi gerekir:
  - SOALT
  - SORES
  - SOCRT
- Aşağıdakilerin yalnızca birine izin verilir:
  - SODUR
  - SONDUR

**Not:** Daha önce listelenen seçenekler karşılıklı olarak birbirini dışlar. Ancak, SONDUR değerinin sıfır olduğu için, bu değeri SODUR ile belirtme, RC2046neden koduyla sonuçlanmaz. SONDUR, program belgelerine yardımcı olmak için sağlanmıştır.

- SOGRP ve SOMAN ' ın birleşimine izin verilmez.
- SOGRP, SOSCID ' nin belirtilmesini gerektirir.
- Şu öge için yalnızca birine izin verilir: SOAUID SOFUID
- SONEWP ve SOPUBR birleşimine izin verilmez.
- SONEWP ' nin yalnızca SOCRT ile birlikte kullanılmasına izin verilir.
- Aşağıdakilerin yalnızca birine izin verilir:
  - SOWCHR
  - SOWTOP

## Machine encodings on IBM i

İleti tanımlayıcısında *MDENC* alanının yapısı hakkında bilgi edinmek için bu bilgileri kullanın.

İleti tanımlayıcısına ilişkin daha fazla bilgi için bkz. [“MQMD \(Message descriptor\) on IBM i” sayfa 1092.](#)

*MDENC* alanı, dört ayrı alt alana bölünen 32 bitlik bir tamsayıdır; bu alt alanlar şunlardır:

- İkili tamsayılar için kullanılan kodlama
- Paketlenmiş ondalık tamsayılar için kullanılan kodlama
- Kayan noktalı sayılar için kullanılan kodlama
- Ayrılmış bit

Her alt alan, alt alana karşılık gelen konumlarda 1 bit ve başka bir yerde 0 biti olan bit maskesi ile tanımlanır. Bitler, 0 bit 'in en önemli bit, bit 31 'i ise en az anlamlı bit gibi numaralandırılmıştır. Aşağıdaki maskeleri tanımlıyor:

### ENIMSK

İkili tamsayı kodlaması maskesi.

Bu alt alan, *MDENC* alanı içinde 28-31 arasında bit konumlarını kaplar.

### SMSK

Paketlenmiş ondalık-tamsayı kodlaması için maske.

Bu alt alan, *MDENC* alanı içinde 24-27 arasında bit konumlarını kaplar.

### ENFMSK

Kayan noktalı kodlama için maske.

Bu alt alan, *MDENC* alanı içinde 20-23 arasında bit konumlarını kaplar.

### GİRİŞ MSK

Ayrılmış bitler için maske.

Bu alt alan, *MDENC* alanı içinde 0-19 arasındaki konumları bit olarak kaplar.

## IBM i IBM üzerinde ikili tamsayı kodlaması

İkili tamsayı kodlaması için geçerli değerler.

İkili tamsayı kodlaması için geçerli olan değerler şunlardır:

### ENIUND

Tanımsız tamsayı kodlaması.

İkili tamsayılar, tanımsız bir kodlama kullanılarak temsil edilir.

### ENINOR

Normal tamsayı kodlaması.

İkili tamsayılar geleneksel şekilde gösterilir:

- Sayıdaki en az önemli bayt, sayıdaki herhangi bir bayttan en yüksek adrese sahiptir; en önemli bayt en düşük adrese sahiptir.
- Her bir bayta ilişkin en az önemli bit, sonraki üst adrese sahip baytın yanında; her bir baytın en önemli biti, bir sonraki alt adresle birlikte bayt 'ın yanında yer alıyor.

#### **ENIREV**

Ters tamsayı kodlaması.

İkili tamsayılar, ENINOR ile aynı şekilde, ancak ters sırada düzenlenmiş byte 'lar ile gösterilir. Her bir baytın içindeki bitler ENINOR ile aynı şekilde düzenlenir.

#### **IBM i**

### **Packed-decimal-integer encoding on IBM i**

Paketlenmiş ondalık-tamsayı kodlaması için geçerli değerler

Aşağıdaki değerler paketlenmiş ondalık-tamsayı kodlaması için geçerlidir:

#### **ENDUND**

Tanımlanmamış paket-ondalık kodlaması.

Paketlenmiş ondalık tamsayılar, tanımsız bir kodlama kullanılarak gösterilir.

#### **ENDNOR**

Paketlenmiş normal onlu kodlama.

Paketlenmiş onlu tamsayılar geleneksel şekilde gösterilir:

- Sayının yazdırılabilir biçimindeki her bir ondalık basamağı, X' 0 'ile X' 9' aralığındaki tek bir onaltılı sayı ile paketlenmiş onlu sayıyla temsil edilir. Her onaltılı sayı 4 biti kaplar ve paketlenmiş onlu sayıdaki her bayt, sayının yazdırılabilir biçiminde iki ondalık basamağı temsil eder.
- Paketli onlu sayıdaki en az önemli bayt, en az önemli ondalık basamağı içeren byte 'tır. Bu bayt içinde, en önemli 4 bit, en az anlamlı ondalık basamak içerir ve en az önemli 4 bit, işareti içerir. İşaret, X'C '(artı), X 'D' (negatif) ya da X'F ' (işaretsiz) olur.
- Sayıdaki en az önemli bayt, sayıdaki herhangi bir bayttan en yüksek adrese sahiptir; en önemli bayt en düşük adrese sahiptir.
- Her bir bayta ilişkin en az önemli bit, sonraki üst adrese sahip baytın yanında; her bir baytın en önemli biti, bir sonraki alt adresle birlikte bayt 'ın yanında yer alıyor.

#### **ERDİDR**

Tersine çevrildi-ondalık kodlama.

Paketlenmiş onlu tamsayılar, ENDNOR ile aynı şekilde, ancak ters sırada düzenlenmiş byte 'lar ile gösterilir. Her bir baytın içindeki bitler, ENDNOR ile aynı şekilde düzenlenir.

#### **IBM i**

### **IBM üzerinde kayan noktalı kodlama**

Kayan noktalı kodlama için geçerli değerler

Kayan noktalı kodlama için geçerli olan değerler şunlardır:

#### **GİRİŞ**

Kayar noktalı sayı kodlaması tanımsız.

Kayan noktalı sayılar, tanımlanmamış bir kodlama kullanılarak gösterilir.

#### **ENFNOR**

Normal IEEE (Elektrik ve Elektronik Mühendisleri Enstitüsü) kayar noktalı (float) kodlama.

Kayan noktalı sayılar, standart IEEE kayan nokta biçimi kullanılarak, baytlar aşağıdaki şekilde ayarlanışlarla gösterilir:

- mantissa 'da en az önemli bayt, sayıdaki herhangi bir baytın en yüksek adresine sahip; üsküdar' ı içeren bayt en düşük adrese sahip.

- Her bir bayta ilişkin en az önemli bit, sonraki üst adrese sahip baytın yanında; her bir baytta en önemli bit, sonraki alt adrese sahip byte 'ın yanında yer alıyor.

IEEE kayar noktalı sayı kodlamasının ayrıntıları IEEE Standard 754 'te bulunabilir.

#### ENFREV

Ters IEEE kayar noktalı sayı kodlaması.

Kayan noktalı sayılar, ENFNOR ile aynı şekilde, ancak ters sırada düzenlenmiş byte 'lar ile gösterilir. Her bir baytın içindeki bitler, ENFNOR ile aynı şekilde düzenlenir.

#### ENF390

System/390 mimarisi kayar noktalı sayı kodlaması.

Kayan noktalı sayılar, standart System/390 kayan noktalı biçimi kullanılarak gösterilir; bu değer System/370 tarafından da kullanılır.

### IBM i IBM üzerinde kodlamalar oluşturuluyor

MQMD ' de MDENC alanı için bir değer oluşturmak üzere, gerekli kodlamaları açıklayan ilgili sabitler eklenmelidir.

Be sure to combine only one of the ENI\* encodings with one of the END\* encodings and one of the ENF\* encodings.

### IBM i IBM üzerindeki kodlamaların çözümleniyor

MDENC alanı alt alanlar içerir; bunun nedeni, tamsayıyı, paketlenmiş ondalığı ya da kayar noktalı sayı kodlamasını incelemek için gereken uygulamaların bu konuda açıklanan tekniği kullanmaları gerekir.

#### Aritmetik kullanılması

Tamsayı aritmetiği kullanılarak aşağıdaki adımlar gerçekleştirilmelidir:

1. Gereken kodlama tipine göre, aşağıdaki değerlerden birini seçin:

- İkili tamsayı kodlaması için 1
- Paketlenmiş onlu tamsayı kodlaması için 16
- 256 kayan noktalı kodlama için

Adeğerini arayın.

2. MDENC alanının değerini A ile bölün; Bsonucunu çağırın.

3. Divide B by 16; call the result C.

4. C ile16arasındaki çarpı ve B ' tan çıkarma; Dsonucunu çağırın.

5. Multiply D by A ; call the result E.

6. E , gereken kodlamadır ve bu kodlama tipi için geçerli olan her bir değer için eşitlik için test edilebilir.

### IBM i IBM üzerindeki makine mimarisi kodlamalarının özeti

Makine mimarileri için kodlamaları özetleyen bir tablo.

Makine mimarilerine ilişkin kodlamalar Çizelge 816 sayfa 1409 içinde gösterilir.

Çizelge 816. Makine mimarilerine ilişkin kodlamaların özeti			
Makine mimarisi	İkili tamsayı kodlaması	Paketli-onlu tamsayı kodlaması	Kayan noktalı kodlama
IBM i	normal	normal	IEEE olağan

Çizelge 816. Makine mimarilerine ilişkin kodlamaların özeti (devamı var)

Makine mimarisi	İkili tamsayı kodlaması	Paketli-onlu tamsayı kodlaması	Kayan noktalı kodlama
Intel x86	Tersine Çevrilmiş	Tersine Çevrilmiş	IEEE ters
PowerPC	normal	normal	IEEE olağan
System/390	normal	normal	System/390

## IBM i

## IBM i' ta rapor seçenekleri ve ileti işaretleri

Bu konu, MQGET, MQPUT ve MQPUT1 çağrılarında belirtilen MQMD ileti tanımlayıcısı MQMD ' nin bir parçası olan MDREP ve MDMFL alanlarını ilgilendirir.

İleti tanımlayıcısına ilişkin daha fazla bilgi için bkz. [“MQMD \(Message descriptor\) on IBM i” sayfa 1092](#). Bu bilgiler şunları açıklar:

- Rapor alanının yapısı ve kuyruk yöneticisi tarafından nasıl işlem yapılma
- Bir uygulamanın rapor alanını nasıl analiz etmesi gerekir
- İletinin yapısı-işaretler alanı

### Rapor alanının yapısı

MDREP alanı, üç ayrı alt alana bölünen 32 bitlik bir tamsayıdır.

Bu alt alanlar şunları tanımlar:

- Yerel kuyruk yöneticisi tarafından tanınmadığında reddedilen rapor seçenekleri
- Yerel kuyruk yöneticisi bunları tanımıyorsa bile her zaman kabul edilen rapor seçenekleri
- Yalnızca belirli bazı koşullar karşılanırsa kabul edilen rapor seçenekleri

Her alt alan, alt alana karşılık gelen konumlarda 1 bit ve başka bir yerde 0 biti olan bit maskesi ile tanımlanır. Alt alanlardaki bitlerin bitişik olması gerekmediği unutulmalıdır. Bitler, 0 bit 'in en önemli bit, bit 31 'i ise en az anlamlı bit gibi numaralandırılmıştır. Alt alanları tanımlamak için aşağıdaki maskeleri tanımlar:

#### RORUM

Reddedilen desteklenmeyen rapor seçenekleri için maske.

Bu maske, yerel kuyruk yöneticisi tarafından desteklenmeyen rapor seçeneklerinin MQPUT ya da MQPUT1 çağrısının tamamlanma kodu CCFAIL ve neden kodu RC2061 ile başarısız olmasına neden olacağı için, MDREP alanındaki bit konumlarını tanımlar.

Bu alt alan, 3, 11 ile 13 arasında bit pozisyonları kaplıyor.

#### ROAUM

Kabul edilen desteklenmeyen rapor seçenekleri için maske.

Bu maske, MDREP alanında, yerel kuyruk yöneticisinin desteklemediği rapor seçeneklerinin yine de MQPUT ya da MQPUT1 çağrılarında kabul edileceğine ilişkin bit konumlarını tanımlar. Bu durumda, tamamlanma kodu CCWARN neden kodu RC2104 ile döndürülür.

Bu alt alan, 0-2, 4-10 ve 24-31 arasında bit pozisyonları içerir.

Bu alt alana aşağıdaki rapor seçenekleri dahil edilir:

- ROCMTC
- RODLQ
- RODISC
- ROEXC
- ROEXCD

- ROEXCF
- ROEXP
- GENİŞLET
- GENİŞLET
- RONAN
- RONMI
- ROYOK
- ROPAN
- ROPCI
- ROPMI

## ROAUXM

Yalnızca belirli durumlarda kabul edilen desteklenmeyen rapor seçenekleri için maske.

This mask identifies the bit positions within the *MDREP* field where report options which are not supported by the local queue manager will nevertheless be accepted on the MQPUT or MQPUT1 calls *sağlandı* that both of the following conditions are satisfied:

- İletin kaderi uzak bir kuyruk yöneticisine ait.
- Uygulama iletiyi doğrudan yerel bir iletim kuyruğuna koymuyor (yani, MQOPER ya da MQPUT1 çağrısında belirlenen nesne tanımlayıcısındaki *ODMN* ve *ODON* alanları tarafından tanımlanan kuyruk, yerel bir iletim kuyruğu değil).

Completion code CCWARN with reason code RC2104 are returned if these conditions are satisfied, and CCFAIL with reason code RC2061 if not.

Bu alt alan 14-23 arasında bit pozisyonları kaplıyor.

Bu alt alana aşağıdaki rapor seçenekleri dahil edilir:

- ROCOA
- ROCOAD
- ROCOAF
- ROCOD
- ROCODD
- ROCODF

If there are any options specified in the *MDREP* field which the queue manager does not recognize, the queue manager checks each subfield in turn by using the bitwise AND operation to combine the *MDREP* field with the mask for that subfield. Bu işlemin sonucu sıfır değilse, daha önce açıklanan tamamlanma kodu ve neden kodları döndürülür.

CCWARN değeri döndürülürse, diğer uyarı koşulları varsa, bu neden kodlarının döndürüldüğü tanımlanmaz.

Yerel kuyruk yöneticisi tarafından tanınmayan rapor seçeneklerini belirleme ve kabul etme yeteneği, bir *uzak* kuyruk yöneticisi tarafından tanınacağı ve işlenecek bir rapor seçeneği içeren bir ileti göndermek gerektiğinde yararlı olur.

## IBM i Analyzing the report field on IBM i

MDREP alanı alt alanlar içerir. Bu yüzden bazı uygulamaların, iletinin göndericisinin belirli bir rapor isteyip istemediğini kontrol etmesi gerekir. Bu uygulamalar, bu konuda açıklanan tekniği kullanmalıdır.

### Aritmetik kullanılması

Tamsayı aritmetiği kullanılarak aşağıdaki adımlar gerçekleştirilmelidir:

1. Denetlenecek rapor tipine göre, aşağıdaki değerlerden birini seçin:

- COA raporu için ROCOA
- COD raporu için ROCOD
- Kural dışı durum raporu için ROEXC
- Süre bitim raporu için ROEXP

Adeğerini arayın.

2. MDREP alanını A ile bölün; Bsonucunu çağırın.

3. Divide B by 8 ; call the result C.

4. C 'u 8 ile çarpın ve B ' dan çıkarma; Dsonucunu çağırın.

5. Multiply D by A ; call the result E.

6. Bu rapor türü için mümkün olan her bir değer ile eşitlik için E ' yi test edin.

Örneğin, A ROEXC ise, iletinin gönderici tarafından nelerin belirtildiğini saptamak için, aşağıdakilerin her biriyle eşitlik için E sınavasını sınavın:

- ROYOK
- ROEXC
- ROEXCD
- ROEXCF

Testler, uygulama mantığı için en uygun sırayla gerçekleştirilebilecek şekilde gerçekleştirilebilir.

Aşağıdaki sözde kod, kural dışı durum raporu iletileri için bu tekniği gösterir:

```
A = ROEXC
B = Report/A
C = B/8
D = B - C*8
E = D*A
```

A similar method can be used to test for the ROPMI or ROPCI options; select as the value A whichever of these two constants is appropriate, and then proceed as described previously, but replacing the value 8 in the previous steps by the value 2.

IBM i

## IBM üzerindeki ileti işaretlerinin yapısı

MDMFL alanı, üç ayrı alt alana bölünen 32 bitlik bir tamsayıdır.

Bu alt alanlar şunları tanımlar:

- Yerel kuyruk yöneticisi tarafından tanınmadığında reddedilen ileti işaretleri
- Yerel kuyruk yöneticisi bunları tanımıyorsa bile, her zaman kabul edilen ileti işaretleri
- Yalnızca belirli bazı koşullar karşılanırsa kabul edilen ileti işaretleri

**Not:** All subfields in MDMFL are reserved for use by the queue manager.

Her alt alan, alt alana karşılık gelen konumlarda 1 bit ve başka bir yerde 0 biti olan bit maskesi ile tanımlanır. Bitler, 0 bit 'in en önemli bit, bit 31 'i ise en az anlamlı bit gibi numaralandırılmıştır. Alt alanları tanımlamak için aşağıdaki maskeleri tanımlar:

### MFRUM

Reddedilen ileti işaretleri için reddedilen maske.

Bu maske, yerel kuyruk yöneticisi tarafından desteklenmeyen ileti işaretlerinin MQPUT ya da MQPUT1 çağırısının tamamlanma kodu CCFAIL ve neden kodu RC2249 ile başarısız olmasına neden olacağı için, MDMFL alanındaki bit konumlarını tanıtır.

Bu alt alan, 20 ile 31 arasında bit konumlarını kaplar.



Bu alt alana aşağıdaki ileti işaretleri dahil edilir:

- MFLMIG
- MFLSEG
- MFMIG
- MFSEG
- MFSEGA
- MFSEGI

#### **MFAUM**

Kabul edilen desteklenmeyen ileti işaretleri için maske.

Bu maske, *MDMFL* alanında, yerel kuyruk yöneticisi tarafından desteklenmeyen ileti işaretlerinin, MQPUT ya da MQPUT1 çağrılarında kabul edileceğine ilişkin bit konumlarını tanıtır. Tamamlanma kodu CCOK ' tır.

Bu alt alan 0 ile 11 arasında bit pozisyonları kaplıyor.

#### **MFAUXM**

Yalnızca belirli durumlarda kabul edilen ileti işaretleri için maske.

This mask identifies the bit positions within the *MDMFL* field where message flags which are not supported by the local queue manager will nevertheless be accepted on the MQPUT or MQPUT1 calls *sağlandı* that both of the following conditions are satisfied:

- İletin kaderi uzak bir kuyruk yöneticisine ait.
- Uygulama iletiyi doğrudan yerel bir iletim kuyruğuna koymuyor (yani, MQOPER ya da MQPUT1 çağrısında belirlenen nesne tanımlayıcısındaki *ODMN* ve *ODON* alanları tarafından tanımlanan kuyruk, yerel bir iletim kuyruğu değil).

Bu koşullar karşılanırsa, tamamlanma kodu CCOK döndürülür ve değilse neden kodu RC2249 olan CCFAIL döndürülür.

Bu alt alan 12 ile 19 arasındaki pozisyonları kaplıyor.

*MDMFL* alanında, kuyruk yöneticisinin tanımadığı işaretler varsa, kuyruk yöneticisi, *MDMFL* alanını o alt alanın maskesiyle birleştirmek için bit ve işlemi kullanarak sırayla her alt alanı denetler. Bu işlemin sonucu sıfır değilse, daha önce açıklanan tamamlanma kodu ve neden kodları döndürülür.

IBM i

## **IBM üzerinde veri dönüştürme**

Bu konuda, veri dönüştürme çıkışa ilişkin arabirim ve veri dönüştürme gerektiğinde kuyruk yöneticisi tarafından gerçekleştirilen işleme anlatılır.

Veri dönüştürme çıkışı, MQGET çağrısının işlenmesinin bir parçası olarak çağrılır. Uygulama iletileri verilerini, alma uygulamasının gerektirdiği gösteriye dönüştürmek için kullanılır. Uygulama iletilerinin dönüştürülmesi isteğe bağlıdır ve MQGET çağrısında GMCONV seçeneğinin belirtilmesini gerektirir.

Veri dönüştürmenin aşağıdaki yönlerini açıklanmıştır:

- GMCONV seçeneğine yanıt olarak kuyruk yöneticisi tarafından gerçekleştirilen işleme; bkz. [“IBM üzerinde dönüştürme işlemi” sayfa 1414](#).
- Yerleşik bir biçim işlenirken kuyruk yöneticisi tarafından kullanılan işleme kuralları; kullanıcı tarafından yazılan çıkışlar için de bu kurallar önerilir. Bkz. [“IBM üzerindeki işleme kuralları” sayfa 1415](#).
- Rapor iletilerinin dönüştürülmesine özel dikkat edilmesi gereken noktalar; bkz. [“Conversion of report messages on IBM i” sayfa 1418](#).
- Veri dönüştürme çıkışa aktarılan parametreler; bkz. [“IBM üzerinde MQCONVX \(Veri dönüştürme çıkışı\)” sayfa 1429](#).
- Karakter verilerini farklı gösterimler arasında dönüştürmek için çıkışta kullanılacak bir çağrı; bkz. [“MQXCNCV \(Convert characters\) on IBM i” sayfa 1424](#).

- Çıkışa özel veri yapısı parametresi; bkz. “MQDXP (Data-conversion exit parameter) on IBM i” sayfa 1419.

## IBM i IBM üzerinde dönüştürme işlemi

Bu bilgilerde, kuyruk yöneticisi tarafından GMCONV seçeneğine yanıt olarak gerçekleştirilen işleme anlatılır.

MQGET çağrısında GMCONV seçeneği belirtilirse ve uygulamaya döndürülemeniz için bir ileti varsa, kuyruk yöneticisi aşağıdaki işlemleri gerçekleştirir:

1. Aşağıdakilerden biri ya da birkaçı doğruysa, dönüştürme gerekli değildir:

- İleti verileri, MQGET çağrısını yayınlayan uygulamanın gerektirdiği karakter kümesiyle ve kodlamalarında zaten var. Uygulama, çağrıyı yayınlamadan önce, MQGET çağrısının **MSGDSC** parametresindeki *MDCSI* ve *MDENC* alanlarını gerekli değerlere ayarlamalıdır.
- İleti verilerinin uzunluğu sıfır.
- MQGET çağrısına ilişkin **BUFFER** değiştirgesinin uzunluğu sıfır.

Bu durumda, MQGET çağrısını yayınlayan uygulamaya dönüştürme yapılmadan ileti döndürülür; **MSGDSC** parametresindeki *MDCSI* ve *MDENC* değerleri, iletteki denetim bilgilerindeki değerlere ayarlanır ve arama, aşağıdaki tamamlanma kodu ve neden kodlarından biriyle tamamlanır:

**Tamamlanma kodu**  
**Neden kodu**

**CCOK**

YOK

**CCWARN**

RC2079

**CCWARN**

RC2080

Aşağıdaki adımlar yalnızca, ileti verilerinin karakter kümesi ya da kodlaması **MSGDSC** parametresindeki karşılık gelen değerden farklıysa ve dönüştürülecek veriler varsa gerçekleştirilir:

1. İletideki denetim bilgilerindeki *MDFMT* alanı FMNONE değerine sahipse, ileti dönüştürülmeden, tamamlanma kodu CCWARN ve neden kodu RC2110 döndürülür.

Diğer tüm durumlarda dönüştürme işlemleri devam eder.

2. İleti kuyruktan kaldırılır ve **BUFFER** parametresiyle aynı boyutta olan geçici bir arabelleğe yerleştirilir. Göz atma işlemleri için, ileti kuyruktan kaldırılma yerine geçici arabelleğe kopyalanır.

3. İletin, arabelleğe sığması için kesilmesi gerekiyorsa, aşağıdaki yapılı:

- GGMATM seçeneği belirtilmediyse, ileti, tamamlanma kodu CCWARN ve neden kodu RC2080 ile dönüştürülemez olarak döndürülür.
- GMATM seçeneği *belirtilmişse*, tamamlanma kodu CCWARN olarak ayarlanır, neden kodu RC2079 olarak ayarlanır ve dönüştürme işlemi devam eder.

4. İleti, kesilmeden arabelleğe sığabiliyorsa ya da GGMATM seçeneği belirtilmişse, aşağıdaki yapılı:

- Biçim yerleşik bir biçimse, arabellek kuyruk yöneticisinin veri dönüştürme hizmetine geçirilir.
- Biçim, yerleşik bir biçim değilse, arabellek, biçimle aynı adı taşıyan bir kullanıcı tarafından yazılmış bir çıkışa geçirilir. Çıkış bulunamazsa, ileti, tamamlanma kodu CCWARN ve neden kodu RC2110 ile çevrilmemiş olarak döndürülür.

Hiçbir hata oluşmazsa, veri dönüştürme hizmetindeki ya da kullanıcı tarafından yazılan çıkıştan alınan çıkış, dönüştürülen iletidir ve MQGET çağrısını yayınlayan uygulamaya döndürülecek tamamlanma kodu ve neden kodu.

5. Dönüştürme başarılı olursa, kuyruk yöneticisi dönüştürülen iletiyi uygulamaya geri döndürür. Bu durumda, MQGET çağrısının döndürdüğü tamamlanma kodu ve neden kodu tipik olarak aşağıdaki birleşimlerden biri olur:

**Tamamlanma kodu**  
**Neden kodu**

**CCOK**  
YOK

**CCWARN**  
RC2079

Ancak, dönüştürme, kullanıcı tarafından yazılan bir çıkış tarafından gerçekleştirilirse, dönüştürme başarılı olduğunda da diğer neden kodları geri döndürülebilir.

Dönüştürme başarısız olursa (herhangi bir nedenden dolayı), kuyruk yöneticisi, **MSGDSC** parametresindeki *MDCSI* ve *MDENC* alanları, iletteki denetim bilgilerindeki değerlere ve tamamlanma kodu CCWARN değerine ayarlanmış olarak, dönüştürülemez iletiyi uygulamaya geri döndürür.

## IBM i IBM üzerindeki işleme kuralları

Yerleşik bir biçimi dönüştürürken, kuyruk yöneticisi bu konuda açıklanan işleme kurallarını izler.

Bu kuralları, kuyruk yöneticisi tarafından uygulanmamış olsa da, kullanıcı tarafından yazılan çıkışlara uygulamayı düşünün. Kuyruk yöneticisi tarafından dönüştürülen yerleşik biçimler aşağıdaki gibidir:

- FMADMN
- FMMDE
- FMCICS
- FMPCF
- FMCMD1
- FMRMH
- FMCMD2
- FNRFH
- FBIDLH
- FMRFH2
- FMDH
- FMSTR
- FMEVNT
- FMTM
- FMIMS
- FMXQH
- FMIMVS

1. İleti dönüştürme sırasında genişler ve **BUFFER** parametresinin boyutunu aşarsa, aşağıdaki yapılır:

- GGATM seçeneği belirtilmediyse, ileti, tamamlanma kodu CCWARN ve neden kodu RC2120 ile dönüştürülemez olarak döndürülür.
- GMATM seçeneği *belirtilmişse*, ileti kısaltılır, tamamlanma kodu CCWARN olarak ayarlanır, neden kodu RC2079 olarak ayarlanır ve dönüştürme işlemi devam eder.

2. Kesme gerçekleşirse (dönüştürme öncesinde ya da sırasında), **BUFFER** değiştirgesinde döndürülen geçerli byte sayısı, arabelleğin uzunluğunun *küçüktür* olarak döndürülebileceği bir sayı olabilir.

Bu durum, örneğin, 4 baytlık bir tamsayı ya da bir DBCS karakter arabelleğindeki arabelleğin sonunu getirirse ortaya çıkabilir. Bilgilerin tamamlanmamış ögesi dönüştürülmedi ve döndürülen iletide bu

byte 'lar geçerli bilgiler içermiyor. Dönüştürme sırasında dönüştürme daraltılmadan önce kesilmiş bir ileti varsa bu durum da oluşabilir.

Döndürülen geçerli byte sayısı arabelleğin uzunluğundan azsa, arabelleğin sonundaki kullanılmayan byte 'lar boş değere ayarlıdır.

3. Bir dizi ya da dizgi arabelleğin sonunu getirirse, olabildiğince çok veri dönüştürülür; yalnızca, eksik olan belirli bir dizi ögesi ya da DBCS karakteri dönüştürülmez-önceki dizi ögeleri ya da karakterler dönüştürülür.
  4. If truncation occurs (either before or during conversion), the length returned for the **DATLEN** parameter is the length of the *dönüştürülemez* message before truncation.
  5. Dizgiler, SBCS (tek baytlık karakter takımları), çift baytlık karakter takımları (DBCS) ya da çok baytlı karakter kümeleri (MBCS) arasında dönüştürüldüğünde, dizgiler genişletilebilir ya da daraltabilir.
    - PCF ' de FMADMN, FMEVNT ve FMPCF biçimlerinde, MQCFST ve MQCFSL yapılarındaki dizgiler, dönüştürmenin ardından dizgiyi sığdırmak için gereken şekilde genişletilecek ya da sözleşmeye devam eder.

MQCFSL dizgi listesi yapısı için, listedeki dizgiler farklı miktarlara göre genişletilebilir ya da daraltabilirler. Bu gerçekleşirse, kuyruk yöneticisi, dönüştürmeden sonraki en uzun dizgiyle aynı uzunlukta olacak şekilde, daha kısa dizgileri de boşluk karakteriyle destekler.
    - In the format FMRMH, the strings addressed by the RMSEO, RMSNO, RMDEO, and RMDNO fields expand or contract as necessary to accommodate the strings after conversion.
    - In the format FMRFH, the RFNVS field expands or contracts as necessary to accommodate the name-value pairs after conversion.
    - Değişmez alan büyüklükleriyle yapılan yapılarda, kuyruk yöneticisi önemli bir bilgi kaybolmadıysa, dizgilerin sabit alanları içinde genişletmesine ya da sözleşmeye izin verir. Bu bakımdan, alandaki ilk boş karakteri izleyen sondaki boşluk ve karakterler önemsiz olarak değerlendirilir.
      - Dizgi genişleirse, ancak yalnızca önemsiz karakterlerin, dönüştürülen dizgiyi alana sığdırmak için atılması gerekiyorsa, dönüştürme başarılı olur ve çağrı CCOK ile tamamlanır ve neden kodu RCNONE (başka hata bulunmadığını varsayarlar).
      - Dizgi genişleirse, ancak dönüştürülen dizgi, alana sığabilmek için önemli karakterler kullanılmasını gerektiriyorsa, ileti dönüştürülmez olarak döndürülür ve çağrı, CCWARN ve neden kodu RC2190 ile tamamlanır.
- Not:** Neden kodu RC2190 , bu durumda, GMATM seçeneğinin belirlenip belirlenmediğini ortaya atsin.
- Dizgi sözleşmeleri ise, kuyruk yöneticisi dizeyi, alanın uzunluğuna sahip olacak şekilde boşluklarla doldurur.
6. Kullanıcı verilerinin izlediği bir ya da daha çok IBM MQ üstbilgi yapısından oluşan iletiler için, bir ya da daha çok başlık yapısı dönüştürülebilirken, iletinin geri kalan kısmı da dönüştürülmez. Ancak, iki kural dışı durumla birlikte, her üstbilgi yapısındaki MDCSI ve MDENC alanları her zaman, üstbilgi yapısını izleyen verilerin karakter kümesini ve kodlamasını doğru bir şekilde gösterir.

Bu iki kural dışı durum, MQCIH ve MQIIH yapılarıdır; bu yapılar, bu yapılardaki MDCSI ve MDENC alanlarındaki değerlerin anlamlı olmamalarıdır. Bu yapılar için, yapıdan sonraki veriler, MQCIH ya da MQIIH yapısının kendisi ile aynı karakter kümesinde ve kodlamadır.

7. Alınmakta olan iletinin denetim bilgilerindeki MDCSI ya da MDENC alanları ya da **MSGDSC** değiştirilmesinde, tanımsız ya da desteklenmeyen değerler belirtin; tanımsız ya da desteklenmeyen değerlerin ileti dönüştürülmesinde kullanılması gerekmiyorsa, kuyruk yöneticisi hatayı yoksayabilir.

Örneğin, iletteki MDENC alanı desteklenmeyen bir kayar noktalı sayı kodlamasını belirtiyorsa, ancak ileti yalnızca tamsayı verileri içeriyorsa ya da dönüştürme gerektirmeyen kayar noktalı veri içeriyorsa (kaynak ve hedef kayar noktalı kodlamalar özdeş olduğu için), hata ortaya konabilir ya da saptanmayabilir.

Hata tanılanırsa, ileti dönüştürülemez olarak döndürülür; tamamlanma kodu CCWARN ve RC2111, RC2112, RC2113, RC2114 ya da RC2115, RC2116, RC2117, RC2118 neden kodları (uygun olduğu

şekilde); **MSGDSC** parametresindeki MDCSI ve MDENC alanları, iletteki denetim bilgilerindeki değerlere ayarlanır.

Hata tanısı konmazsa ve dönüştürme başarılı bir şekilde tamamlanırsa, **MSGDSC** parametresindeki MDCSI ve MDENC alanlarında döndürülen değerler, MQGET çağrısını yayınlayan uygulama tarafından belirtilenler olur.

8. Tüm durumlarda, ileti uygulamaya döndürülürse, tamamlanma kodu, tamamlanma kodu CCWARN olarak ayarlanır ve **MSGDSC** parametresindeki MDCSI ve MDENC alanları, dönüştürülenmemiş veriler için uygun olan değerlere ayarlanır. Bu, FMNONE için de yapılır.

**REASON** parametresi, iletinin kesilmesini de gerekemediği sürece, dönüştürmenin neden gerçekleştirilemediğini gösteren bir kod olarak ayarlanır; kesme ile ilgili neden kodları, dönüştürme ile ilgili neden kodlarına göre önceliklidir. (Kesilen bir iletinin dönüştürülmüş olup olmadığını saptamak için, **MSGDSC** parametresindeki MDCSI ve MDENC alanlarında döndürülen değerleri denetleyin.)

Bir hata tanıldığında, belirli bir neden kodu döndürülür ya da genel neden kodu RC2119' dur. Döndürülen neden kodu, temeldeki veri-dönüştürme hizmetinin tanılama yeteneklerine bağlıdır.

9. Tamamlanma kodu CCWARN döndürülürse ve birden fazla neden kodu uygun ise, öncelik sırası aşağıdaki gibidir:

a. Aşağıdaki neden, diğerlerinden öncelikli olarak uygulanır:

- RC2079

b. Sonraki öncelik, aşağıdaki nedendir:

- RC2110

c. Geri kalan neden kodlarına göre öncelik sırası tanımlanmadı.

10. MQGET çağrısının tamamlandığı tarih:

- Aşağıdaki neden kodu, iletinin başarıyla dönüştürülmüş olduğunu gösterir:
  - YOK
- The following reason code indicates that the message *may* have been converted successfully (check the MDCSI and MDENC fields in the **MSGDSC** parameter to find out):
  - RC2079
- Diğer tüm neden kodları, iletinin dönüştürülmediğini gösterir.

Aşağıdaki işlemler yerleşik biçimler için özgüdür; kullanıcı tanımlı biçimler için geçerli değildir:

1. Aşağıdaki biçimler dışındadır:

- FMADMN
- FMEVNT
- FMIMVS
- FMPCF
- FMSTR

Yerleşik biçimlerin hiçbiri, kuyruk adlarında geçerli olan karakterler için SBCS karakterleri olmayan karakter kümelerinden ya da bu biçimlerden dönüştürülemez. Böyle bir dönüştürmeyi gerçekleştirmek için bir girişimde bulunulursa, ileti dönüştürülmez olarak döndürülür; tamamlanma kodu CCWARN ve neden kodu RC2111 ya da RC2115' in uygun olduğu şekilde döndürülür.

Unicode karakter takımı UTF-16 , kuyruk adlarında geçerli olan karakterler için SBCS karakterleri olmayan bir karakter takımı örneğidir.

2. Yerleşik bir biçimin ileti verileri kesilirse, ileti içindeki alanlar ya da öge ya da yapı sayısı, uygulamaya döndürülen verilerin uzunluğunu yansıtacak şekilde ayarlanmaz; ileti verileri içinde bu tür alanlar için döndürülen değerler, kesilmeden önce ileti için geçerli olan değerlerdir.

Kesilen bir FMADMN iletisi gibi iletiler işlenirken, uygulamanın döndürülen verilerin sonuna kadar verilere erişmeye çalışmadığından emin olmak için dikkatli olmanız gerekir.

3. Biçim adı FMDLH ise, ileti verileri bir MQLH yapısıyla başlar ve bu, sıfır ya da daha fazla sayıda uygulama iletisi verisi izlemiş olabilir. Uygulama iletisi verilerinin biçimi, karakter kümesi ve kodlaması, iletinin başlangıcındaki MQLH yapısındaki DLFMT, DLCSİ ve DLENC alanları tarafından tanımlanır. MQLH yapısı ve uygulama iletisi verilerinin farklı karakter kümeleri ve kodlamalar olabildiği için, MQLH yapısının biri ya da her ikisi ya da her ikisi için de dönüştürme gerektirecek veriler olabilir.

Kuyruk yöneticisi, önce MQLH yapısını gerektiği gibi dönüştürür. Dönüştürme başarılı olursa ya da MQLH yapısı dönüştürme gerektirmiyorsa, kuyruk yöneticisi, MQLH yapısındaki DLCSİ ve DLENC alanlarını, uygulama iletisi verilerinin dönüştürülmesinin gerekli olup olmadığını görmek için denetler. Dönüştürme gerekiyorsa, kuyruk yöneticisi kullanıcı tarafından yazılan çıkışı, MQLH yapısındaki DLFMT alanı tarafından belirtilen adla çağırır ya da dönüştürmeyi gerçekleştirir ( DLFMT , yerleşik bir biçimin adıdır).

MQGET çağırısı, CCWARN tamamlanma kodunu döndürürse ve neden kodu, dönüştürme işleminin başarısız olduğunu gösteren kodlardan biri olursa, aşağıdakilerden biri uygulanır:

- MQLH yapısı dönüştürülemedi. Bu durumda, uygulama iletisi verileri de dönüştürülmez.
- MQLH yapısı dönüştürüldü, ancak uygulama iletisi verileri dönüştürülmedi.

The application can examine the values returned in the MDCSI and MDENC fields in the **MSGDSC** parameter, and those in the MQLH structure, in order to determine which of the previous applies.

4. Biçim adı FMXQH ise, ileti verileri bir MQXQH yapısıyla başlar ve bu, sıfır ya da daha fazla bayt ek veriyle izlenebilir. Bu ek veriler tipik olarak, uygulama iletisi verilerinden (sıfır uzunluklu olabilir), ancak ek verilerin başlangıcındaki bir ya da daha fazla sayıda IBM MQ üstbilgi yapısı da olabilir.

MQXQH yapısı, kuyruk yöneticisinin karakter kümesinde ve kodlamasında olmalıdır. MQXQH yapısını izleyen verilerin biçimi, karakter kümesi ve kodlaması, MQXQH içinde yer alan MQMD yapısındaki MDFMT, MDCSİ ve MDENC alanları tarafından verilir. Sonraki her IBM MQ üstbilgi yapısı için, yapıdaki MDFMT, MDCSİ ve MDENC alanları, o yapıyı izleyen verileri açıklar; bu veriler, başka bir IBM MQ üstbilgi yapısı ya da uygulama iletisi veridir.

Bir FMXQH iletisi için GMCONV seçeneği belirtilirse, uygulama iletisi verileri ve MQ üstbilgi yapılarının belli bir değeri dönüştürülür, ancak MQXQH yapısındaki veriler dönüştürülür. Bu nedenle, MQGET çağırısından geri dönün:

- The values of the MDFMT, MDCSI, and MDENC fields in the **MSGDSC** parameter, describe the data in the MQXQH structure, and not the application message data; the values will therefore not be the same as those specified by the application that issued the MQGET call.

The effect of this is that an application which repeatedly gets messages from a transmission queue with the GMCONV option specified must reset the MDCSI and MDENC fields in the **MSGDSC** parameter to the values necessary for the application message data, before each MQGET call.

- Son MQ üstbilgi yapısındaki MDFMT, MDCSİ ve MDENC alanlarının değerleri, uygulama iletisi verilerini açıklar. Başka bir IBM MQ üstbilgi yapısı yoksa, uygulama iletisi verileri bu alanlar tarafından MQXQH yapısındaki MQMD yapısıyla tanımlanır. Dönüştürme başarılı olursa, değerler, MQGET çağırısını yayınlayan uygulama tarafından **MSGDSC** parametresindeki belirtilerle aynı olur.

İleti bir dağıtım listesi iletirse, MQXQH yapısının ardından bir MQDH yapısı (artı MQOR ve MQPMR kayıtları dizileri) gelir. Bu yapı, sıfır ya da daha fazla ek IBM MQ üstbilgi yapısı ve sıfır ya da daha fazla sayıda uygulama iletisi verilerinin izlenebileceği bir MQDH yapısı (MQOR ve MQPMR kayıtları dizileridir). MQXQH yapısı gibi, MQDH yapısı kuyruk yöneticisinin karakter kümesinde ve kodlamasında olmalıdır; GMCONV seçeneği belirtilse bile, MQGET çağırısına dönüştürülmez.

Daha önce açıklanan MQXQH ve MQDH yapılarının işlenmesi öncelikli olarak ileti kanalı araçları tarafından iletim kuyruklarından ileti aldıklarında kullanılmak üzere tasarlanmıştır.

Bir rapor iletisi, özgün iletinin göndericisinin belirttiği rapor seçeneklerine göre değişen miktarlarda uygulama iletisi verisi içerebilir.

Belirli bir durumda, bir rapor iletisi aşağıdaki gibi olabilir:

1. Uygulama iletisi verisi yok
2. Özgün iletiden bazı uygulama iletisi verileri

Bu durum, özgün iletinin göndericisinin RO\* D belirttiğinde ve iletinin 100 byte 'tan uzun olduğunu gösterir.

3. Özgün iletiden gelen tüm uygulama iletisi verileri

Bu durum, özgün iletiyi gönderen RO\* F 'yi belirlerken ya da RO\* D 'yi belirlerken ve iletinin 100 byte ya da daha kısa olduğunu belirtir.

When the queue manager or message channel agent generates a report message, it copies the format name from the original message into the *MDFMT* field in the control information in the report message. Bu nedenle, rapor iletisinde biçim adı, rapor iletisinde bulunan uzunluktan farklı olan bir veri uzunluğu anlamına gelebilir (daha önce açıklanan 1. ve 2. durumlar).

Rapor iletisi alındığında GMCONV seçeneği belirtilirse:

- Daha önce açıklanan 1 numaralı vaka için, veri dönüştürme çıkışı çağrılmaz (rapor iletisine veri verilmeyecek).
- Daha önce açıklanan 3. vakada, biçim adı ileti verilerinin uzunluğunu doğru biçimde ifade eder.
- Ancak daha önce açıklanan 2. vaka için, veri dönüştürme çıkışı, biçim adının belirttiği uzunluktan *daha kısa* olan bir iletiyi dönüştürmek için çağrılır.

Buna ek olarak, çıkışa geçirilen neden kodu genellikle RCNONE (yani, iletinin kesildiğini belirtmez) olur. Bu durum, ileti verilerinin, rapor iletisinin *gönderen* tarafından kesildiğinden ve MQGET çağrısına yanıt olarak alıcının kuyruk yöneticisi tarafından kesilmediği için ortaya çıktı.

Bu olasılıklardan dolayı, veri-dönüştürme çıkışı, aktarılan verilerin uzunluğunu nasıl algılayabilmek için biçim adını kullanmamalıdır; bunun yerine, çıkış, sağlanan verilerin uzunluğunu denetleyerek, biçim adının belirttiği uzunluktan daha az veri dönüştürmeye hazır olmasını sağlamalıdır. Veriler başarılı bir şekilde dönüştürülebiliyorsa, çıkış kodu CCOK ve neden kodu RCA 'nın çıkış tarafından döndürülmesi gerekir. The length of the message data to be converted is passed to the exit as the **INLEN** parameter.

## Ürüne duyarlı programlama arabirimi

Bir rapor iletisi, gerçekleşen bir etkinle ilgili bilgi içeriyorsa, bu, etkinlik raporu olarak bilinir. Etkinlik örnekleri:

- Bir kanala bir kuyruktan ileti gönderen MCA
- MCA bir kanaldan ileti alan ve bunu bir kuyruğa yerleştiren
- Bir MCA ölü mektubu teslim edilemeyen bir ileti kuyruğa giriyor
- MCA bir iletiyi kuyruğundan alma ve atma
- Kuyruğun üzerine bir ileti yerleştiren bir ölü-harf işleyici
- Komut sunucusu bir PCF isteğini işliyor-yayınlama isteğini işleyen bir aracı
- Kuyruktan ileti alan bir kullanıcı uygulaması-bir kullanıcının kuyruğunda bir iletiye göz atma kullanıcı uygulaması

Kuyruk yöneticisi de içinde olmak üzere herhangi bir uygulama, rapor üstbilgisinden sonra ileti verilerinin bir kısmını etkinlik raporuna ekleyebilir. Bazı veriler gönderilirse sağlanması gereken veri miktarı sabit değildir ve uygulama tarafından kararlaştırılır. Döndürülen bilgiler, etkinlik raporunu işleyen uygulama için yararlı olmalıdır. Kuyruk yöneticisi etkinlik raporları, özgün iletide yer alan standart IBM MQ üstbilgi yapılarını ('MQH' başlangıcı) geri döndürür. Bu, örneğin, özgün iletide yer alan herhangi bir MQRFH2 üstbilgileri içerir. Ayrıca, kuyruk yöneticisi bir MQCFH üstbilgisi döndürür, ancak bununla ilişkili PCF değiştirgelerini döndürür. Bu, izleme uygulamalarının, iletinin ne hakkında olduğuna ilişkin bir fikir verir.

## IBM i MQDXP (Data-conversion exit parameter) on IBM i

Veri dönüştürme çıkış parametresi öbeği.

## Genel Bakış

**Amaç:** MQDXP yapısı, kuyruk yöneticisinin, ileti verilerini MQGET çağrısının işlenmesinin bir parçası olarak dönüştürmek için çıkış çağrıldığında veri dönüştürme çıkışa geçtiği bir parametredir. Veri dönüştürme çıkışa ilişkin ayrıntılar için MQCONVX çağrısının açıklamasına bakın.

**Karakter kümesi ve kodlama:** MQDXP ' deki karakter verileri, yerel kuyruk yöneticisinin karakter takımında yer alıyor; bu, **CodedCharSetId** kuyruk yöneticisi özniteliği tarafından verilir. MQDXP ' deki sayısal veriler yerel makine kodlamasıdır; bu, ENNAT tarafından verilir.

**Kullanım:** MQDXP ' de yalnızca *DXLEN*, *DXCC*, *DXREA* ve *DXRES* alanları, çıkışta değiştirilebilir; diğer alanlardaki değişiklikler yoksayılr. Ancak dönüştürülmekte olan ileti, mantıksal iletinin yalnızca bir kısmını içeren bir bölümse, *DXLEN* alanı değiştirilemez.

Denetim çıkış kuyruğundan kuyruk yöneticisine geri döndüğünde, kuyruk yöneticisi MQDXP ' de döndürülen değerleri denetler. If the values returned are not valid, the queue manager continues processing as though the exit had returned XRFAIL in *DXRES* ; however, the queue manager ignores the values of the *DXCC* and *DXREA* fields returned by the exit in this case, and uses instead the values those fields had on *giriş* to the exit. MQDXP ' de aşağıdaki değerler, bu işlemin gerçekleşmesine neden olur:

- *DXRES* alanı XROK değil ve XRFAIL değil
- *DXCC* alanı CCOK değil ve CCWARN değil
- Dönüştürülmekte olan ileti bir mantıksal iletinin yalnızca bir kısmını içeren bir bölümse, *DXLEN* alanı sıfırdan küçük ya da *DXLEN* alanı değişmektedir.
- [“Alanlar” sayfa 1420](#)
- [“RPG bildirim \(CMQDXPH dosyasını kopyala\)” sayfa 1424](#)

## Alanlar

MQDXP yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

### **DXAOP (10 basamaklı işaretli tamsayı)**

Uygulama seçenekleri.

Bu, MQGET çağrısını yayınlayan uygulama tarafından belirtilen MQGMO yapısının *GMOPT* alanının bir kopyasıdır. Çıkışta, GMATM seçeneğinin belirlenip belirlenmediğini saptamak için bu çıkışa bakmanız gerekebilir.

Bu, çıkışa giriş alanıdır.

### **DXCC (10 basamaklı işaretli tamsayı)**

Tamamlanma kodu.

Çıkış çağrıldığında, çıkış hiçbir şey yapmamayı seçerse, MQGET çağrısını yayınlayan uygulamaya döndürülecek tamamlanma kodunu içerir. Bu her zaman CCWARN, çünkü ya ileti kesildi ya da ileti dönüştürülmeyi gerektiriyor ve bu henüz yapılamadı.

Çıkıştan çıkışta, bu alan MQGET çağrısının **CMPCOD** parametresindeki uygulamaya döndürülebilme için tamamlanma kodunu içerir; yalnızca CCOK ve CCWARN geçerli olur. Çıkışta çıkışın bu alanı nasıl ayarlamaması gerektiğine ilişkin öneriler için *DXREA* alanının açıklamasına bakın.

Bu, çıkışa giriş/çıkış alanıdır.

### **DXCSI (10 basamaklı işaretli tamsayı)**

Uygulama için karakter kümesi gerekiyor.

Bu, MQGET çağrısını yayınlayan uygulama tarafından gerekli olan karakter kümesinin kodlanmış karakter takımı tanıtıcısıdır; ek ayrıntılar için MQMD yapısındaki *MDCSI* alanına bakın. Uygulama, MQGET çağrısındaki CSQM özel değerini belirtiyorsa, kuyruk yöneticisi bu değeri, çıkış çağrılmadan



önce kuyruk yöneticisi tarafından kullanılan karakter kümesinin gerçek karakter kümesi tanıtıcısında değiştirir.

Dönüştürme başarılı olursa, çıkışta bu değeri ileti tanımlayıcısındaki *MDCSI* alanına kopyalamalıdır.

Bu, çıkışa giriş alanıdır.

#### **DXENC (10 basamaklı işaretli tamsayı)**

Uygulama için sayısal kodlama gerekiyor.

Bu, MQGET çağrısını yayınlayan uygulamanın gerektirdiği sayısal kodlamadır; ek ayrıntılar için MQMD yapısındaki *MDENC* alanına bakın.

Dönüştürme başarılı olursa, çıkışta bu değeri ileti tanımlayıcısındaki *MDENC* alanına kopyalamalıdır.

Bu, çıkışa giriş alanıdır.

#### **DXHCN (10 basamaklı işaretli tamsayı)**

Bağlantı tanıtıcısı.

Bu, MQXCNCV çağrısında kullanılacak bir bağlantı tanıtıcısıdır. Bu tanıtıcı değeri, MQGET çağrısını yayınlayan uygulama tarafından belirlenen tanıtıcı ile aynı değil.

#### **DXLEN (imzalanmış 10 basamaklı tamsayı)**

İleti verilerinin bayt cinsinden uzunluğu.

Çıkış çağrıldığında, bu alanda uygulama iletisi verilerinin özgün uzunluğu yer alır. İleti, uygulama tarafından sağlanan arabelleğe sığacak şekilde kısaltıldıysa, çıkışa sağlanan iletinin boyutu *DXLEN*değerinden *daha küçük* olur. Çıkışa sağlanan iletinin boyutu, her zaman çıkışa ilişkin **INLEN** parametresi tarafından verilir; ortaya çıkan herhangi bir kesmeden bağımsız olarak, çıkışa ilişkin ileti.

Kesme, çıkışa giriş sırasında RC2079 değerinin sahip olduğu *DXREA* alanı tarafından gösterilir.

Çoğu dönüştürmenin bu uzunluğu değiştirmesi gerekmez, ancak gerekirse bir çıkış yapabilir; çıkışa göre ayarlanan değer, MQGET çağrısının **DATLEN** parametresindeki uygulamaya geri döndürülür. Ancak, dönüştürülmekte olan ileti, mantıksal iletinin yalnızca bir kısmını içeren bir bölümse bu uzunluk değiştirilemez. Bunun nedeni, uzunluğun değiştirilmesi, mantıksal iletinin sonraki bölümlerinin yanlış olarak belirlenmesine neden olur.

Çıkış, verilerin uzunluğunu değiştirmek isterse, kuyruk yöneticisinin *dönüştürülemez* verilerinin uzunluğuna dayalı olarak, ileti verilerinin uygulamanın arabelleğiyle uyup uymayacağına karar verdiğiinden haberdar olun. Bu karar, iletinin kuyruktan kaldırılıp kaldırılmayacağını (ya da bir göz atma isteği için imleci hareket ettirerek) belirler ve dönüştürmenin neden olduğu veri uzunluğunda yapılan herhangi bir değişiklikten etkilenmez. Bu nedenle, dönüştürme çıkışlarının, uygulama iletisi verilerinin uzunluğunda bir değişikliğe neden olmamasını önermektedir.

Karakter dönüştürmesi bir uzunluk değişikliği anlamına geldiyse, bir dizgi byte olarak aynı uzunluğa sahip başka bir dizgiye dönüştürülebilir, sondaki boşlukları kesilebilir ya da gerektiği gibi boşluklarla doldurulabilir.

The exit is not invoked if the message contains no application message data; hence *DXLEN* is always greater than zero.

Bu, çıkışa giriş/çıkış alanıdır.

#### **DXREA (10 basamaklı işaretli tamsayı)**

Neden kodu ön eleme *DXCC*.

Çıkış çağrıldığında, bu durum, çıkış hiçbir şey yapmayı seçerse, MQGET çağrısını yayınlayan uygulamaya döndürülebilir neden kodunu içerir. Olası değerler arasında, iletinin, uygulamanın sağladığı arabelleğe sığması için kesildiğini belirten RC2079ve iletinin dönüştürülmesini gerektirdiğini, ancak bunun henüz gerçekleştirilmediğini belirten RC2119.

Çıkışta, bu alan, MQGET çağrısının **REASON** parametresindeki uygulamaya döndürülmesinin nedenini içerir; aşağıdaki durum önerilir:

- If *DXREA* had the value *RC2079* on input to the exit, the *DXREA* and *DXCC* fields should not be altered, irrespective of whether the conversion succeeds or fails.

(*DXCC* alanı *CCOK* değilse, iletiyi alan uygulama, ileti tanımlayıcısındaki döndürülen *MDENC* ve *MDCSI* değerlerini istenen değerlerle karşılaştırarak bir dönüştürme hatasını tanımlayabilir; buna karşılık, uygulama, kesilen bir iletiyi, arabelleği yeni yerleştiren bir iletiden ayırt edemez. For this reason, *RC2079* should be returned in preference to any of the reasons that indicate conversion failure.)

- *DXREA* ' un çıkışa giriş üzerinde başka bir değeri varsa:
    - Dönüştürme başarılı olursa, *DXCC* *CCOK* olarak ayarlanmalıdır ve *DXREA* , *RCNONE* olarak ayarlanmış olmalıdır.
    - Dönüştürme başarısız olursa ya da ileti genişleirse ve arabelleğe sığmak için kesilmek zorunda kaldıysa, *DXCC* ' un *CCWARN* (değişmez) olarak ayarlanması (ya da değiştirilmeden bırakılmalıdır) ve *DXREA* , hatanın doğasını belirtmek için aşağıdaki listedeki değerlerden birine ayarlanmalıdır.
- Dönüştürme işleminden sonraki ileti arabellek için çok büyük olursa, yalnızca *MQGET* çağrısını yayınlayan uygulama *GMATM* seçeneğini belirttiyse kısaltılmalıdır:
- Bu seçenek belirlendiyse, *RC2079* neden döndürülmelidir.
  - Bu seçeneğin belirlenmemesi durumunda, iletinin dönüştürülmemiş olarak döndürülmesi gerekir; neden kodu *RC2120*.

Aşağıdaki listede yer alan neden kodlarının, dönüştürmenin neden başarısız olduğunu belirtmek için çıkışta kullanılması önerilir; ancak, çıkış uygun görülürse, çıkış, *RCA* kodları kümesinden diğer değerleri döndürebilir. Bunun yanı sıra, çıkışa göre, çıkışta *MQGET* çağrısını veren uygulamayla iletişim kurmak istediği koşulları belirtmek için, *RC0900* - *RC0999* arasındaki değerler de bu çıkış tarafından kullanılmak üzere ayrılır.

**Not:** İleti başarıyla dönüştürülemezse, kuyruk yöneticisinin dönüştürülenmemiş iletiyi döndürmesi için, çıkışa *DXRES* alanında *XRFAIL* döndürülmesi gerekir. Bu, *DXREA* alanında döndürülen neden kodundan bağımsız olarak geçerlidir.

#### **RC0900**

(900, X'384 ') Uygulama tanımlı neden kodu için En Düşük değeri.

#### **RC0999**

(999, X'3E7') Uygulama tanımlı neden kodlarına ilişkin en yüksek değer.

#### **RC2120**

(2120, X'848 ') Dönüştürülen veriler arabellek için çok büyük.

#### **RC2119**

(2119, X'847 ') İleti verileri dönüştürülmedi.

#### **RC2111**

(2111, X'83F') Kaynak kodlanmış karakter takımı tanıtıcısı geçerli değil.

#### **RC2113**

(2113, X'841 ') İletide paketli ondalık kodlama tanınmadı.

#### **RC2114**

(2114, X'842 ') İletide kayan noktalı kodlama tanınmadı.

#### **RC2112**

(2112, X'840 ') Kaynak tamsayı kodlaması tanınmadı.

#### **RC2115**

(2115, X'843 ') Hedef kodlanmış karakter takımı tanıtıcısı geçerli değil.

#### **RC2117**

(2117, X'845 ') Packed-receiver tarafından belirtilen paket-ondalık kodlama tanınmıyor.

#### **RC2118**

(2118, X'846 ') Alıcı tarafından belirtilen kayan noktalı kodlama tanınmıyor.

**RC2116**

(2116, X'844 ') Hedef tamsayı kodlaması tanınmadı.

**RC2079**

(2079, X'81F') Kesilen ileti döndürüldü (işleme tamamlandı).

Bu, çıkışa giriş/çıkış alanıdır.

**DXRES (10 basamaklı işaretli tamsayı)**

Çıkıştan yanıt.

Bu, başarıyı ya da dönüştürmenin başka bir şekilde belirtilmesini sağlamak için çıkışa göre ayarlanır. Aşağıdakilerden biri olmalıdır:

**XROK**

Dönüştürme başarılı oldu.

Çıkış bu değeri belirtiyorsa, kuyruk yöneticisi, MQGET çağrısını yayınlayan uygulamaya aşağıdaki gibi döner:

- Çıkışta çıkışta bulunan *DXCC* alanının değeri
- Çıkışta çıkışta bulunan *DXREA* alanının değeri
- Çıkışta çıkışta bulunan *DXLEN* alanının değeri
- The contents of the exit's output buffer *OUTBUF*. Döndürülen bayt sayısı, çıkışın **OUTLEN** parametresinden daha küçük olan ve çıkışta *DXLEN* alanının değeri.

If the *MDENC* and *MDCSI* fields in the exit's message descriptor parameter are *her ikisi* unchanged, the queue manager returns:

- Çıkışa *giriş* üzerindeki MQDXP yapısındaki *MDENC* ve *MDCSI* alanlarının değeri

Çıkışta, ileti tanımlayıcısı değiştirilmesindeki *MDENC* ve *MDCSI* alanlarının biri ya da her ikisi değiştirildiyse, kuyruk yöneticisi şunları döndürür:

- Çıkıştaki çıkışta çıkış iletisi tanımlayıcısı parametresindeki *MDENC* ve *MDCSI* alanlarının değeri
- 

**XRFAIL**

Dönüştürme başarısız oldu.

Çıkış bu değeri belirtiyorsa, kuyruk yöneticisi, MQGET çağrısını yayınlayan uygulamaya aşağıdaki gibi döner:

- Çıkışta çıkışta bulunan *DXCC* alanının değeri
- Çıkışta çıkışta bulunan *DXREA* alanının değeri
- Çıkışa *girişte* ilişkin *DXLEN* alanının değeri
- The contents of the exit's input buffer *INBUF*. Döndürülen bayt sayısı **INLEN** parametresiyle verilir.

Çıkış *INBUF*değiştirildiyse, sonuçlar tanımsız olur.

*DXRES*, çıkışa ait bir çıkış alanıdır.

**DXSID (4 byte 'lık karakter dizisi)**

Yapı tanıtcısı.

Değer şu olmalıdır:

**DXSIDV**

Veri dönüştürme çıkış parametresi yapısıyla ilgili tanıtcı.

Bu, çıkışa giriş alanıdır.

## DXVER (10 basamaklı işaretli tamsayı)

Yapı sürüm numarası.

Değer şu olmalıdır:

### DXVER1

Veri dönüştürme çıkış parametresi yapısı için sürüm numarası.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

### DXVERC

Veri dönüştürme çıkış değiştirgesi yapısının yürürlükteki sürümü.

**Not:** Bu yapının yeni bir sürümü kullanıma sunulduğunda, var olan parçanın düzeni değiştirilmez. Bu nedenle, çıkışta, *DXVER* alanının, çıkışa gerek duyduğu alanları içeren en düşük sürüme eşit ya da bu sürümden daha büyük olduğunu doğrulayın.

Bu, çıkışa giriş alanıdır.

## DXXOP (10 basamaklı işaretli tamsayı)

Ayrılmış.

Bu ayrılmış bir alandır; değeri 0' dir.

## RPG bildirim (CMQDXPH dosyasını kopyala)

```
D* .1.....2.....3.....4.....5.....6.....7..
D* MQDXP Structure
D*
D* Structure identifier
D DXSID 1 4
D* Structure version number
D DXVER 5 8I 0
D* Reserved
D DXXOP 9 12I 0
D* Application options
D DXAOP 13 16I 0
D* Numeric encoding required by application
D DXENC 17 20I 0
D* Character set required by application
D DXCSI 21 24I 0
D* Length in bytes of message data
D DXLEN 25 28I 0
D* Completion code
D DXCC 29 32I 0
D* Reason code qualifying DXCC
D DXREA 33 36I 0
D* Response from exit
D DXRES 37 40I 0
D* Connection handle
D DXHCN 41 44I 0
```

IBM i

## MQXCNCV (Convert characters) on IBM i

MQXCNCV çağrısı, karakterleri bir karakter kümesinden başka bir karaktere dönüştürür.

Bu çağrı, IBM MQ Framework arabirimlerinden biri olan DCI ( IBM MQ Data Conversion Interface; Veri Dönüştürme Arabirimi) ' nin bir parçasıdır. Not: Bu çağrı yalnızca veri dönüştürme çıkışından kullanılabilir.

- “Sözdizimi” sayfa 1425
- “Parametreler” sayfa 1425
- “RPG çağrısı (ILE)” sayfa 1429

## Sözdizimi

**MQXCNCV HCONN, OPTS, SRCCSI, SRCLEN, SRCBUF, TGTCSI, TGTLEN, TGTBUF, DATLEN, CMPCOD, REASON)**

## Parametreler

MQXCNCV çağrısında aşağıdaki parametreler yer aldı:

### **HCONN (10 basamaklı işaretli tamsayı)-giriş**

Bağlantı tanıtıcısı.

Bu tanıtıcı, kuyruk yöneticisiyle olan bağlantıyı gösterir. Olağan durumda, MQDXP yapısının DXHCN alanındaki veri dönüştürme çıkışından geçirilen tanıtıcı olmalıdır; bu tanıtıcı, MQGET çağrısını yayınlayan uygulama tarafından belirlenen tanıtıcı ile aynı olmayabilir.

IBM üzerinde, HCONN için aşağıdaki özel değer belirtilebilir:

### **HCDEFH**

Varsayılan bağlantı tanıtıcısı.

### **OPTS (10 basamaklı işaretli tamsayı)-giriş**

MQXCNCV işlemini denetleyen seçenekler.

Bu kısımda daha sonra açıklanan seçeneklerden sıfır ya da daha fazlası belirtilebilir. Birden çok değer gerekliyse, değerler eklenebilir (aynı değışmezi bir kereden fazla eklemeyin).

**Default-conversion option**(Varsayılan dönüştürme seçeneđi): Aşğıdaki seçenek, varsayılan karakter dönüştürmenin kullanımını denetler:

### **DCCDEF**

Varsayılan dönüştürme.

Bu seçenek, çağrıda belirtilen karakter kümelerinden biri ya da her ikisi desteklenmiyorsa, varsayılan karakter dönüştürmenin kullanılabilmesini belirtir. Bu, kuyruk yöneticisinin dizeyi dönüştürürken belirtilen karakter kümesine yaklaşan, kuruluş tarafından belirlenmiş bir varsayılan karakter kümesi kullanmasını sağlar.

**Not:** Dizgiyi dönüştürmek için yaklaşık bir karakter kümesi kullanılmasının sonucu, bazı karakterlerin yanlış dönüştürülebileceğini ifade eder. Yalnızca, hem belirlenen karakter takımı, hem de varsayılan karakter kümesi için ortak olan karakterler dizgi içinde kullanılmaktan kaçınılabılır.

Varsayılan karakter kümeleri, kuyruk yöneticisi kurulu ya da yeniden başlatıldığında bir yapılandırma seçeneđi tarafından tanımlanır.

DCCDEF belirtilmezse, kuyruk yöneticisi dizgiyi dönüştürmek için yalnızca belirtilen karakter kümelerini kullanır ve karakter kümelerinden biri ya da her ikisi desteklenmiyorsa çağrı başarısız olur.

**Doldurma seçeneđi:** Aşğıdaki seçenek, dönüştürülen dizginin hedef arabelleđe sığmasını sağlamak için, kuyruk yöneticisinin dönüştürülen dizeyi boşluklarla doldurmasına ya da önemsiz sondaki karakterleri atmasına olanak sağlar.

### **DCCFIL**

Hedef arabelleđi doldur.

Bu seçenek, hedef arabelleđin tam olarak doldurulduđunu böyle bir şekilde dönüştürmenin gerçekleşmesini ister:

- Dönüştürüldüğünde dizgi sözleşmeleri varsa, hedef arabelleđi doldurmak için sondaki boşluklar eklenir.
- Dizgi dönüştürüldüğünde genişlerse, dönüştürülen dizginin hedef arabelleđe sığmasını sağlamak için anlamlı olmayan sondaki karakterler atılır. Bu işlem başarıyla gerçekleştirilebilecekse, çağrı CCOK ile tamamlanır ve neden kodu RCNONE.

Çok az sayıda önemsiz sonda karakter varsa, dizginin sığacağı kadar dize hedef arabelleğe yerleştirilir ve arama, CCWARN ve neden kodu RC2120 ile tamamlanır.

Önemsiz karakterler şunlardır:

- Sondaki boşluklar
- Dizgideki ilk boş karakteri izleyen karakterler (ilk boş karakterin kendisi dışında)
- Dizgi, TGTCSI ve TGTLEN , hedef arabellek geçerli karakterler ile tamamen ayarlanamazsa, çağrı CCFAIL ve neden kodu RC2144 ile başarısız olur. TGTCSI salt DBCS karakter kümesi (UTF-16 gibi) olduğunda bu durum oluşabilir; ancak TGTLEN , tek sayıda byte olan bir uzunluk belirtir.
- TGTLEN , SRCLENDegerinden küçük ya da daha büyük olabilir. MQXCNCV ' den dönüşte DATLEN , TGTLEN ile aynı değere sahiptir.

Bu seçenek belirlenmezse:

- Dizilimin, hedef arabellek içinde gerektiği şekilde sözleşmesine ya da genişletmesine izin verilir. Önemsiz sondaki karakterler eklenmez ya da atılmaz.

Dönüştürülen dizgi hedef arabelleğe sığarsa, çağrı CCOK ile tamamlanır ve neden kodu RCNONE olur.

Dönüştürülen dizgi hedef arabellek için çok büyükse, dizginin büyük bölümü hedef arabelleğe sığacak şekilde yerleştirilir ve çağrı, CCWARN ve neden kodu RC2120 ile tamamlanır. Bu durumda TGTLEN baytından daha az bayt döndürülebileceğini unutmayın.

- TGTLEN , SRCLENDegerinden küçük ya da daha büyük olabilir. MQXCNCV ' den dönüşte DATLEN , TGTLENDegerinden küçük ya da ona eşit.

**Kodlama seçenekleri:** Kaynak ve hedef dizgilerinin tamsayı kodlamasını belirtmek için aşağıdaki seçenekler kullanılabilir. İlgili kodlama yalnızca, ilgili karakter takımı tanıtıcısı, ana saklama alanında belirlenen karakter kümesinin ikili tamsayılar için kullanılan kodlamaya bağımlı olduğunu gösterdiğinde kullanılır. Bu, yalnızca bazı çok baytlı karakter kümelerini (örneğin, UTF-16 karakter kümeleri gibi) etkiler.

Karakter kümesi tek baytlık karakter takımı (SBCS) ise ya da ana saklama alanında, tamsayı kodlamasına bağımlı olmayan bir çok baytlı karakter kümesi ise, kodlama yoksayıdır.

DCCS\* değerlerinden yalnızca biri belirtilmelidir; DCCT\* değerlerinden biriyle birleştirilebilir:

#### **DCCSNA**

Kaynak kodlama, ortam ve programlama dili için varsayılan değerdir.

#### **DCCSNO**

Kaynak kodlama olağandır.

#### **DCCSID ' LER**

Kaynak kodlama tersine çevrildi.

#### **DCCSID**

Kaynak kodlama tanımsız.

#### **DCCTNA**

Hedef kodlama, ortam ve programlama dili için varsayılan değerdir.

#### **DCCTNO**

Hedef kodlama normal.

#### **DCCTRE**

Hedef kodlama tersine çevrildi.

#### **DCCTUN**

Hedef kodlama tanımsız.

Daha önce tanımlanan kodlama değerleri doğrudan OPTS alanına eklenmelidir. Ancak, kaynak ya da hedef kodlama, MQMD ' de ya da başka bir yapıdaki MDENC alanından alınırsa, aşağıdaki işlem gerçekleştirilmelidir:

1. The integer encoding must be extracted from the MDENC field by eliminating the float and packed-decimal encodings; see [“IBM üzerindeki kodlamaların çözümleniyor” sayfa 1409](#) for details of how to do this.
2. 1. adımdan kaynaklanan tamsayı kodlaması, OPTS alanına eklenmeden önce uygun etkenle çarpılmalıdır. Bu etkenler şunlardır:

**DCCFA**

Kaynak kodlama katsayısı

**DCCTFA**

Hedef kodlama katsayısı

Belirtilmezse, kodlama seçenekleri varsayılan olarak tanımsız (DCC\* UN) değerine ayarlanacak şekilde ayarlanabilir. Çoğu durumda bu, MQXCNVC çağrısının başarıyla tamamlanmadığını etkilemez. Ancak, ilgili karakter kümesi, kodlamaya (örneğin, bir UTF-16 karakter takımı gibi) bağımlı bir çok baytlı bir karakter kümesine, çağrı neden kodu RC2112 ya da RC2116 uygun olduğu şekilde başarısız olur.

**Varsayılan seçenek:** Önceden açıklanan seçeneklerden hiçbiri belirtilmediyse, aşağıdaki seçenek kullanılabilir:

**DCCON**

Seçenek belirtilmedi.

DCCON, program belgelerine yardımcı olmak için tanımlıdır. Bu seçeneğin başka biriyle birlikte kullanılması amaçlanmaz, ancak değeri sıfır olduğu için bu tür bir kullanım saptanamaz.

**SRCS CSI (10 basamaklı işaretli tamsayı)-giriş**

Dönüştürmeden önce, dizginin kodlanmış karakter takımı tanıtıcısı.

Bu, SRCBUF içindeki giriş dizgisinin kodlanmış karakter takımı tanıtıcısıdır.

**SRCSLEN (imzalanmış 10 basamaklı tamsayı)-giriş**

Dönüştürmeden önce dizginin uzunluğu.

Bu, SRCBUF içindeki giriş dizgisinin bayt cinsinden uzunluğudur; Sıfır ya da daha büyük olmalıdır.

**SRCBUF (1 baytlık karakter dizilimi x SRCLEN)-giriş**

Dönüştürülecek dizgi.

Bu, bir karakter kümesinden diğerine dönüştürülecek dizeyi içeren arabelleğidir.

**TGTCSI (10 basamaklı işaretli tamsayı)-giriş**

Dönüştürmeden sonra dizginin kodlanmış karakter takımı tanıtıcısı.

Bu, SRCBUF ' un dönüştürüleceği karakter kümesinin kodlanmış karakter takımı tanıtıcısıdır.

**TGTLEN (10 basamaklı işaretli tamsayı)-giriş**

Çıkış arabelleğindeki uzunluk.

This is the length in bytes of the output buffer TGTBUF ; it must be zero or greater. SRCLENDegerinden küçük ya da daha büyük olabilir.

**TGTBUF (1 baytlık karakter dizilimi x TGTLEN)-çıkış**

Dönüştürmeden sonraki dizgi.

Bu, TGTCSI tarafından tanımlanan karakter kümesine dönüştürüldükten sonra dizgidir. Dönüştürülen dize, dönüştürülenmemiş dizgiden daha kısa ya da daha uzun olabilir. **DATLEN** parametresi, döndürülen geçerli baytların sayısını gösterir.

**DATLEN (10 basamaklı işaretli tamsayı)-çıkış**

Çıkış dizgisinin uzunluğu.

This is the length of the string returned in the output buffer TGTBUF. Dönüştürülen dize, dönüştürülenmemiş dizgiden daha kısa ya da daha uzun olabilir.

## **CMPCOD (10 basamaklı işaretli tamsayı)-çıkış**

Tamamlanma kodu.

Bu, aşağıdakilerden biridir:

### **CCOK**

İşlem başarıyla tamamlandı.

### **CCWARN**

Uyarı (kısmi tamamlama).

### **CCCFAIL**

Arama başarısız oldu.

## **REASON (10 basamaklı işaretli tamsayı)-çıkış**

Neden kodu ön eleme CMPCOD.

CMPCOD CCOK ise:

### **YOK**

(0, X'000 ') Raporlamak için bir neden yok.

CMPCOD CCWARN ise:

### **RC2120**

(2120, X'848 ') Dönüştürülen veriler arabellek için çok büyük.

CMPCOD CCFAIL ise:

### **RC2010**

(2010, X'7DA') Veri uzunluğu parametresi geçerli değil.

### **RC2150**

(2150, X'866 ') DBCS dizgisi geçerli değil.

### **RC2018**

(2018, X'7E2') Bağlantı tanıtıcısı geçerli değil.

### **RC2046**

(2046, X'7FE') Seçenekler geçerli değil ya da tutarlı değil.

### **RC2102**

(2102, X'836 ') Yetersiz sistem kaynakları var.

### **RC2145**

(2145, X'861 ') Kaynak arabellek parametresi geçerli değil.

### **RC2111**

(2111, X'83F') Kaynak kodlanmış karakter takımı tanıtıcısı geçerli değil.

### **RC2112**

(2112, X'840 ') Kaynak tamsayı kodlaması tanınmadı.

### **RC2143**

(2143, X'85F') Kaynak uzunluğu parametresi geçerli değil.

### **RC2071**

(2071, X'817 ') Kullanılabilir bellek yetersiz.

### **RC2146**

(2146, X'862 ') Hedef arabellek parametresi geçerli değil.

### **RC2115**

(2115, X'843 ') Hedef kodlanmış karakter takımı tanıtıcısı geçerli değil.

### **RC2116**

(2116, X'844 ') Hedef tamsayı kodlaması tanınmadı.

### **RC2144**

(2144, X'860 ') Hedef uzunluk parametresi geçerli değil.



## RC2195

(2195, X'893 ') Beklenmeyen bir hata oluştu.

Bu neden kodlarıyla ilgili daha fazla bilgi için bkz. "[IBM i için dönüş kodları \(ILE RPG\)](#)" sayfa 1403.

## RPG çağırısı (ILE)

```
C*..1.....2.....3.....4.....5.....6.....7..
C          CALLP      MQXCNV (HCONN : OPTS : SRCCSI :
C                               SRCLN : SRCBUF : TGTCSI :
C                               TGTLEN : TGTBUF : DATLEN :
C                               CMPCOD : REASON)
```

Çağrıya ilişkin prototip tanımı:

```
D*..1.....2.....3.....4.....5.....6.....7..
DMQXCNV      PR          EXTPROC('MQXCNV')
D* Connection handle
D HCONN          10I 0 VALUE
D* Options that control the action of MQXCNV
D OPTS          10I 0 VALUE
D* Coded character set identifier of string before conversion
D SRCCSI        10I 0 VALUE
D* Length of string before conversion
D SRCLN         10I 0 VALUE
D* String to be converted
D SRCBUF        * VALUE
D* Coded character set identifier of string after conversion
D TGTCSI        10I 0 VALUE
D* Length of output buffer
D TGTLEN        10I 0 VALUE
D* String after conversion
D TGTBUF        * VALUE
D* Length of output string
D DATLEN        10I 0
D* Completion code
D CMPCOD        10I 0
D* Reason code qualifying CMPCOD
D REASON        10I 0
```

## IBM i

## IBM üzerinde MQCONVX (Veri dönüştürme çıkışı)

Bu çağrı tanımı, veri dönüştürme çıkışa geçirilen parametreleri açıklar.

Kuyruk yöneticisi tarafından sağlanan MQCONVX adlı bir giriş noktası yok (bkz. kullanım notu "[11](#)" sayfa [1431](#)).

This definition is part of the IBM MQ Data Conversion Interface (DCI), which is one of the IBM MQ framework interfaces.

- "[Sözdizimi](#)" sayfa [1429](#)
- "[Kullanım notları](#)" sayfa [1429](#)
- "[Parametreler](#)" sayfa [1431](#)
- "[RPG çağırısı \(ILE\)](#)" sayfa [1432](#)

## Sözdizimi

**MQCONVX (MQDXP, MQMD, INLEN, INBUF, OUTLEN, OUTBUF)**

## Kullanım notları

1. Veri dönüştürme çıkışı, bir MQGET çağırısının işlenmesi sırasında denetimi alan, kullanıcı tarafından yazılmış bir çıkışıdır. Veri dönüştürme çıkışı tarafından gerçekleştirilen işlem, çıkışa ilişkin sağlayıcı tarafından tanımlanır; ancak çıkış, burada açıklanan kurallara ve MQDXP ' nin ilişkili değiştirgeyle uyumlu olmalıdır.

Veri dönüştürme çıkışı için kullanılacak programlama dilleri ortam tarafından belirlenir.

2. Çıkış yalnızca, aşağıdaki deyimlerin *tümü* true olduğunda çağrılır:

- MQGET çağrısında GMCONV seçeneği belirtildi.
- İleti tanımlayıcısındaki *MDFMT* alanı FMNONE değil
- İleti gerekli gösterimde değil; yani, iletinin *MDCSI* ve *MDENC* bir ya da her ikisi, MQGET çağrısında belirtilen ileti tanımlayıcısında uygulama tarafından belirtilen değerden farklı.
- Kuyruk yöneticisi dönüştürmeyi başarıyla yapmamış
- Uygulamanın arabelleğindeki uzunluk sıfırdan büyük.
- İleti verilerinin uzunluğu sıfırdan büyük.
- MQGET işlemi sırasında şu ana kadar neden kodu RCNONE ya da RC2079' tır

3. Bir çıkış yazılmakta olduğunda, çıkışı kesilen iletilerin dönüştürülmesine izin verecek şekilde kodlamak için bir çıkış dikkate alınmalıdır. Kısaltılmış iletiler aşağıdaki şekillerde ortaya çıkabilir:

- Alma uygulaması, iletiden küçük bir arabellek sağlar, ancak MQGET çağrısında GMATM seçeneğini belirtir.

Bu durumda, çıkışa girilen **MQDXP** parametresindeki *DXREA* alanı, RC2079 değerine sahip olur.

- İletiyi gönderen, göndermeden önce kesmiş olur. Bu, rapor iletileriyle olabilir, örneğin (daha ayrıntılı bilgi için bkz. ["Conversion of report messages on IBM i"](#) sayfa 1418 ).

Bu durumda, çıkışa girişteki **MQDXP** parametresindeki *DXREA* alanı, RNONE değerine sahip olur (alan uygulama, ileti için yeterince büyük bir arabellek sağlandıysa).

Dolayısıyla, çıkışa giriş üzerindeki *DXREA* alanının değeri, iletinin kesilip kesilmediğine karar vermek için her zaman kullanılamaz.

Kesilen bir iletinin ayırt edici özelliği, **INLEN** parametresindeki çıkışa sağlanan uzunluğun, ileti tanımlayıcısındaki *MDFMT* alanında bulunan biçim adının belirttiği uzunluk uzunluğunun *küçüktür* olduğu anlamına gelir. The exit should therefore check the value of *INLEN* before attempting to convert any of the data; the exit *bunu yapmak için* assume that the full amount of data implied by the format name has been provided.

If the exit has not been written to convert truncated messages, and **INLEN** is less than the value expected, the exit should return XRFAIL in the *DXRES* field of the **MQDXP** parameter, with the *DXCC* field set to CCWARN and the *DXREA* field set to RC2110.

If the exit *has* been written to convert truncated messages, the exit should convert as much of the data as possible (see next usage note), taking care not to attempt to examine or convert data beyond the end of *INBUF*. Dönüştürme başarıyla tamamlanırsa, çıkış **MQDXP** parametresindeki *DXREA* alanını değiştirmeden bırakmalıdır. İleti, alıcının kuyruk yöneticisi tarafından kesildiyse, RC2079 dizgisini döndürür; ileti, iletiyi gönderen tarafından kesilmişse, RCNONE değerini döndürür.

It is also possible for a message to expand *sırasında* conversion, to the point where it is bigger than *OUTBUF*. In this case the exit must decide whether to truncate the message; the *DXAOP* field in the **MQDXP** parameter will indicate whether the receiving application specified the GMATM option.

4. Generally it is recommended that all of the data in the message provided to the exit in *INBUF* is converted, or that none of it is. Ancak, bu durumda ileti kesilirse, dönüştürmeden önce ya da dönüştürme sırasında bir kural dışı durum oluşur; bu durumda, arabelleğin sonunda eksik bir öge olabilir (örneğin: çift byte karakterinin bir baytı ya da 4 baytlık bir tamsayıya ilişkin 3 byte). Bu durumda, tamamlanmamış ögenin atlanması ve *OUTBUF* içindeki kullanılmayan byte 'ları boş değere ayarlamaması önerilir. However, complete elements or characters within an array or string *gerekir* be converted.
5. İlk kez bir çıkış gerektiğinde, kuyruk yöneticisi, biçimiyle aynı adı taşıyan bir nesneyi (uzantılar dışında) yüklemeyi dener. Yüklenen nesne, o biçim adını taşıyan iletileri işleyen çıkışı içermelidir. Çıkış adının ve çıkışı içeren nesnenin adının aynı olması önerilir; ancak, tüm ortamlar bunu gerektirmese de, bu adın aynı olması gerekir.

6. Bir uygulama kuyruk yöneticisine bağlı uygulamadan bu yana *MDFMT* ' u kullanan ilk iletiyi almayı denediğinde çıkışa ilişkin yeni bir kopya yüklenir. Kuyruk yöneticisi önceden yüklenmiş bir kopyayı atmışsa, başka zamanlarda yeni bir kopya da yüklenebilir. Bu nedenle, bir çıkıştan çıkışa ilişkin bilgilerin bir sonraki çağrıya iletilmesi için statik saklama alanı kullanma girişiminde bulunulmamalıdır; bu, iki çağırma arasında bir çıkışa boşaltılabilir.
7. Kuyruk yöneticisi tarafından desteklenen yerleşik biçimlerden biriyle aynı adı taşıyan bir kullanıcı tarafından sağlanan bir çıkış varsa, kullanıcı tarafından sağlanan çıkış yerleşik dönüştürme yordamlarından birini değiştirmez. Bu tür bir çıkışa çağrılan tek koşul şunlardır:
  - Yerleşik dönüştürme yordamı *MDCSI* ya da *MDENC* ilişkisinden ya dainvolved' e dönüştürmeleri işleyemiyorsa ya da
  - Yerleşik dönüştürme yordamı verileri dönüştüremediye (örneğin, dönüştürülemeyen bir alan ya da karakter olduğu için).
8. Çıkışa ilişkin kapsam ortamdan bağımsız olur. *MDFMT* names should be chosen so as to minimize the risk of clashes with other formats. Biçim adını tanımlayan uygulamayı tanıtan karakterlerle başlamaları önerilir.
9. Veri dönüştürme çıkışı, MQGET çağrısını yayınlayan programdaki gibi bir ortamda çalışır; ortam, adres alanını ve kullanıcı tanımını içerir (geçerli olduğu yerlerde). Bu program, ileti kanalı aracısızın ileti dönüştürmesini desteklemeyen bir hedef kuyruk yöneticisine ileti göndermesine yol açabilirdi. Çıkış, kuyruk yöneticisinin ortamında çalıştırılmadığından, kuyruk yöneticisinin bütünlüğünden ödün veremiyor.
10. Çıkış tarafından kullanılacak tek MQI çağrısı MQXCNCV; diğer MQI çağrılarını kullanma girişiminde bulunulması, neden kodu RC2219ya da öngörülemez diğer hatalar ile başarısız olur.
11. Kuyruk yöneticisi tarafından sağlanan bir MQCONVX giriş noktası sağlanmadı. Çıkışa ilişkin ad, tüm ortamlarda gerekmede de, bu adın biçim adıyla (MQMD ' de *MDFMT* alanında bulunan ad) aynı olması gerekir.

## Parametreler

MQCONVX çağrısında şu deęiřtirgeler var:

### MQDXP (MQDXP)-giriř/çıkıř

Veri dönüřtürme çıkıř parametresi öbeęi.

Bu yapı, çıkıřa iliřkin çağrıyla ilgili bilgileri içerir. Çıkıř, dönüřtürmenin sonucunu belirtmek için bu yapıdaki bilgileri ayarlar. Bu yapıdaki alanların ayrıntıları için bkz. "[MQDXP \(Data-conversion exit parameter\) on IBM i](#)" sayfa 1419 .

### MQMD (MQMD)-giriř/çıkıř

İleti tanımlayıcısı.

Çıkıřa giriř sırasında, dönüřtürme gerçekteřtirilmezse, uygulamaya döndürüleceęini bildiren ileti tanımlayıcısı budur. Bu nedenle, *INBUF* içinde yer alan dönüřtürülemez iletinin *MDFMT*, *MDENC* ve *MDCSI* deęerini içerir.

**Not:** Çıkıřa aktarılan **MQMD** parametresi her zaman, çıkıřı çağırın kuyruk yöneticisi tarafından desteklenen en son MQMD sürümüdür. If the exit is intended to be portable between different environments, the exit should check the *MDVER* field in *MQMD* to verify that the fields that the exit needs to access are present in the structure.

IBM üzerinde, çıkıř bir version-2 MQMD geçirilir.

Çıkıřta, çıkıř *MDENC* ve *MDCSI* alanlarını uygulama tarafından istenen deęerlere deęiřtirmeli, dönüřtürme başarılı olursa, bu deęiřiklikler uygulamaya geri yansıtılır. Çıkıřta yapılan dięer deęiřiklikler yoksayılır; bu deęiřiklikler uygulamaya geri yansıtılmaz.

Çıkıř, MQDXP yapısının *DXRES* alanında *XROK* deęerini döndürürse, ancak ileti tanımlayıcısındaki *MDENC* ya da *MDCSI* alanlarını deęiřtirmezse, kuyruk yöneticisi bu alanlar için, MQDXP yapısındaki ilgili alanların çıkıřa giriřte sahip olduęu deęerleri döndürür.

## INLEN (10 basamaklı işaretli tamsayı)-giriş

Length in bytes of *INBUF*.

This is the length of the input buffer *INBUF*, and specifies the number of bytes to be processed by the exit. *INLEN*, dönüştürme işleminden önce ileti verilerinin uzunluğunun küçüktür ve MQGET çağrısında uygulama tarafından sağlanan arabelleğin uzunluğudur.

Değer her zaman sıfırdan büyük olur.

## INBUF (1-byte bit dizgisi x INLEN)-giriş

Dönüştürülemez iletiyi içeren arabellek.

Bu, dönüştürmeden önce ileti verilerini içerir. Çıkış, verileri dönüştüremiyorsa, kuyruk yöneticisi, çıkış tamamlandıktan sonra bu arabelleğin içeriğini uygulamaya geri döndürür.

**Not:** Çıkış *INBUF*; değiştirmemelidir; bu parametre değiştirilirse, sonuçlar tanımsız olur.

## OUTLEN (10 basamaklı işaretli tamsayı)-giriş

Length in bytes of *OUTBUF*.

This is the length of the output buffer *OUTBUF*, and is the same as the length of the buffer provided by the application on the MQGET call.

Değer her zaman sıfırdan büyük olur.

## OUTBUF (1 byte 'lık bit dizgisi x OUTLEN)-çıkış

Dönüştürülen iletiyi içeren arabellek.

Çıkışta çıkışta, dönüştürme başarılı olursa (**MQDXP** parametresinin *DXRES* alanındaki *XROK* değeri ile belirtildiği gibi), **OUTBUF** istenen gösterimde uygulamaya teslim edilecek ileti verilerini içerir. Dönüştürme başarısız olduysa, çıkışta bu arabelleğe ilişkin yapılan değişiklikler yoksayılr.

## RPG çağrısı (ILE)

```
C*..1.....2.....3.....4.....5.....6.....7..
C          CALLP      exitname(MQDXP : MQMD : INLEN :
C                               INBUF : OUTLEN : OUTBUF)
```

Çağrıya ilişkin prototip tanımı:

```
D*..1.....2.....3.....4.....5.....6.....7..
Dexitname      PR          EXTPROC('exitname')
D* Data-conversion exit parameter block
D MQDXP                44A
D* Message descriptor
D MQMD                  364A
D* Length in bytes of INBUF
D INLEN                10I 0 VALUE
D* Buffer containing the unconverted message
D INBUF                * VALUE
D* Length in bytes of OUTBUF
D OUTLEN               10I 0 VALUE
D* Buffer containing the converted message
D OUTBUF               * VALUE
```

Ürüne duyarlı programlama arabiriminin sonu

## Kullanıcı çıkışları, API çıkışları ve kurulabilir hizmetler başvurusu

Kullanıcı çıkışlarınızı, API çıkışlarınızı ve kurulabilir hizmet uygulamalarınızı geliştirmenize yardımcı olması için bu bölümdeki bilgileri kullanın:

- “MQIEP yapısı” sayfa 1433

- [“Veri dönüştürme çıkış başvurusu” sayfa 1436](#)
- [“MQ\\_PUBLISH\\_EXIT-Yayınlama çıkışı” sayfa 1440](#)
- [“Kanal-çıkış çağrıları ve veri yapıları” sayfa 1448](#)
- [“API çıkış başvurusu” sayfa 1535](#)
- [“Kurulabilir hizmetler arabirimi başvuru bilgileri” sayfa 1595](#)

### **İlgili kavramlar**

[Kullanıcı çıkışları, API çıkışları ve IBM MQ kurulabilir hizmetleri](#)

### **İlgili görevler**

[Kuyruk yöneticisi olanaklarının genişletmesi](#)

## **MQIEP yapısı**

MQIEP yapısı, çıkışa izin verilen her işlev çağrısı için bir giriş noktası içerir.

### **Alanlar**

#### **StrucId**

Tip: MQCHAR4 -giriş

Yapı tanıtıcısı. Değer aşağıdaki gibidir:

**MQIEP\_STRUC\_ID**

#### **S\u00fcr\u00fcm**

Tip: MQHOME-girişi

Yapı sürüm numarası. Değer aşağıdaki gibidir:

**MQIEP\_VERSION\_1**

Sürüm 1 yapısı sürüm numarası.

**MQIEP\_XX\_ENCODE\_CASE\_ONE current\_version**

Yapının geçerli sürümü.

#### **StrucLength**

Tip: MQUZE

MQIEP yapısının bayt cinsinden büyüklüğü. Değer aşağıdaki gibidir:

**MQIEP\_LENGTH\_1**

### **İşaretler**

Tip: MQUZE

İşlev adreslerine ilişkin bilgi sağlar. Kitaplığın, kitaplığın bir istemci ya da sunucu kitaplığı olup olmadığını göstermek için bir işaretle kullanılıp kullanılamayabileceğini belirten işaret.

Kitaplık bilgisi belirtmek için aşağıdaki değer kullanılır:

**MQIEPF\_NONE**

Paylaşılan kitaplığın iş parçacıklı ya da iş parçacıklı olup olmadığını belirtmek için aşağıdaki değerlerden biri kullanılır:

**MQIEPF\_NON\_THREADED\_LIBRARY**

Yivli olmayan bir paylaşılan kitaplık

**MQIEPF\_THREADED\_KITAPLIK**

Yivli paylaşılan kitaplık

Paylaşılan kitaplığın istemci ya da sunucu paylaşılan kitaplığı olup olmadığını belirtmek için aşağıdaki değerlerden biri kullanılır:

**MQIEPF\_CLIENT\_KITAP**

Bir istemci paylaşılan kitaplığı

## **MQIEPF\_LOCAL\_KİTAPLI**

Bir sunucu paylaşılan kitaplığı

### **Ayrıldı**

Tip: MQPTR

### **MQBACK\_Çağrısı**

Tip: PMQ\_BACK\_CALL

MQBACK çağrısının adresi.

### **MQBEGIN\_Call**

Tip: PMQ\_BEGIN\_CALL

MQBEGIN çağrısının adresi.

### **MQBUFMH\_Call**

Tip: PMQ\_BUFMH\_CALL

MQBUFMH çağrısının adresi.

### **MQCB\_Çağrısı**

Tip: PMQ\_CB\_CALL

MQCB çağrısının adresi.

### **MQCLOSE\_Call**

Tip: PMQ\_CLOSE\_CALL

MQCLOSE çağrısının adresi.

### **MQCMIT\_Çağrısı**

Tip: PMQ\_CMIT\_CALL

MQCMIT çağrısının adresi.

### **MQCONN\_Çağrısı**

Tip: PMQ\_CONN\_CALL

MQCONN çağrısının adresi.

### **MQCONNX\_Ç**

Tip: PMQ\_CONNX\_CALL

MQCONNX çağrısının adresi.

### **MQCRTMH\_Çağrısı**

Tip: PMQ\_CRTMH\_CALL

MQCRTMH çağrısının adresi.

### **MQCTL\_Call**

Tip: PMQ\_CTL\_CALL

MQCTL çağrısının adresi.

### **MQDISC\_Call**

Tip: PMQ\_DISC\_CALL

MQDISC çağrısının adresi.

### **MQDLTMH\_Call**

Tip: PMQ\_DLTMH\_CALL

MQDLTMH çağrısının adresi.

### **MQDLTMP\_Call**

Tip: PMQ\_DLTMP\_CALL

MQDLTMP çağrısının adresi.

**MQGET\_Çağrısı**

Tip: PMQ\_GET\_CALL

MQGET çağrısının adresi.

**MQINQ\_Call**

Tip: PMQ\_INQ\_CALL

MQINQ çağrısının adresi.

**MQINQMP\_Call**

Tip: PMQ\_INQMP\_CALL

MQINQMP çağrısının adresi.

**MQMHBUF\_Call**

Tip: PMQ\_MHBUF\_CALL

MQMHBUF çağrısının adresi.

**MQOPEN\_Call**

Tip: PMQ\_OPEN\_CALL

MQOPER çağrısının adresi.

**MQPUT\_Call**

Tip: PMQ\_PUT\_CALL

MQPUT çağrısının adresi.

**MQPUT1\_Call**

Tip: PMQ\_PUT1\_CALL

MQPUT1 çağrısının adresi.

**MQSET\_Call**

Tip: PMQ\_SET\_CALL

MQSET çağrısının adresi.

**MQSETMP\_Call**

Tip: PMQ\_SETMP\_CALL

MQSETMP çağrısının adresi.

**MQSTAT\_Call**

Tip: PMQ\_STAT\_CALL

MQSTAT çağrısının adresi.

**MQSUB\_CALL**

Tip: PMQ\_SUB\_CALL

MQSUB çağrısının adresi.

**MQSUBRQ\_Call**

Tip: PMQ\_SUBRQ\_CALL

MQSUBRQ çağrısının adresi.

**MQXCNVC\_Call**

Tip: PMQ\_XCNVC\_CALL

MQXCNVC çağrısının adresi.

**MQXCLWLN\_Call**

Tip: PMQ\_XCLWLN\_CALL

MQXCLWLN çağrısının adresi.

**MQXDX\_Call**

Tip: PMQ\_XDX\_CALL

MQXDX çağrısının adresi.

### **MQXEP\_Call**

Tip: PMQ\_XEP\_CALL

MQXEP çağrısının adresi.

### **MQZEP\_Call**

Tip: PMQ\_ZEP\_CALL

MQZEP çağrısının adresi.

## **C Bildirimi**

```
struct tagMQIEP {
MQCHAR4      StrucId;          /* Structure identifier */
MQLONG       Version;         /* Structure version number */
MQLONG       StrucLength;     /* Structure length */
MQLONG       Flags;          /* Flags */
MQPTR        Reserved;       /* Reserved */
PMQ_BACK_CALL MQBACK_Call;   /* Address of MQBACK */
PMQ_BEGIN_CALL MQBEGIN_Call; /* Address of MQBEGIN */
PMQ_BUFMH_CALL MQBUFMH_Call; /* Address of MQBUFMH */
PMQ_CB_CALL   MQCB_Call;     /* Address of MQCB */
PMQ_CLOSE_CALL MQCLOSE_Call; /* Address of MQCLOSE */
PMQ_CMIT_CALL MQCMIT_Call;   /* Address of MQCMIT */
PMQ_CONN_CALL MQCONN_Call;   /* Address of MQCONN */
PMQ_CONNX_CALL MQCONNX_Call; /* Address of MQCONNX */
PMQ_CRTMH_CALL MQCRTMH_Call; /* Address of MQCRTMH */
PMQ_CTL_CALL  MQCTL_Call;    /* Address of MQCTL */
PMQ_DISC_CALL MQDISC_Call;   /* Address of MQDISC */
PMQ_DLTMH_CALL MQDLTMH_Call; /* Address of MQDLTMH */
PMQ_DLTMP_CALL MQDLTMP_Call; /* Address of MQDLTMP */
PMQ_GET_CALL  MQGET_Call;    /* Address of MQGET */
PMQ_INQ_CALL  MQINQ_Call;    /* Address of MQINQ */
PMQ_INQMP_CALL MQINQMP_Call; /* Address of MQINQMP */
PMQ_MHBUF_CALL MQMHBUF_Call; /* Address of MQMHBUF */
PMQ_OPEN_CALL MQOPEN_Call;   /* Address of MQOPEN */
PMQ_PUT_CALL  MQPUT_Call;    /* Address of MQPUT */
PMQ_PUT1_CALL MQPUT1_Call;   /* Address of MQPUT1 */
PMQ_SET_CALL  MQSET_Call;    /* Address of MQSET */
PMQ_SETMP_CALL MQSETMP_Call; /* Address of MQSETMP */
PMQ_STAT_CALL MQSTAT_Call;   /* Address of MQSTAT */
PMQ_SUB_CALL  MQSUB_Call;    /* Address of MQSUB */
PMQ_SUBBRQ_CALL MQSUBBRQ_Call; /* Address of MQSUBBRQ */
PMQ_XCLWLN_CALL MQXCLWLN_Call; /* Address of MQXCLWLN */
PMQ_XCNVC_CALL MQXCNCV_Call; /* Address of MQXCNCV */
PMQ_XDX_CALL  MQXDX_Call;    /* Address of MQXDX */
PMQ_XEP_CALL  MQXEP_Call;    /* Address of MQXEP */
PMQ_ZEP_CALL  MQZEP_Call;    /* Address of MQZEP */
};
```



## **Veri dönüştürme çıkış başvurusu**

z/OS için, çevirici diline veri dönüştürme çıkışları yazmanız gerekir. Diğer platformlarda, C programlama dilini kullanmanız önerilir.

Bir veri dönüştürme çıkış programı yaratmanıza yardımcı olması için aşağıdaki kaynaklar sağlanır:

- İskelet kaynak dosyası
- Karakter dönüştürme çağrısı
- Veri tipi yapılarında veri dönüştürme işlemini gerçekleştiren bir kod parçası yaratan bir yardımcı program. Bu yardımcı program yalnızca C girişini alır. z/OS üzerinde, çevirici kodu üretir.

Programların yazılmasına ilişkin yordam için bkz:

-  [IBM MQ for IBM için veri dönüştürme çıkış programı yazılıyor](#)
-  [IBM MQ for z/OS için veri dönüştürme çıkış programı yazılıyor](#)








- [IBM MQ for AIX or Linux sistemleri için veri dönüştürme çıkışı yazılması](#)
- [IBM MQ for Windows için veri dönüştürme çıkışı yazılması](#)

## Çatı kaynak dosyası

Bunlar, bir veri dönüştürme çıkış programı yazarken başlangıç noktanız olarak kullanılabilir.

Sağlanan dosyalar [Çizelge 817 sayfa 1437](#) içinde listelenir.

Çizelge 817. Çatı kaynak dosyaları	
Altyapı	Dosya
 AIX	amqsvfc0.c
 IBM i	QMOMSAMP/QCSRC (AMQSVFC4)
 Linux	amqsvfc0.c
 Windows sistemleri	amqsvfc0.c
 z/OS	CSQ4BAX8 ( "1" sayfa 1437 ) CSQ4BAX9 ( "2" sayfa 1437 ) CSQ4CAX9 ( "3" sayfa 1437 )
<p><b>Notlar:</b></p> <ol style="list-style-type: none"> <li>1. MQXCVNC çağrısını gösterir.</li> <li>2. Yardımcı program tarafından CICS dışındaki tüm ortamlarda kullanılmak üzere oluşturulan kod parçalarına ilişkin bir sarıcı.</li> <li>3. Yardımcı program tarafından CICS ortamında kullanılmak üzere oluşturulan kod parçaları için bir sarıcı.</li> </ol>	

## Karakter çağrısını dönüştür

Karakter iletisi verilerini bir karakter kümesinden diğerine dönüştürmek için, bir veri dönüştürme çıkış programı içindeki MQXCNVC (karakterleri dönüştürme) çağrısını kullanın. Bazı çok baytlı karakter kümeleri (örneğin, UTF-16 karakter kümeleri) için, uygun seçeneklerin kullanılması gerekir.




Çıkış içinden başka bir MQI çağrısı yapılamayabilir; bu tür bir çağrıyı yapma girişimi, MQRC\_CALL\_IN\_PROGRESS neden kodu ile başarısız olur.

MQXCNVC çağrısına ve uygun seçeneklere ilişkin ek bilgi için ["MQXCNVC-Karakterleri dönüştür" sayfa 903](#) başlıklı konuya bakın.

## Dönüştürme-çıkış kodu yaratmak için kullanılan yardımcı program

Dönüştürme çıkış kodu oluşturma hakkında daha fazla bilgi edinmek için bu bilgileri kullanın.

Dönüştürme-çıkış kodu yaratılmasına ilişkin komutlar şunlardır:

 IBM i	CVTMQMDTA ( IBM MQ Veri Tipini Dönüştür)
 ALW	AIX, Linux, and Windows sistemleri crtmqcvx (Create IBM MQ conversion-exit)
 z/OS	CSQUCVX

Altyapınıza ilişkin komut, veri dönüştürme çıkış programınızda kullanılmak üzere, veri tipi yapılarında veri dönüştürme işlemini gerçekleştiren bir kod parçası üretir. Komut, bir ya da daha çok C dili yapısı tanımlaması içeren bir dosyayı alır. **z/OS** z/OS üzerinde, daha sonra, çevirici kod parçaları ve dönüştürme işlevlerini içeren bir veri kümesi oluşturur. Diğer platformlarda, her yapı tanımlamasını dönüştürmek için C işlevi içeren bir dosya oluşturur. z/OS üzerinde, yardımcı program LE/370 çalıştırma zamanı kitaplığı SCEERUN' e erişim gerektirir.

## z/OS üzerinde CSQUCVX yardımcı programı çağırılıyor

**z/OS**

Şekil 10 sayfa 1438 , CSQUCVX yardımcı programını çağırarak için kullanılan JCL ' ye ilişkin bir örnek gösterir.

```
//CVX EXEC PGM=CSQUCVX
//STEPLIB DD DISP=SHR,DSN=thlqua1.SCSQANLE
// DD DISP=SHR,DSN=thlqua1.SCSQLOAD
// DD DISP=SHR,DSN=le370qua1.SCEERUN
//SYSPRINT DD SYSOUT=*
//CSQUINP DD DISP=SHR,DSN=MY.MQSERIES.FORMATS(MSG1)
//CSQUOUT DD DISP=OLD,DSN=MY.MQSERIES.EXIT(SMSG1)
```

Şekil 10. CSQUCVX yardımcı programını çağırarak için kullanılan JCL örneği

## z/OS veri tanımlama deyimleri

**z/OS**

CSQUCVX yardımcı programı, Çizelge 818 sayfa 1438 içinde gösterilen şu DD adlarına sahip DD deyimleri gerektirir:

Çizelge 818. Veri tanımlama deyimleri adları ve açıklamaları	
DD bildirimi	Tanım
SYSPRINT	Raporlar ve hata iletileri için bir veri kümesi ya da yazdırma kuyruğu sınıfı belirler.
CSQUINP	Dönüştürülecek veri yapılarının tanımlamalarını içeren bölümlenmiş veri kümesini belirtir.
CSQUOUT	Dönüştürme kodu parçalarının yazılacağı bölümlenmiş veri kümesini belirtir. Mantıksal kayıt uzunluğu (LRECL) 80 olmalı ve kayıt biçimi (RECFM) FB olmalıdır.

## AIX, Linux, and Windows sistemlerindeki hata iletileri

The `crtmqcvx` command returns messages in the range AMQ7953 through AMQ7970.

Bu iletiler, *Messages and reason codes IBM MQ İletileri*' nde listelenir.

İki ana hata tipi vardır:

- İşleme devam edemediğinde sözdizimi hataları gibi önemli hatalar.

Giriş kütüğündeki hatanın satır numarasını veren ekranda bir ileti görüntülenir. Çıkış dosyası kısmen yaratılmış olabilir.

- Bir sorunun saptandığını, ancak yapının ayrıştırılması devam edebileceğinin belirtildiği bir ileti görüntülendiğinde diğer hatalar.

Çıkış dosyası yaratıldı ve ortaya çıkan sorunlarla ilgili hata bilgileri içeriyor. Bu hata bilgisinin başına `#error` öneki eklenir; böylece, üretilen kod, sorunları gidermek için araya girmeden derleyici tarafından kabul edilmez.

## Geçerli sözdizimi

Yardımcı programa ilişkin giriş dosyan C dili sözdizimine uygun olmalıdır.

C ile ilgili bilgi sahibi olmadıysanız, bu konudaki [C örneği](#) ' ne bakın.

Ayrıca, aşağıdaki kurallardan haberdar olun:

- typedef, struct anahtar sözcüğünden önce tanınır.
- Yapı bildirimlerinizde bir yapı etiketi gereklidir.
- Bir iletinin sonunda değişken uzunluklu bir dizi ya da dizgi belirtmek için boş köşeli ayraçları [] kullanabilirsiniz.
- Çok boyutlu diziler ve dizeler dizileri desteklenmiyor.
- Aşağıdaki ek veri tipleri tanınır:

- MQBOOL
- MQBYTE
- MQCHAR
- MQFLOAT32
- MQFLOAT64
- MQSHORT
- MQLONG
- MQINT8
- MQUINT8
- MQINT16
- MQUINT16
- MQINT32
- MQUINT32
- MQINT64
- MQUINT64

MQCHAR alanları kod sayfası dönüştürüldü, ancak MQBYTE, MQINT8 ve MQUINT8 dokunulmamış bırakılıyor. Kodlama farklıysa, MQSHORT, MQLong, MQINT16, MQUINT16, MQINT32, MQUINT32, MQINT64, MQUINT64, MQFLOAT32, MQFLOAT64 ve MQBOOL buna göre dönüştürülür.

- Aşağıdaki veri tiplerini kullanmayın:

- çift
- işaretçiler
- bit alanları

Bunun nedeni, dönüştürme-çıkış kodu yaratmak için kullanılan yardımcı programın bu veri tiplerini dönüştürmesi için gereken olanağı sağlamasıdır. Bunu aşmak için kendi rutinlerinizi yazabilir ve çıkışlardan arayabilirsiniz.

Notlara ilişkin diğer noktalar:

- Giriş verileri kümesinde sıra numaralarını kullanmayın.
- Kendi dönüştürme yordamlarınızı sağlamak istediğiniz alanlar varsa, bunları MQBYTE olarak bildirin ve daha sonra, oluşturulan CMQXCFBA makrolarını kendi dönüştürme kodunuzla değiştirin.

## C örneği

```
struct TEST { MQLONG    SERIAL_NUMBER;  
              MQCHAR    ID[5];  
              MQINT16   VERSION;
```

```
MQBYTE    CODE[4];
MQLONG    DIMENSIONS[3];
MQCHAR    NAME[24];
} ;
```

Bu, diğer programlama dillerindeki aşağıdaki bildirimlere karşılık gelir:

## COBOL

```
10 TEST.
  15 SERIAL-NUMBER PIC S9(9) BINARY.
  15 ID             PIC X(5).
  15 VERSION       PIC S9(4) BINARY.
  * CODE IS NOT TO BE CONVERTED
  15 CODE          PIC X(4).
  15 DIMENSIONS    PIC S9(9) BINARY OCCURS 3 TIMES.
  15 NAME          PIC X(24).
```

## System/390

```
TEST          EQU *
SERIAL_NUMBER DS F
ID            DS CL5
VERSION       DS H
CODE         DS XL4
DIMENSIONS    DS 3F
NAME         DS CL24
```

## PL/I

**Yalnızca z/OS üzerinde desteklenir**

```
DCL 1 TEST,
  2 SERIAL_NUMBER FIXED BIN(31),
  2 ID            CHAR(5),
  2 VERSION       FIXED BIN(15),
  2 CODE          CHAR(4), /* not to be converted */
  2 DIMENSIONS(3) FIXED BIN(31),
  2 NAME          CHAR(24);
```

## MQ\_PUBLISH\_EXIT-Yayınlama çıkışı

MQ\_PUBLISH\_EXIT çağırısı, abonelere teslim edilen iletileri inceleyebilir ve değiştirebilir.

### Amaç

Abonelere teslim edilen iletileri incelemek ve değiştirmek için yayınlama çıkışını kullanın:

- Her abonede yayınlanan bir iletinin içeriğini inceler
- Her abonede yayınlanan bir iletinin içeriğini değiştirir
- İletin konacağı kuyruğu değiştirir
- Bir iletiyi aboneye teslim etmeyi durdurur

Bu çıkış IBM MQ for z/OS üzerinde kullanılamaz.

### Sözdizimi

**MQ\_PUBLISH\_EXIT** (*ExitParms, PubContext, SubContext*)

## Parametreler

### **ExitParms (MQPSXP) - Input/Output**

*ExitParms* , çıkışa ilişkin çağrıya ilişkin bilgileri içerir.

### **PubContext (MQPBC) - Input**

*PubContext* , yayınının yayıncısına ilişkin bağlamsal bilgiler içerir.

### **SubContext (MQSBC) - Input/Output**

*SubContext* , yayını alan aboneye ilişkin bağlamsal bilgiler içerir.

## MQPSXP-Çıkış veri yapısını yayınla

MQPSXP yapısı, yayınlama çıkışından geçirilen ve yayınlama çıkışından döndürülen bilgileri açıklar.

Çizelge 819 sayfa 1441 , yapıdaki alanları özetler:

Çizelge 819. MQPSXP 'deki alanlar	
Alan	Tanım
<u>StrucID</u>	Yapı tanıtcısı
<u>Version</u>	Yapı sürüm numarası
<u>ExitId</u>	Çağrılmakta olan çıkış tipi
<u>ExitReason</u>	Çıkış çağrısının nedeni
<u>ExitResponse</u>	Çıkıştan yanıt
<u>ExitResponse2</u>	Çıkıştan ikincil yanıt
<u>Feedback</u>	Feedback kodu
<u>ExitUserArea</u>	Kullanıcı alanından çık
<u>ExitData</u>	Verilerden çık
<u>QMgrName</u>	Yerel kuyruk yöneticisinin adı
<u>Hconn</u>	Bağlantı tanıtcı
<u>MsgDescPtr</u>	İleti tanımlayıcısının adresi (MQMD)
<u>MsgHandle</u>	İleti özellikleriyle tanıtcı değeri (MQHMSG)
<u>MsgInPtr</u>	Giriş iletisinin adresi
<u>MsgInLength</u>	Giriş iletisinin uzunluğu
<u>MsgOutPtr</u>	Çıkış iletisinin adresi
<u>MsgOutLength</u>	Çıkış iletisinin uzunluğu
<u>pEntryPoints</u>	MQIEP yapısının adresi

## Alanlar

### **StrucID (MQCHAR4)**

*StrucID* yapı tanıtcısıdır. Değer aşağıdaki gibidir:

#### **MQPSXP\_STRUCID**

MQPSXP\_STRUCID , yayınlama çıkış parametresi yapısına ilişkin tanıtcıdır. For the C programming language, the constant MQPSXP\_STRUC\_ID\_ARRAY is also defined; it has the same value as MQPSXP\_STRUC\_ID, but is an array of characters instead of a string.

*StrucID* , çıkışa ilişkin bir giriş alanıdır.

### **Version (MQLONG)**

*Version* , yapı sürüm numarasıdır. Değer aşağıdaki gibidir:

#### **MQPSXP\_VERSION\_1**

MQPSXP\_VERSION\_1 , Sürüm 1 yayınlama çıkış parametre yapısıdır. The constant MQPSXP\_CURRENT\_VERSION is also defined with the same value.

*Version* , çıkışa ilişkin bir giriş alanıdır.

### **ExitId (MQLONG)**

*ExitId* , çağrılmakta olan çıkış tipidir. Değer aşağıdaki gibidir:

#### **MQXT\_PUBLISH\_EXIT**

Çıkış yayınlayın.

*ExitId* , çıkışa ilişkin bir giriş alanıdır.

### **ExitReason (MQLONG)**

*ExitReason* , çıkışa çağrılmasına ilişkin nedendir. Olası değerler şunlardır:

#### **MQXR\_INIT**

Bu bağlantıya ilişkin çıkış, kullanıma hazırlama için çağrılır. Çıkış, gereksinim duyduğu kaynakları edinebilir ve kullanıma hazırlayabilir; örneğin, ana depolama alanı.

#### **MQXR\_TERM**

Bu bağlantıya ilişkin çıkış, çıkışta durdurulacak olduğu için çağrılır. Çıkış, ilk kullanıma hazırlandığından bu yana edindiği kaynakları serbest (örneğin, ana saklama alanı) serbest olmalıdır.

#### **MQXR\_PUBLICATION**

Çıkış, bir yayın bir abonenin ileti kuyruğuna yayın yerleştirmeden önce kuyruk yöneticisi tarafından çağrılır. Çıkış iletiyi değiştirebilir, kuyruğa ileti yerleştiremez ya da yayını durduramaz.

*ExitReason* , çıkışa ilişkin bir giriş alanıdır.

### **ExitResponse (MQLONG)**

İşlemin nasıl devam etmesi gerektiğini belirtmek için çıkışta *ExitResponse* değerini ayarlayın.

*ExitResponse* aşağıdaki değerlerden biridir:

#### **MQXCC\_OK**

Set MQXCC\_OK to continue processing normally. Set MQXCC\_OK in response to any values of ExitReason.

ExitReason değeri MQXR\_PUBLICATION ise, MQSBC yapısının *DestinationQName* ve *DestinationQMgrName* alanları, iletinin gönderildiği hedefi tanımlar.

#### **MQXCC\_FAILED**

Yayınlama işlemini durdurmak için MQXCC\_FAILED ögesini ayarlayın. Tamamlanma kodu MQCC\_FAILED ve neden kodu 2557 (09FD) (RC2557): MQRC\_PSAPISH\_EXIT\_ERROR , çıkıştan geri dönüş işlemi için ayarlandı.

#### **MQXCC\_SUPPRESS\_FUNCTION**

MQXCC\_SUPPRESS\_FUNCTION ' un iletinin olağan işlenmesini durdurmak için ayarlayın. Only set MQXCC\_SUPPRESS\_FUNCTION if *ExitReason* has the value MQXR\_PUBLICATION.

İleti, iletinin ileti tanımlayıcısındaki *Report* alanında bulunan MQRO\_DISCARD\_MSG seçeneğine göre kuyruk yöneticisi tarafından işlenmeye devam eder.

- MQRO\_DISCARD\_MSG seçeneği belirtilirse, ileti aboneye teslim edilmez.
- MQRO\_DISCARD\_MSG seçeneği belirtilmediyse, ileti, ölü-mektup kuyruğuna yerleştirilir. Herhangi bir ölü-mektup kuyruğu yoksa ya da ileti, ölü-mektup kuyruğuna başarıyla yerleştirilemezse, yayın aboneye teslim edilmez. Yayının diğer abonelere teslim edilmesi, PMSGDLV ve NPMSGDLV konu nesnesi özniteliklerinin değerlerine bağlıdır. Bu özniteliklere ilişkin açıklamalar için, DEFINE TOPIC komutuna ilişkin parametre açıklamalarına bakın.

*ExitResponse* , çıkışa ait bir çıkış alanıdır.

### **ExitResponse2 (MQLONG)**

*ExitResponse2* , ileride kullanılmak üzere ayrılmıştır.

### **Feedback (MQLONG)**

*Feedback* is the feedback code to be used if the exit returns MQXCC\_SUPPRESS\_FUNCTION in *ExitResponse*.

On input to the exit, *Feedback* always has the value MQFB\_NONE. Çıkış MQXCC\_SUPPRESS\_FUNCTIONdeğerini döndürürse, kuyruk yöneticisi iletiyi ölüme mektup kuyruğuna yerleştirdiğinde, ileti için kullanılacak değeri *Feedback* olarak ayarlayın. Çıkıştan dönüşte, *Feedback* özgün değeri MQFB\_NONEise, kuyruk yöneticisi *Feedback* değerini MQFB\_STOPPED\_BY\_PUBSUB\_EXITolarak ayarlar.

*Feedback* , çıkışa giriş/çıkış alanıdır.

### **ExitUserArea (MQBYTE16)**

*ExitUserArea* , çıkışta kullanılacak bir alandır. Her bağlantının ayrı bir *ExitUserArea*değeri vardır. The length of *ExitUserArea* is given by MQ\_EXIT\_USER\_AREA\_LENGTH.

*ExitReason* alanı, çıkışa ilişkin ilk çağrıda MQXR\_INIT değerine sahiptir. *ExitUserArea* , bağlantıya ilişkin çıkışa ilişkin ilk çağrıda MQXUA\_NONE ile başlatılır. *ExitUserArea* ' da yapılan sonraki değişiklikler, çıkışa yönelik çağrılar boyunca korunur.

*ExitUserArea* , çıkışa giriş/çıkış alanıdır.

### **ExitData (MQCHAR32)**

*ExitData* , kuyruk yöneticisinin kullanıma hazırlama kütüğündeki stanza **PublishExitData** değiştirgesi tarafından tanımlanan değişmez çıkış verileri. Veriler, alanın tam uzunluğuna kadar boşluklarla dolduruludur. Kullanıma hazırlama kütüğünde değişmez çıkış verisi tanımlı değilse, *ExitData* boş olur. The length of *ExitData* is given by MQ\_EXIT\_DATA\_LENGTH.

*ExitData* , çıkışa ilişkin bir giriş alanıdır.

### **QMgrName (MQCHAR48)**

*QMgrName* , yerel kuyruk yöneticisinin adıdır. Ad, alanın tam uzunluğuna kadar boşluklarla dolduruludur. Bu alanın uzunluğu MQ\_Q\_MGR\_NAME\_LENGTHtarafından verilir.

*QMgrName* , çıkışa ilişkin bir giriş alanıdır.

### **Hconn (MQHCONN)**

*Hconn* , kuyruk yöneticisiyle bir bağlantıyı gösteren tanıtıcıdır. Only use *Hconn* as a parameter to the MQSETMP, MQINQMMP, or MQDLTMP message property function calls to work with message properties.

*Hconn* , çıkışa ilişkin bir giriş alanıdır.

### **MsgDescPtr (PMQMD)**

*MsgDescPtr* , işlenmekte olan iletinin ileti tanımlayıcısının ( MQMD) adresidir ve MQPUT çağrısından döndürülen MQMD ' nin bir kopyasıdır. Çıkış, ileti tanımlayıcısının içeriğini değiştirebilir. İleti tanımlayıcısının içindekilerle ilgili herhangi bir değişiklik, dikkatli bir şekilde yapılmalıdır. Özellikle, MQSBC yapısının *SubType* alanının MQSUBTYPE\_PROXYdeğerine sahip olduğu durumlarda, ileti tanımlayıcısındaki *CorrelId* alanının değiştirilmemesi gerekir.

No message descriptor is passed to the exit if *ExitReason* is MQXR\_INIT or MQXR\_TERM ; in these cases, *MsgDescPtr* is the null pointer.

*MsgDescPtr* , çıkışa ilişkin bir giriş alanıdır.

### **MsgHandle (MQHMSG)**

*MsgHandle* , ileti özelliklerinin tanıtıcısıdır. İleti özellikleriyle çalışmak için yalnızca MQSETMP, MQINQMMPya da MQDLTMP ileti özelliği işlev çağrılarını ile *MsgHandle* kullanın.

*MsgHandle* , çıkışa ilişkin bir giriş alanıdır.

### **MsgInPtr (PMQVOID)**

*MsgInPtr* , giriş iletisi verilerinin adresidir. The contents of the buffer addressed by *MsgInPtr* can be modified by the exit; see [MsgOutPtr](#) .

*MsgInPtr* , çıkışa ilişkin bir giriş alanıdır.

### **MsgInLength (MQLONG)**

*MsgInLength* , çıkışa aktarılan ileti verilerinin bayt cinsinden uzunluğudur. Verilerin adresi *MsgInPtr* tarafından verilir.

*MsgInLength* , çıkışa ilişkin bir giriş alanıdır.

### **MsgOutPtr (PMQVOID)**

*MsgOutPtr* , çıkıştan döndürülen ileti verilerini içeren bir arabelleğin adresidir. Çıkışa girişte *MsgOutPtr* boş değerli. Çıkıştan dönüşte, değer yine de boşsa, kuyruk yöneticisi *MsgInPtr* tarafından belirtilen iletiyi, *MsgInLength* ile belirtilen uzunluğa gönderir.

Çıkış, ileti verilerini değiştirirse, aşağıdaki yordamlardan birini kullanın:

- Verilerin uzunluğu değişmezse, veriler *MsgInPtr* tarafından adreslenen arabellekte değiştirilebilir. Bu durumda, *MsgOutPtr* ve *MsgOutLength* seçeneğini değiştirmeyin.
- Değiştirilen veriler özgün verilerden daha kısaysa, veriler *MsgInPtr* tarafından adreslenen arabellekte değiştirilebilir. Bu durumda, *MsgOutPtr* giriş iletisi arabelleğindeki adresin adresi olarak ayarlanmalıdır ve *MsgOutLength* , ileti verilerinin yeni uzunluğuna ayarlanır.
- Değiştirilen veriler özgün verilerden daha uzun olursa, çıkışta yeni bir ileti arabelleği edinilmesi gerekir. Değiştirilen verileri bu kopyaya kopyalayın. *MsgOutPtr* seçeneğini yeni arabelleğin adresine ayarlayın ve *MsgOutLength* değerini yeni ileti verilerinin uzunluğuna ayarlayın. Çıkış, bir sonraki çağrıldığında *MsgOutPtr* tarafından adreslenen arabelleğin serbest bırakılmasından çıkış sorumludur.

**Not:** *MsgOutPtr* her zaman çıkışa girilen boş değerli işaretir ve daha önce elde edilen bir ileti arabelleğindeki adresin değil. Daha önce elde edilen arabelleği serbest etmek için, çıkışta adresini ve uzunluğunu saklaması gerekir. Bilgileri *ExitUserArea*'de ya da adresi *ExitUserArea*' de kayıtlı olan bir denetim blokunda saklayın.

*MsgOutPtr* , çıkışa giriş/çıkış alanıdır.

### **MsgOutLength (MQLONG)**

*MsgOutLength* , çıkış tarafından döndürülen ileti verilerinin bayt cinsinden uzunluğudur. Çıkışa giriş sırasında, bu alan her zaman sıfırdır. Çıkıştan dönüşte, *MsgOutPtr* boş değerse bu alan yoksayıdır. İleti verilerinin değiştirilmesiyle ilgili bilgi için [MsgOutPtr](#) başlıklı konuya bakın.

*MsgOutLength* , çıkışa giriş/çıkış alanıdır.

### **pEntryPoints (PMQIEP)**

*pEntryPoints* , MQI ve DCI çağrılarının yapılabileceği bir MQIEP yapısının adresidir.

## **C dil bildirimi-MQPSXP**

```
typedef struct tagMQPSXP {
    MQCHAR4      StructId;          /* Structure identifier */
    MQLONG       Version;           /* Structure version number */
    MQLONG       ExitId;            /* Type of exit */
    MQLONG       ExitReason;        /* Reason for invoking exit */
    MQLONG       ExitResponse;      /* Response from exit */
    MQLONG       ExitResponse2;     /* Reserved */
    MQLONG       Feedback;          /* Feedback code */
    MQBYTE16     ExitUserArea;      /* Exit user area */
    MQCHAR32     ExitData;          /* Exit data */
    MQCHAR48     QMgrName;          /* Name of local queue manager */
    MQHCONN      Hconn;             /* Connection handle */
    MQHMSG       MsgHandle;         /* Handle to message properties */
    PMQMD        MsgDescPtr;        /* Address of message descriptor */
    PMQVOID      MsgInPtr;          /* Address of input message data */
    MQLONG       MsgInLength;       /* Length of input message data */
    PMQVOID      MsgOutPtr;         /* Address of output message data */
    MQLONG       MsgOutLength;     /* Length of output message data */
    /* Ver:1 */
    PMQIEP       pEntryPoints;      /* Address of the MQIEP structure */
    /* Ver:2 */
} MQPSXP;
```



## MQPBC-Yayın bağlamı veri yapısı

MQPBC yapısı, yayının yayıncısıyla ilgili bağlamsal bilgileri içerir; yayınlama çıkışa iletilir.

Çizelge 820 sayfa 1445 , yapıdaki alanları özetler:

Çizelge 820. MQPBC 'deki alanlar	
Alan	Tanım
<i>StrucID</i>	Yapı tanıttıcısı
<i>Version</i>	Yapı sürüm numarası
<i>PubTopicString</i>	Konu dizgisini yayınlama
<i>MsgDescPtr</i>	İleti tanımlayıcısının adresi (MQMD)

### Alanlar

#### **StrucID (MQCHAR4)**

*StrucID* yapı tanıttıcısıdır. Değer aşağıdaki gibidir:

#### **MQPBC\_STRUCID**

MQPBC\_STRUCID , yayın bağlamı yapısına ilişkin tanıttıcısıdır. For the C programming language, the constant MQPBC\_STRUC\_ID\_ARRAY is also defined; it has the same value as MQPBC\_STRUC\_ID, but is an array of characters instead of a string.

*StrucID* , çıkışa ilişkin bir giriş alanıdır.

#### **Version (MQLONG)**

*Version* , yapı sürüm numarasıdır. Değer aşağıdaki gibidir:

#### **MQPBC\_VERSION\_1**

MQPBC\_VERSION\_1 , Sürüm 1 yayınlama çıkış parametre yapısıdır.

#### **MQPBC\_VERSION\_2**

MQPBC\_VERSION\_2 , Sürüm 2 yayınlama çıkış parametre yapısıdır. The constant MQPBC\_CURRENT\_VERSION is also defined with the same value.

*Version* , çıkışa ilişkin bir giriş alanıdır.

#### **PubTopicString (MQCHARV)**

*PubTopicString* , yayınlanmakta olan konu dizisidir.

*PubTopicString* , çıkışa ilişkin bir giriş alanıdır.

#### **MsgDescPtr (PMQMD)**

*MsgDescPtr* , işlenmekte olan iletiye ilişkin ileti tanımlayıcısının (MQMD) bir kopyasının adresidir.

*MsgDescPtr* , çıkışa ilişkin bir giriş alanıdır.

## C dil bildiriimi-MQPBC

```
typedef struct tagMQPBC {
    MQCHAR4    StrucId;           /* Structure identifier */
    MQLONG     Version;          /* Structure version number */
    MQCHARV    PubTopicString;   /* Publish topic string */
    PMQMD      MsgDescPtr;       /* Address of message descriptor */
} MQPBC;
```

## MQSBC-Abonelik bağlamı veri yapısı

MQSBC yapısı, yayınlamayı alan aboneye ilişkin bağlamsal bilgileri içerir; yayınlama çıkışa geçer.

Çizelge 821 sayfa 1446 , yapıdaki alanları özetler:

Çizelge 821. MQSBCiçindeki alanlar	
Alan	Tanım
<u>StrucID</u>	Yapı tanıtıcısı
<u>Version</u>	Yapı sürüm numarası
<u>DestinationQMgrName</u>	Hedef kuyruk yöneticisinin adı
<u>DestinationQName</u>	Hedef kuyruğun adı
<u>SubType</u>	Abonelik tipi
<u>SubOptions</u>	Abonelik seçenekleri
<u>ObjectName</u>	Nesne adı
<u>ObjectString</u>	Nesne dizgisi
<u>SubTopicString</u>	Abonelik konusu dizgisi
<u>SubName</u>	Abonelik adı
<u>SubId</u>	Abonelik tanıtıcısı
<u>SelectionString</u>	Seçim dizgisinin adresi
<u>SubLevel</u>	Abonelik düzeyi
<u>PSPProperties</u>	Yayınlama/abone olma özellikleri

## Alanlar

### **StrucID (MQCHAR4)**

Yapı tanıtıcısı. Değer aşağıdaki gibidir:

#### **MQSBC\_STRUCID**

MQSBC\_STRUCID , yayınlama çıkış parametresi yapısına ilişkin tanıtıcıdır. For the C programming language, the constant MQSBC\_STRUC\_ID\_ARRAY is also defined; MQSBC\_STRUC\_ID\_ARRAY has the same value as MQSBC\_STRUC\_ID, but is an array of characters instead of a string.

*StrucID* , çıkışa ilişkin bir giriş alanıdır.

### **Version (MQLONG)**

Yapı sürüm numarası. Değer aşağıdaki gibidir:

#### **MQSBC\_VERSION\_1**

Sürüm 1 yayınlama çıkış parametre yapısı. The constant MQSBC\_CURRENT\_VERSION is also defined with the same value.

*Version* , çıkışa ilişkin bir giriş alanıdır.

### **DestinationQMgrName (MQCHAR48)**

*DestinationQMgrName* , iletinin gönderilmekte olduğu kuyruk yöneticisinin adıdır. Ad, alanın tam uzunluğuna kadar boşluklarla dolduruludur. Ad, çıkışa göre değiştirilebilir. Bu alanın uzunluğu MQ\_Q\_MGR\_NAME\_LENGTHtarafından verilir.

*DestinationQMgrName* , çıkışa giriş/çıkış alanıdır; bkz. [not](#).

### **DestinationQName (MQCHAR48)**

*DestinationQName* , iletinin gönderilmekte olduğu kuyruğun adıdır. Ad, alanın tam uzunluğuna kadar boşluklarla dolduruludur. Ad, çıkışa göre değiştirilebilir. Bu alanın uzunluğu MQ\_Q\_NAME\_LENGTHtarafından verilir.

*DestinationQName* , çıkışa giriş/çıkış alanıdır; bkz. [not](#).

**SubType (MQLONG)**

*SubType* , aboneliğin nasıl oluşturulduyla ilgili olduğunu belirtir. Geçerli değerler şunlardır: MQSUBTYPE\_API, MQSUBTYPE\_ADMIN ve MQSUBTYPE\_PROXY ; Bkz. [Abonelik Durumunu Sorgula \(Yanıt\)](#).

*SubType* , çıkışa ilişkin bir giriş alanıdır.

**SubOptions (MQLONG)**

*SubOptions* are the subscription options; see [“Seçenekler \(MQUZE\)”](#) sayfa 562 for a description of values this field can take.

*SubOptions* , çıkışa ilişkin bir giriş alanıdır.

**ObjectName (MQCHAR48)**

*ObjectName* , yerel kuyruk yöneticisinde tanımlandığı şekilde, konu nesnesinin adıdır. Bu alanın uzunluğu MQ\_TOPIC\_NAME\_LENGTH tarafından verilir. Nesne adı, kuyruk yöneticisinin konu dizgisiyle ilişkilendirildiği denetim konusu nesnesinin adıdır. Abone, aboneliğin bir parçası olarak bir konu nesnesi sağlasa bile, *ObjectName* farklı bir konu nesnesi olabilir. Bir konu nesnesinin aboneliğiyle ilişkilendirilmesi, *SubTopicString*' in tam çözümünün üzerine bağlıdır.

*ObjectName* , çıkışa ilişkin bir giriş alanıdır.

**ObjectString (MQCHARV)**

*ObjectString* , abone olunan yayının tam konu dizesidir. Özgün abonelik dizgisinde genel arama karakterleri çözülmüş olur. It is different to the MQSD subscription *ObjectString* field described in [“ObjectString \(MQCHARV\)”](#) sayfa 571, which might contain wildcards, and is exclusive of any object name provided by the subscriber.

*ObjectString* , çıkışa ilişkin bir giriş alanıdır.

**SubTopicString (MQCHARV)**

*SubTopicString* , abonenin sağladığı eksiksiz konu dizesidir. *SubTopicString* , bir konu nesnesinde tanımlanan konu dizgisinin birleşimidir ve bir konu dizesidir. Bir abone, bir konu nesnesini, bir konu dizgisini ya da her ikisini birden sağlamalıdır. Abone bir konu dizgisi sağlıyorsa, genel arama karakterleri de içerebilir.

*SubTopicString* , çıkışa ilişkin bir giriş alanıdır.

**SubName (MQCHARV)**

*SubName* , abone tarafından sağlanan ya da üretilmiş bir ad olan abonelik adıdır.

*SubName* , çıkışa ilişkin bir giriş alanıdır.

**SubId (MQBYTE 24)**

*SubId* , benzersiz iç abonelik tanıtıcısıdır.

*SubId* , çıkışa ilişkin bir giriş alanıdır.

**SelectionString (MQCHARV)**

*SelectionString* , bir konudaki iletiler için abone olduğunda kullanılan seçim ölçütüdür; bkz. [Seçiciler](#).

*SelectionString* , çıkışa ilişkin bir giriş alanıdır.

**SubLevel (MQLONG)**

*SubLevel* , abonelik ile ilişkili başlangıç düzeyi düzeyidir; ek ayrıntılar için [“SubLevel \(MQUZE\)”](#) sayfa 575 ' e bakın.

*SubLevel* , çıkışa ilişkin bir giriş alanıdır.

**PSPProperties (MQLONG)**

*PSPProperties* are the publish/subscribe properties. Bu abonelik ile ilgili ileti özelliklerinin yayınlama/abone olma ile ilgili ileti özelliklerinin nasıl ekleneceğini belirtirler. Olası değerler şunlardır: MQPSPROP\_NONE, MQPSPROP\_COMPAT, MQPSPROP\_RFH2, MQPSPROP\_MSGPROP. Bu değerlerin bir açıklaması için [İsteğe bağlı parametreler \(Değişiklik, Kopyalama ve Abonelik\)](#) konusuna bakın.

*PSPProperties* , çıkışa ilişkin bir giriş alanıdır.

**Not:** Yetkilendirme denetimleri, yayınlama çıkışa geçirilmeden önce yalnızca *DestinationQMgrName* ve *DestinationQName* özgün değerlerinde gerçekleştirilir. Çıkış, *DestinationQMgrName* ya da *DestinationQName* olarak değiştirilerek, çıkış hedef kuyruğunu değiştirdiğinde, yeni yetki denetimi gerçekleştirilmez.

## C dil bildirimi-MQSB

```
typedef struct tagMQSB {
    MQCHAR4   StrucId;           /* Structure identifier */
    MQLONG    Version;          /* Structure version number */
    MQCHAR48  DestinationQMgrName; /* Destination queue manager */
    MQCHAR48  DestinationQName; /* Destination queue name */
    MQLONG    SubType;          /* Type of subscription */
    MQLONG    SubOptions;       /* Subscription options */
    MQCHAR48  ObjectName;       /* Object name */
    MQCHARV   ObjectString;     /* Object string */
    MQCHARV   SubTopicString;    /* Subscription topic string */
    MQCHARV   SubName;          /* Subscription name */
    MQBYTE24  SubId;            /* Subscription identifier */
    MQCHARV   SelectionString;   /* Subscription selection string */
    MQLONG    SubLevel;         /* Subscription level */
    MQLONG    PSProperties;     /* Publish/subscribe properties */
} MQSB;
```

## Kanal-çıkış çağrılar ve veri yapıları

Bu konu derlemi, kanal çıkış programları yazdığınızda kullanabileceğiniz özel IBM MQ çağrılar ve veri yapılarıyla ilgili başvuru bilgileri sağlar.

Bu bilgiler, ürüne duyarlı programlama arabirimi bilgileridir. IBM MQ kullanıcı çıkışlarını aşağıdaki programlama dillerinde yazabilirsiniz:

<i>Çizelge 822. IBM MQ kullanıcı çıkışları: platformlar ve programlama dilleri</i>	
<b>Altyapı</b>	<b>Programlama dilleri</b>
IBM MQ for z/OS	Assembler and C (which must conform to the C system programming environment for system exits, described in the <i>z/OS C/C++ Programlama Kılavuzu</i> .)
IBM MQ for IBM i	ILE C, ILE COBOL ve ILE RPG
Diğer tüm IBM MQ platformları	C

You can also write user exits in Java for use only with Java and JMS applications. For more information about creating and using channel exits with the IBM MQ classes for Java, see [Using channel exits in IBM MQ classes for Java](#) and for IBM MQ classes for JMS, see [Using channel exits with IBM MQ classes for JMS](#).

IBM MQ kullanıcı çıkışlarını TAL ya da Visual Basic içinde yazamazsınız. However, a declaration for the MQCD structure is provided in Visual Basic for use on the MQCONN call from an IBM MQ MQI client program.

Aşağıdaki açıklamalardaki bazı durumlarda, parametreler, sabit olmayan bir boyutu olan diziler ya da karakter dizilimleri olur. Bu parametreler için küçük harf "n", sayısal bir sabiti göstermek için kullanılır. When the declaration for that parameter is coded, the "n" must be replaced by the numeric value required. Bu tanımlarda kullanılan kurullarla ilgili ek bilgi için ["Temel veri tipleri" sayfa 235'](#) e bakın.

## Veri tanımlama dosyaları

Veri tanımlama dosyaları, desteklenen programlama dillerinin her biri için IBM MQ ile birlikte sağlanır. Bu dosyaların ayrıntıları için [Copy, header, include, and module files](#) başlıklı konuya bakın.

## MQ\_CHANNEL\_EXIT-Kanal çıkışı

MQ\_CHANNEL\_EXIT çağrısı, Message Channel Agent tarafından çağrılan tüm kanal çıkışlarına geçirilen parametreleri açıklar.

Kuyruk yöneticisi tarafından MQ\_CHANNEL\_EXIT adlı bir giriş noktası sağlanmadı; kanal tanımında MQCD ' de bulunan kanal çıkışlarının adları belirtildiğinden, MQ\_CHANNEL\_EXIT adı özel bir önem içermiyor.

Kanal çıkışı beş tipi vardır:

- Kanal güvenlik çıkışı
- Kanal ileti çıkışı
- Kanal gönderme çıkışı
- Kanal alma çıkışı
- Kanal iletisi-yeniden deneme çıkışı

Parametreler her çıkış tipi için benzerdir ve burada verilen açıklama, özellikle belirtilenler dışında, tüm bunlar için geçerlidir.

### Sözdizimi

**MQ\_CHANNEL\_EXIT** (*ChannelExitParms*, *ChannelDefinition*, *DataLength*, *AgentBufferLength*, *AgentBuffer*, *ExitBufferLength*, *ExitBufferAddr*)

### Parametreler

MQ\_CHANNEL\_EXIT çağrısında aşağıdaki deęiřtirgeler vardır.

#### **ChannelExitParms (MQCXP)-giriř/çıkıř**

Kanal çıkışı parametre bloęu.

Bu yapı, çıkıřa iliřkin çağrıyla ilgili ek bilgi içerir. Çıkıř, MCA iřleminin nasıl devam eteceęini belirtmek için bu yapıdaki bilgileri ayarlar.

#### **ChannelDefinition (MQCD)-giriř/çıkıř**

Kanal tanımı.

Bu yapı, kanalın davranıřını denetleyebilmek için yönetici tarafından ayarlanan parametreleri içerir.

#### **DataLength (MQULength)-giriř/çıkıř**

Veri uzunluęu.

Veriler çıkıřa göre deęiřir:

- Bir kanal güvenlięi çıkıřı için, çıkıř çağrıldıęında bu deęiřtirge *AgentBuffer* alanında herhangi bir güvenlik iletisinin uzunluęunu içerir ( *ExitReason* MQXR\_SEC\_MSG ise). Herhangi bir ileti yoksa, bu deęer sıfır olur. Çıkıř, bu alanı, *ExitResponse* ' u MQXCC\_SEND\_SEC\_MSG ya da MQXCC\_SEND\_AND\_REQUEST\_SEC\_MSG olarak ayarlarsa, iř ortaęına gönderilecek herhangi bir güvenlik iletisinin uzunluęuna ayarlamalıdır. The message data is in either *AgentBuffer* or *ExitBufferAddr*.

güvenlik mesajlarının içerięi, güvenlik çıkıřlarının tek sorumluluęunda.

- Bir kanal iletisi çıkıřı için, çıkıřa çağrıldıęında bu parametre iletinin uzunluęunu (iletim kuyruęu üstbilgisi de içinde olmak üzere) içerir. Çıkıř, bu alanı, devam etmek için *AgentBuffer* ya da *ExitBufferAddr* ' da iletinin uzunluęuna ayarlamalıdır. Bu deęer, iletim kuyruęu üstbilgisinin (MQXQH) uzunluęundan büyük ya da bu uzunluęa eřit olmalıdır.
- Kanal gönderme ya da kanal alma çıkıřıysa, çıkıř çağrıldıęında bu parametre iletim uzunluęunu içerir. Çıkıř, bu alanı, devam etmek için *AgentBuffer* ya da *ExitBufferAddr* ' deki iletiminin uzunluęuna ayarlamalıdır.

If a security exit sends a message, and there is no security exit at the other end of the channel, or the other end sets an *ExitResponse* of MQXCC\_OK, the initiating exit is re-invoked with MQXR\_SEC\_MSG and a null response (*DataLength* =0).

### **AgentBufferUzunluđu (MQUZE)-giriş**

Aracı arabelleğindeki uzunluk.

Bu parametre, başlatma sırasında *DataLength* değerinden büyük olabilir.

Kanal iletisi, gönderme ve alma çıkışları için, başlatma sırasında kullanılmayan herhangi bir boşluk, verileri yerinde genişletmek için bu çıkış tarafından kullanılabilir. If this is done, the **DataLength** parameter must be set appropriately by the exit.

C programlama dilinde, bu parametre adrese göre iletilir.

### **AgentBuffer (MQBYTE x AgentBufferUzunluđu)-giriş/çıkış**

Aracı arabelleği.

Bu parametrenin içeriği çıkış tipine bağlıdır:

- For a channel security exit, on invocation of the exit it contains a security message if *ExitReason* is MQXR\_SEC\_MSG. Bir güvenlik iletisini geri göndermek için, çıkış bu arabelleği ya da kendi arabelleğini (*ExitBufferAddr*) kullanabilir.
- Bir kanal iletisi çıkışı için, bu değıştirgenin içerdiği çıkışa ilişkin olarak aşağıdaki komutu içerir:
  - İleti tanımlayıcısını (ileti için bağlam bilgilerini içeren) içeren iletim kuyruđu üstbilgisi (MQXQH), hemen ardından
  - İleti verileri

İleti devam etmek ise, çıkışta aşağıdakilerden birini yapabilirsiniz:

- Dokunulmamış arabellek içeriğini bırak
- Modify the contents in place (returning the new length of the data in *DataLength* ; this must not be greater than *AgentBufferLength*)
- Gerekli değışiklikleri yapmak için içeriği *ExitBufferAddr'* e kopyalayın.

Çıkışta iletim kuyruđu üstbilgisinde yapılan değışiklikler denetlenmez; ancak, hatalı değışiklikler, iletinin hedefe konamadığı anlamına gelebilir.

- Bir kanal gönderme ya da alma çıkışı için, çıkışa ilişkin başlatma sırasında iletim verilerini içeren verileri içerir. Çıkış aşağıdakilerden birini yapabilir:
  - Dokunulmamış arabellek içeriğini bırak
  - Modify the contents in place (returning the new length of the data in *DataLength* ; this must not be greater than *AgentBufferLength*)
  - Gerekli değışiklikleri yapmak için içeriği *ExitBufferAddr'* e kopyalayın.

Verilerin ilk 8 baytı çıkış tarafından değıştirilmemelidir.

### **ExitBufferUzunluk (MQUZE)-giriş/çıkış**

Çıkış arabelleğindeki uzunluk.

Çıkışa ilişkin ilk çağrıda bu parametre sıfır olarak ayarlanır. Bundan sonra her bir değer, her çağrıda çıkış tarafından geri geçirilirse, çıkışa bir sonraki çağrıldığında gösterilir. Değer MCA tarafından kullanılmaz.

**Not:** Bu parametre, işaretçi veri tipini desteklemeyen programlama dillerinde yazılan çıkışlar tarafından kullanılmamalıdır.

### **ExitBufferAddr (MQPTR)-giriş/çıkış**

Çıkış arabelleğindeki adres.

Bu deęiřtirge, ıkıřta ynetilen depolama arabelleęindeki adresin adresidir; aracının arabelleęi varsa ya da yeterince byk deęilse ya da ıkıř iin daha uygun olduęunda, aracıya ileti ya da iletim verilerini (ıkıř tipine baęlı olarak) dndrmeyi seebileceęi bir bellek arabelleęidir.

ıkıřa iliřkin ilk aęırma sırasında, ıkıřa geirilen adres boř deęerli. Bundan sonra, her bir aęrıdan ıkıřta hangi adres aktarılırsa, ıkıřa bir sonraki aęrıldıęında bu ıkıřa sunulur.

ExitBufferAddr boř deęer ise, kullanılan veriler AgentBuffer deęiřtirgesinden alınır.

ExitBufferAddr boř deęerli deęilse, kullanılan veriler ExitBufferAddr parametresine iřaret edilen arabellekten alınır.

**Not:** Bu parametre, iřareti veri tipini desteklemeyen programlama dillerinde yazılan ıkıřlar tarafından kullanılmamalıdır.

## C aęırma

```
exitname (&ChannelExitParms, &ChannelDefinition,  
&DataLength, &AgentBufferLength, AgentBuffer,  
&ExitBufferLength, &ExitBufferAddr);
```

ıkıřa geirilen parametreler ařaęıdaki gibi bildirilir:

```
MQCXP  ChannelExitParms; /* Channel exit parameter block */  
MQCD   ChannelDefinition; /* Channel definition */  
MQLONG DataLength; /* Length of data */  
MQLONG AgentBufferLength; /* Length of agent buffer */  
MQBYTE AgentBuffer[n]; /* Agent buffer */  
MQLONG ExitBufferLength; /* Length of exit buffer */  
MQPTR  ExitBufferAddr; /* Address of exit buffer */
```

## Cobol aęırma

```
CALL 'exitname' USING CHANNELEXITPARMS, CHANNELDEFINITION,  
DATALENGTH, AGENTBUFFERLENGTH, AGENTBUFFER,  
EXITBUFFERLENGTH, EXITBUFFERADDR.
```

ıkıřa geirilen parametreler ařaęıdaki gibi bildirilir:

```
** Channel exit parameter block  
01 CHANNELEXITPARMS.  
COPY CMQCXPV.  
** Channel definition  
01 CHANNELDEFINITION.  
COPY CMQCDV.  
** Length of data  
01 DATALENGTH PIC S9(9) BINARY.  
** Length of agent buffer  
01 AGENTBUFFERLENGTH PIC S9(9) BINARY.  
** Agent buffer  
01 AGENTBUFFER PIC X(n).  
** Length of exit buffer  
01 EXITBUFFERLENGTH PIC S9(9) BINARY.  
** Address of exit buffer  
01 EXITBUFFERADDR POINTER.
```

## RPG aęırısı (ILE)

```
C*.1.....2.....3.....4.....5.....6.....7..  
C CALL exitname(MQCXP : MQCD : DATLEN :  
C ABUFL : ABUF : EBUFL :  
C EBUF)
```

aęırma iliřkin prototip tanımı:

```

D*..1.....2.....3.....4.....5.....6.....7..
Dexitname          PR          EXTPROC('exitname')
D* Channel exit parameter block
D MQCXP            160A
D* Channel definition
D MQCD            1328A
D* Length of data
D DATLEN          10I 0
D* Length of agent buffer
D ABUFL           10I 0
D* Agent buffer
D ABUF            *   VALUE
D* Length of exit buffer
D EBUFL           10I 0
D* Address of exit buffer
D EBUF            *

```

## System/390 çevirici çağırma

```

CALL EXITNAME, (CHANNELEXITPARMS, CHANNELDEFINITION, DATALENGTH, X
AGENTBUFFERLENGTH, AGENTBUFFER, EXITBUFFERLENGTH, X
EXITBUFFERADDR)

```

Çıkışa geçirilen parametreler aşağıdaki gibi bildirilir:

CHANNELEXITPARMS	CMQCXPA	,	Channel exit parameter block
CHANNELDEFINITION	CMQCDA	,	Channel definition
DATALENGTH	DS	F	Length of data
AGENTBUFFERLENGTH	DS	F	Length of agent buffer
AGENTBUFFER	DS	CL(n)	Agent buffer
EXITBUFFERLENGTH	DS	F	Length of exit buffer
EXITBUFFERADDR	DS	F	Address of exit buffer

## Kullanım notları

1. Kanal çıkışı tarafından gerçekleştirilen işlev, çıkışa ilişkin sağlayıcıya göre tanımlanır. Ancak çıkış, burada ve ilişkili denetim blokunda, MQCXP ' de tanımlanan kurallara uymalıdır.
2. Kanal çıkışa aktarılan **ChannelDefinition** parametresi birkaç sürümden biri olabilir. Daha fazla bilgi için, MQCD yapısındaki *Version* alanına bakın.
3. If the channel exit receives an MQCD structure with the *Version* field set to a value greater than MQCD\_VERSION\_1, the exit must use the *ConnectionName* field in MQCD, in preference to the *ShortConnectionName* field.
4. Genel olarak, kanal çıkışlarının ileti verilerinin uzunluğunu değiştirmesine izin verilir. Bu, iletiye veri eklenmesi ya da iletiden veri kaldırılması ya da iletinin sıkıştırılması ya da şifrelenmesi sonucunda ortaya çıkabilir. Ancak, ileti, mantıksal iletinin yalnızca bir kısmını içeren bir bölümse, özel kısıtlamalar geçerli olur. özellikle, tamamlayıcı gönderme ve alma çıkışlarının işlemlerinin bir sonucu olarak, mesajın uzunluğunun net bir şekilde net bir değişiklik olmaması gerekiyor.

Örneğin, iletiyi sıkıştırarak kısaltmak için bir gönderen çıkışa izin verilebilir, ancak iletiyi tamamlayıcı alan çıkışı, iletinin uzunluğunun net bir şekilde değiştirilmediği için, iletinin özgün uzunluğuna basarak iletinin özgün uzunluğunu geri yüklememelidir.

Bu kısıtlama, bir kesimin uzunluğunun değiştirilmesi iletisinde daha sonraki bölümlerin görelî konumlarının yanlış olmasına neden olacağı için, kuyruk yöneticisinin kesimlerin tam bir mantıksal ileti oluşturup oluşturmadığını algılayabilme yeteneğini engeller.

## MQ\_CHANNEL\_AUTO\_DEF\_EXIT-Kanal otomatik tanımlama çıkışı

MQ\_CHANNEL\_AUTO\_DEF\_EXIT çağırısı, Message Channel Agent tarafından çağrılan kanal otomatik tanımlama çıkışa geçirilen değıştirmeleri tanımlar.



Kuyruk yöneticisi tarafından MQ\_CHANNEL\_AUTO\_DEF\_EXIT adlı bir giriş noktası sağlanmadı; kuyruk yöneticisinde otomatik tanımlama çıkışlarının adları belirtildiğinden, MQ\_CHANNEL\_AUTO\_DEF\_EXIT adı özel bir önem içermiyor.

## Sözdizimi

**MQ\_CHANNEL\_AUTO\_DEF\_EXIT (*ChannelExitParms*, *ChannelDefinition*)**

## Parametreler

MQ\_CHANNEL\_AUTO\_DEF\_EXIT çağrısının aşağıdaki parametreleri vardır.

### ChannelExitParms (MQCXP)-giriş/çıkış

Kanal çıkışı parametre bloğu.

Bu yapı, çıkışa ilişkin çağrıyla ilgili ek bilgi içerir.Çıkış, MCA işleminin nasıl devam eteceğini belirtmek için bu yapıdaki bilgileri ayarlar.

### ChannelDefinition (MQCD)-giriş/çıkış

Kanal tanımı.

Bu yapı, otomatik olarak oluşturulan kanalların işleyişini denetlemek için yönetici tarafından ayarlanan parametreleri içerir.Çıkış, yönetici tarafından ayarlanan varsayılan davranışı değiştirmek için bu yapıdaki bilgileri ayarlar.

Listelenen MQCD alanlarının çıkış tarafından değiştirilmemesi gerekir:

- *ChannelName*
- *ChannelType*
- *StrucLength*
- *Version*

Diğer alanlar değiştirilirse, çıkışa göre ayarlanan değer geçerli olmalıdır. Değer geçerli değilse, hata günlüğü dosyasına bir hata iletisi yazılır ya da konsolda görüntülenir (ortama uygun).



**Uyarı:** Kanal otomatik tanımlaması (CHAD) çıkışı tarafından oluşturulan otomatik tanımlı kanallar, kanal yaratıldığı zaman TLS anlaşması olduğu için sertifika etiketini ayarlayamaz. Gelen kanallara ilişkin CHAD çıkışta sertifika etiketinin ayarlanması hiçbir etkiye sahip değildir.

## C çağırma

```
exitname (&ChannelExitParms, &ChannelDefinition);
```

Çıkışa geçirilen parametreler aşağıdaki gibi bildirilir:

```
MQCXP ChannelExitParms; /* Channel exit parameter block */
MQCD ChannelDefinition; /* Channel definition */
```

## Cobol çağırma

```
CALL 'exitname' USING CHANNELEXITPARMS, CHANNELDEFINITION.
```

Çıkışa geçirilen parametreler aşağıdaki gibi bildirilir:

```
** Channel exit parameter block
01 CHANNELEXITPARMS.
COPY CMQCXPV.
** Channel definition
```

```
01 CHANNELDEFINITION.  
COPY CMQCDV.
```

## RPG çağrısı (ILE)

```
C*.1.....2.....3.....4.....5.....6.....7..  
C          CALLP          exitname(MQCXP : MQCD)
```

Çağrıya ilişkin prototip tanımı:

```
D*.1.....2.....3.....4.....5.....6.....7..  
Dexitname PR          EXTPROC('exitname')  
D* Channel exit parameter block  
D MQCXP          160A  
D* Channel definition  
D MQCD          1328A
```

## System/390 çevirici çağırma

```
CALL EXITNAME, (CHANNELEXITPARMS, CHANNELDEFINITION)
```

Çıkışa geçirilen parametreler aşağıdaki gibi bildirilir:

```
CHANNELEXITPARMS CMQCXPA , Channel exit parameter block  
CHANNELDEFINITION CMQCDA , Channel definition
```


## Kullanım notları

1. Kanal çıkışı tarafından gerçekleştirilen işlev, çıkışa ilişkin sağlayıcıya göre tanımlanır. Ancak çıkış, burada ve ilişkili denetim blokunda, MQCXP ' de tanımlanan kurallara uymalıdır.
2. Kanala otomatik tanımlama çıkışa geçirilen **ChannelExitParms** parametresi bir MQCXP yapısıdır. İletilen MQCXP sürümü, çıkışa bağlı ortama bağlıdır; ayrıntılar için [“MQCXP-Kanal çıkış değiştirgesi” sayfa 1495](#) içindeki *Version* alanının tanımına bakın.
3. Kanala otomatik tanımlama çıkışa geçirilen **ChannelDefinition** parametresi bir MQCD yapısıdır. İletilen MQCD sürümü, çıkışa bağlı ortama bağlıdır; ayrıntılar için [“MQCD-Kanal tanımlaması” sayfa 1456](#) içindeki *Version* alanının tanımına bakın.

## MQXWIN-Çıkışta bekle

MQXWINE çağrısı bir olayın oluşmasını bekler. Yalnızca z/OS üzerindeki bir kanal çıkışından kullanılabilir.

MQXWAIN kullanımı, bir kanal çıkışı bir beklenmeye neden olan bir şey yaparsa, başka bir şekilde oluşabilecek başarım sorunlarını önlemeye yardımcı olur. MQXBEKE olayı, MVS ECB (olay denetim bloğu) tarafından işaretlenir. ECB, MQXWD denetim bloğu tanımında açıklanmıştır.

 MQXWAST kullanımı ve kanal çıkışı programlarının kullanımı hakkında daha fazla bilgi için bkz. [Writing channel exit programs on z/OS](#)

## Sözdizimi

**MQXDUR (Hconn, WaitDesc, CompCode, Reason)**

## Parametreler

MQXBEKE çağrısının değiştirgeleri şunlardır:

## Hconn (MQHCONN)-giriş

Bağlantı tanıtıcısı.

Bu tanıtıcı, kuyruk yöneticisiyle olan bağlantıyı gösterir. The value of *Hconn* was returned by a previous MQCONN call issued in the same or earlier invocation of the exit.

## WaitDesc (MQXWD)-giriş/çıkış

Bekle bir tanımlayıcı.

Bu değıştirge, beklenecek olayı açıklar. Bu yapıdaki alanların ayrıntıları için bkz. "[MQXWD-Çıkış bekleme tanımlayıcısı](#)" sayfa 1509 .

## CompCode (MQHOT)-çıkış

Tamamlanma kodu.

Bu, aşağıdaki kodlardan biridir:

### MQCC\_OK

İşlem başarıyla tamamlandı.

### MQCC\_FAILED

Arama başarısız oldu.

## Neden (MQUZE)-çıkış

Neden kodu ön eleme *CompCode*.

*CompCode* MQCC\_OK ise:

### MQRC\_NONE

(0, X'000 ') Raporlamak için bir neden yok.

### MQRC\_ADAPTER\_NOT\_AVAM

(2204, X'89C') Bağdaştırıcı kullanılamıyor.

### MQRC\_OPTIONS\_ERROR

(2046, X'7FE') Seçenekler geçerli değil ya da tutarlı değil.

### MQRC\_XWAIT İPTAL EDİLDİ

(2107, X'83B') MQXBEKE çağrısı iptal edildi.

### MQRC\_XWAIT\_ERROR

(2108, X'83C') MQXBEKE çağrısının çağrılması geçerli değil.

## C çağırma

```
MQXWAIT (Hconn, &WaitDesc, &CompCode, &Reason);
```

Değıştirgeleri aşağıdaki gibi bildirin:

```
MQHCONN Hconn; /* Connection handle */
MQXWD WaitDesc; /* Wait descriptor */
MQLONG CompCode; /* Completion code */
MQLONG Reason; /* Reason code qualifying CompCode */
```

## System/390 çevirici çağırma

```
CALL MQXWAIT, (HCONN, WAITDESC, COMPCODE, REASON)
```

Değıştirgeleri aşağıdaki gibi bildirin:

```
HCONN DS F Connection handle
WAITDESC CMQXWDA , Wait descriptor
```

COMPCODE	DS	F	Completion code
REASON	DS	F	Reason code qualifying COMPCODE

## MQCD-Kanal tanımlaması

MQCD yapısı, bir kanalın yürütülmesini denetleyen deęiřtirgeleri ierir. İleti kanalı aracısından (MCA) aęrılan her kanal ıkıřa geilir.

Kanal ıkıřlarına iliřkin daha fazla bilgi iin bkz. "MQ\_CHANNEL\_EXIT-Kanal ıkıřı" sayfa 1449. Bu konudaki aıklama hem ileti kanallarıyla, hem de MQI kanallarıyla iliřkilendirir.

## ıkıř adı alanları

Bir ıkıř aęrıldıęında, *SecurityExit*, *MsgExit*, *SendExit*, *ReceiveExit* ve *MsgRetryExit* 'daki ilgili alan řu anda aęrılmakta olan ıkıřa iliřkin adı ierir. Bu alanlardaki adın anlamı, MCA 'nın alıřmakta olduęu ortama baęlıdır. Burada, dikkat edilenler dıřında, ad alanda sola hizalanmıř ve bořluk bırakılmaz; ad, alanın uzunluęuna kadar bořluklarla doldurulur. Ařaęıdaki aıklamalarda, kşeli ayrılar ([ ]) isteęe baęlı bilgileri gsterir:

### AIX and Linux

ıkıř adı, dinamik olarak yklenebilir bir modln ya da kitaplıęın adı, o kitaplıkta bulunan bir iřlevin adını tařıyan bir kitaplık adıdır. İřlev adının ayrı iine alınması gerekir. Kitaplık adı isteęe baęlı olarak bir dizin yolu ile nek olarak nek olarak kullanılabilir:

```
[ path ] library ( function )
```

Ad en ok 128 karakterle sınırlanmıřtır.

### z/OS

ıkıř adı, LINK ya da LOAD makrosu iin EP parametresindeki belirtim iin geerli olan bir ykleme modlnn adıdır. Ad en ok sekiz karakterden oluřan bir deęer ile sınırlanmıřtır.

### Windows

ıkıř adı, o kitaplıkta bulunan bir iřlevin adını tařıyan, devingen baęlı bir kitaplıęın adıdır. İřlev adının ayrı iine alınması gerekir. Kitaplık adı, isteęe baęlı olarak bir dizin yolu ve src neki ile nek olarak kullanılabilir:

```
[d:][ path ] library ( function )
```

Ad en ok 128 karakterle sınırlanmıřtır.

### IBM i

ıkıř adı, 10 baytlık bir program adı ve ardından 10 baytlık bir kitaplık adıdır. Adlar 10 bayttan daha uzunsa, her ad 10 bayt yapmak iin bořluklarla doldurulsun. The library name can be \*LIBL except when calling a channel auto-definition exit, in which case a fully qualified name is required.

## Kanal ıkıřındaki MQCD alanlarının deęiřtirilmesi

Bir kanal ıkıřı, MQCD 'deki alanları deęiřtirebilir. Deęiřtirilen deęer MQCD 'de kalır ve bir ıkıř zincirindeki geri kalan ıkıřlara ve kanal ynetim ortamını paylařan herhangi bir konuřmaya geirilir. Deęiřtirilen MQCD, kanalda devam eden yařam sresi boyunca olaęan iřlemesi iin MCA tarafından da kullanılır.

Ařaęıdaki MQCD alanlarının ıkıř tarafından deęiřtirilmemesi gerekir:

- ChannelName
- ChannelType
- StructLength
- S\u00fcr\u00fcm

## İlgili başvurular

“Alanlar” sayfa 1457

Bu konu, MÇCD yapısındaki tüm alanları listeler ve her bir alanı açıklar.

“C bildirim” sayfa 1482

Bu bildirim, MÇCD yapısına ilişkin C bildirimidir.

“COBOL bildirim” sayfa 1484

Bu bildirim, MÇCD yapısıyla ilgili COBOL bildirimidir.

“RPG bildirim (ILE)” sayfa 1487

Bu bildirim, MÇCD yapısına ilişkin RPG bildirimidir.

“System/390 çevirici bildirim” sayfa 1489

Bu bildirim, MÇCD yapısı için System/390 çevirici bildirimidir.

“Visual Basic bildirim” sayfa 1491

Bu bildirim, MÇCD yapısındaki Visual Basic bildirimidir.

“Kanal çıkışındaki MÇCD alanlarının değiştirilmesi” sayfa 1492

Bir kanal çıkışı, MÇCD ' deki alanları değiştirebilir. Ancak, listelenen koşullar dışında, bu değişiklikler genellikle üzerinde işlem yapmamış olur.

## Alanlar

Bu konu, MÇCD yapısındaki tüm alanları listeler ve her bir alanı açıklar.

*BatchDataSınırı (MQUUR)*

Bu alan, bir eşitleme noktası almadan önce bir kanaldan gönderilebilecek veri miktarının kilobayt cinsinden sınırını belirtir.

Bir eşitleme noktası, sınıra ulaşılmasına neden olan iletinin kanaldan aktarılmasından sonra alınır.

Aşağıdaki koşullardan biri karşılandığında toplu iş sonlandırılır:

- **BatchSize** iletileri gönderildi.
- **BatchDataLimit** byte 'ları gönderildi.
- İletim kuyruğu boş ve **BatchInterval** aşıldı.

Değer 0-999999 aralığında olmalıdır. Varsayılan değer 5000 'dir.

Bu özniteliğe ilişkin sıfır değeri, bu kanal üzerinden toplu işler için veri sınırı uygulanmadığı anlamına gelir.

Bu parametre yalnızca, *ChannelType* MQCHT\_SENDER, MQCHT\_SERVER, MQCHT\_CLUUSRCVR ya da MQCHT\_CLUSSDR içeren kanallar için geçerlidir.

Bu, çıkışa giriş alanıdır. *Version* değeri MQCD\_VERSION\_11değerinden küçükse bu alan yoktur.

*BatchHeartbeat (MQUZA)*

Bu alan, kanala ilişkin toplu iş işaretini tetiklemek için kullanılan zaman aralığını belirtir.

Toplu kalp atımı, belirsiz bir şekilde devam etmeden önce, gönderen kanallarının uzak kanal eşgörünümünün hala etkin olup olmadığını belirlemesine olanak sağlar. Bir gönderen kanalı, belirtilen zaman aralığı içindeki uzak kanal örneğiyle iletişim kurmadıysa, toplu iş bildirim oluşur.

Değer, 0-999 999 aralığında; birimler ise milisaniye cinsinden. Sıfır değeri, toplu kalp atışlarının etkinleştirilmediğini gösterir.

Bu alan yalnızca, *ChannelType* MQCHT\_ENDER, MQCHT\_SERVER, MQCHT\_CLUUSSDR ya da MQCHT\_CLAUSRCVR ile ilişkili kanallar için anlamlıdır.

Bu, çıkışa giriş alanıdır. *Version* değeri MQCD\_VERSION\_7değerinden küçükse bu alan yoktur.

*BatchInterval (MQUZE)*

Bu alan, yürürlükteki toplu işte *BatchSize* ' den az ileti iletildiyse, bir kanalın toplu olarak açık kalmasını sağlayan yaklaşık süreyi milisaniye cinsinden belirtir.

*BatchInterval* sıfırdan büyükse, ilk önce aşağıdaki olaylardan hangisi gerçekleşirse, toplu iş sonlandırılır:

- *BatchSize* iletileri gönderildi ya da
- Toplu işin başlangıcından bu yana geçen *BatchInterval* milisaniye sayısı.

*BatchInterval* sıfırsa, ilk önce aşağıdaki olaylardan hangisi gerçekleşirse, toplu iş sonlandırılır:

- *BatchSize* iletileri gönderildi ya da
- İletim kuyruğu boş olur.

*BatchInterval* , 0 ile 999 999 999 arasında bir değer aralığında olmalıdır.

Bu alan yalnızca, *ChannelType* MQCHT\_ENDER, MQCHT\_SERVER, MQCHT\_CLUUSDR ya da MQCHT\_CLAUSRCVR ile olan kanallar için geçerlidir.

Bu, çıkışa giriş alanıdır. *Version* değeri MQCD\_VERSION\_4 değerinden küçükse bu alan yoktur.

#### *BatchSize (MQUZA)*

Bu alan, kanalı eşitlemeden önce bir kanaldan gönderilebilecek ileti sayısı üst sınırını belirtir.

This field is not relevant for channels with a *ChannelType* of MQCHT\_SVRCONN or MQCHT\_CLNTCONN.

#### *CertificateLabel (MQCHAR64)*

Bu alan, kullanılmakta olan sertifika etiketinin ayrıntılarını verir.

IBM MQ , *CertificateLabel* alanı için varsayılan değeri boşluk olarak başlatır.

Bu, yürütüm sırasında varsayılan değer olarak yorumlanır ve geriye doğru uyumludur.

Örneğin, 11 'den küçük bir MQCD sürümü belirtilmesi ya da *CertificateLabel* alanı için varsayılan boşluk değerinin kullanılması, bu alanın yoksayıldığı anlamına gelir.

Bu alanın uzunluğu MQ\_CERT\_LABEL\_LENGTH tarafından verilir.

#### *ChannelMonitoring (MQUZE)*

Bu alan, kanala ilişkin izleme verileri derleminin yürürlükteki düzeyini belirtir.

This field is not relevant for channels with a *ChannelType* of MQCHT\_CLNTCONN.

Bu değer aşağıdaki değerlerden biridir:

- MQMON\_OFF
- MQMON\_LOW
- MQMON\_ORTAMI
- MQMON\_YükSEK

Bu, çıkışa giriş alanıdır. *Version* , MQCD\_VERSION\_8 değerinden küçükse, bu ad yoktur.

#### *ChannelName (MQCHAR20)*

Bu alan kanal tanımlama adını belirtir.

İletişim kurabilmek için, uzak makineden aynı adı içeren bir kanal tanımlaması olmalıdır.

Ad yalnızca şu karakterleri kullanmalıdır:

- Büyük harf A
- Küçük harf a-z
- Sayısal 0-9
- Nokta (.)
- Eğik Çizgi (/)
- Alt çizgi (\_)
- Yüzde işareti (%)

ve boşluklarla doldurulmalıdır. Baştaki ya da gömülü boşluklara izin verilmez.

Bu alanın uzunluğu MQ\_CHANNEL\_NAME\_LENGTH tarafından verilmektedir.

#### *ChannelStatistics (MQUZE)*

Bu alan, kanala ilişkin istatistik verileri derleminin yürürlükteki düzeyini belirtir.

This field is not relevant for channels with a ChannelType of MQCHT\_CLNTCONN or MQCHT\_SVRCONN.

Bu değer aşağıdaki değerlerden biridir:

- MQMON\_OFF
- MQMON\_LOW
- MQMON\_ORTAMI
- MQMON\_YükSEK

Bu, çıkışa giriş alanıdır. *Version* , MQCD\_VERSION\_8değerinden küçükse, bu ad yoktur.

#### *ChannelType (MQUZE)*

Bu alan kanal tipini belirtir.

Bu değer aşağıdaki değerlerden biridir:

#### **MQCHT\_SENDER**

Gönderen.

#### **MQCHT\_SERVER**

server.

#### **MQCHT\_RECEI**

Alıcı.

#### **MQCHT\_REQUESTER**

İstekte bulunanın.

#### **MQCHT\_CLNTCONN**

İstemci bağlantısı.

#### **MQCHT\_SVRCONN**

Sunucu bağlantısı (istemciler tarafından kullanılmak üzere).

#### **MQCHT\_CLUSSDR**

Küme gönderen.

#### **MQCHT\_CLAUSRCVR**

Küme alıcısı.

#### *ClientChannelAğırlık (MQUZE)*

Bu alan, hangi istemci bağlantısı kanal tanımının kullanılacağı etki alanını etkileyecek bir ağırlıklandırma belirtir.

ClientChannelAğırlık özneliği, birden çok uygun tanımlama kullanılabilir olduğunda, müşteri kanalı tanımlamalarının ağırlıklandırmalarına dayalı olarak rasgele seçilebilmesi için kullanılır. Bir istemci, bir MQCONN ile bir kuyruk yöneticisi grubuyla bağlantı isteğinde bulunurken, bir yıldız işaretiyle başlayan bir kuyruk yöneticisi adı belirtilerek ve istemci kanal tanımlama çizelgesinde (CCDT) birden çok uygun kanal tanımlaması varsa, kullanılacak tanımlama, ilk olarak alfabetik sırayla seçilen geçerli ClientChannelAğırlık (0) tanımlarıyla birlikte ağırlıklandırma temelinde rasgele seçilir.

0-99 aralığında bir değer belirtin. Varsayılan, 0'dır.

0 değeri, herhangi bir yük dengelemenin gerçekleştirilmediği ve geçerli tanımların alfabetik sırayla seçildiğini gösterir. Yük dengelemeyi etkinleştirmek için 1-99 aralığında bir değer seçin; burada 1, en düşük ağırlıklandırma ve 99 en yüksek ağırlıklandırma olur. Sıfır olmayan ağırlıklandırmalar içeren iki ya da daha fazla kanal arasındaki iletilerin dağılımı, bu ağırlıklandırmaların oranı ile orantılıdır. Örneğin, ClientChannelAğırlık değeri 2, 4 ve 14 olan üç kanal yaklaşık %10, 20% ve %70 'i seçilir. Bu dağıtım garanti edilemez.

Bu öznitelik, yalnızca istemci-bağlantı kanalı tipi için geçerlidir.

Bu, çıkışa giriş alanıdır. *Sürüm* , MQCD\_VERSION\_9değerinden küçükse, alan yoktur.

#### *ClusterPtr (MQPTR)*

Bu alan, küme adlarının listesini belirtir.

*ClustersDefined* sıfırdan büyükse, bu adres küme adları listesinin adresidir. Kanal, listelenen her kümeye aittir.

Bu alan yalnızca, bir MQCHT\_CLUSSDR ya da MQCHT\_CLAUSRCVR *ChannelType* olan kanallar için anlamlıdır.

Bu, çıkışa giriş alanıdır. *Version* değeri MQCD\_VERSION\_5değerinden küçükse bu alan yoktur.

#### *ClustersDefined (MQHOMER)*

Bu alan, kanalın ait olduğu kümelerin sayısını belirtir.

Bu alan, *ClusterPtr* ile gösterilen küme adlarının sayısıdır. Sıfır ya da daha büyük bir değer.

Bu alan yalnızca, bir MQCHT\_CLUSSDR ya da MQCHT\_CLAUSRCVR *ChannelType* olan kanallar için anlamlıdır.

Bu, çıkışa giriş alanıdır. *Version* değeri MQCD\_VERSION\_5değerinden küçükse bu alan yoktur.

#### *CLWLChannelPriority (MQUZE)*

Bu alan, küme iş yükü kanal önceliğini belirtir.

İş yükü yöneticisi seçme algoritması, sıra temelinde seçilen hedefler kümesinden en yüksek önceliğe sahip bir hedef seçer. Olası iki hedef kuyruk yöneticisi varsa, bu öznitelik, bir kuyruk yöneticisini diğer kuyruk yöneticisine hata durumunda yedek sisteme geçiş yapmak için kullanılabilir. Tüm iletiler, sonuna kadar en yüksek önceliğe sahip kuyruk yöneticisine gider, sonra iletiler bir sonraki en yüksek önceliğe sahip kuyruk yöneticisine gider.

Değer, 0 ile 9 arasında bir aralıktır. Varsayılan, 0'dır.

Bu, çıkışa giriş alanıdır. *Version* değeri MQCD\_VERSION\_8değerinden küçükse bu alan yoktur.

Ek bilgi için [Kuyruk yöneticisi kümesinin yapılandırılması](#) başlıklı konuya bakın.

#### *CLWLChannelRank (MQUZE)*

Bu alan, küme iş yükü kanal sırasını belirtir.

İş yükü yöneticisi, en yüksek derecenin sahip olduğu bir hedef seçmeyi seçiyor. Son hedef, farklı bir kümede kuyruk yöneticisi olduğunda, ara ağ geçidi kuyruk yöneticilerinin (komşu kümelerin kesişme noktasında) sırasını ayarlayabilirsiniz, böylece seçme algoritması hedef kuyruk yöneticisini son varış noktasına daha yakın bir hedef kuyruk yöneticisini seçer.

Değer, 0 ile 9 arasında bir aralıktır. Varsayılan, 0'dır.

Bu, çıkışa giriş alanıdır. *Version* değeri MQCD\_VERSION\_8değerinden küçükse bu alan yoktur.

Ek bilgi için [Kuyruk yöneticisi kümesinin yapılandırılması](#) başlıklı konuya bakın.

#### *CLWLChannelWeight (MQUZE)*

Bu alan, küme iş yükü kanal ağırlığını belirtir.

Küme iş yükü kanal ağırlığı.

İş yükü yöneticisi, algoritmanın hedef seçimi sapma noktasına kadar olan "ağırlık" özneliğini kullanarak, belirli bir makineye daha fazla ileti gönderebilmesini sağlar. For example, you can give a channel on a large UNIX server a larger "weight" than another channel on small desktop PC, and the choose algorithm chooses the UNIX server more frequently than the PC.

Değer, 1 ile 99 arasında bir aralıktır. Varsayılan değer 50 'dir.

Bu, çıkışa giriş alanıdır. *Version* değeri MQCD\_VERSION\_8değerinden küçükse bu alan yoktur.



Ek bilgi için Kuyruk yöneticisi kümesinin yapılandırılması başlıklı konuya bakın.

#### *ConnectionAffinity (MQUBE)*

Bu alan, aynı kuyruk yöneticisi adını kullanarak birden çok kez bağlantı kullanan istemci uygulamalarının aynı istemci kanalını kullanıp kullanmadığını belirler.

Birden çok geçerli kanal tanımlaması kullanılabilir olduğunda bu özneliği kullanın.

Değer aşağıdakilerden biridir:

#### **MQCAFTY\_PREFERDI**

İstemci kanal tanımlama çizelgesini (CCDT) okuyan bir işlemdeki ilk bağlantı, ilk ve alfabetik sırayla, geçerli tüm CLNTWGHT (0) tanımlamalarıyla ağırlıklandırma esasına dayalı olarak geçerli tanımların bir listesini oluşturur. Süreçteki her bağlantı, listedeki ilk tanımlamayı kullanarak bağlanmayı dener. Bir bağlantı başarısız olursa, sonraki tanım kullanılır. 0 dışındaki CLNTWGHT değerlerine sahip başarısız tanımlamalar, listenin sonuna taşınır. CLNTWGHT (0) tanımlamaları, listenin başında kalır ve her bağlantı için ilk olarak seçilir.

Her istemci işlemi, aynı anasistem adına sahip her zaman aynı listeyi yaratır.

For client applications written in C, C++, or the .NET programming framework (including fully managed .NET) the list is updated if the CCDT has been modified since the list was created.

Bu değer, varsayılan değerdir.

#### **MQCAFTY\_NONE**

CCDT 'i okuyan bir işlemdeki ilk bağlantı, geçerli tanımların bir listesini oluşturur. Bir süreçteki tüm bağlantılar, alfabetik sırayla ilk olarak seçilen geçerli tüm CLNTWGHT (0) tanımlarıyla ağırlıklandırma esasına dayalı olarak geçerli bir tanımlı seçer.

For client applications written in C, C++, or the .NET programming framework (including fully managed .NET) the list is updated if the CCDT has been modified since the list was created.

Bu öznelik, yalnızca istemci-bağlantı kanalı tipi için geçerlidir.

Bu, çıkışa giriş alanıdır. *Sürüm* , MQCD\_VERSION\_9 değerinden küçükse, alan yoktur.

#### *ConnectionName (MQCHAR264)*

Bu alan, kanala ilişkin bağlantı adını belirtir.

Küme alıcı kanalları için (belirtildiğinde) CONNAME, yerel kuyruk yöneticisiyle ve diğer kanallarda hedef kuyruk yöneticisiyle ilişkilendirir. Belirlediğiniz değer, kullanılacak iletim protokolünün (*TransportType*) bağlı olduğu değere bağlıdır:

- MQXPT\_LU62 için, iş ortağı Mantıksal Biriminin tam olarak nitelenmiş adıdır.
- MQXPT\_NETBIOS için, uzak makinede tanımlanan NetBIOS adıdır.
- MQXPT\_TCP için, anasistem adı, IPv4 noktalı onlu ya da IPv6 onaltılı biçiminde belirtilen uzak makinenin ağ adresi ya da küme-alıcı kanallarına ilişkin yerel makine.
- MQXPT\_SPX için, 4 baytlık bir ağ adresi, 6 baytlık bir düğüm adresi ve 2 baytlık yuva numaralarından oluşan SPX stili bir adrestir.

When defining a channel, this field is not relevant for channels with a *ChannelType* of MQCHT\_SVRCONN or MQCHT\_RECEIVER. Ancak, kanal tanımlaması bir çıkışa geçirildiğinde, bu alan kanal tipi ne olursa olsun, ortağın adresini içerir.

Bu alanın uzunluğu MQ\_CONN\_NAME\_LENGTH değeri tarafından verilir. *Version* , MQCD\_VERSION\_2 değerinden küçükse bu alan yoktur.

#### *DataConversion (MQUZE)*

Bu alan, alıcı ileti kanalı aracısının bu dönüşümü gerçekleştirmediğinde, ileti kanalı aracısının uygulama ileti verilerini dönüştürmeye çalışamayacağını belirler.

Bu alan yalnızca mantıksal iletilerin kesimleri olmayan iletiler için geçerlidir; MCA hiçbir zaman bölüm olan iletileri dönüştürme girişiminde bulunmaz.

Bu alan yalnızca, *ChannelType* MQCMT\_ENDER, MQCMT\_SERVER, MQCMT\_CLUUSDR ya da MQCMT\_CLAUSRCVR ile olan kanallar için geçerlidir. Bu, aşağıdakilerden biridir:

#### **MQCDC\_SENDER\_CONVERSION**

Gönderene göre dönüştürme.

#### **MQCDC\_NO\_SENDER\_CONVERSION**

Gönderen tarafından dönüştürme yok.

#### *DefReconnect ( MQLONG)*

DefReconnect kanal özneliği, bir istemci bağlantı kanalına ilişkin varsayılan yeniden bağlantı özneliği değerini ayarlar.

Varsayılan otomatik istemci yeniden bağlanma seçeneği. Bir istemci uygulamasını otomatik olarak yeniden bağlamak için IBM MQ MQI client yapılandırabilirsiniz. IBM MQ MQI client , bir bağlantı hatasından sonra bir kuyruk yöneticisine yeniden bağlanmayı dener. Uygulama istemcisi bir MQCONN ya da MQCONNX MQI çağrısı yayınlamadan yeniden bağlanmayı dener.

Yeniden bağlanma bir MQCONNX seçeneğidir. DefReconnect kanal özneliğini kullanarak, MQCONNkullanan var olan uygulamalara yeniden bağlantı davranışı ekleyebilirsiniz. MQCONNXkullanan uygulamaların yeniden bağlanma davranışını da değiştirebilirsiniz.

Yeniden bağlanma davranışını ayarlamak ya da değiştirmek için mqclient . ini dosyasından DefRecon değerini de ayarlayabilirsiniz. mqclient . ini dosyasındaki DefRecon değeri, DefReconnect kanal özneliğinden önceliklidir.

## **Syntax**

**DefReconnect** ( MQRCN\_NO (default) |MQRCN\_YES|MQRCN\_Q\_MGR|MQRCN\_DISABLED )

## **Parametreler**

### **MQRCN\_NO**

MQRCN\_NO varsayılan değerdir.

MQCONNXtarafından geçersiz kılınmadıkça, istemci otomatik olarak yeniden bağlanmaz.

### **MQRCN\_YES**

MQCONNXtarafından geçersiz kılınmadıkça, istemci otomatik olarak yeniden bağlanır.

### **MQRCN\_Q\_MGR**

MQCONNXtarafından geçersiz kılınmadıkça, istemci otomatik olarak yeniden bağlanır, ancak yalnızca aynı kuyruk yöneticisine bağlanır. QMGR seçeneği, MQCNO\_RECONNECT\_Q\_MGRile aynı etkiye sahiptir.

### **MQRCN\_DISABLED**

MQCONNX MQI çağrısı kullanılarak istemci programı tarafından istense de, yeniden bağlanma geçersiz kılır.

IBM MQ classes for Javaotomatik istemci yeniden bağlantısını desteklemez.

<i>Çizelge 823. Otomatik yeniden bağlanma, uygulamada ve kanal tanımında ayarlanan değerlere bağlıdır</i>				
<b>DefReconnect</b>	<b>Uygulamada ayarlanan yeniden bağlanma seçenekleri</b>			
	MQCNO_RECONNE CT	MQCNO_RECONNE CT_Q_MGR	MQCNO_RECONNE CT_AS_DEF	MQCNO_RECONNE CT_DISABLED
MQRCN_NO	EVET	QMGR	HAYIR	HAYIR
MQRCN_YES	EVET	QMGR	EVET	HAYIR
MQRCN_Q_MGR	EVET	QMGR	QMGR	HAYIR
MQRCN_DISABLED	HAYIR	HAYIR	HAYIR	HAYIR

## İlgili kavramlar

[Otomatik istemci yeniden bağlantısı](#)

[Kanal ve istemci yeniden bağlantısı](#)

## İlgili başvurular

[İstemci yapılandırma dosyasının KANAL kısmı](#)

[Bağlanma seçenekleri](#)

MQCONNX 'in işlemini denetleyen seçenekler.

### Tanım (MQCHAR64)

Bu alan açıklayıcı açıklamalar için kullanılabilir.

Alanın içeriği, İleti Kanalı Aracıları için önem göstermiyor. Ancak, yalnızca görüntülenebilen karakterler içermeli. Boş değerli karakterler içeremez; gerekirse, boşlukla doldurulabilir. Bir DBCS kuruluşunda, alan DBCS karakterleri içerebilir (en çok 64 baytlık alan uzunluğuna tabi).

**Not:** Bu alan, kuyruk yöneticisinin karakter kümesinde olmayan karakterler içeriyorsa ( **CodedCharSetId** kuyruk yöneticisi özniteliğiyle tanımlandığı gibi), bu alan başka bir kuyruk yöneticisine gönderilirse, bu karakterler yanlış çevrilebilir.

Bu alanın uzunluğu MQ\_CHANNEL\_DESC\_LENGTH tarafından verilmektedir.

### DiscInterval (MQUZE)

Bu alan, kanalın, kanalı sonlandırmadan önce iletim kuyruğuna gelmesi için bekleyeceği sürenin saniye cinsinden üst sınırını belirler.

Diğer bir deyişle, bağlantı kesme aralığını belirtir.

Sıfır değeri, MCA ' nın süresiz olarak beklemesine neden olur.

TCP iletişim kuralını kullanan sunucu bağlantısı kanalları için aralık, saniye olarak belirtilen istemci boşa durma bağlantı kesme değerini gösterir. Bir sunucu bağlantısı, bu süre için iş ortağı istemcisinden iletişim alamazsa, bağlantıyı sona erdirir. Sunucu bağlantısı boşa durma aralığı, bir istemciden gelen IBM MQ API çağrıları arasında geçerlidir; bu nedenle, uzun süredir çalışan bir MQGET işlemi bekleme çağrısıyla hiçbir istemci bağlantısı kesilmez.

Bu öznitelik, TCP dışındaki protokolleri kullanan sunucu bağlantısı kanalları için geçerli değildir.

Bu alan yalnızca, *ChannelType* MQCHT\_ENDER, MQCHT\_SERVER, MQCHT\_CLAUSSDR, MQCHT\_CLAUSRCVR ya da MQCHT\_SVRCONN ile kanallar için anlamlıdır.

### ExitDataUzunluk (MQUZE)

Bu alan, *MsgUserDataPtr*, *SendUserDataPtr* ve *ReceiveUserDataPtr* alanları tarafından adreslenen çıkış kullanıcı veri ögeleri listelerindeki her bir kullanıcı veri ögesinin bayt cinsinden uzunluğunu belirtir.

Bu uzunluğun MQ\_EXIT\_DATA\_LENGTH ile aynı olması gerekmez.

Bu, çıkışa giriş alanıdır. *Version* değeri MQCD\_VERSION\_4değerinden küçükse bu alan yoktur.

### ExitNameUzunluk (MQUZE)

Bu alan, *MsgExitPtr*, *SendExitPtr* ve *ReceiveExitPtr* alanları tarafından adreslenen çıkış adları listelerindeki her bir adın uzunluğunu bayt cinsinden belirtir.

Bu uzunluğun, MQ\_EXIT\_NAME\_LENGTH ile aynı uzunlukta olması gerekmez.

Bu, çıkışa giriş alanıdır. *Version* değeri MQCD\_VERSION\_4değerinden küçükse bu alan yoktur.

### HdrCompListe [ 2] (MQUZE)

Bu alan, kanal tarafından desteklenen üstbilgi veri sıkıştırma tekniklerinin listesini belirtir.

Liste, aşağıdaki değerlerden birini ya da birkaçını içerir:

### MQCOMPRESS\_NONE

Üstbilgi veri sıkıştırma işlemi gerçekleştirilmedi.

## **MQCOMPRESS\_SYSTEM**

Üstbilgi veri sıkıştırması gerçekleştirilir.

Dizideki kullanılmayan değerler MQCOMPRESS\_NOT\_AVAM değerine ayarlanır.

Bu, çıkışa giriş alanıdır. *Version* değeri MQCD\_VERSION\_8değerinden küçükse bu alan yoktur.

### *HeartbeatInterval (MQUZE)*

Bu alan, sağlıklı işletim bildirim akışları arasındaki süre/saniye cinsinden belirtir.

Bu alanın yorumlanması kanal tipine bağlıdır; aşağıdaki gibi:

- Kanal tipi için (MQCHT\_SENDER, MQCHT\_SERVER, MQCHT\_RECEIVER MQCHT\_REQUESTER, MQCHT\_CLUSSDR ya da MQCHT\_CLAUSRCVR), bu alan, iletim kuyruğunda ileti olmadığı, gönderen MCA ' dan geçirilen sağlıklı işletim bildirim akışları arasındaki süre (saniye). Bu işlem, MCA ' yı kanalı susturma fırsatı verir. Yararlı olmak için *HeartbeatInterval* , *DiscInterval* değerinden küçük olmalıdır.
- MQCHT\_CLNTCONN ya da MQCHT\_SVRCONN 'ın bir kanal tipi için, MQCD Paylaşımı Conversations alanı sıfır olarak ayarlıdır; bu alan, MCA' nın bir istemci uygulaması adına MQGMO\_WEKE seçeneğiyle MQGET çağrısı yayınladığında, sunucu MCA ' dan geçirilen sağlıklı işletim bildirim akışları arasındaki süre (saniye). Bu, sunucu MCA ' nın MQGMO\_WATE ile MQGET işlemi sırasında istemci bağlantısının başarısız olduğu durumları işlemlerini sağlar.
- MQCHT\_CLNTCONN ya da MQCHT\_SVRCONN ' ın bir kanal tipi için, MQCD Paylaşımı Conversations alanı sıfır dışında bir değere ayarlanıyor; bu alan, gönderilen ya da alınan veri akımları olmadığı, sağlıklı işletim bildirim akışı arasındaki süre/saniye cinsinden süredir. Bu, kanalın verimli bir şekilde susturulmasını sağlar.

Değer, 0 ile 999 999 arasında bir aralıktır. Kullanılan değer, her iki tarafta da 0 değeri belirlenmediği sürece, gönderme tarafında ve giriş tarafında belirtilen değerlerin daha büyük olması durumunda, herhangi bir sinyal alışverişi gerçekleştirilmez.

Bu, çıkışa giriş alanıdır. *Version* değeri MQCD\_VERSION\_4değerinden küçükse bu alan yoktur.

### *KeepAliveAralığı (MQUZE)*

Bu alan, kanala ilişkin canlı tutma zamanlaması için iletişim yığınına geçirilen değeri belirtir.

Bu değer, tüm somutlamalar tarafından desteklenmese de, bu değer TCP/IP ve SPX iletişim protokolleri için geçerlidir.

Değer, 0-99 999 aralığında; birimlerin sayısı saniye cinsinden. Sıfır değeri, kanal canlı tutma özelliğinin etkinleştirilmediğini gösterir; ancak, canlı tutma (keepalive), TCP/IP canlı tutma (keepalive yerine) (kanal canlı tutma (keepalive yerine)) etkinleştirilmişse de (keepalive) gerçekleşebilir. Aşağıdaki özel değer geçerlidir:

## **MQKI\_AUTO**

Otomatik.

Bu değer, canlı tutma aralığının, kararlaştırılan sağlıklı işletim bildirim aralığından aşağıdaki gibi hesaplandığı anlamına gelir:

- Kararlaştırılan sağlıklı işletim bildirim aralığı sıfırdan büyükse, kullanılan canlı tutma aralığı, sağlıklı işletim bildirim aralığı artı 60 saniyedir.
- Kararlaştırılan sağlıklı işletim bildirim aralığı sıfırsa, kullanılan canlı tutma aralığı sıfırdır.
- z/OSüzerinde, kuyruk yöneticisi nesnesinde TCPKEEP (YES) belirtildiğinde TCP/IP canlı tutma işlemi gerçekleştirilir.
- Diğer ortamlarda, TCP/IP canlı tutma işlemi, dağıtılmış kuyruğa alma yapılandırma dosyasındaki TCP stanza içinde **KEEPALIVE=YES** değiştirgesi belirtildiğinde gerçekleşir.

Bu alan yalnızca MQXPT\_TCP ya da MQXPT\_SPX *TransportType* içeren kanallar için anlamlıdır.

Bu, çıkışa giriş alanıdır. *Version* değeri MQCD\_VERSION\_7değerinden küçükse bu alan yoktur.

### *LocalAddress (MQCHAR48)*

Bu alan, giden iletişim için kanal için tanımlanan yerel TCP/IP adresini belirtir.

Giden iletişim için belirli bir adres tanımlanmadıysa, bu alan boştur. Adres isteğe bağlı olarak bir kapı numarası ya da kapı numarası aralığı içerebilir. Bu adresin biçimi şöyledir:

```
[ip-addr][(low-port[,high-port])]
```

where square brackets ( [ ] ) denote optional information, *ip-addr* is specified in IPv4 dotted decimal, IPv6 hexadecimal, or alphanumeric form, and *low-port* and *high-port* are port numbers enclosed in parentheses. Her şey isteğe bağlıdır.

Giden iletişim için belirli bir IP adresi, kapı ya da kapı aralığı, bir kanalın farklı bir TCP/IP yığnında yeniden başlatıldığı kurtarma senaryolarında kullanışlıdır.

*LocalAddress* is similar in form to *ConnectionName*, but must not be confused with it. *LocalAddress* specifies the characteristics of the local communications, whereas *ConnectionName* specifies how to reach a remote queue manager.

**V9.2.2** **V9.2.0.2** From Long Term Support için IBM MQ 9.2.0 Fix Pack 2 ve Continuous Delivery için IBM MQ 9.2.2, the Java Message Queueing Interface (JMQUI) has been updated to ensure that the local address field is set on an MQCD object after a channel instance has been created and is connected to a queue manager. Bu, Java ' ta yazılan bir kanal çıkışı MQCD. *getLocalAddress()* yöntemini çağırdığında, yöntemin, kanal örneğinin kullandığı yerel adresi döndüreceği anlamına gelir. IBM MQ 9.2.0 Fix Pack 2 ve IBM MQ 9.2.2 öncesinde kanal güvenliği çıkışı, kanal yönetim ortamı tarafından kullanılmakta olan yerel adrese erişemedi ve MQCD. *getLocalAddress()* yöntemi boş değer döndürdü.

Bu alan yalnızca, bir MQXPT\_TCP *TransportType* olan kanallar ve *ChannelType* MQCHT\_SENDER, MQCHT\_SERVER, MQCHT\_REQUESTER, MQCHT\_CLNTCONN, MQCHT\_CLUSSDR ya da MQCHT\_CLAUSRCVR kanalları için geçerlidir.

Bu alanın uzunluğu MQ\_LOCAL\_ADDRESS\_LENGTH tarafından verilir. *Version*, MQCD\_VERSION\_7 değerinden küçükse bu alan yoktur.

### *LongMCAUserIdLength (MQUZE)*

Bu alan, *LongMCAUserIdPtr* tarafından işaret edilen tam MCA kullanıcı kimliğinin bayt cinsinden uzunluğunu belirtir.

This field is not relevant for channels with a *ChannelType* of MQCHT\_CLNTCONN.

Bu, çıkışa giriş/çıkış alanıdır. *Version* değeri MQCD\_VERSION\_6 değerinden küçükse bu alan yoktur.

### *LongMCAUserIdPtr (MQPTR)*

Bu alan, uzun MCA kullanıcı tanımlayıcının adresini belirtir.

*LongMCAUserIdLength* sıfırdan büyükse, bu alan tam MCA kullanıcı kimliğinin adresidir. Tam tanıtıcının uzunluğu *LongMCAUserIdLength* tarafından verilir. The first 12 bytes of the MCA user identifier are also contained in the field *MCAUserIdentifier*.

MCA kullanıcı tanımlayıcıyla ilgili ayrıntılar için *MCAUserIdentifier* alanının açıklamasına bakın.

This field is not relevant for channels with a *ChannelType* of MQCHT\_SDR, MQCHT\_SVR, MQCHT\_CLNTCONN, or MQCHT\_CLUSSDR.

Bu, çıkışa giriş/çıkış alanıdır. *Version* değeri MQCD\_VERSION\_6 değerinden küçükse bu alan yoktur.

### *LongRemoteUserIdUzunluğu (MQUZE)*

Bu alan, *LongRemoteUserIdPtr* tarafından işaret edilen tam uzak kullanıcı kimliğinin bayt cinsinden uzunluğunu belirtir.

Bu alan yalnızca, bir MQCHT\_CLNTCONN ya da MQCHT\_SVRCONN *ChannelType* içeren kanallar için anlamlıdır.

Bu, çıkışa giriş alanıdır. *Version* değeri MQCD\_VERSION\_6değerinden küçükse bu alan yoktur.

#### *LongRemoteUserIdPtr (MQPTR)*

Bu alan, uzun uzak kullanıcı tanıtıcının adresini belirtir.

*LongRemoteUserIdLength* sıfırdan büyükse, bu işaret tam uzak kullanıcı kimliğinin adresidir. Tam tanıtıcının uzunluğu *LongRemoteUserIdLength* tarafından verilir. Uzak kullanıcı kimliğinin ilk 12 baytı da *RemoteUserIdentifier* alanında yer alır.

Uzak kullanıcı tanıtıcının ayrıntıları için *RemoteUserIdentifier* alanının açıklamasına bakın.

Bu alan yalnızca, bir MQCHT\_CLNTCONN ya da MQCHT\_SVRCONN *ChannelType* içeren kanallar için anlamlıdır.

Bu, çıkışa giriş alanıdır. *Version* değeri MQCD\_VERSION\_6değerinden küçükse bu alan yoktur.

#### *LongRetryCount (MQLONG)*

Bu alan, *ShortRetryCount* tarafından belirtilen sayıdan sonra kullanılan sayıyı belirtir.

İşletmene bir hata kaydedilmeden önce, *LongRetryInterval* tarafından belirlenen aralıklarda, uzak makineye bağlanmak için yapılan deneme sayısı üst sınırını belirler.

Bu alan yalnızca, *ChannelType* MQCHT\_ENDER, MQCHT\_SERVER, MQCHT\_CLUUSSDR ya da MQCHT\_CLAUSRCVR ile olan kanallar için geçerlidir.

#### *LongRetryAralığı (MQUZE)*

Bu alan, uzak makineden yeniden bağlantı kurulmadan önce beklenecek süre üst sınırını saniye cinsinden belirtir.

Kanal etkin olmak için beklemesi gerekiyorsa, yeniden denemeler arasındaki aralık uzatılabilir.

Bu alan yalnızca, *ChannelType* MQCHT\_ENDER, MQCHT\_SERVER, MQCHT\_CLUUSSDR ya da MQCHT\_CLAUSRCVR ile olan kanallar için geçerlidir.

#### *MaxInstances (MQUZE)*

Bu alan, tek bir sunucu bağlantısı kanalının başlatılabilecek koşut zamanlı eşgörünümlerine ilişkin üst sınır sayısını belirtir.

Bu alan yalnızca sunucu-bağlantı kanallarında kullanılır.

Alan 0-999 999 999 aralığında bir değere sahip olabilir. Sıfır değeri, tüm istemci erişimini engeller.

Bu alanın varsayılan değeri 999 999 999 numaradır.

Bu alanın değeri, çalışmakta olan sunucu bağlantı kanalının eşgörünümlerinden az olan bir sayıya indirgenirse, çalışmakta olan eşgörünümler etkilenmez. Ancak, çalışmakta olan eşgörünümlerin sayısının alanın değerinden küçük olması için, var olan eşgörünümlerin çalıştırılması yeterli oluncaya kadar yeni yönetim ortamları başlatılamaz.

#### *MaxInstancesPerClient (MQUBE)*

Bu alan, tek bir istemciden başlatılabilecek tek bir sunucu bağlantısı kanalının eş zamanlı eşgörünümlerinin maksimum sayısını belirtir.

Bu bağlamda, aynı uzak ağ adresinden kaynaklanan bağlantılar aynı istemciden gelmekte kabul edilir.

Bu alan yalnızca sunucu-bağlantı kanallarında kullanılır.

Alan 0-999 999 999 aralığında bir değere sahip olabilir. Sıfır değeri, tüm istemci erişimini engeller.

Bu alanın varsayılan değeri 999 999 999 numaradır.

Bu alanın değeri, tek tek istemcilerden çalışmakta olan sunucu bağlantısı kanalının eşgörünümlerinden az olan bir sayıya indirgenirse, çalışmakta olan eşgörünümler etkilenmez. Ancak, bu istemcilerin herhangi birinden yeni eşgörünümler, var olan eşgörünümlerin sayısı yeterli oluncaya kadar başlayamaz. Bu tür,

çalışmakta olan eşgörünümlerin sayısı, istemciden yeni bir başlangıç girişiminde bulunmaya çalışılan eşgörünümler, alanın değerinden küçük olur.

#### *MaxMsgUzunluğu (MQUZE)*

Bu alan, kanalda iletilebilecek ileti uzunluğu üst sınırını belirtir.

Bu, uzak kanala ilişkin değerle karşılaştırılır ve gerçek üst sınır iki değer alt sınırıdır.

#### *MCAAdı (MQCHAR20)*

Bu alan ayrılmış bir alandır.

Bu alanın değeri boş.

Bu alanın uzunluğu MQ\_MCA\_NAME\_LENGTH değeri tarafından verilir.

#### *MCASecurityId (MQBYTE40)*

Bu alan, MCA ' ya ilişkin güvenlik tanıtıcısını belirtir.

This field is not relevant for channels with a *ChannelType* of MQCHT\_CLNTCONN.

Aşağıdaki özel değer, güvenlik tanıtıcısı olmadığını gösterir:

#### **MQSID\_NONE**

Güvenlik tanıtıcısı belirtilmedi.

Değer, alanın uzunluğuna ilişkin ikili sıfır değeridir.

C programlama dili için, constant MQSID\_NONE\_ARRAY de tanımlanır; bu değişmez, MQSID\_NONE ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisi olur.

Bu, çıkışa giriş/çıkış alanıdır. Bu alanın uzunluğu MQ\_SECURITY\_ID\_LENGTH tarafından verilmektedir. *Version* , MQCD\_VERSION\_6değerinden küçükse bu alan yoktur.

#### *MCAType (MQHOUT)*

Bu alan, ileti kanalı aracısı programının tipini belirtir.

This field is relevant only for channels with a *ChannelType* of MQCHT\_SENDER, MQCHT\_SERVER, MQCHT\_REQUESTER, MQCHT\_CLUSSDR, or MQCHT\_CLUSRCVR.

Değer aşağıdakilerden biridir:

#### **MQMCAT\_PROCESI**

İşlem.

İleti kanalı aracısı ayrı bir işlem olarak çalışır.

#### **MQMCAT\_THREAD**

İş parçacığı (Multiplatforms).

İleti kanalı aracısı ayrı bir iş parçacığı olarak çalışır.

*Sürüm* , MQCD\_VERSION\_2değerinden küçükse bu alan yoktur.

#### *MCAUserIdentifier (MQCHAR12)*

Bu alan, ileti kanalı aracısına (MCA) ilişkin kullanıcı kimliğini belirtir.

Bu alan MCA kullanıcı tanıtıcısının ilk 12 baytı kullanır ve bir güvenlik aracısıyla ayarlanabilir.

MCA kullanıcı tanımlayıcıyla ilgili iki alan vardır:

- *MCAUserIdentifier* , MCA kullanıcı tanıtıcısının ilk 12 baytı içerir ve tanıtıcı 12 bayttan kısaysa boşluklarla doldurulur. *MCAUserIdentifier* boş olabilir.
- *LongMCAUserIdPtr* , 12 bayttan uzun olabilen tam MCA kullanıcı tanımlayıcısından işaret eder. Uzunluğu *LongMCAUserIdLength* ile verilir. Tam tanıtıcı sonda boşluk içermiyor ve boş değerle sonlandırılmadı. Tanıtıcı boşsa, *LongMCAUserIdLength* sıfırdır ve *LongMCAUserIdPtr* değeri tanımsız olur.

**Not:** *Version* , MQCD\_VERSION\_6değerinden küçükse *LongMCAUserIdPtr* yoktur.

If the MCA user identifier is nonblank, it specifies the user identifier to be used by the message channel agent for authorization to access IBM MQ resources. MQCHT\_REQUESTER, MQCHT\_RECEIVER ve MQCHT\_CLAUSRCVR kanal tipleri için, PutAuthority MQPA\_XX\_ENCODE\_CASE\_ONE default ise, bu, hedef kuyruklara koyma işlemi için yetki denetimleri için kullanılan kullanıcı tanıtıcısıdır.

MCA kullanıcı kimliği boşsa, ileti kanalı aracısının varsayılan kullanıcı kimliğini kullanır.

MCA kullanıcı tanıtıcısı, ileti kanalı aracısının kullanması gereken kullanıcı kimliğini belirtmek için bir güvenlik çıkışı tarafından ayarlanabilir. Çıkış, *MCAUserIdentifier* ya da *LongMCAUserIdPtr* tarafından gösterilen dizeyi değiştirebilir. If both are changed but differ from each other, the MCA uses *LongMCAUserIdPtr* in preference to *MCAUserIdentifier*. If the exit changes the length of the string addressed by *LongMCAUserIdPtr*, *LongMCAUserIdLength* must be set correspondingly. Çıkış, tanıtıcının uzunluğunu artırıyor, çıkışta gerekli uzunluğun depolama alanını ayırmalı, bu depolamayı gerekli tanıtıcıya ayarlamalıdır ve bu depolamanın adresini *LongMCAUserIdPtr* içine yerleştirmelidir. Çıkış, daha sonra MQXR\_TERM nedeniyle çağrıldığında, bu depolamanın serbest bırakılmasından sorumlu olur.

Bir MQCHT\_SVRCONN *ChannelType* içeren kanallar için, kanal tanımlamasındaki *MCAUserIdentifier* boş bırakılırsa, istemciden aktarılan tüm kullanıcı tanımlayıcısı bu kanala kopyalanır. Bu kullanıcı kimliği (sunucudaki güvenlik çıkışıyla yapılan herhangi bir değişikliğin ardından), istemci uygulamasının çalışmakta olduğu varsayılan bir tanıtıcıdır.

MCA kullanıcı kimliği, *ChannelType* MQCHT\_SDR, MQCHT\_SVR, MQCHT\_CLNTCONN, MQCHT\_CLAUSSDR olan kanallar için uygun değildir.

Bu, çıkışa giriş/çıkış alanıdır. Bu alanın uzunluğu MQ\_USER\_ID\_LENGTH tarafından verilir. *Version*, MQCD\_VERSION\_2 değerinden küçükse bu alan yoktur.

*ModeName (MQCHAR8)*

Bu alan, LU 6.2 kip adını belirtir.

Bu alan, yalnızca iletim protokolü (*TransportType*) MQXPT\_LU62 ise ve *ChannelType* MQCHT\_SVRCONN ya da MQCHT\_RECEIVER değilse anlamlıdır.

Bu alan her zaman boş olur. Bilgi, iletişim Side nesnesinde yer alır.

Bu alanın uzunluğu MQ\_MODE\_NAME\_LENGTH tarafından verilir.

*MsgCompListesi [ 16] (MQUBE)*

Bu alan, kanal tarafından desteklenen ileti veri sıkıştırma tekniklerinin listesini belirtir.

Liste, aşağıdaki değerlerden birini ya da birkaçını içerir:

#### **MQCOMPRESS\_NONE**

İleti veri sıkıştırma işlemi gerçekleştirilmedi.

#### **MQCOMPRESS\_RLE**

İleti veri sıkıştırması, çalıştırma uzunluğu kodlaması kullanılarak gerçekleştirilir.

#### **MQCOMPRESS\_ZLIBFAST**

İleti veri sıkıştırması, zlib sıkıştırma tekniği kullanılarak gerçekleştirilir. Hızlı sıkıştırma süresi tercih edilir.

#### **MQCOMPRESS\_ZLIBHIGH**

İleti veri sıkıştırması, zlib sıkıştırma tekniği kullanılarak gerçekleştirilir. Yüksek düzeyde sıkıştırma tercih edilir.

Dizideki kullanılmayan değerler MQCOMPRESS\_NOT\_AVAM değerine ayarlanır.

Bu, çıkışa giriş alanıdır. *Version* değeri MQCD\_VERSION\_8 değerinden küçükse bu alan yoktur.

*MsgExit (MQCHARn)*

Bu alan kanal ileti çıkış adını belirtir.

Bu ad boş değilse, çıkış şu saatlerde çağrılır:



- İletim kuyruğundan (gönderen ya da sunucu) bir ileti alındıktan hemen sonra ya da bir ileti hedef kuyruğuna (alıcı ya da istekte bulunan) alınmadan hemen önce.

Çıkışa, değişiklik için tüm uygulama iletisi ve iletim kuyruğu üstbilgisi verilir.

- Kanalın kullanıma hazırlanması ve sona erdirilmesi.

This field is not relevant for channels with a *ChannelType* of MQCMT\_SVRCONN or MQCMT\_CLNTCONN; a message exit is never invoked for such channels.

Çeşitli ortamlardaki bu alanın içeriğine ilişkin açıklamalar için [“MQCD-Kanal tanımlaması” sayfa 1456](#) ' e bakın.

Bu alanın uzunluğu MQ\_EXIT\_NAME\_LENGTH değeri tarafından verilir.

**Not:** Bu değişimin değeri ortamdan özgüdür.

#### *MsgExitPtr (MQPTR)*

Bu alan, ilk *MsgExit* alanının adresini belirtir.

*MsgExitsDefined* sıfırdan büyükse, bu adres zincirdeki her bir kanal iletisi çıkışının adları listesinin adresidir.

Her ad, *ExitNameLength* uzunluktaki bir alanda, boşluklarla doldurulmuşta doldurulur. Her bir çıkış için birden bire bir *MsgExitsDefined* alanı bitişik olan fields alanları vardır.

Bir çıkış tarafından bu adlarda yapılan değişiklikler korunur, ancak ileti kanalı çıkışı belirttik bir işlem yapmasa da, hangi çıkışlar çağrıldığı değişmez.

*MsgExitsDefined* sıfırsa, bu alan boş değerli göstergedir.

Programlama dilinin işaretçi veri tipini desteklemediği altyapılarda, bu alan uygun uzunluğun byte dizgisi olarak bildirilir.

Bu, çıkışa giriş alanıdır. *Version* değeri MQCD\_VERSION\_4 değerinden küçükse bu alan yoktur.

#### *MsgExitsTanımlı (MQUSEK)*

Bu alan, zincirde tanımlanan kanal ileti çıkışlarının sayısını belirtir.

Sıfır değerinden büyük ya da sıfıra eşit.

Bu, çıkışa giriş alanıdır. *Version* değeri MQCD\_VERSION\_4 değerinden küçükse bu alan yoktur.

#### *MsgRetryCount (MQLONG)*

Bu alan, ilk deneme başarısız olduktan sonra MCA ' nın iletiyi ne zaman yerleştirmeyi denediğini belirtir.

Bu alan, MQOPED ya da MQPUT ilk tamamlanma kodu MQCC\_FAILED ile başarısız olursa, MCA ' nın açma ya da koyma işlemini denemesinin kaç kez denediğini belirtir. Bu özneliğin etkisi, *MsgRetryExit* boş mu yoksa boş mu değil mi:

- *MsgRetryExit* boş bırakılırsa, **MsgRetryCount** özneliği MCA ' nın yeniden deneme girişiminde bulunmayacağını denetler. Öznelik değeri sıfırsa, yeniden deneme girişiminde bulunulmaz. Öznelik değeri sıfırdan büyükse, yeniden denemeler, **MsgRetryInterval** özneliği tarafından verilen aralıklarda denenir.

Yeniden denemeler, yalnızca aşağıdaki neden kodları için denenir:

- MQRC\_PAGEET\_FULL
- MQRC\_PUT\_INHIBITED
- MQRC\_Q\_FULL

Diğer neden kodları için MCA, başarısız olan iletiyi yeniden denemeden, hemen olağan arızalarına devam eder.

- *MsgRetryExit* boş değilse, **MsgRetryCount** özneliği MCA ' yı etkilemez; bunun yerine, yeniden deneme işleminin kaç kez denendiğini ve hangi aralıklarda olduğunu belirleyen ileti yeniden deneme çıkışıdır; **MsgRetryCount** özneliği sıfır olsa da çıkış çağrılır.

The **MsgRetryCount** attribute is made available to the exit in the MQCD structure, but the exit it not required to honor it - retries continue indefinitely until the exit returns MQXCC\_SUPPRESS\_FUNCTION in the *ExitResponse* field of MQCXP.

Bu alan yalnızca, *ChannelType* MQCHT\_REQUESTER, MQCHT\_RECEIVER ya da MQCHT\_CLAUSRCVR ile olan kanallar için anlamlıdır.

*Version*, MQCD\_VERSION\_3değerinden küçükse bu alan yoktur.

#### *MsgRetryExit (MQCHARn)*

Bu alan, kanal iletisi yeniden deneme çıkış adını belirtir.

İleti yeniden deneme çıkışı, MCA ' nın MQOPED ya da MQPUT çağrısından MQCC\_FAILED tamamlanma kodu aldığıında, MCA tarafından çağrılan bir çıkışıdır. Çıkışa ilişkin amaç, MCA ' nın MQOL ya da MQPUT işlemini yeniden denemeden önce bekleyeceği bir zaman aralığı belirtmendir. Diğer bir seçenek olarak, çıkış, işlemi yeniden denememek için ayarlanabilir.

Çıkış, tamamlanma kodu MQCC\_FAILED olan tüm neden kodları için çağrılır-çıkışa ilişkin ayarlar, hangi neden kodlarının MCA ' nın yeniden denenmesini istediğini, kaç deneme denemesini ve hangi zaman aralıklarında çalışacağını belirler.

İşlem daha fazla denilemediğinde, MCA olağan başarısızlık işlemlerini gerçekleştirir; bu işlem, bir kural dışı durum raporu iletisi (gönderen tarafından belirtilmişse) oluşturulmasını ve özgün iletisi ölüme mektup kuyruğuna yerleştirmesini ya da iletinin (gönderenin MQRO\_DEAD\_LETTER\_Q ya da MQRO\_DISCARD\_MSG belirtmesine göre) atılacağını içerir. Ölü-mektup kuyruğunu içeren hatalar (örneğin, ölü-mektup kuyruğu dolu), iletinin yeniden deneme çıkışısının çağrılmasına neden olmaz.

Çıkış adı boş değilse, çıkış şu saatlerde çağrılır:

- Bir iletisi yeniden teslim etme girişiminde bulunmadan önce beklemeyi gerçekleştirmeden hemen önce
- Kanalın kullanıma hazırlanması ve sona erdirilmesi

Çeşitli ortamlardaki bu alanın içeriğine ilişkin açıklamalar için [“MQCD-Kanal tanımlaması” sayfa 1456 ' e](#) bakın.

Bu alan yalnızca, *ChannelType* MQCHT\_REQUESTER, MQCHT\_RECEIVER ya da MQCHT\_CLAUSRCVR ile olan kanallar için anlamlıdır.

Bu alanın uzunluğu MQ\_EXIT\_NAME\_LENGTH değeri tarafından verilir.

**Not:** Bu değışmezin değeri ortamdan özgüdür.

*Version*, MQCD\_VERSION\_3değerinden küçükse bu alan yoktur.

#### *MsgRetryInterval (MQHOT)*

Bu alan, açma ya da koyma işleminin yeniden denenmesi için geçen milisaniye cinsinden aralık alt sınırını belirtir.

Bu özniteliğin etkisi, *MsgRetryExit* boş mu yoksa boş mu değil mi:

- *MsgRetryExit* boşsa, **MsgRetryInterval** özniteliği MCA ' nın bir iletisi yeniden denemeden önce bekleyeceği minimum süreyi belirtir; ilk MQOUT ya da MQPUT işlemi tamamlanma kodu MQCC\_FAILED ile başarısız olur. Sıfır değeri, önceki denemeden sonra yeniden deneme işleminin en kısa sürede gerçekleştirileceği anlamına gelir. Yeniden denemeler, *MsgRetryCount* sıfırdan büyükse gerçekleştirilir.

Bu öznitelik, iletinin yeniden deneme çıkışı MQCXP ' deki *MsgRetryInterval* alanında geçersiz bir değeri döndürürse bekleme süresi olarak da kullanılır.

- *MsgRetryExit* boş değilse, **MsgRetryInterval** özniteliği MCA ' yı etkilemez; bunun yerine, MCA ' nın ne kadar bekleyeceğini belirleyen, iletisiyle yeniden deneme çıkışıdır. The **MsgRetryInterval** attribute is made available to the exit in the MQCD structure, but the exit it not required to honor it.

Değer, 0 ile 999 999 999 aralığında olmalıdır.

Bu alan yalnızca, *ChannelType* MQCHT\_REQUESTER, MQCHT\_RECEIVER ya da MQCHT\_CLAUSRCVR ile olan kanallar için anlamlıdır.

*Version* , MQCD\_VERSION\_3değerinden küçükse bu alan yoktur.

*Version* değeri MQCD\_VERSION\_4değerinden küçükse, bu yapıdaki şu alanlar yoktur.

#### *MsgRetryUserData (MQCHAR32)*

Bu alan, kanal iletisi yeniden deneme çıkış kullanıcı verilerini belirtir.

Bu veriler kanal iletisine geçirilir- **ChannelExitParms** parametresinin *ExitData* alanındaki yeniden deneme çıkışıdır (MQ\_CHANNEL\_EXIT 'e bakın).

Başlangıçta bu alan, kanal tanımlamasında belirlenen verileri içerir. Ancak, bu MCA eşgörünümünün geçerlilik süresi boyunca, bu alanın herhangi bir çıkıştan çıkılarak bu alanın içeriğinde yapılan değişiklikler MCA tarafından korunur ve bu MCA eşgörünümü için sonraki çıkışlar (tipten bağımsız olarak) tarafından görülebilir olarak görünür. Bu tür değişiklikler, diğer MCA eşgörünümleri tarafından kullanılan kanal tanımlamasını etkilemez. Herhangi bir karakter (ikili veri de içinde olmak üzere) kullanılabilir.

Bu alan yalnızca, *ChannelType* MQCHT\_REQUESTER, MQCHT\_RECEIVER ya da MQCHT\_CLAUSRCVR ile olan kanallar için anlamlıdır.

Bu alanın uzunluğu MQ\_EXIT\_DATA\_LENGTH tarafından verilir. *Version* , MQCD\_VERSION\_3değerinden küçükse bu alan yoktur.

Bu alan IBM MQ for IBM i ile ilgili değildir.

#### *MsgUserVerileri (MQCHAR32)*

Bu alan, kanal iletisi çıkışı kullanıcı verilerini belirtir.

This data is passed to the channel message exit in the *ExitData* field of the **ChannelExitParms** parameter (see MQ\_CHANNEL\_EXIT).

Başlangıçta bu alan, kanal tanımlamasında belirlenen verileri içerir. Ancak, bu MCA eşgörünümünün geçerlilik süresi boyunca, bu alanın herhangi bir çıkıştan çıkılarak bu alanın içeriğinde yapılan değişiklikler MCA tarafından korunur ve bu MCA eşgörünümü için sonraki çıkışlar (tipten bağımsız olarak) tarafından görülebilir olarak görünür. Bu tür değişiklikler, diğer MCA eşgörünümleri tarafından kullanılan kanal tanımlamasını etkilemez. Herhangi bir karakter (ikili veri de içinde olmak üzere) kullanılabilir.

Bu alanın uzunluğu MQ\_EXIT\_DATA\_LENGTH tarafından verilir.

Bu alan IBM MQ for IBM i ile ilgili değildir.

#### *MsgUserDataPtr (MQPTR)*

Bu alan, ilk *MsgUserData* alanının adresini belirtir.

*MsgExitsDefined* sıfırdan büyükse, bu adres zincirdeki her kanal iletisi çıkışa ilişkin kullanıcı veri öğeleri listesinin adresidir.

Each user data item is in a field of length *ExitDataLength*, padded to the right with blanks. Her bir çıkış için birden bire bir *MsgExitsDefined* alanı bitişik olan fieldsalanları vardır. Tanımlanan kullanıcı veri öğesi sayısı çıkış adlarından küçükse, tanımlanmamış kullanıcı veri öğeleri boşluk olarak ayarlanır. Ters durumda, tanımlanan kullanıcı veri öğeleri sayısı çıkış adlarından büyükse, fazla kullanıcı veri öğeleri yoksayılar ve çıkışa sunulmaz.

Bu değerler üzerinde bir çıkış tarafından yapılan değişiklikler korunur. Bu, bir çıkışa bilgileri başka bir çıkışa geçirmenizi sağlar. Herhangi bir değişiklik üzerinde geçerlilik denetimi gerçekleştirilmez; örneğin, gerekirse ikili veriler bu alanlara yazılabilir.

*MsgExitsDefined* sıfırsa, bu alan boş değerli göstergedir.

Programlama dilinin işaretçi veri tipini desteklemediği altyapılarda, bu alan uygun uzunluğun byte dizgisi olarak bildirilir.

Bu, çıkışa giriş alanıdır. *Version* değeri MQCD\_VERSION\_4değerinden küçükse bu alan yoktur.

#### *NetworkPriority (MQUBE)*

Bu alan, kanala ilişkin ağ bağlantısının önceliğini belirler.

Belirli bir hedefe giden birden çok yol varsa, en yüksek önceliğe sahip yol seçilir. Değer 0 ile 9 arasındaki aralıklarda; 0 en düşük önceliğe sahip olur.

Bu alan yalnızca, bir MQCHT\_CLUSSDR ya da MQCHT\_CLAUSRCVR *ChannelType* olan kanallar için anlamlıdır.

Bu, çıkış girişi alanıdır. *Version* değeri MQCD\_VERSION\_5değerinden küçükse bu alan yoktur.

*Version* değeri MQCD\_VERSION\_6değerinden küçükse, bu yapıdaki şu alanlar yoktur.

#### *NonPersistentMsgSpeed (MQHOT)*

Bu alan, kanaldan kalıcı olmayan iletilerin hangi hızda seyahat ettiği anlamına gelir.

Bu alan yalnızca, *ChannelType* MQCHT\_ENDER, MQCHT\_SERVER, MQCHT\_RECEIVER, MQCHT\_REQUESTER, MQCHT\_CLUSSDR ya da MQCHT\_CLAUSRCVR olan kanallar için anlamlıdır.

Değer aşağıdakilerden biridir:

#### **MQNPMS\_NORMAL**

Normal hız.

Bir kanal MQNPMS\_NORMAL olarak tanımlandıysa, kalıcı olmayan iletiler kanaldan normal hızda seyahat eder. Bu, bir kanal hatası varsa, bu iletilerin kaybedilmemesi avantajına sahiptir. Ayrıca, aynı iletim kuyruğunda kalıcı ve kalıcı olmayan iletiler, sıralarını birbirlerine göre korur.

#### **MQNPMS\_FAST**

Hızlı hız.

Bir kanal MQNPMS\_FAST olarak tanımlandıysa, kalıcı olmayan iletiler kanaldan hızlı bir şekilde seyahat eder. Bu, kanalın verimini artırır, ancak bir kanal hatası varsa, kalıcı olmayan iletilerin kaybedileceği anlamına gelir. Ayrıca, kalıcı olmayan iletilerin aynı iletim kuyruğunda bekleyen kalıcı iletilerin önüne atlayabileceği, yani kalıcı olmayan iletilerin sırasının kalıcı iletilere göre korunmamasına da olanak sağlar. Ancak, kalıcı olmayan iletilerin her birine göre sırası korunur. Benzer şekilde, her biri ile göreceli kalıcı iletilerin sırası korunur.

#### *Parola (MQCHAR12)*

Bu alan, uzak ileti kanalı aracısıyla güvenli bir SNA oturumu başlatma girişimi sırasında ileti kanalı aracısının kullandığı parolayı belirtir.

Bu alan yalnızca AIX, Linux, and Windows' ta boş bırakılabilir ve yalnızca *ChannelType* MQCHT\_SENDER, MQCHT\_SERVER, MQCHT\_REQUESTER ya da MQCHT\_CLNTCONN ile olan kanallar için anlamlıdır. z/OS üzerinde, bu alan ilgili değildir.

Bu alanın uzunluğu MQ\_PASSWORD\_LENGTH tarafından verilir. Ancak, yalnızca ilk 10 karakter kullanılır.

*Version* , MQCD\_VERSION\_2değerinden küçükse bu alan yoktur.

#### *PropertyControl (MQUZE)*

Bu alan, ileti, V6 ya da önceki bir kuyruk yöneticisine (özellik tanımlayıcısı kavramını anlamayan bir kuyruk yöneticisi) gönderilmek üzereyken iletilerin özelliklerinde ne olacağını belirtir.

Değer, aşağıdaki değerlerden herhangi biri olabilir:

#### **MQPROP\_UYUMLULUK**

If the message contains a property with a prefix of **mcd.**, **jms.**, **usr.**, or **mnext.**, all message properties are delivered to the application in an MQRFH2 header. Ters durumda, ileti tanımlayıcısında (ya da uzantısında) yer alan özellikler dışında, iletinin tüm özellikleri atılır ve artık bu uygulamanın erişilemeyecek şekilde olur.

Bu değer, varsayılan değerdir; JMS ile ilgili özelliklerin ileti verilerindeki bir MQRFH2 üstbilgisinde yer almasını ve değiştirilmemiş çalışmaya devam etmesini sağlar.

#### **MQPROP\_NONE**

İleti tanımlayıcısındaki (ya da uzantısındaki) özellikler dışında, iletinin tüm özellikleri, ileti uzak kuyruk yöneticisine gönderilmeden önce iletiden kaldırılır.

## **MQPROP\_ALL**

İletinin tüm özellikleri, ileti uzak kuyruk yöneticisine gönderildiğinde iletiyle birlikte verilir. İleti tanımlayıcısındaki (ya da uzantıdaki) özellikler dışında, özellikler ileti verilerinde bir ya da daha çok MQRFH2 üstbilgilerine yerleştirilir.

Bu öznitelik, Gönderen, Sunucu, Küme Göndericisi ve Küme Alıcı kanalları için geçerlidir.

“MQIA\_ \* (Tamsayı Özniteliği Seçicileri)” sayfa 129

“MQPROP\_ \* (Kuyruk ve Kanal Özelliği Denetim Değerleri ve Maksimum Özellik Uzunluğu)” sayfa 169

### *PutAuthority (MQUZE)*

Bu alan, iletiyi hedef kuyruğa koyma yetkisi oluşturmak için bir iletiyle ilişkili bağlam bilgilerinde bulunan kullanıcı kimliğinin kullanılıp kullanılmayacağını belirtir.

Bu alan yalnızca, *ChannelType* MQCHT\_REQUESTER, MQCHT\_RECEIVER ya da MQCHT\_CLAUSRCVR ile olan kanallar için anlamlıdır. Bu, aşağıdakilerden biridir:

### **MQPA\_DEFAULT**

Varsayılan kullanıcı kimliği kullanılır.

### **MQPA\_CONTEXT**

Bağlam kullanıcı kimliği kullanıldı.

### **MQPA\_ALTERNATE\_OR\_MCA**

İleti tanımlayıcısının UserIdentifier (Kullanıcı Kimliği) alanındaki kullanıcı kimliği kullanılır. Ağdan alınan herhangi bir kullanıcı kimliği kullanılmaz. Bu değer yalnızca z/OS üzerinde desteklenir.

### **MQPA\_ONLY\_MCA**

Varsayılan kullanıcı kimliği kullanılır. Ağdan alınan herhangi bir kullanıcı kimliği kullanılmaz. Bu değer yalnızca z/OS üzerinde desteklenir.

### *QMgrName (MQCHAR48)*

Bu alan, bir çıkışta bağlantı kurabileceği kuyruk yöneticisinin adını belirtir.

For channels with a *ChannelType* other than MQCHT\_CLNTCONN, this field is the name of the queue manager that an exit can connect to, which on AIX, Linux, and Windows, is always nonblank.

Bu alanın uzunluğu MQ\_Q\_MGR\_NAME\_LENGTH değeri tarafından verilir.

### *ReceiveExit (MQCHARn)*

Bu alan, kanal alma çıkış adını belirtir.

Bu ad boş değilse, çıkış şu saatlerde çağrılır:

- Alınan ağ verileri işlenmeden hemen önce.

Çıkış, alınan tüm iletim arabelleğiyle verilir. Arabelleğin içeriği gerektiği gibi değiştirilebilir.

- Kanalın kullanıma hazırlanması ve sona erdirilmesi.

Çeşitli ortamlardaki bu alanın içeriğine ilişkin açıklamalar için “MQCD-Kanal tanımlaması” sayfa 1456 ' e bakın.

Bu alanın uzunluğu MQ\_EXIT\_NAME\_LENGTH değeri tarafından verilir.

**Not:** Bu değişmezin değeri ortamdan özgüdür.

### *ReceiveExitPtr (MQPTR)*

Bu alan, ilk *ReceiveExit* alanının adresini belirtir.

*ReceiveExitsDefined* sıfırdan büyükse, bu adres zincirdeki her bir kanal alma çıkışının adlarının adresidir.

Her ad, *ExitNameLength* uzunluktaki bir alanda, boşluklarla doldurulmuş doldurulur. Her bir çıkış için birden bire bir *ReceiveExitsDefined* alanı bitişik olan fields alanları vardır.

Bir çıkış tarafından bu adlarda yapılan değişiklikler korunur, ancak ileti kanalı çıkışı belirttik bir işlem yaparsa da, hangi çıkışlar çağrıldığı değişmez.

*ReceiveExitsDefined* sıfırda, bu alan boş değerli göstergedir.

Programlama dilinin işaretçi veri tipini desteklemediği altyapılarda, bu alan uygun uzunluğun byte dizgisi olarak bildirilir.

Bu, çıkış girişi alanıdır. *Version* değeri MQCD\_VERSION\_4değerinden küçükse bu alan yoktur.

*ReceiveExitsTanımlı (MQUZE)*

Bu alan, zincirde tanımlanan kanal alma çıkışlarının sayısını belirtir.

Sıfır değerinden büyük ya da sıfıra eşit.

Bu, çıkış girişi alanıdır. *Version* değeri MQCD\_VERSION\_4değerinden küçükse bu alan yoktur.

*ReceiveUserVerileri (MQCHAR32)*

Bu kanal, kanal alma çıkış kullanıcı verilerini belirtir.

This data is passed to the channel receive exit in the *ExitData* field of the **ChannelExitParms** parameter (see MQ\_CHANNEL\_EXIT).

Başlangıçta bu alan, kanal tanımlamasında belirlenen verileri içerir. Ancak, bu MCA eşgörünümünün geçerlilik süresi boyunca, bu alanın herhangi bir çıkıştan çıkılarak bu alanın içeriğinde yapılan değişiklikler MCA tarafından korunur ve bu MCA eşgörünümü için sonraki çıkışlar (tipten bağımsız olarak) tarafından görülebilir olarak görünür. Bu, farklı etkileşimlerdeki çıkışlar için geçerlidir. Bu tür değişiklikler, diğer MCA eşgörünümleri tarafından kullanılan kanal tanımlamasını etkilemez. Herhangi bir karakter (ikili veri de içinde olmak üzere) kullanılabilir.

Bu alanın uzunluğu MQ\_EXIT\_DATA\_LENGTH tarafından verilir.

Bu alan IBM MQ for IBM i ile ilgili değildir.

*Version* , MQCD\_VERSION\_2değerinden küçükse, bu yapıdaki şu alanlar yoktur.

*ReceiveUserDataPtr (MQPTR)*

Bu alan, ilk *ReceiveUserData* alanının adresini belirtir.

*ReceiveExitsDefined* sıfırdan büyükse, bu adres zincirdeki her bir kanal alma çıkışa ilişkin kullanıcı veri ögesi listesinin adresidir.

Each user data item is in a field of length *ExitDataLength*, padded to the right with blanks. Her bir çıkış için birden bire bir *ReceiveExitsDefined* alanı bitişik olan fieldsalanları vardır. Tanımlanan kullanıcı veri ögesi sayısı çıkış adlarından küçükse, tanımlanmamış kullanıcı veri ögeleri boşluk olarak ayarlanır. Ters durumda, tanımlanan kullanıcı veri ögeleri sayısı çıkış adlarından büyükse, fazla kullanıcı veri ögeleri yoksayılar ve çıkışa sunulmaz.

Bu değerler üzerinde bir çıkış tarafından yapılan değişiklikler korunur. Bu, bir çıkışa bilgileri başka bir çıkışa geçirmenizi sağlar. Herhangi bir değişiklik üzerinde geçerlilik denetimi gerçekleştirilmez; örneğin, gerekirse ikili veriler bu alanlara yazılabilir.

*ReceiveExitsDefined* sıfırda, bu alan boş değerli göstergedir.

Programlama dilinin işaretçi veri tipini desteklemediği altyapılarda, bu alan uygun uzunluğun byte dizgisi olarak bildirilir.

Bu, çıkış girişi alanıdır. *Version* değeri MQCD\_VERSION\_4değerinden küçükse bu alan yoktur.

*Version* değeri MQCD\_VERSION\_5değerinden küçükse, bu yapıdaki şu alanlar yoktur.

*RemotePassword (MQCHAR12)*

Bu alan, bir ortaktan parolayı belirtir.

Bu alan yalnızca, *ChannelType* MQCHT\_CLNTCONN ya da MQCHT\_SVRCONN ise geçerli bilgi içerir.

- MQCHT\_CLNTCONN kanalından bir güvenlik çıkışı için bu parola, ortamdan elde edilen bir paroladır. Çıkış, sunucudaki güvenlik çıkışa göndermeyi seçebilir.

- MQCHT\_SVRCONN kanalından bir güvenlik çıkışı için, istemci güvenlik çıkışı yoksa, bu alanda istemciden alınan bir parola bulunabilir. Çıkış, *RemoteUserIdentifier* içindeki kullanıcı kimliğini doğrulamak için bu parolayı kullanabilir.

İstemcide bir güvenlik çıkışı varsa, bu bilgiler istemcideki bir güvenlik akışında elde edilebilir.

Bu alanın uzunluğu MQ\_PASSWORD\_LENGTH tarafından verilir. *Version* , MQCD\_VERSION\_2 değerinden küçükse bu alan yoktur.

#### *RemoteSecurityTanıtıcısı (MQBYTE40)*

Bu alan, uzak kullanıcıya ilişkin güvenlik tanıtıcısını belirtir.

Bu alan yalnızca, bir MQCHT\_CLNTCONN ya da MQCHT\_SVRCONN *ChannelType* içeren kanallar için anlamlıdır.

Aşağıdaki özel değer, güvenlik tanıtıcısı olmadığını gösterir:

#### **MQSID\_NONE**

Güvenlik tanıtıcısı belirtilmedi.

Değer, alanın uzunluğuna ilişkin ikili sıfır değeridir.

C programlama dili için, constant MQSID\_NONE\_ARRAY de tanımlanır; bu değişmez, MQSID\_NONE ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisi olur.

Bu, çıkışa giriş alanıdır. Bu alanın uzunluğu MQ\_SECURITY\_ID\_LENGTH tarafından verilmektedir.

*Version* , MQCD\_VERSION\_6 değerinden küçükse bu alan yoktur.

*Version* değeri MQCD\_VERSION\_7 değerinden küçükse, bu yapıdaki şu alanlar yoktur.

#### *RemoteUserTanıtıcısı (MQCHAR12)*

Bu alan, bir ortaktan kullanıcı kimliğinin ilk 12 baytını belirtir.

Uzak kullanıcı kimliğini içeren iki alan vardır:

- *RemoteUserIdentifier* , uzak kullanıcı tanıtıcısının ilk 12 baytı içerir ve tanıtıcı 12 bayttan kısa olduğunda boşluklarla doldurulur. *RemoteUserIdentifier* boş olabilir.
- *LongRemoteUserIdPtr* , 12 byte 'tan uzun olabilen tam uzak kullanıcı tanıtıcısını gösterir. Uzunluğu *LongRemoteUserIdLength* ile verilir. Tam tanıtıcı sonda boşluk içermiyor ve boş değerle sonlandırılmadı. Tanıtıcı boşsa, *LongRemoteUserIdLength* sıfırdır ve *LongRemoteUserIdPtr* değeri tanımsız olur.

*Version* , MQCD\_VERSION\_6 değerinden küçükse *LongRemoteUserIdPtr* yoktur.

Uzak kullanıcı kimliği yalnızca, bir MQCHT\_CLNTCONN ya da MQCHT\_SVRCONN *ChannelType* içeren kanallar için anlamlıdır.

- MQCHT\_CLNTCONN kanalındaki bir güvenlik çıkışı için, bu değer, ortamdan elde edilen bir kullanıcı kimliğidir. Çıkış, sunucudaki güvenlik çıkışa göndermeyi seçebilir.
- MQCHT\_SVRCONN kanalındaki bir güvenlik çıkışı için, istemci güvenlik çıkışı yoksa, bu alanda, istemcideki ortamdan elde edilen bir kullanıcı kimliği bulunabilir. Çıkış, bu kullanıcı kimliğinin geçerliliğini denetleyebilir (olasılıkla *RemotePassword* parolasıyla birlikte) ve değeri *MCAUserIdentifier*(sayfa.) olarak güncelleyin.

İstemcide bir güvenlik çıkışı varsa, bu bilgiler istemcideki bir güvenlik akışında elde edilebilir.

Bu alanın uzunluğu MQ\_USER\_ID\_LENGTH tarafından verilir. *Version* , MQCD\_VERSION\_2 değerinden küçükse bu alan yoktur.

#### *SecurityExit (MQCHARn)*

Bu alan kanal güvenliği çıkış adını belirtir.

Bu ad boş değilse, çıkış şu saatlerde çağrılır:

- Bir kanal kurduktan hemen sonra.

Herhangi bir ileti aktarılmadan önce, çıkış, bağlantı yetkilendirmesini doğrulamak için güvenlik akışlarının aşılması için bir fırsat verilir.

- Bir güvenlik iletisi akışına yanıt aldıktan sonra.

Uzak makineden uzak işlemciden alınan herhangi bir güvenlik iletisi akışı çıkışa verilir.

- Kanalın kullanıma hazırlanması ve sona erdirilmesi.

Çeşitli ortamlardaki bu alanın içeriğine ilişkin açıklamalar için [“MQCD-Kanal tanımlaması” sayfa 1456](#) ' e bakın.

Bu alanın uzunluğu MQ\_EXIT\_NAME\_LENGTH değeri tarafından verilir.

**Not:** Bu değişmezin değeri ortamdan özgüdür.

*SecurityUserVerileri (MQCHAR32)*

Bu kanal, kanal güvenliği çıkış kullanıcı verilerini belirtir.

Bu veriler, **ChannelExitParms** parametresinin *ExitData* alanındaki kanal güvenliği çıkışa geçirilir (bkz. MQ\_CHANNEL\_EXIT).

Başlangıçta bu alan, kanal tanımlamasında belirlenen verileri içerir. Ancak, bu MCA eşgörünümünün geçerlilik süresi boyunca, bu alanın herhangi bir çıkıştan çıkılarak bu alanın içeriğinde yapılan değişiklikler MCA tarafından korunur ve bu MCA eşgörünümü için sonraki çıkışlar (tipten bağımsız olarak) tarafından görülebilir olarak görünür. Bu, farklı etkileşimlerdeki çıkışlar için geçerlidir. Bu tür değişiklikler, diğer MCA örnekleri tarafından kullanılan kanal tanımında yürürlüğe girmez. Herhangi bir karakter (ikili veri de içinde olmak üzere) kullanılabilir.

Bu alanın uzunluğu MQ\_EXIT\_DATA\_LENGTH tarafından verilir.

Bu alan IBM MQ for IBM i ile ilgili değildir.

*SendExit (MQCHARn)*

Bu alan, kanal gönderme çıkış adını belirtir.

Bu ad boş değilse, çıkış şu saatlerde çağrılır:

- Ağ üzerinden veri gönderilmeden hemen önce.

Çıkış, iletilmeden önce tam iletim arabelleğiyle verilir. Arabelleğin içeriği gerektiği gibi değiştirilebilir.

- Kanalın kullanıma hazırlanması ve sona erdirilmesi.

Çeşitli ortamlardaki bu alanın içeriğine ilişkin açıklamalar için [“MQCD-Kanal tanımlaması” sayfa 1456](#) ' e bakın.

Bu alanın uzunluğu MQ\_EXIT\_NAME\_LENGTH değeri tarafından verilir.

**Not:** Bu değişmezin değeri ortamdan özgüdür.

*SendExitPtr (MQPTR)*

Bu alan, ilk *SendExit* alanının adresini belirtir.

*SendExitsDefined* sıfırdan büyükse, bu adres zincirdeki her bir kanal gönderme çıkışının adlarının bulunduğu adrestir.

Her ad, *ExitNameLength* uzunluktaki bir alanda, boşluklarla doldurulmuşta doldurulur. Her bir çıkış için birden bire bir *SendExitsDefined* alanı bitişik olan alanları vardır.

Bir çıkışa göre bu adlarda yapılan değişiklikler korunur; ancak, ileti gönderme çıkışı belirttik bir işlem gerçekleştirmez; ancak, çağrılan çıkışlar değişmez.

*SendExitsDefined* sıfırsa, bu alan boş değerli göstergedir.

Programlama dilinin işaretçi veri tipini desteklemediği altyapılarda, bu alan uygun uzunluğun byte dizgisi olarak bildirilir.

Bu, çıkışa giriş alanıdır. *Version* değeri MQCD\_VERSION\_4 değerinden küçükse bu alan yoktur.



### *SendExitsTanımlı (MQLONG)*

Bu alan, zincirde tanımlanan kanal gönderme çıkışlarının sayısını belirtir.

Sıfır değerinden büyük ya da sıfıra eşit.

Bu, çıkışa giriş alanıdır. *Version* değeri MQCD\_VERSION\_4değerinden küçükse bu alan yoktur.

### *SendUserVerileri (MQCHAR32)*

Bu alan, kanal gönderme çıkış kullanıcı verilerini belirtir.

This data is passed to the channel send exit in the *ExitData* field of the **ChannelExitParms** parameter (see MQ\_CHANNEL\_EXIT).

Başlangıçta bu alan, kanal tanımlamasında belirlenen verileri içerir. Ancak, bu MCA eşgörünümünün geçerlilik süresi boyunca, bu alanın herhangi bir çıkıştan çıkılarak bu alanın içeriğinde yapılan değişiklikler MCA tarafından korunur ve bu MCA eşgörünümü için sonraki çıkışlar (tipten bağımsız olarak) tarafından görülebilir olarak görünür. Bu, farklı etkileşimlerdeki çıkışlar için geçerlidir. Bu tür değişiklikler, diğer MCA eşgörünümleri tarafından kullanılan kanal tanımlamasını etkilemez. Herhangi bir karakter (ikili veri de içinde olmak üzere) kullanılabilir.

Bu alanın uzunluğu MQ\_EXIT\_DATA\_LENGTH tarafından verilir.

Bu alan IBM MQ for IBM ile ilgili değildir.

### *SendUserDataPtr (MQPTR)*

Bu alan, *SendUserData* alanının adresini belirtir.

*SendExitsDefined* sıfırdan büyükse, bu adres zincirdeki her kanal iletili çıkışa ilişkin kullanıcı veri öğeleri listesinin adresidir.

Each user data item is in a field of length *ExitDataLength*, padded to the right with blanks. Her bir çıkış için birden bire bir *MsgExitsDefined* alanı bitişik olan fieldsalanları vardır. Tanımlanan kullanıcı veri öğesi sayısı çıkış adlarından küçükse, tanımlanmamış kullanıcı veri öğeleri boşluk olarak ayarlanır. Ters durumda, tanımlanan kullanıcı veri öğeleri sayısı çıkış adlarından büyükse, fazla kullanıcı veri öğeleri yoksayılır ve çıkışa sunulmaz.

Bu değerler üzerinde bir çıkış tarafından yapılan değişiklikler korunur. Bu, bir çıkışa bilgileri başka bir çıkışa geçirmenizi sağlar. Herhangi bir değişiklik üzerinde geçerlilik denetimi gerçekleştirilmez; örneğin, gerekirse ikili veriler bu alanlara yazılabilir.

*SendExitsDefined* sıfırsa, bu alan boş değerli göstergedir.

Programlama dilinin işaretçi veri tipini desteklemediği altyapılarda, bu alan uygun uzunluğun byte dizgisi olarak bildirilir.

Bu, çıkışa giriş alanıdır. *Version* değeri MQCD\_VERSION\_4değerinden küçükse bu alan yoktur.

### *SeqNumberKaydır (MQUZE)*

Bu alan, izin verilen en yüksek ileti sıra numarasını belirtir.

Bu değere ulaşıldığında, sıra numaraları 1 'de yeniden başlayacak şekilde kaydırılır.

Bu değer, pazarlığa açık değildir ve hem yerel hem de uzak kanal tanımlamalarında eşleşmelidir.

This field is not relevant for channels with a *ChannelType* of MQCHT\_SVRCONN or MQCHT\_CLNTCONN.

### *SharingConversations (MQUZE)*

Bu alan, bu kanalla ilişkilendirilmiş bir kanal eşgörünümünü paylaşabilecek etkileşim sayısı üst sınırını belirtir.

Bu alan, istemci bağlantısı ve sunucu-bağlantı kanallarında kullanılır.

0 değeri, kanalın aşağıdaki özniteliklere göre IBM WebSphere MQ 7.0 öncesindeki sürümlerde olduğu gibi çalıştığı anlamına gelir:

- Etkileşim paylaşımı

- Önden okuma
- STOP CHANNEL (*channelname*) MODE (QUIESCE)
- Kalp atımı
- İstemci zamanuyumsuz tüketimi

1 değeri, IBM WebSphere MQ 7.0 davranışı için değer alt sınırı değeridir. Kanal yönetim ortamında yalnızca bir sohbete izin verilse de, ileriye doğru okuma, zamanuyumsuz tüketim ve CLNTCONN-SVRCONN ' in kalp atımı ve susturucu kanal durdurma davranışlarının IBM WebSphere MQ 7.0 davranışı kullanılabilir.

Bu, çıkış girişi alanıdır. *Version* değeri MQCD\_VERSION\_9değerinden küçükse, bu değer yoktur.

Bu alanın varsayılan değeri 10 'tır.

**Not:** *MaxInstances* ve *MaxInstancesPerClient* bir kanala uygulanan sınırlar, bu eşgörünümleri paylaşabilecek etkileşimlerin sayısını değil, kanal eşgörünümlerinin sayısını kısıtlıyor.

*ShortConnectionAdı* (MQCHAR20)

Bu alan, bir bağlantı adının ilk 20 baytını belirtir.

*Version* alanı MQCD\_VERSION\_1ise, *ShortConnectionName* tam bağlantı adını içerir.

*Version* alanı MQCD\_VERSION\_2 ya da daha büyükse, *ShortConnectionName* , bağlantı adının ilk 20 karakterini içerir. Tam bağlantı adı *ConnectionName* alanı tarafından verilir; *ShortConnectionName* ve *ConnectionName* ' un ilk 20 karakteri aynı olur.

Bu alanın içindekilerin ayrıntıları için bkz. *ConnectionName* .

**Not:** Bu alanın adı MQCD\_VERSION\_2 ve sonraki MQCD sürümleri için değiştirildi; bu alan önceden *ConnectionName*olarak adlandırıldı.

Bu alanın uzunluğu MQ\_SHORT\_CONN\_NAME\_LENGTH tarafından verilir.

*ShortRetryCount* (MQLONG)

Bu alan, uzak bir makineye bağlanmak için yapılan deneme sayısı üst sınırını belirtir.

Bu alan, (normalde daha uzun) *LongRetryCount* ve *LongRetryInterval* kullanılmadan önce, *ShortRetryInterval*ile belirtilen aralıklarda, uzak makineye bağlanmak için yapılan deneme sayısı üst sınısıdır.

Bu alan yalnızca, *ChannelType* MQCHT\_ENDER, MQCHT\_SERVER, MQCHT\_CLUUSDR ya da MQCHT\_CLAUSRCVR ile olan kanallar için geçerlidir.

*ShortRetryInterval* (MQHOMER)

Bu alan, uzak makineden yeniden bağlantı kurulmadan önce beklenecek süre üst sınırını saniye cinsinden belirtir.

Kanal etkin olmak için beklemek zorunda kalacaksa, yeniden denemeler arasındaki aralık uzatılabilir.

Bu alan yalnızca, *ChannelType* MQCHT\_ENDER, MQCHT\_SERVER, MQCHT\_CLUUSDR ya da MQCHT\_CLAUSRCVR ile olan kanallar için geçerlidir.

 *SPLProtection* (MQHOT)

Bu alan, AMS güvenlik ilkesi korumasının değerini belirtir.

Değer aşağıdakilerden biridir:

#### **MQSP\_PASSSON**

Bu kanal için MCA tarafından gönderilen ya da alınan tüm iletiler, değiştirilmeden, değiştirilmeden geçer.

Bu değer yalnızca, *ChannelType* MQCHT\_ENDER, MQCHT\_SERVER, MQCHT\_RECEIVER ya da MQCHT\_REQUESTER ile ilişkili kanallar için geçerlidir ve varsayılan değerdir.

## **MQSPL\_REMOVE**

MCA ' nın iletim kuyruğundan alınan iletilerden AMS korumasını kaldırın ve iletileri iş ortağına gönderin.

This value is relevant only for channels with a *ChannelType* of MQCHT\_SENDER or MQCHT\_SERVER.

## **MQSP\_ASPOICILANCE**

Based on the policy defined for the target queue, apply AMS protection to inbound messages prior to putting them on to the target queue.

Bu değer yalnızca, *ChannelType* MQCHT\_RECEIVER ya da MQCHT\_REQUESTER içeren kanallar için anlamlıdır.

Bu, çıkışa giriş alanıdır. *Version* , MQCD\_VERSION\_12değerinden küçükse bu alan yoktur.

### *SSLCipherSpec (MQCHAR32)*

Bu alan, TLS kullanılırken kullanımda olan Cipher Spec 'i belirtir.

SSLCipherSpec boş bırakılırsa, kanal TLS kullanmaz. Bu alan boş değilse, bu alan, kullanımda CipherSpec belirtimini belirten bir dizgi içerir.

Bu parametre tüm kanal tipleri için geçerlidir. Aşağıdaki altyapılarda desteklenir:

-  AIX
-  IBM i
-  Linux
-  Windows
-  z/OS

Yalnızca, TCP ' nin TRPTYPE (iletim tipi) kanal tipleri için geçerlidir.

Bu, çıkışa giriş alanıdır. Bu alanın uzunluğu MQ\_SSL\_CIPHER\_SPEC\_LENGTH tarafından verilir. *Version* değeri MQCD\_VERSION\_7değerinden küçükse bu alan yoktur.

### *SSLClientAuth (MQUZE)*

Bu alan, TLS istemcisi kimlik doğrulamasının gerekli olup olmadığını belirtir.

Bu alan yalnızca SVRCONN kanal tanımlamalarıyla ilişkilidir.

Bu değer aşağıdaki değerlerden biridir:

## **MQSCA\_REQUIREM**

İstemci kimlik doğrulaması gerekli.

## **MQSCA\_OPTIONAL**

İstemci kimlik doğrulaması isteğe bağlıdır.

Bu, çıkışa giriş alanıdır. *Version* değeri MQCD\_VERSION\_7değerinden küçükse bu alan yoktur.

### *SSLPeerNameUzunluğu (MQUZE)*

This field specifies the length in bytes of the TLS peer name pointed to by *SSLPeerNamePtr*.

Bu, çıkışa giriş alanıdır. *Version* değeri MQCD\_VERSION\_7değerinden küçükse bu alan yoktur.

### *SSLPeerNamePtr (MQPTR)*

Bu alan, TLS eşdüzey adının adresini belirtir.

Başarılı bir TLS anlaşması sırasında bir sertifika alındığında, sertifika konularının Ayırt Edici Adı, sertifikayı alan kanal sonunda SSLPeerNamePtr 'si tarafından erişilen MQCD alanına kopyalanır. Bu değer, yerel kullanıcının kanal tanımlamasında varsa, kanala ilişkin SSLPeerName değerini geçersiz kılar. Kanal sonunda bir güvenlik çıkışı belirtilirse, bu çıkış, MQCD ' deki eşdüzey sertifikadan ayırt edici adı alır.

Bu, çıkışa giriş alanıdır. *Version* değeri MQCD\_VERSION\_7değerinden küçükse bu alan yoktur.

**Not:** IBM WebSphere MQ 7.1 yayın düzeyinden önce oluşturulan güvenlik çıkış uygulamalarının güncellenmesi gerekebilir. Ek bilgi için bkz. [Kanal güvenlik çıkış programları](#).

*StrucLength (MQUZE)*

Bu alan, MQCD yapısının bayt cinsinden uzunluğunu belirler.

Uzunluk, yapı içinde bulunan işaretçi alanları tarafından adreslenen dizgilerden hiçbirini içermez. Değer aşağıdakilerden biridir:

**MQCD\_LENGTH\_4**

version-4 kanal tanımlama yapısının uzunluğu.

**MQCD\_LENGTH\_5**

version-5 kanal tanımlama yapısının uzunluğu.

**MQCD\_LENGTH\_6**

version-6 kanal tanımlama yapısının uzunluğu.

**MQCD\_LENGTH\_7**

version-7 kanal tanımlama yapısının uzunluğu.

**MQCD\_LENGTH\_8**

version-8 kanal tanımlama yapısının uzunluğu.

**MQCD\_LENGTH\_9**

version-9 kanal tanımlama yapısının uzunluğu.

**MQCD\_LENGTH\_10**

version-10 kanal tanımlama yapısının uzunluğu.

**MQCD\_LENGTH\_11**

version-11 kanal tanımlama yapısının uzunluğu.

 **MQCD\_LENGTH\_12**

version-12 kanal tanımlama yapısının uzunluğu.

Aşağıdaki değişmez, yürürlükteki sürümün uzunluğunu belirtir:

**MQCD\_CURRENT\_LENGTH**

Kanal tanımlama yapısının yürürlükteki sürümü uzunluğu.

**Not:** Bu değişmezlerin, ortama özgü değerleri vardır.

*Version* değeri MQCD\_VERSION\_4değerinden küçükse bu alan yoktur.

*TpName (MQCHAR64)*

Bu alan, LU 6.2 hareket programı adını belirtir.

Bu alan, yalnızca iletim protokolü (*TransportType*) MQXPT\_LU62ise ve *ChannelType* MQCHT\_SVRCONN ya da MQCHT\_RECEIVER değilse anlamlıdır.

Bu alan her zaman, onun yerine iletişim Side nesnesinde yer alan altyapılarda boş olur.

Bu alanın uzunluğu MQ\_TP\_NAME\_LENGTH tarafından verilir.

*TransportType (MQLONG)*

Bu alan, kullanılacak iletim protokolünü belirler.

Kanal başka bir uçtan başlatıldıysa, değer denetlenmez.

Bu değer aşağıdaki değerlerden biridir:

**MQXPT\_LU62**

LU 6.2 iletimi protokolü.

**MQXPT\_TCP**

TCP/IP iletim protokolü.

**MQXPT\_NETBIOS**

NetBIOS iletim protokolü.

Bu değer, şu ortamlarda desteklenir: Windows.

**MQXPT\_SPX**

SPX iletim protokolü.

Bu değer, şu ortamlarda desteklenir: Windows, ayrıca bu sistemlere bağlı IBM MQ istemcileri.

*UseDLQ (MQUZY)*

Bu alan, iletiler kanallar tarafından teslim edilemediğinde, kullanılmayan ileti kuyruğunun (ya da teslim edilmemiş ileti kuyruğunun) kullanılıp kullanılmayacağını belirtir.

Aşağıdaki değerlerden birini içerebilir:

**MQUSEDLQ\_NO**

Bir kanal tarafından teslim edilemeyen iletiler, bir hata olarak değerlendirilir. Kanal, NPMSPEED ayarına uygun olarak iletiyi ya da kanal uçlarını atar.

**MQUSEDLQ\_YES**

DEADQ kuyruk yöneticisi özniteliği, bir ölü-mektup kuyruğunun adını sağladığında, o zaman kullanılır; tersi durumda, davranış NO (HAYIR) olarak olur. Varsayılan değer YES (Evet) değeridir.

*UserIdentifier (MQCHAR12)*

Bu alan, uzak ileti kanalı aracısıyla güvenli bir SNA oturumu başlatma girişimi sırasında ileti kanalı aracısı tarafından kullanılan kullanıcı kimliğini belirtir.

Bu alan yalnızca AIX, Linux, and Windows' ta boş bırakılabilir ve yalnızca *ChannelType* MQCHT\_SENDER, MQCHT\_SERVER, MQCHT\_REQUESTER ya da MQCHT\_CLNTCONN ile olan kanallar için anlamlıdır. z/OS üzerinde, bu alan ilgili değildir.

Bu alanın uzunluğu MQ\_USER\_ID\_LENGTH tarafından verilir. Ancak, yalnızca ilk 10 karakter kullanılır.

*Version*, MQCD\_VERSION\_2 değerinden küçükse bu alan yoktur.

*Sürüm (MQLONG)*

The *Version* field specifies the highest version number that you can set for the structure.

Değer, ortama bağlıdır:

**MQCD\_VERSION\_1**

Sürüm 1 kanal tanımlama yapısı.

**MQCD\_VERSION\_2**

Sürüm 2 kanal tanımlama yapısı.

**MQCD\_VERSION\_3**

Sürüm 3 kanal tanımlama yapısı.

**MQCD\_VERSION\_4**

Sürüm 4 kanal tanımlama yapısı.

**MQCD\_VERSION\_5**

Sürüm 5 kanal tanımlama yapısı.

**MQCD\_VERSION\_6**

Sürüm 6 kanal tanımlama yapısı.

**MQCD\_VERSION\_7**

Sürüm 7 kanal tanımlama yapısı.

**MQCD\_VERSION\_8**

Sürüm 8 kanal tanımlama yapısı.

**MQCD\_VERSION\_9**

Sürüm 9 kanal tanımlama yapısı.

Sürüm 9, alanı tüm platformlarda IBM WebSphere MQ 7.0 ve IBM WebSphere MQ 7.0.1 platformunda ayarlayabileceğiniz en yüksek sürümdür.

#### **MQCD \_VERSION\_10**

Sürüm 10 kanal tanımlama yapısı.

Sürüm 10, alanı tüm platformlarda IBM WebSphere MQ 7.1 ve IBM WebSphere MQ 7.5 platformunda ayarlayabileceğiniz en yüksek sürüm olan 10 'tır.

#### **MQCD \_VERSION\_11**

Sürüm 11 kanal tanımlama yapısı.

Sürüm 11, alanı tüm platformlarda IBM MQ 8.0 ' ta ayarlayabileceğiniz en yüksek sürüm.

#### **z/OS V 9.2.0 MQCD \_VERSION\_12**

Sürüm 12 kanal tanımlama yapısı.

Sürüm 12, alanı IBM MQ 9.1.3' ta ayarlayabileceğiniz en yüksek sürüm.

Yalnızca yapının daha yeni sürümlerinde bulunan alanlar, alanların açıklamalarında olduğu gibi tanımlanır. Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

#### **MQCD\_CURRENT\_VERSION**

MQCD\_CURRENT\_VERSION içinde ayarlanan değer, kullanılmakta olan kanal tanımlama yapısının yürürlükteki sürüsüdür.

MQCD\_CURRENT\_VERSION değeri ortama bağlıdır. Bu, platform tarafından desteklenen en yüksek değeri içerir.

MQCD\_CURRENT\_VERSION , üstbilgide sağlanan varsayılan yapıları başlatmak, kopyalamak ve farklı programlama dilleri için sağlanan dosyaları içermek için kullanılmaz. *Version* ' in varsayılan kullanıma hazırlanması altyapıya ve yayına bağlıdır.

IBM WebSphere MQ 7.0 ve sonraki sürümler için, üstbilgideki MQCD bildirimleri, kopyalanır ve içerme dosyaları MQCD\_VERSION\_6' a başlatılır. Ek MQCD alanları kullanmak için, uygulamaların sürüm numarasını MQCD\_CURRENT\_VERSION olarak ayarlaması gerekir. Birkaç ortam arasında taşınabilir bir uygulama yazıyorsanız, tüm ortamlarda desteklenen bir sürüm seçmeniz gerekir.

**İpucu:** MQCD yapısının yeni bir sürümü kullanıma sunulduğunda, var olan parçanın düzeni değiştirilmez. Çıkış, sürüm numarasını denetlemeli. Çıkışa eşit ya da daha büyük olmalıdır; bu, çıkışa ilişkin alanların kullanılması gereken en düşük sürüme eşit ya da daha büyük olmalıdır.

#### *XmitQName (MQCHAR48)*

Bu alan, iletilerin alınacağı iletim kuyruğunun adını belirler.

This field is relevant only for channels with a *ChannelType* of MQCHT\_SENDER or MQCHT\_SERVER.

Bu alanın uzunluğu MQ\_Q\_NAME\_LENGTH değeri tarafından verilir.

#### **C bildirimi**

Bu bildirim, MQCD yapısına ilişkin C bildirimidir.

#### **V 9.2.0**

```
typedef struct tagMQCD MQCD;  
typedef MQCD MQPOINTER PMQCD;  
typedef PMQCD MQPOINTER PPMQCD;
```

```
struct tagMQCD {  
    MQCHAR    ChannelName[20];           /* Channel definition name */  
    MQLONG    Version;                  /* Structure version number */  
    MQLONG    ChannelType;              /* Channel type */  
    MQLONG    TransportType;            /* Transport type */  
    MQCHAR    Desc[64];                 /* Channel description */  
    MQCHAR    QMgrName[48];             /* Queue manager name */  
    MQCHAR    XmitQName[48];            /* Transmission queue name */  
    MQCHAR    ShortConnectionName[20]; /* First 20 bytes of */  
                                           /* connection name */  
    MQCHAR    MCAName[20];              /* Reserved */  
    MQCHAR    ModeName[8];              /* LU 6.2 Mode name */  
};
```

```

MQCHAR    TpName[64];                /* LU 6.2 transaction program */
                                                /* name */
MQLONG    BatchSize;                /* Batch size */
MQLONG    DiscInterval;            /* Disconnect interval */
MQLONG    ShortRetryCount;        /* Short retry count */
MQLONG    ShortRetryInterval;     /* Short retry wait interval */
MQLONG    LongRetryCount;         /* Long retry count */
MQLONG    LongRetryInterval;     /* Long retry wait interval */
MQCHAR    SecurityExit[128];      /* Channel security exit name */
MQCHAR    MsgExit[128];           /* Channel message exit name */
MQCHAR    SendExit[128];         /* Channel send exit name */
MQCHAR    ReceiveExit[128];      /* Channel receive exit name */
MQLONG    SeqNumberWrap;          /* Highest allowable message */
                                                /* sequence number */

MQLONG    MaxMsgLength;           /* Maximum message length */
MQLONG    PutAuthority;          /* Put authority */
MQLONG    DataConversion;        /* Data conversion */
MQCHAR    SecurityUserData[32];   /* Channel security exit user */
                                                /* data */
MQCHAR    MsgUserData[32];        /* Channel message exit user */
                                                /* data */
MQCHAR    SendUserData[32];      /* Channel send exit user */
                                                /* data */
MQCHAR    ReceiveUserData[32];   /* Channel receive exit user */
                                                /* data */

/* Ver:1 */
MQCHAR    UserIdentifier[12];     /* User identifier */
MQCHAR    Password[12];          /* Password */
MQCHAR    MCAUserIdentifier[12]; /* First 12 bytes of MCA user */
                                                /* identifier */

MQLONG    MCAType;                /* Message channel agent type */
MQCHAR    ConnectionName[264];   /* Connection name */
MQCHAR    RemoteUserIdentifier[12]; /* First 12 bytes of user */
                                                /* identifier from partner */
MQCHAR    RemotePassword[12];    /* Password from partner */

/* Ver:2 */
MQCHAR    MsgRetryExit[128];     /* Channel message retry exit */
                                                /* name */
MQCHAR    MsgRetryUserData[32];  /* Channel message retry exit */
                                                /* user data */
MQLONG    MsgRetryCount;         /* Number of times MCA will */
                                                /* try to put the message, */
                                                /* after first attempt has */
                                                /* failed */
MQLONG    MsgRetryInterval;     /* Minimum interval in */
                                                /* milliseconds after which */
                                                /* the open or put operation */
                                                /* will be retried */

/* Ver:3 */
MQLONG    HeartbeatInterval;     /* Time in seconds between */
                                                /* heartbeat flows */
MQLONG    BatchInterval;        /* Batch duration */
MQLONG    NonPersistentMsgSpeed; /* Speed at which */
                                                /* nonpersistent messages are */
                                                /* sent */
MQLONG    StrucLength;          /* Length of MQCD structure */
MQLONG    ExitNameLength;       /* Length of exit name */
MQLONG    ExitDataLength;      /* Length of exit user data */
MQLONG    MsgExitsDefined;     /* Number of message exits */
                                                /* defined */
MQLONG    SendExitsDefined;     /* Number of send exits */
                                                /* defined */
MQLONG    ReceiveExitsDefined;  /* Number of receive exits */
                                                /* defined */
MQPTR     MsgExitPtr;           /* Address of first MsgExit */
                                                /* field */
MQPTR     MsgUserDataPtr;       /* Address of first */
                                                /* MsgUserData field */
MQPTR     SendExitPtr;          /* Address of first SendExit */
                                                /* field */
MQPTR     SendUserDataPtr;     /* Address of first */
                                                /* SendUserData field */
MQPTR     ReceiveExitPtr;      /* Address of first */
                                                /* ReceiveExit field */
MQPTR     ReceiveUserDataPtr;  /* Address of first */
                                                /* ReceiveUserData field */

/* Ver:4 */
MQPTR     ClusterPtr;           /* Address of a list of */
                                                /* cluster names */
MQLONG    ClustersDefined;      /* Number of clusters to */
                                                /* which the channel belongs */
MQLONG    NetworkPriority;      /* Network priority */

```

```

/* Ver:5 */
MQLONG    LongMCAUserIdLength;    /* Length of long MCA user */
/* identifier */
MQLONG    LongRemoteUserIdLength; /* Length of long remote user */
/* identifier */
MQPTR     LongMCAUserIdPtr;       /* Address of long MCA user */
/* identifier */
MQPTR     LongRemoteUserIdPtr;    /* Address of long remote */
/* user identifier */
MQBYTE40  MCASecurityId;          /* MCA security identifier */
MQBYTE40  RemoteSecurityId;       /* Remote security identifier */
/* Ver:6 */
MQCHAR    SSLCipherSpec[32];      /* TLS CipherSpec */
MQPTR     SSLPeerNamePtr;         /* Address of TLS peer name */
MQLONG    SSLPeerNameLength;      /* Length of TLS peer name */
MQLONG    SSLClientAuth;          /* Whether TLS client */
/* authentication is required */
MQLONG    KeepAliveInterval;      /* Keepalive interval */
MQCHAR    LocalAddress[48];       /* Local communications */
/* address */
MQLONG    BatchHeartbeat;         /* Batch heartbeat interval */
/* Ver:7 */
MQLONG    HdrCompList[2];         /* Header data compression */
/* list */
MQLONG    MsgCompList[16];        /* Message data compression */
/* list */
MQLONG    CLWLChannelRank;        /* Channel rank */
MQLONG    CLWLChannelPriority;     /* Channel priority */
MQLONG    CLWLChannelWeight;      /* Channel weight */
MQLONG    ChannelMonitoring;      /* Channel monitoring */
MQLONG    ChannelStatistics;      /* Channel statistics */
/* Ver:8 */
MQLONG    SharingConversations;   /* Limit on sharing */
/* conversations */
MQLONG    PropertyControl;        /* Message property control */
MQLONG    MaxInstances;           /* Limit on SVRCONN channel */
/* instances */
MQLONG    MaxInstancesPerClient;  /* Limit on SVRCONN channel */
/* instances per client */
MQLONG    ClientChannelWeight;    /* Client channel weight */
MQLONG    ConnectionAffinity;     /* Connection affinity */
/* Ver:9 */
MQLONG    BatchDataLimit;         /* Batch data limit */
MQLONG    UseDLQ;                /* Use Dead Letter Queue */
MQLONG    DefReconnect;          /* Default client reconnect */
/* option */
/* Ver:10 */
MQCHAR64  CertificateLabel;       /* Certificate label */
/* Ver:11 */
MQLONG    SPLProtection           /* AMS Security policy protection */
/* Ver:12 */
};

```

## COBOL bildirim

Bu bildirim, MQCD yapısıyla ilgili COBOL bildirimidir.

```

V9.2.0
** MQCD structure
  10 MQCD.
    ** Channel definition name
    15 MQCD-CHANNELNAME PIC X(20).
    ** Structure version number
    15 MQCD-VERSION PIC S9(9) BINARY.
    ** Channel type
    15 MQCD-CHANNELTYPE PIC S9(9) BINARY.
    ** Transport type
    15 MQCD-TRANSPORTTYPE PIC S9(9) BINARY.
    ** Channel description
    15 MQCD-DESC PIC X(64).
    ** Queue manager name
    15 MQCD-QMGRNAME PIC X(48).
    ** Transmission queue name
    15 MQCD-XMITQNAME PIC X(48).
    ** First 20 bytes of connection name
    15 MQCD-SHORTCONNECTIONNAME PIC X(20).
    ** Reserved
    15 MQCD-MCANAME PIC X(20).
    ** LU 6.2 Mode name

```



```

15 MQCD-MODENAME PIC X(8).
** LU 6.2 transaction program name
15 MQCD-TPNAME PIC X(64).
** Batch size
15 MQCD-BATCHSIZE PIC S9(9) BINARY.
** Disconnect interval
15 MQCD-DISCINTERVAL PIC S9(9) BINARY.
** Short retry count
15 MQCD-SHORTRETRYCOUNT PIC S9(9) BINARY.
** Short retry wait interval
15 MQCD-SHORTRETRYINTERVAL PIC S9(9) BINARY.
** Long retry count
15 MQCD-LONGRETRYCOUNT PIC S9(9) BINARY.
** Long retry wait interval
15 MQCD-LONGRETRYINTERVAL PIC S9(9) BINARY.
** Channel security exit name
15 MQCD-SECURITYEXIT PIC X(20).
** Channel message exit name
15 MQCD-MSGEXIT PIC X(20).
** Channel send exit name
15 MQCD-SENDEXIT PIC X(20).
** Channel receive exit name
15 MQCD-RECEIVEEXIT PIC X(20).
** Highest allowable message sequence number
15 MQCD-SEQNUMBERWRAP PIC S9(9) BINARY.
** Maximum message length
15 MQCD-MAXMSGLLENGTH PIC S9(9) BINARY.
** Put authority
15 MQCD-PUTAUTHORITY PIC S9(9) BINARY.
** Data conversion
15 MQCD-DATACONVERSION PIC S9(9) BINARY.
** Channel security exit user data
15 MQCD-SECURITYUSERDATA PIC X(32).
** Channel message exit user data
15 MQCD-MSGUSERDATA PIC X(32).
** Channel send exit user data
15 MQCD-SENDUSERDATA PIC X(32).
** Channel receive exit user data
15 MQCD-RECEIVEUSERDATA PIC X(32).
** Ver:1 **
** User identifier
15 MQCD-USERIDENTIFIER PIC X(12).
** Password
15 MQCD-PASSWORD PIC X(12).
** First 12 bytes of MCA user identifier
15 MQCD-MCAUSERIDENTIFIER PIC X(12).
** Message channel agent type
15 MQCD-MCATYPE PIC S9(9) BINARY.
** Connection name
15 MQCD-CONNECTIONNAME PIC X(264).
** First 12 bytes of user identifier from partner
15 MQCD-REMOTEUSERIDENTIFIER PIC X(12).
** Password from partner
15 MQCD-REMOTEPASSWORD PIC X(12).
** Ver:2 **
** Channel message retry exit name
15 MQCD-MSGRETRYEXIT PIC X(20).
** Channel message retry exit user data
15 MQCD-MSGRETRYUSERDATA PIC X(32).
** Number of times MCA will try to put the message, after first
** attempt has failed
15 MQCD-MSGRETRYCOUNT PIC S9(9) BINARY.
** Minimum interval in milliseconds after which the open or put
** operation will be retried
15 MQCD-MSGRETRYINTERVAL PIC S9(9) BINARY.
** Ver:3 **
** Time in seconds between heartbeat flows
15 MQCD-HEARTBEATINTERVAL PIC S9(9) BINARY.
** Batch duration
15 MQCD-BATCHINTERVAL PIC S9(9) BINARY.
** Speed at which nonpersistent messages are sent
15 MQCD-NONPERSISTENTMSGSPPEED PIC S9(9) BINARY.
** Length of MQCD structure
15 MQCD-STRUCLLENGTH PIC S9(9) BINARY.
** Length of exit name
15 MQCD-EXITNAMELENGTH PIC S9(9) BINARY.
** Length of exit user data
15 MQCD-EXITDATALENGTH PIC S9(9) BINARY.
** Number of message exits defined
15 MQCD-MSGEXITSDEFINED PIC S9(9) BINARY.
** Number of send exits defined
15 MQCD-SENDEXITSDEFINED PIC S9(9) BINARY.

```

```

** Number of receive exits defined
  15 MQCD-RECEIVEEXITSDEFINED PIC S9(9) BINARY.
** Address of first MsgExit field
  15 MQCD-MSGEXITPTR POINTER.
** Address of first MsgUserData field
  15 MQCD-MSGUSERDATAPTR POINTER.
** Address of first SendExit field
  15 MQCD-SENDEXITPTR POINTER.
** Address of first SendUserData field
  15 MQCD-SENDUSERDATAPTR POINTER.
** Address of first ReceiveExit field
  15 MQCD-RECEIVEEXITPTR POINTER.
** Address of first ReceiveUserData field
  15 MQCD-RECEIVEUSERDATAPTR POINTER.
** Ver:4 **
** Address of a list of cluster names
  15 MQCD-CLUSTERPTR POINTER.
** Number of clusters to which the channel belongs
  15 MQCD-CLUSTERSDEFINED PIC S9(9) BINARY.
** Network priority
  15 MQCD-NETWORKPRIORITY PIC S9(9) BINARY.
** Ver:5 **
** Length of long MCA user identifier
  15 MQCD-LONGMCAUSERIDLENGTH PIC S9(9) BINARY.
** Length of long remote user identifier
  15 MQCD-LONGREMOTEUSERIDLENGTH PIC S9(9) BINARY.
** Address of long MCA user identifier
  15 MQCD-LONGMCAUSERIDPTR POINTER.
** Address of long remote user identifier
  15 MQCD-LONGREMOTEUSERIDPTR POINTER.
** MCA security identifier
  15 MQCD-MCASECURITYID PIC X(40).
** Remote security identifier
  15 MQCD-REMOTESECURITYID PIC X(40).
** Ver:6 **
** TLS CipherSpec
  15 MQCD-SSLCIPHERSPEC PIC X(32).
** Address of TLS peer name
  15 MQCD-SSLPEERNAMEPTR POINTER.
** Length of TLS peer name
  15 MQCD-SSLPEERNAMELENGTH PIC S9(9) BINARY.
** Whether TLS client authentication is required
  15 MQCD-SSLCLIENTAUTH PIC S9(9) BINARY.
** Keepalive interval
  15 MQCD-KEEPALIVEINTERVAL PIC S9(9) BINARY.
** Local communications address
  15 MQCD-LOCALADDRESS PIC X(48).
** Batch heartbeat interval
  15 MQCD-BATCHHEARTBEAT PIC S9(9) BINARY.
** Ver:7 **
** Header data compression list
  15 MQCD-HDRCOMPLIST PIC S9(9) BINARY.
** Message data compression list
  15 MQCD-MSGCOMPLIST PIC S9(9) BINARY.
** Channel rank
  15 MQCD-CLWLCHANNELRANK PIC S9(9) BINARY.
** Channel priority
  15 MQCD-CLWLCHANNELPRIORITY PIC S9(9) BINARY.
** Channel weight
  15 MQCD-CLWLCHANNELWEIGHT PIC S9(9) BINARY.
** Channel monitoring
  15 MQCD-CHANNELMONITORING PIC S9(9) BINARY.
** Channel statistics
  15 MQCD-CHANNELSTATISTICS PIC S9(9) BINARY.
** Ver:8 **
** Limit on sharing conversations
  15 MQCD-SHARINGCONVERSATIONS PIC S9(9) BINARY.
** Message property control
  15 MQCD-PROPERTYCONTROL PIC S9(9) BINARY.
** Limit on SVRCONN channel instances
  15 MQCD-MAXINSTANCES PIC S9(9) BINARY.
** Limit on SVRCONN channel instances per client
  15 MQCD-MAXINSTANCESPERCLIENT PIC S9(9) BINARY.
** Client channel weight
  15 MQCD-CLIENTCHANNELWEIGHT PIC S9(9) BINARY.
** Connection affinity
  15 MQCD-CONNECTIONAFFINITY PIC S9(9) BINARY.
** Ver:9 **
** Batch data limit
  15 MQCD-BATCHDATALIMIT PIC S9(9) BINARY.
** Use Dead Letter Queue
  15 MQCD-USEDLQ PIC S9(9) BINARY.

```

```

** Default client reconnect option
 15 MQCD-DEFRECONNECT PIC S9(9) BINARY.
** Ver:10 **
** Certificate Label
 15 MQCD-CERTLABL PIC X (64)
** Ver:11 **
** AMS Security policy protection
 15 MQCD-SPLPROTECTION PIC S9(9) BINARY
** Ver:12 **

```

## **RPG bildirim (ILE)**

Bu bildirim, MQCD yapısına ilişkin RPG bildirimidir.

```

D* MQCD Structure
D*
D* Channel definition name
D CDCHN      1      20
D* Structure version number
D CDVER      21      24I 0
D* Channel type
D DCCHT      25      28I 0
D* Transport type
D CDTRT      29      32I 0
D* Channel description
D CDDDES     33      96
D* Queue manager name
D CDQM       97     144
D* Transmission queue name
D CDXQ      145     192
D* First 20 bytes of connection name
D CDSCN     193     212
D* Reserved
D CDMCA     213     232
D* LU 6.2 Mode name
D CDMOD     233     240
D* LU 6.2 transaction program name
D CDTP      241     304
D* Batch size
D CDBS      305     308I 0
D* Disconnect interval
D CDDI      309     312I 0
D* Short retry count
D CDSRC     313     316I 0
D* Short retry wait interval
D CDSRI     317     320I 0
D* Long retry count
D CDLRC     321     324I 0
D* Long retry wait interval
D CDLRI     325     328I 0
D* Channel security exit name
D CDSCX     329     348
D* Channel message exit name
D CDMSX     349     368
D* Channel send exit name
D CDSNX     369     388
D* Channel receive exit name
D CDRCX     389     408
D* Highest allowable message sequence number
D CDSNW     409     412I 0
D* Maximum message length
D CDMML     413     416I 0
D* Put authority
D CDPA      417     420I 0
D* Data conversion
D CDDC      421     424I 0
D* Channel security exit user data
D CDSCD     425     456
D* Channel message exit user data
D CDMSD     457     488
D* Channel send exit user data
D CDSND     489     520
D* Channel receive exit user data
D CDRCDD    521     552
D* Ver:1 **
D* User identifier
D CDUID     553     564
D* Password
D CDPW      565     576

```

```

D* First 12 bytes of MCA user identifier
D CDAUI          577    588
D* Message channel agent type
D CDCAT          589    592I 0
D* Connection name
D CDCON          593    848
D CDCN2          849    856
D* First 12 bytes of user identifier from partner
D CDRUI          857    868
D* Password from partner
D CDRPW          869    880
D* Ver:2 **
D* Channel message retry exit name
D CDMRX          881    900
D* Channel message retry exit user data
D CDMRD          901    932
D* Number of times MCA will try to put the message, after first
D* attempt has failed
D CDMRC          933    936I 0
D* Minimum interval in milliseconds after which the open or put
D* operation will be retried
D CDMRI          937    940I 0
D* Ver:3 **
D* Time in seconds between heartbeat flows
D CDHBI          941    944I 0
D* Batch duration
D CDBI           945    948I 0
D* Speed at which nonpersistent messages are sent
D CDNPM          949    952I 0
D* Length of MQCD structure
D CDLEN          953    956I 0
D* Length of exit name
D CDXNL          957    960I 0
D* Length of exit user data
D CDXDL          961    964I 0
D* Number of message exits defined
D CDMXD          965    968I 0
D* Number of send exits defined
D CDSXD          969    972I 0
D* Number of receive exits defined
D CDRXD          973    976I 0
D* Address of first MsgExit field
D CDMXP          977    992*
D* Address of first MsgUserData field
D CDMUP          993   1008*
D* Address of first SendExit field
D CDSXP         1009   1024*
D* Address of first SendUserData field
D CDSUP         1025   1040*
D* Address of first ReceiveExit field
D CDRXP         1041   1056*
D* Address of first ReceiveUserData field
D CDRUP         1057   1072*
D* Ver:4 **
D* Address of a list of cluster names
D CDCLP         1073   1088*
D* Number of clusters to which the channel belongs
D CDCLD         1089   1092I 0
D* Network priority
D CDNP          1093   1096I 0
D* Ver:5 **
D* Length of long MCA user identifier
D CDLML         1097   1100I 0
D* Length of long remote user identifier
D CDLRL         1101   1104I 0
D* Address of long MCA user identifier
D CDLMP         1105   1120*
D* Address of long remote user identifier
D CDLRP         1121   1136*
D* MCA security identifier
D CDMSI         1137   1176
D* Remote security identifier
D CDRSI         1177   1216
D* Ver:6 **
D* TLS CipherSpec
D CDSCS         1217   1248
D* Address of TLS peer name
D CDSPN         1249   1264*
D* Length of TLS peer name
D CDSPL         1265   1268I 0
D* Whether TLS client authentication is required
D CDSCA         1269   1272I 0

```

```

D* Keepalive interval
D CDKAI 1273 1276I 0
D* Local communications address
D CDLOA 1277 1324
D* Batch heartbeat interval
D CDBHB 1325 1328I 0
D* Ver:7 **
D* Header data compression list
D CDHCL0
D CDHCL1 1329 1332I 0
D CDHCL2 1333 1336I 0
D CDHCL 10I 0 DIM(2) OVERLAY(CDHCL0)
D* Message data compression list
D CDMCL0
D CDMCL1 1337 1340I 0
D CDMCL2 1341 1344I 0
D CDMCL3 1345 1348I 0
D CDMCL4 1349 1352I 0
D CDMCL5 1353 1356I 0
D CDMCL6 1357 1360I 0
D CDMCL7 1361 1364I 0
D CDMCL8 1365 1368I 0
D CDMCL9 1369 1372I 0
D CDMCL10 1373 1376I 0
D CDMCL11 1377 1380I 0
D CDMCL12 1381 1384I 0
D CDMCL13 1385 1388I 0
D CDMCL14 1389 1392I 0
D CDMCL15 1393 1396I 0
D CDMCL16 1397 1400I 0
D CDMCL 10I 0 DIM(16) OVERLAY(CDMCL0)
D* Channel rank
D CDCWCR 1401 1404I 0
D* Channel priority
D CDCWCP 1405 1408I 0
D* Channel weight
D CDCWCW 1409 1412I 0
D* Channel monitoring
D CDCHLMON 1413 1416I 0
D* Channel statistics
D CDCHLST 1417 1420I 0
D* Ver:8 **
D* Limit on sharing conversations
D CDSHC 1421 1424I 0
D* Message property control
D CDPRC 1425 1428I 0
D* Limit on SVRCONN channel instances
D CDMXIN 1429 1432I 0
D* Limit on SVRCONN channel instances per client
D CDMXIC 1433 1436I 0
D* Client channel weight
D CDCLNCHLW 1437 1440I 0
D* Connection affinity
D CDCONNAFF 1441 1444I 0
D* Ver:9 **
D* Batch data limit
D CDBDL 1445 1448I 0
D* Use Dead Letter Queue
D CDUDLQ 1449 1452I 0
D* Default client reconnect option
D CDDRCN 1453 1456I 0
D* Ver:10 **

```

## System/390 çevirici bildirimi

Bu bildirim, MQCD yapısı için System/390 çevirici bildirimidir.

### V9.2.0

MQCD	DSECT		
MQCD_CHANNELNAME	DS	CL20	Channel definition name
MQCD_VERSION	DS	F	Structure version number
MQCD_CHANNELTYPE	DS	F	Channel type
MQCD_TRANSPORTTYPE	DS	F	Transport type
MQCD_DESC	DS	CL64	Channel description
MQCD_QMGRNAME	DS	CL48	Queue manager name
MQCD_XMITQNAME	DS	CL48	Transmission queue name
MQCD_SHORTCONNECTIONNAME	DS	CL20	First 20 bytes of connection name
*			name
MQCD_MCANAME	DS	CL20	Reserved

MQCD_MODENAME	DS	CL8	LU 6.2 Mode name
MQCD_TPNAME	DS	CL64	LU 6.2 transaction program name
MQCD_BATCHSIZE	DS	F	Batch size
MQCD_DISCINTERVAL	DS	F	Disconnect interval
MQCD_SHORTRETRYCOUNT	DS	F	Short retry count
MQCD_SHORTRETRYINTERVAL	DS	F	Short retry wait interval
MQCD_LONGRETRYCOUNT	DS	F	Long retry count
MQCD_LONGRETRYINTERVAL	DS	F	Long retry wait interval
MQCD_SECURITYEXIT	DS	CLn	Channel security exit name
MQCD_MSGEXIT	DS	CLn	Channel message exit name
MQCD_SENDEXIT	DS	CLn	Channel send exit name
MQCD_RECEIVEEXIT	DS	CLn	Channel receive exit name
MQCD_SEQNUMBERWRAP	DS	F	Highest allowable message sequence number
*			
MQCD_MAXMSGLLENGTH	DS	F	Maximum message length
MQCD_PUTAUTHORITY	DS	F	Put authority
MQCD_DATACONVERSION	DS	F	Data conversion
MQCD_SECURITYUSERDATA	DS	CL32	Channel security exit user data
MQCD_MSGUSERDATA	DS	CL32	Channel message exit user data
MQCD_SENDUSERDATA	DS	CL32	Channel send exit user data
MQCD_RECEIVEUSERDATA	DS	CL32	Channel receive exit user data
MQCD_USERIDENTIFIER	DS	CL12	User identifier
MQCD_PASSWORD	DS	CL12	Password
MQCD_MCAUSERIDENTIFIER	DS	CL12	First 12 bytes of MCA user identifier
*			
MQCD_MCATYPE	DS	F	Message channel agent type
MQCD_CONNECTIONNAME	DS	CL264	Connection name
MQCD_REMOTEUSERIDENTIFIER	DS	CL12	First 12 bytes of user identifier from partner
*			
MQCD_REMOTEPASSWORD	DS	CL12	Password from partner
MQCD_MSGRETRYEXIT	DS	CLn	Channel message retry exit name
MQCD_MSGRETRYUSERDATA	DS	CL32	Channel message retry exit user data
*			
MQCD_MSGRETRYCOUNT	DS	F	Number of times MCA will try to put the message, after the first attempt has failed
*			
MQCD_MSGRETRYINTERVAL	DS	F	Minimum interval in milliseconds after which the open or put operation will be retried
*			
MQCD_HEARTBEATINTERVAL	DS	F	Time in seconds between heartbeat flows
*			
MQCD_BATCHINTERVAL	DS	F	Batch duration
MQCD_NONPERSISTENTMSGSPPEED	DS	F	Speed at which nonpersistent messages are sent
*			
MQCD_STRUCLNGTH	DS	F	Length of MQCD structure
MQCD_EXITNAMELENGTH	DS	F	Length of exit name
MQCD_EXITDATALENGTH	DS	F	Length of exit user data
MQCD_MSGEXITSDEFINED	DS	F	Number of message exits defined
MQCD_SENDEXITSDEFINED	DS	F	Number of send exits defined
MQCD_RECEIVEEXITSDEFINED	DS	F	Number of receive exits defined
MQCD_MSGEXITPTR	DS	F	Address of first MSGEXIT field
MQCD_MSGUSERDATAPTR	DS	F	Address of first MSGUSERDATA field
*			
MQCD_SENDEXITPTR	DS	F	Address of first SENDEXIT field
MQCD_SENDUSERDATAPTR	DS	F	Address of first SENDUSERDATA field
*			
MQCD_RECEIVEEXITPTR	DS	F	Address of first RECEIVEEXIT field
*			
MQCD_RECEIVEUSERDATAPTR	DS	F	Address of first RECEIVEUSERDATA field
*			
MQCD_CLUSTERPTR	DS	F	Address of a list of cluster names
*			
MQCD_CLUSTERSDEFINED	DS	F	Number of clusters to which the channel belongs
*			
MQCD_NETWORKPRIORITY	DS	F	Network priority
MQCD_LONGMCAUSERIDLENGTH	DS	F	Length of long MCA user identifier
*			
MQCD_LONGREMOTEUSERIDLENGTH	DS	F	Length of long remote user identifier
*			
MQCD_LONGMCAUSERIDPTR	DS	F	Address of long MCA user identifier
*			
MQCD_LONGREMOTEUSERIDPTR	DS	F	Address of long remote user identifier
*			
MQCD_MCASECURITYID	DS	XL40	MCA security identifier
MQCD_REMOTESECURITYID	DS	XL40	Remote security identifier
MQCD_SSLCIPHERSPEC	DS	CL32	TLS CipherSpec
MQCD_SSLPEERNAMEPTR	DS	F	Address of TLS peer name
MQCD_SSLPEERNAMELENGTH	DS	F	Length of TLS peer name
MQCD_SSLCLIENTAUTH	DS	F	Whether TLS client authentication is required
*			
MQCD_KEEPLIVEINTERVAL	DS	F	Keepalive interval

MQCD_LOCALADDRESS	DS	CL48	Local communications address
MQCD_BATCHHEARTBEAT	DS	F	Batch heartbeat interval
MQCD_HDRCOMPLIST	DS	CL2	Header data compression list
MQCD_MSGCOMPLIST	DS	CL16	Message data compression list
MQCD_CLWLCHANNELRANK	DS	F	Channel rank
MQCD_CLWLCHANNELPRIORITY	DS	F	Channel priority
MQCD_CLWLCHANNELWEIGHT	DS	F	Channel weight
MQCD_CHANNELMONITORING	DS	F	Channel monitoring
MQCD_CHANNELSTATISTICS	DS	F	Channel statistics
MQCD_SHARINGCONVERSATIONS	DS	F	Limit on sharing conversations
* MQCD_PROPERTYCONTROL	DS	F	Message property control
* MQCD_SHARINGCONVERSATIONS	DS	F	Limit on sharing conversations
MQCD_PROPERTYCONTROL	DS	F	Message property control
MQCD_MAXINSTANCES	DS	F	Limit on SVRCONN chl instances
MQCD_MAXINSTANCESPERCLIENT	DS	F	Limit on SVRCONN chl instances per client
MQCD_CLIENTCHANNELWEIGHT	DS	F	Channel weight
MQCD_CONNECTIONAFFINITY	DS	F	Connection Affinity
MQCD_BATCHDATALIMIT	DS	F	Batch data limit
MQCD_USEDLQ	DS	F	Use dead-letter queue
MQCD_DEFRECONNECT	DS	F	Default client reconnect option
MQCD_CERTLABL	DS	F	Certificate label
MQCD_SPLPROTECTION	DS	F	AMS Security policy protection
MQCD_LENGTH	EQU	*-MQCD	
	ORG	MQCD	
MQCD_AREA	DS	CL(MQCD_LENGTH)	

## Visual Basic bildirim

Bu bildirim, MQCD yapısındaki Visual Basic bildirimidir.

Visual Basic 'te, MQCONNX çağrısında MQCNO yapıyla MQCD yapıyla kullanılabilir.

Type	MQCD	
ChannelName	As String*20	'Channel definition name'
Version	As Long	'Structure version number'
ChannelType	As Long	'Channel type'
TransportType	As Long	'Transport type'
Desc	As String*64	'Channel description'
QMgrName	As String*48	'Queue manager name'
XmitQName	As String*48	'Transmission queue name'
ShortConnectionName	As String*20	'First 20 bytes of connection name'
MCAName	As String*20	'Reserved'
ModeName	As String*8	'LU 6.2 Mode name'
TpName	As String*64	'LU 6.2 transaction program name'
BatchSize	As Long	'Batch size'
DiscInterval	As Long	'Disconnect interval'
ShortRetryCount	As Long	'Short retry count'
ShortRetryInterval	As Long	'Short retry wait interval'
LongRetryCount	As Long	'Long retry count'
LongRetryInterval	As Long	'Long retry wait interval'
SecurityExit	As String*128	'Channel security exit name'
MsgExit	As String*128	'Channel message exit name'
SendExit	As String*128	'Channel send exit name'
ReceiveExit	As String*128	'Channel receive exit name'
SeqNumberWrap	As Long	'Highest allowable message sequence number'
MaxMsgLength	As Long	'Maximum message length'
PutAuthority	As Long	'Put authority'
DataConversion	As Long	'Data conversion'
SecurityUserData	As String*32	'Channel security exit user data'
MsgUserData	As String*32	'Channel message exit user data'
SendUserData	As String*32	'Channel send exit user data'
ReceiveUserData	As String*32	'Channel receive exit user data'
UserIdentifier	As String*12	'User identifier'
Password	As String*12	'Password'
MCAUserIdentifier	As String*12	'First 12 bytes of MCA user identifier'
MCAType	As Long	'Message channel agent type'
ConnectionName	As String*264	'Connection name'
RemoteUserIdentifier	As String*12	'First 12 bytes of user identifier from partner'
RemotePassword	As String*12	'Password from partner'
MsgRetryExit	As String*128	'Channel message retry exit name'
MsgRetryUserData	As String*32	'Channel message retry exit user data'

MsgRetryCount	As Long	'Number of times MCA will try to 'put the message, after the' 'first attempt has failed'
MsgRetryInterval	As Long	'Minimum interval in' 'milliseconds after which the' 'open or put operation will be' 'retried'
HeartbeatInterval	As Long	'Time in seconds between' 'heartbeat flows'
BatchInterval	As Long	'Batch duration'
NonPersistentMsgSpeed	As Long	'Speed at which nonpersistent' 'messages are sent'
StrucLength	As Long	'Length of MQCD structure'
ExitNameLength	As Long	'Length of exit name'
ExitDataLength	As Long	'Length of exit user data'
MsgExitsDefined	As Long	'Number of message exits defined'
SendExitsDefined	As Long	'Number of send exits defined'
ReceiveExitsDefined	As Long	'Number of receive exits defined'
MsgExitPtr	As MQPTR	'Address of first MsgExit field'
MsgUserDataPtr	As MQPTR	'Address of first MsgUserData' 'field'
SendExitPtr	As MQPTR	'Address of first SendExit field'
SendUserDataPtr	As MQPTR	'Address of first SendUserData' 'field'
ReceiveExitPtr	As MQPTR	'Address of first ReceiveExit' 'field'
ReceiveUserDataPtr	As MQPTR	'Address of first' 'ReceiveUserData field'
ClusterPtr	As MQPTR	'Address of a list of cluster' 'names'
ClustersDefined	As Long	'Number of clusters to which the' 'channel belongs'
NetworkPriority	As Long	'Network priority'
LongMCAUserIdLength	As Long	'Length of long MCA user' 'identifier'
LongRemoteUserIdLength	As Long	'Length of long remote user' 'identifier'
LongMCAUserIdPtr	As MQPTR	'Address of long MCA user' 'identifier'
LongRemoteUserIdPtr	As MQPTR	'Address of long remote user' 'identifier'
MCASecurityId	As MQBYTE40	'MCA security identifier'
RemoteSecurityId	As MQBYTE40	'Remote security identifier'
SSLCipherSpec	As String*32	'TLS CipherSpec'
SSLPeerNamePtr	As MQPTR	'Address of TLS peer name'
SSLPeerNameLength	As Long	'Length of TLS peer name'
SSLClientAuth	As Long	'Whether TLS client' 'authentication is required'
KeepAliveInterval	As Long	'Keepalive interval'
LocalAddress	As String*48	'Local communications address'
BatchHeartbeat	As Long	'Batch heartbeat interval'
HdrCompList(0 to 1)	As Long2	'Header data compression list'
MsgCompList(0 to 15)	As Long16	'Message data compression list'
CLWLChannelRank	As Long	'Channel Rank'
CLWLChannelPriority	As Long	'Channel priority'
CLWLChannelWeight	As Long	'Channel Weight'
ChannelMonitoring	As Long	'Channel Monitoring control'
ChannelStatistics	As Long	'Channel Statistics'
End Type		

### ***Kanal çıkışındaki MQCD alanlarının değiştirilmesi***

Bir kanal çıkışı, MQCD 'deki alanları değiştirebilir. Ancak, listelenen koşullar dışında, bu değişiklikler genellikle üzerinde işlem yapmamış olur.

Bir kanal çıkış programı, MQCD veri yapısındaki bir alanı değiştirirse, yeni değer tipik olarak IBM MQ kanal işlemi tarafından yoksayılır. Ancak, yeni değer MQCD 'de kalır ve bir çıkış zincirindeki geri kalan çıkışlara ve kanal yönetim ortamını paylaşan herhangi bir konuşmaya geçirilir.

MQCXP yapısında SharingConversations değeri FALSE olarak ayarlanırsa, çıkış programının tipine, kanal tipine ve çıkış neden koduna bağlı olarak, belirli alanlarda yapılan değişiklikler üzerinde işlem yapılabilir. Aşağıdaki tabloda, çevrenin işleyişini ve kanalın davranışını etkileyebilecek alanları ve hangi koşullarda etkilenebileceğini gösterir. Bir çıkış programı bu alanlardan birini başka herhangi bir koşulda değiştirirse ya da listede olmayan herhangi bir alanı değiştirirse, yeni değer kanal işlemi tarafından yoksayılır. Yeni değer MQCD 'de kalır ve bir çıkış zincirindeki tüm çıkışlara ve kanal yönetim ortamını paylaşan tüm çıkışlara geçirilir.



Başlatma için çağrıldığında herhangi bir çıkış programı tipi (MQXR\_INIT), MQCXP SharingConverstions FALSE olarak ayarlandığı sürece herhangi bir kanaldaki ChannelName alanını değiştirebilir. MQCXP SharingConverstionsdeğeri ne olursa olsun, yalnızca bir güvenlik çıkışı MCAUserIdentifier alanını değiştirebilir.

<i>Çizelge 824. Çevrenin işleyişini değiştirebilen ve işleyebilen alanlar</i>			
<b>Alan</b>	<b>Çıkış nedeni kodu</b>	<b>Çıkış tipi</b>	<b>Kanal tipi</b>
ChannelName	MQXR_INIT	Tümü	Tümü
TransportType	MQXR_INIT	Tümü	Tümü
XmitQName	MQXR_INIT	Tümü	SDR, RCVR
ModeName	MQXR_INIT	Tümü	Tümü
TpName	MQXR_INIT	Tümü	Tümü
BatchSize	MQXR_INIT	Tümü	SDR, SVR, RCVR, RQSTR, CLUSSDR, CLUSRCVR
DiscInterval	MQXR_INIT	Tümü	SDR, SVR, RCVR, RQSTR, CLUSSDR, CLUSRCVR
ShortRetrySayı	MQXR_INIT	Tümü	SDR, SVR, RCVR, RQSTR, CLUSSDR, CLUSRCVR
ShortRetryAralığı	MQXR_INIT	Tümü	SDR, SVR, RCVR, RQSTR, CLUSSDR, CLUSRCVR
LongRetrySayısı	MQXR_INIT	Tümü	SDR, SVR, RCVR, RQSTR, CLUSSDR, CLUSRCVR
LongRetryAralığı	MQXR_INIT	Tümü	SDR, SVR, RCVR, RQSTR, CLUSSDR, CLUSRCVR
SeqNumberKaydır	MQXR_INIT	Tümü	SDR, SVR, RCVR, RQSTR, CLUSSDR, CLUSRCVR
MaxMsgUzunluğu	MQXR_INIT	Tümü	Tümü

<i>Çizelge 824. Çevrenin işleyişini değiştirebilen ve işleyebilen alanlar (devamı var)</i>			
<b>Alan</b>	<b>Çıkış nedeni kodu</b>	<b>Çıkış tipi</b>	<b>Kanal tipi</b>
PutAuthority	MQXR_INIT	Tümü	SDR, SVR, RCVR, RQSTR, CLUSSDR, CLUSRCVR
DataConversion	MQXR_INIT	Tümü	Tümü
MCAUserIdentifier	MQXR_INIT, MQXR_INIT_SEC, MQXR_SEC_MSG, MQXR_SEC_PARMS	Güvenlik	RCVR, RQSTR, SVRCONN, CLUSRCVR
ConnectionName	MQXR_INIT	Tümü	SDR, SVR, RQSTR, CLNTCONN, CLUSSDR, CLUSRCVR
MsgRetryUserData	MQXR_INIT	Tümü	RCVR, RQSTR, CLUSRCVR
MsgRetrySayısı	MQXR_INIT	Tümü	RCVR, RQSTR, CLUSRCVR
MsgRetryAralığı	MQXR_INIT	Tümü	RCVR, RQSTR, CLUSRCVR
HeartbeatInterval	MQXR_INIT	Tümü	Tümü
BatchInterval	MQXR_INIT	Tümü	SDR, SVR, CLUSSDR, CLUSRCVR
NonPersistentMsgSpeed	MQXR_INIT	Tümü	SDR, SVR, RCVR, RQSTR, CLUSSDR, CLUSRCVR
MCASecurityId	MQXR_INIT, MQXR_INIT_SEC, MQXR_SEC_MSG, MQXR_SEC_PARMS	Güvenlik	SDR, SVR, RCVR, RQSTR, SVRCONN, CLUSSDR, CLUSRCVR
SSLCipherSpec	MQXR_INIT	Tümü	Tümü
SSLPeerNamePtr	MQXR_INIT	Tümü	Tümü
SSLPeerNameUzunluğu	MQXR_INIT	Tümü	Tümü
SSLClientAuth	MQXR_INIT	Tümü	SVR, RCVR, RQSTR, SVRCONN, CLUSRCVR

Çizelge 824. Çevrenin işleyişini değiştirebilen ve işleyebilen alanlar (devamı var)			
Alan	Çıkış nedeni kodu	Çıkış tipi	Kanal tipi
KeepAliveAralığı	MQXR_INIT	Tümü	Tümü
LocalAddress	MQXR_INIT	Tümü	SDR, SVR, RQSTR, CLNTCONN, CLUSSDR, CLUSRCVR
BatchHeartbeat	MQXR_INIT	Tümü	SDR, SVR, CLUSSDR, CLUSRCVR
HdrCompListesi	MQXR_INIT	Tümü	Tümü
MsgCompListesi	MQXR_INIT	Tümü	Tümü
ChannelMonitoring	MQXR_INIT	Tümü	SDR, SVR, RCVR, RQSTR, SVRCONN, CLUSSDR, CLUSRCVR
ChannelStatistics	MQXR_INIT	Tümü	SDR, SVR, RCVR, RQSTR, CLUSSDR, CLUSRCVR
SharingConversations	MQXR_INIT	Tümü	SVRCONN, CLNTCONN
PropertyControl	MQXR_INIT	Tümü	SDR, SVR, CLUSSDR, CLUSRCVR

## MQCXP-Kanal çıkış değiştirgesi

MQCXP yapısı, Message Channel Agent (MCA), istemci-bağlantı kanalı ya da sunucu bağlantısı kanalı tarafından çağrılan her bir çıkış tipine geçirilir.

Bkz. MQ\_CHANNEL\_EXIT.

Çıkışta "çıkışa giriş" olarak tanımlanan alanlar, çıkışta kanala denetim döndürdüğünde kanal tarafından dikkate alınmaz. Kanal çıkışı parametre blokundaki çıkış değişikliklerinin sonraki çağırması için korunmayacak olan giriş alanları korunmaz. Giriş/çıkış alanlarında (örneğin, *ExitUserArea* alanı) yapılan değişiklikler, yalnızca çıkışın o örneğine ilişkin çağrılar için korunur. Bu tür değişiklikler, aynı kanalda tanımlanan farklı çıkışlar arasında ya da farklı kanallarda tanımlanan aynı çıkış arasında veri aktarmak için kullanılamaz.

### İlgili başvurular

["Alanlar" sayfa 1496](#)

Bu konu, MQCXP yapısındaki tüm alanları listeler ve her bir alanı açıklar.

["C bildirim" sayfa 1506](#)

Bu bildirim, MQCXP yapısına ilişkin C bildirimidir.

["COBOL bildirim" sayfa 1507](#)

Bu bildirim, MQCXP yapısına ilişkin COBOL bildirimidir.

“RPG bildirim (ILE)” sayfa 1508

Bu bildirim, MÖCXP yapısına ilişkin RPG bildirimidir.

“System/390 çevirici bildirim” sayfa 1509

Bu bildirim, MÖCXP yapısına ilişkin System/390 çevirici bildirimidir.

## **Alanlar**

Bu konu, MÖCXP yapısındaki tüm alanları listeler ve her bir alanı açıklar.

*StrucId (MÖCHAR4)*

Bu alan, yapı tanıtcısını belirtir.

Değer şu olmalıdır:

### **MÖCXP\_STRUC\_ID**

Kanal çıkış parametresi yapısıyla ilgili tanıtcı.

C programlama dili için, sabit MÖCXP\_STRUC\_ID\_ARRAY değişmezi de tanımlanır; bu değişmez, MÖCXP\_STRUC\_ID ile aynı değere sahiptir, ancak dizgi yerine bir dizi karakter dizisidir.

Bu, çıkışa giriş alanıdır.

*Sürüm (MÖUZE)*

Bu alan, yapı sürüm numarasını belirtir.



Değer, ortama bağlıdır:

### **MÖCXP\_VERSION\_1**

Version-1 kanal çıkışı parametre yapısı.

### **MÖCXP\_VERSION\_3**

Version-3 kanal çıkışı parametre yapısı.

  Bu değer, AIX and Linux sistemlerinde başka bir yerde listelenmeyen bir değer içerir.

### **MÖCXP\_VERSION\_4**

Version-4 kanal çıkış parametresi yapısı.

### **MÖCXP\_VERSION\_5**

Version-5 kanal çıkışı parametre yapısı.

### **MÖCXP\_VERSION\_6**

Version-6 kanal çıkışı parametre yapısı.

### **MÖCXP\_VERSION\_8**

Version-8 kanal çıkışı parametre yapısı.

 Alan, z/OSiçinde bu değere sahiptir.

### **MÖCXP\_VERSION\_9**

Version-9 kanal çıkış parametresi yapısı.

Alan, aşağıdaki ortamlarda bu değere sahiptir:

-  AIX
-  IBM i
-  Linux
-  Windows
-  z/OS

Yalnızca yapının daha yeni sürümlerinde var olan alanlar, alanların açıklamalarında olduğu gibi tanımlanır. Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

**MQCXP\_XX\_ENCODE\_CASE\_ONE current\_version**

Kanal çıkışı deęiřtirge yapısının yürürlükteki sürümü.

Deęer, ortama baęlıdır.

**Not:** MQCXP yapısının yeni bir sürümü kullanıma sunulduğunda, var olan parçanın yerleřim düzeni deęiřtirilmez. Bu nedenle, çıkıřta, sürüm numarasının, çıkıřa gereken alanları içeren en düşük sürüme eřit ya da bu sürümden daha büyük olduğunu kontrol etmelidir.

Bu, çıkıřa giriř alanıdır.

*ExitId (MQUZE)*

Bu alan çağrılmakta olan çıkıř tipini belirtir ve çıkıř yordamlarına giriř için ayarlanır.

Ařaęıdaki deęerler olanaklı:

**MQXT\_CHANNEL\_SEC\_EXIT**

Kanal güvenlik çıkıřı.

**MQXT\_CHANNEL\_MSG\_EXIT**

Kanal ileti çıkıřı.

**MQXT\_CHANNEL\_SEND\_EXIT**

Kanal çıkıřı çıkıřı.

**MQXT\_CHANNEL\_RCV\_EXIT**

Kanal çıkıř çıkıřı.

**MQXT\_CHANNEL\_MSG\_RETRY\_EXIT**

Kanal iletisi-yeniden deneme çıkıřı.

**MQXT\_CHANNEL\_AUTO\_DEF\_EXIT**

Kanal otomatik tanımlama çıkıřı.

z/OSüzerinde, bu çıkıř tipi yalnızca MQCHT\_CLUSSDR ve MQCHT\_CLAUSRCVR tipi kanallar için desteklenir.

Bu, çıkıřa giriř alanıdır.

*ExitReason (MQUZE)*

Bu alan, çıkıřa neden çağrılmakta olduğunu ve çıkıř yordamlarına giriř için ayarlanıp belirlenmemesine neden olur.

Otomatik tanımlama çıkıřı tarafından kullanılmaz. Ařaęıdaki deęerler olanaklı:

**MQXR\_INIT**

Başlatma işleminden çıkılıyor.

Bu deęer, çıkıřa ilk kez çağrılmakta olduğunu gösterir. Çıkıřa, gereksinim duyduğu kaynakları (örneğin: bellek) edinip kullanıma hazırlamaya olanak tanır.

**MQXR\_TERM**

Sondan çıkın.

Bu deęer, çıkıřa son verilmek üzere olduğunu gösterir. Çıkıř, kullanıma hazırlandığı için (örneğin: bellek) kullanıma hazırlandığı tüm kaynakları serbest kılmalıdır.

**MQXR\_MSG**

Bir iletiyi işle.

Bu deęer, bir iletiyi işlemek için çıkıřa çağrıldığını gösterir. Bu deęer yalnızca kanal iletisi çıkıřları için gerçekleşir.

**MQXR\_XMIT**

İletiřimi işleme.

Bu deęer, kanal gönderme ve alma çıkıřları için gerçekleşir.

## **MQXR\_SEC\_MSG**

Güvenlik iletisi alındı.

Bu değer yalnızca kanal güvenliği çıkışları için gerçekleşir.

## **MQXR\_INIT\_SEC**

Güvenlik değiş tokası başlatın.

Bu değer yalnızca kanal güvenliği çıkışları için gerçekleşir.

Alıcısının güvenlik çıkışı, bir güvenlik değiş tokası başlatma olanağı vermek için her zaman bu nedenle MQXR\_INIT ile çağrıldıktan sonra çağrılır. İş fırsatını keserse (MQXCC\_SEND\_SEC\_MSG ya da MQXCC\_SEND\_AND\_REQUEST\_SEC\_MSG yerine MQXCC\_OK döndürerek), gönderenin güvenlik çıkışı MQXR\_INIT\_SEC ile çağrılır.

Alıcının güvenlik çıkışı bir güvenlik değiş tokası başlatırsanız (MQXCC\_SEND\_SEC\_MSG ya da MQXCC\_SEND\_AND\_REQUEST\_SEC\_MSG döndürerek), gönderenin güvenlik çıkışı hiçbir zaman MQXR\_INIT\_SEC; ile çağrılmaz; bunun yerine, alıcıya ilişkin iletiyi işlemek için MQXR\_SEC\_MSG ile çağrılır. (Her iki durumda da, önce MQXR\_INIT ile çağrılır.)

Güvenlik çıkışlarından biri, kanalın sonlanmasını istemezse (*ExitResponse* , MQXCC\_SUPPRESS\_FUNCTION ya da MQXCC\_CLOSE\_CHANNEL değerine ayarlanarak), güvenlik değiş tokası, değiş tokuş başlatan tarafta tamamlanmalıdır. Bu nedenle, bir güvenlik çıkışı MQXR\_INIT\_SEC ile çağrılırsa ve bir değiş tokuş başlatıyorsa, çıkışta bir sonraki başlatılırsa, bu çıkış MQXR\_SEC\_MSG ile olur. Bu durum, çıkışa ilişkin bir güvenlik iletisi olup olmadığı ya da işlememesinin bir güvenlik iletisi olup olmadığı. Ortak MQXCC\_SEND\_SEC\_MSG ya da MQXCC\_SEND\_AND\_REQUEST\_SEC\_MSG değerini döndürürse, ancak ortak MQXCC\_OK değerini döndürürse ya da iş ortağında güvenlik çıkışı yoksa, bir güvenlik iletisi vardır. İşleyecek bir güvenlik iletisi yoksa, başlangıç uçundaki güvenlik çıkışı, sıfır *DataLength* ile yeniden çağrılır.

## **MQXR\_RETRY**

İletiyi yeniden deneyin.

Bu değer, ileti-yeniden deneme çıkışları için gerçekleşir.

## **MQXR\_AUTO\_CLUSSDR**

Bir küme gönderen kanalının otomatik tanımı.

Bu değer, yalnızca kanal otomatik tanımlama çıkışları için gerçekleşir.

## **MQXR\_AUTO\_RECEIVER**

Bir alıcı kanalının otomatik tanımlaması.

Bu değer, yalnızca kanal otomatik tanımlama çıkışları için gerçekleşir.

## **MQXR\_AUTO\_SVRCONN**

Bir sunucu bağlantısı kanalının otomatik tanımlaması.

Bu değer, yalnızca kanal otomatik tanımlama çıkışları için gerçekleşir.

## **MQXR\_AUTO\_CLUSRCVR**

Bir küme alıcı kanalının otomatik tanımlaması.

Bu değer, yalnızca kanal otomatik tanımlama çıkışları için gerçekleşir.

## **MQXR\_SEC\_PARMS**

Güvenlik değiştirgeleri

Bu değer yalnızca güvenlik çıkışları için geçerlidir ve çıkışa MQCSP yapısının geçirildiğini gösterir. Daha fazla bilgi için bkz. [“MQCSP-Güvenlik değiştirgeleri” sayfa 335](#)

### **Not:**

1. Bir kanal için birden fazla çıkış tanımlanmışsa, MCA kullanıma hazırlandığında, bunların her biri MQXR\_INIT ile çağrılır. Ayrıca, MCA sonlandırıldığında her biri MQXR\_TERM ile çağrılır.
2. Kanal otomatik tanımlama çıkışı için, *Version* değeri MQCXP\_VERSION\_4 değerinden küçükse *ExitReason* ayarlı değildir. Bu durumda MQXR\_AUTO\_SVRCONN değeri örtük olarak ifade edilir.

Bu, çıkışa giriş alanıdır.

*ExitResponse (MQUZE)*

Bu alan, çıkıştan gelen yanıtı belirtir.

Bu alan, MCA ile iletişim kurmak için çıkışa göre ayarlanır. Bu değer aşağıdaki değerlerden biri olmalıdır:

#### **MQXCC\_OK**

Çıkış başarıyla tamamlandı.

- Kanal güvenliği çıkışı için bu değer, ileti aktarımlarının artık olağan bir şekilde devam edebileceğini gösterir.
- Kanal iletişi yeniden deneme çıkışı için bu değer, MCA 'nın MQCXP' deki *MsgRetryInterval* alanında çıkış tarafından döndürülen zaman aralığını beklemesi gerektiğini ve daha sonra, iletiyi yeniden deneyin gerektiğini gösterir.

*ExitResponse2* alanı ek bilgi içerebilir.

#### **MQXCC\_SUPPRESS\_FUNCTION**

İşlevi engelle.

- Kanal güvenliği çıkışı için bu değer, kanalın sonlandırılması gerektiğini gösterir.
- Kanal ileti çıkışı için bu değer, iletinin hedefine doğru ilerlemek için olmadığını gösterir. Bunun yerine, MCA bir kural dışı durum raporu iletişi üretir (özgün iletiyi gönderen kişi tarafından istendiyse) ve iletiyi, özgün arabelleğdeki (gönderen MQRO\_DEAD\_LETTER\_Q) özgün arabelleğde yer alır ya da atar (gönderen MQRO\_DISCARD\_MSG belirtirse).

Kalıcı iletiler için, gönderen MQRO\_DEAD\_LETTER\_Q belirtirse, ancak, ölü-mektup kuyruğuna koyma başarısız olursa ya da herhangi bir ölü harf kuyruğu yoksa, özgün ileti iletim kuyruğunda bırakılır ve rapor iletişi oluşturulmaz. Rapor iletişi başarılı bir şekilde oluşturulamazsa, özgün ileti iletim kuyruğunda da bırakılır.

İletinin ölü-harf kuyruğunda iletinin başlangıcındaki MQDLH yapısındaki *Feedback* alanı, iletinin neden ölü-mektup kuyruğuna yerleştirdiğini gösterir; bu geribildirim kodu, (gönderen tarafından istendiyse) kural dışı durum raporu iletisinin ileti tanımlayıcısında da kullanılır.

- Kanal iletişi yeniden deneme çıkışı için, bu değer MCA 'nın beklemediği ve iletiyi yeniden denediğini gösterir; bunun yerine, MCA olağan hata işlemleriyle hemen devam eder (ileti, iletinin göndericisiyle belirtildiği gibi, ileti, ölü-mektup kuyruğuna yerleştirilir ya da atılır).
- Kanal otomatik tanımlama çıkışı için, MQXCC\_OK ya da MQXCC\_SUPPRESS\_FUNCTION belirtilmeli. Bu değerlerin hiçbiri belirtilmediyse, varsayılan olarak MQXCC\_SUPPRESS\_FUNCTION varsayılan olarak kabul edilir ve otomatik tanımlama iptal edilir.

Bu yanıt kanal gönderme ve alma çıkışları için desteklenmiyor.

#### **MQXCC\_SEND\_SEC\_MSG**

Güvenlik iletişi gönder.

Bu değer yalnızca kanal güvenliği çıkışı ile ayarlanabilir. Çıkışta, iş ortağına iletilmesi gereken bir güvenlik iletişi sağlamış olduğunu gösterir.

#### **MQXCC\_SEND\_AND\_REQUEST\_SEC\_MSG**

Yanıt gerektiren güvenlik iletisini gönderin.

Bu değer yalnızca kanal güvenliği çıkışı ile ayarlanabilir. Bu işaret

- çıkışta, iş ortağına iletebilecek bir güvenlik mesajı sunduğunu, ve
- bu çıkışa ortaktan bir yanıt gerekiyor. Herhangi bir yanıt alınmazsa, çıkışta henüz iletişimin devam edip edemeyeceği konusunda karar vermediği için kanal sonlandırılmalıdır.

#### **MQXCC\_SUPPRESS\_EXIT**

Çıkışı engelle.

- Bu değer, güvenlik çıkışı ya da otomatik tanımlama çıkışı dışında tüm kanal çıkışı tipleri tarafından ayarlanabilir. It suppresses any further invocation of that exit (as if its name had been blank in

the channel definition), until termination of the channel, when the exit is again invoked with an *ExitReason* of MQXR\_TERM.

- If a message retry exit returns this value, message retries for subsequent messages are controlled by the *MsgRetryCount* and *MsgRetryInterval* channel attributes as normal. For the current message, the MCA performs the number of outstanding retries, at intervals given by the *MsgRetryInterval* channel attribute, but only if the reason code is one that the MCA would normally retry (see the *MsgRetryCount* field described in “MQCD-Kanal tanımlaması” sayfa 1456 ). Bekleyen yeniden deneme sayısı, **MsgRetryCount** özniteliğinin değeri, çıkışa ilişkin MQXCC\_OK ' un döndürdüğü sayı sayısı daha az; bu sayı negatifse, yürürlükteki ileti için MCA tarafından başka yeniden denemeler gerçekleştirilmez.

### **MQXCC\_CLOSE\_CHANNEL**

Kanalı kapat.

Bu değer, otomatik tanımlama çıkışı dışında herhangi bir kanal çıkışı tipine göre ayarlanabilir.

Paylaşım konuşmaları etkinleştirilmezse, bu değer kanalı kapatır.

Sohbet paylaşımı etkinleştirilirse, bu değer etkileşimi sona erdirir. Kanaldaki tek konuşma bu konuşmaya devam ederse kanal da kapanır.

Bu alan, çıkışa ait bir giriş/çıkış alanıdır.

*ExitResponse2 (MQUZE)*

Bu alan, çıkışa ilişkin ikincil yanıtı belirtir.

Bu alan, çıkış yordamlarına girişte sıfır olarak ayarlanır. Bu, IBM MQ kanal işlevlerine daha fazla bilgi sağlamak için çıkışa göre ayarlanabilir. Otomatik tanımlama çıkışı tarafından kullanılmaz.

Çıkış, aşağıdaki değerlerden birini ya da birkaçını ayarlayabilir. Birden çok değer gerekiyorsa, değerler eklenir. Geçerli olmayan birleşimler not edilir; diğer birleşimlerine izin verilir.

### **MQXR2\_PUT\_WITH\_DEF\_ACTION**

Varsayılan eylemle konun.

Bu değer, alıcı kanal ileti çıkışı tarafından ayarlanır. It indicates that the message is to be put with the default action of the MCA, that is either the default user ID of the MCA, or the context *UserIdentifier* in the MQMD (message descriptor) of the message.

Değer sıfır; bu değer, çıkış çağrıldığında belirlenen ilk değer kümesine karşılık gelir. Sabit, belgeleme amacıyla sağlanır.

### **MQXR2\_PUT\_WITH\_DEF\_USERID**

Varsayılan kullanıcı kimliği girin.

Bu değer yalnızca alıcının kanal ileti çıkışı tarafından ayarlanabilir. Bu ileti, iletinin MCA ' nın varsayılan kullanıcı kimliğiyle konacağı anlamına gelir.

### **MQXR2\_PUT\_WITH\_MSG\_USERID**

İletinin kullanıcı tanımlayıcıyla birlikte girin.

Bu değer yalnızca alıcının kanal ileti çıkışı tarafından ayarlanabilir. It indicates that the message is to be put with the context *UserIdentifier* in the MQMD (message descriptor) of the message (this might have been modified by the exit).

MQXR2\_PUT\_WITH\_DEF\_ACTION, MQXR2\_PUT\_WITH\_DEF\_USERID ve MQXR2\_PUT\_WITH\_MSG\_USERID arasında bir değer ayarlanmalıdır.

### **MQXR2\_USE\_AGENT\_BUFFER**

Aracı arabelleği kullan.

Bu değer, aktarılacağı verilerin *ExitBufferAddr* ' ta değil, *AgentBuffer* içinde olduğunu belirtir.

Değer sıfır; bu değer, çıkış çağrıldığında belirlenen ilk değer kümesine karşılık gelir. Sabit, belgeleme amacıyla sağlanır.



## **MQXR2\_USE\_EXIT\_BUFFER**

Çıkış arabelleği kullan.

Bu değer, aktarılacağı verilerin *AgentBuffer*' ta değil, *ExitBufferAddri*çinde olduğunu belirtir.

MQXR2\_USE\_AGENT\_BUFFER ve MQXR2\_USE\_EXIT\_BUFFER ' in yalnızca biri ayarlanmalıdır.

## **MQXR2\_DEFAULT\_CONTINUATION**

Varsayılan devamı.

Zincirin sonraki çıkışıyla devam et, çağrılan son çıkıştan gelen yanıtla bağlıdır:

- MQXCC\_SUPPRESS\_FUNCTION ya da MQXCC\_CLOSE\_CHANNEL döndürülürse, zincirde başka hiçbir çıkış çağrılmaz.
- Ters durumda, zincirin sonraki çıkışı çağrılır.

## **MQXR2\_CONTINUE\_CHAIN**

Sonraki çıkışa devam edin.

## **MQXR2\_SUPPRESS\_CHAIN**

Zincirdeki diğer çıkışları atlayın.

Bu, çıkışa giriş/çıkış alanıdır.

*Geribildirim (MQUZE)*

Bu alan, geribildirim kodunu belirtir.

Bu alan, çıkış yordamlarına girişte MQFB\_NONE olarak ayarlanır.

Bir kanal iletişi çıkışı *ExitResponse* alanını MQXCC\_SUPPRESS\_FUNCTION değerine ayarlıyorsa, *Feedback* alanı, iletinin ölü harf (teslim edilemeyen ileti) kuyruğuna neden konduğunu tanımlayan geribildirim kodunu belirtir; bir kural dışı durum raporu göndermek için de kullanılır. Bu durumda, *Feedback* alanı MQFB\_NONE ise, aşağıdaki geri bildirim kodu kullanılır:

## **MQFB\_STOPPED\_BY\_MSG\_EXIT**

İleti, kanal iletişi çıkışa göre durduruldu.

Kanal güvenliği, gönderme, alma ve ileti-yeniden deneme çıkışları tarafından bu alanda döndürülen değer MCA tarafından kullanılmaz.

The value returned in this field by auto-definition exits is not used if *ExitResponse* is MQXCC\_OK, but otherwise is used for the *AuxErrorDataInt1* parameter in the event message.

Bu, çıkıştan giriş/çıkış alanıdır.

*MaxSegmentUzunluğu (MQUZE)*

Bu alan, tek bir iletimde gönderilebilecek bayt cinsinden uzunluk üst sınırını belirler.

Otomatik tanımlama çıkışı tarafından kullanılmaz. Bir kanal gönderme çıkışa ilgi gösterdiğinden, bu çıkış iletim kesiminin büyüklüğünü *MaxSegmentLength* değerinden büyük bir değere yükseltmediğinden emin olmalıdır. Uzunluk, çıkışa ilişkin değiştirmemeleri gereken ilk 8 baytı içerir. Bu değer, kanal başlatıldığında IBM MQ kanalı işlevleri arasında kararlaştırılır. Bölüm uzunlukları hakkında ek bilgi için [Kanal çıkış programlarının yazılması](#) başlıklı konuya bakın.

*ExitReason* MQXR\_INIT ise, bu alandaki değer anlamlı değildir.

Bu, çıkışa giriş alanıdır.

*ExitUser(ExitUser) Alanı (MQBYTE16)*

Bu alan, çıkış kullanıcı alanını belirtir-çıkışa ilişkin kullanılabilir alan kullanılır.

Çıkışa ilk çağrılmadan önce ikili sıfır (MQXR\_INIT olarak ayarlanmış bir *ExitReason* kümesi vardır) ikili olarak başlatılır ve bundan sonra çıkış tarafından bu alanda yapılan değişiklikler, çıkışa yönelik çağrılar içinde korunur.

Aşağıdaki değer tanımlıdır:

## **MQXUA\_NONE**

Kullanıcı bilgisi yok.

Değer, alanın uzunluğuna ilişkin ikili sıfır değeridir.

C programlama dili için, sabit MQXUA\_NONE\_DIZISI de tanımlanır; bu değişmez, MQXUA\_NONE ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisidir.

Bu alanın uzunluğu MQ\_EXIT\_USER\_AREA\_LENGTH tarafından verilir. Bu, çıkışa giriş/çıkış alanıdır.

### *ExitData (MQCHAR32)*

Bu alan çıkış verilerini belirtir.

This field is set on entry to the exit routine to information that IBM MQ channel functions took from the channel definition. Böyle bir bilgi yoksa, bu alan tamamen boşluklanır.

Bu alanın uzunluğu MQ\_EXIT\_DATA\_LENGTH tarafından verilir.

Bu, çıkışa giriş alanıdır.

*Version* , MQCXP\_VERSION\_2değerinden küçükse, bu yapıdaki şu alanlar yoktur.

### *MsgRetryCount (MQLONG)*

Bu alan, iletinin kaç kez yeniden denendiğini belirtir.

Çıkış ilk kez belirli bir ileti için çağrıldığında, bu alan sıfır değerine sahiptir (henüz yeniden deneme girişiminde bulunulmadı). Bu iletiye ilişkin çıkışa ilişkin sonraki her çağrıda, değer MCA tarafından bir artırılır.

Bu, çıkışa giriş alanıdır. *ExitReason* MQXR\_INIT ise, bu alandaki değer anlamlı değildir. *Version* değeri MQCXP\_VERSION\_2değerinden küçükse bu alan yoktur.

### *MsgRetryInterval (MQHOT)*

Bu alan, koyma işleminin yeniden denendiği milisaniye cinsinden minimum aralığı belirtir.

Çıkış ilk kez belirli bir ileti için çağrıldığında, bu alan *MsgRetryInterval* kanal özniteliğinin değerini içerir. Çıkış değeri değişmeden bırakabilir ya da milisaniye cinsinden farklı bir zaman aralığı belirtmek için bu değeri değiştirebilirler. Çıkış, *ExitResponse*' de MQXCC\_OK değerini döndürürse, MCA, MQOPER ya da MQPUT işlemini yeniden denemeden önce en az bu zaman aralığını bekler. Belirtilen zaman aralığı sıfır ya da daha büyük olmalıdır.

Çıkışta ikinci ve sonraki zamanlar bu ileti için çağrılır; bu alan, çıkışa ilişkin önceki çağrımın döndürdüğü değeri içerir.

*MsgRetryInterval* alanında döndürülen değer sıfırdan küçük ya da 999 999 'dan büyükse ve *ExitResponse* MQXCC\_OK ise, MCA, MQCXP ' de *MsgRetryInterval* alanını yoksayar ve *MsgRetryInterval* kanal özniteliği tarafından belirtilen aralık yerine bekler.

Bu, çıkışa giriş/çıkış alanıdır. *ExitReason* MQXR\_INIT ise, bu alandaki değer anlamlı değildir. *Version* değeri MQCXP\_VERSION\_2değerinden küçükse bu alan yoktur.

### *MsgRetryNeden (MQUBE)*

Bu alan, iletiyi koymak için önceki girişimden neden kodunu belirtir.

Bu alan, önceki iletiyi koyma girişimindeki neden kodudur; bu, MQRC\_ \* değerlerinden biridir.

Bu, çıkışa giriş alanıdır. *ExitReason* MQXR\_INIT ise, bu alandaki değer anlamlı değildir. *Version* değeri MQCXP\_VERSION\_2değerinden küçükse bu alan yoktur.

*Version* , MQCXP\_VERSION\_3değerinden küçükse, bu yapıdaki şu alanlar yoktur.

### *HeaderLength (MQHOT)*

Bu alan, üstbilgi bilgilerinin uzunluğunu belirtir.

Bu alan yalnızca ileti çıkışı ve ileti-yeniden deneme çıkışı için anlamlıdır. Değer, ileti verilerinin başlangıcındaki yöneltme üstbilgisi yapılarının uzunluğudur; bunlar MQXQH yapısıdır, MQMDE (ileti

tanımlaması uzantısı üstbilgisi) ve (bir dağıtım listesi iletisi için) MQXQH yapısını izleyen MQOR ve MQPMR kayıtlarının MQDH yapısı ve dizileridir.

İleti çıkışı, bu üstbilgi bilgilerini inceleyebilir ve gerekirse değiştirebilir, ancak çıkış döndüren verilerin doğru biçimde olması gerekir. Çıkış, alıcı uçta ileti çıkışı, telafi edici değişiklikler yapsa da, çıkış gönderisinde üstbilgi verilerini şifrelememeli ya da sıkıştırılmamalıdır.

If the message exit modifies the header information in such a way as to change its length (for example, by adding another destination to a distribution-list message), it must change the value of *HeaderLength* correspondingly before returning.

Bu, çıkışa giriş/çıkış alanıdır. *ExitReason* MQXR\_INIT ise, bu alandaki değer anlamlı değildir. *Version* değeri MQCXP\_VERSION\_3değerinden küçükse bu alan yoktur.

*PartnerName* (MQCHAR48)

Bu alan, iş ortağının adını belirtir.

Ortağın adı, aşağıdaki gibi:

- SVRCONN kanallarında, istemcideki oturum açmış kullanıcı kimliğidir.
- Diğer tüm kanal tipleri için, iş ortağının kuyruk yöneticisi adıdır.

Çıkış başlatıldığında, kuyruk yöneticisi iş ortağının adını bilmiyorsa, ilk anlaşma gerçekleşinceye kadar bu alan boş olur.

Bu, çıkışa giriş alanıdır. *Version* değeri MQCXP\_VERSION\_3değerinden küçükse bu alan yoktur.

*FAPLevel* (MQLONG)

Anlaşmalı Biçimler ve İletişim Kuralları düzeyi.

Bu, çıkışa giriş alanıdır. Bu alanda yapılan değişiklikler yalnızca IBM hizmetinin yönlendirmesi altında yapılmalıdır. *Version* değeri MQCXP\_VERSION\_3değerinden küçükse bu alan yoktur.

*CapabilityFlags* (MQUZE)

Yetenek işaretini MQCF\_NONE ya da MQCF\_DIST\_LISTS olarak ayarlayabilirsiniz.

Aşağıdaki yetenek işaretlerinden birini ayarlayabilirsiniz:

#### **MQCF\_NONE**

Bayrak yok.

#### **MQCF\_DIST\_LISTS**

Dağıtım listeleri desteklenir.

Bu, çıkışa giriş alanıdır. *Version* değeri MQCXP\_VERSION\_3değerinden küçükse bu alan yoktur.

*ExitNumber* (MQLONG)

Bu alan, çıkışa ilişkin sıra sayısını belirtir.

Çıkışa ilişkin sıra sayısı, *ExitId* içinde tanımlanan tip içinde. Örneğin, çağrılmakta olan çıkış, üçüncü ileti çıkışıysa, bu alan 3 değerini içerir. Çıkış tipi, çıkış listesinin tanımlanamadığı bir tipse (örneğin, bir güvenlik çıkışı), bu alanın değeri 1 'dir.

Bu, çıkışa giriş alanıdır. *Version* değeri MQCXP\_VERSION\_3değerinden küçükse bu alan yoktur.

*Version* değeri MQCXP\_VERSION\_5değerinden küçükse, bu yapıdaki şu alanlar yoktur.

*ExitSpace* (MQUZE)

Bu alan, çıkışa ayrılan iletim arabelleğindeki bayt sayısını belirler.

Bu alan yalnızca bir gönderme çıkışı için anlamlıdır. It specifies the amount of space in bytes that the IBM MQ channel functions reserve in the transmission buffer for the exit to use. Bu alan, diğer uçta tamamlayıcı bir alma çıkışı tarafından kullanılmak üzere, çıkışa küçük bir veri miktarı (genellikle birkaç yüz bayt aşmayan) veri iletimi arabelleğinden eklenmesine olanak sağlar. Gönderme çıkışıyla eklenen veriler, alma çıkışıyla kaldırılmalıdır.

Değer, z/OS üzerinde her zaman sıfırdır.

**Not:** Bu olanak, büyük miktardaki verileri göndermek için kullanılmamalıdır; bu, performansı düşürebileceği gibi, hatta kanalın çalışmasını engelleyebilir.

*ExitSpace* ayarını yaparak, çıkışa ilişkin iletim arabelleğinde her zaman en az bu sayıda baytın kullanılabilmesini garanti altına aldığından emin olun. Ancak, çıkış ayrılmış miktardan az ya da iletim arabelleğindeki kullanılabilir alan varsa, ayrılan miktardan fazla kullanılabilir. Var olan verilerden sonra arabelleğindeki çıkış alanı sağlanır.

*ExitSpace* , çıkış tarafından yalnızca *ExitReason* MQXR\_INIT; değeri olduğunda çıkışa göre ayarlanabilir; diğer tüm durumlarda, çıkışa döndürülen değer yoksayıdır. Çıkışa giriş sırasında, MQXR\_INIT çağrısı için *ExitSpace* sıfır ve diğer durumlarda MQXR\_INIT çağrısının döndürdüğü değerdir.

MQXR\_INIT çağrısının döndürdüğü değer negatifse ya da zincirdeki tüm çıkışlar için istenen çıkış alanını ayırdıktan sonra ileti verilerine ilişkin iletim arabelleğinde 1024 byte 'tan az sayıda bayt varsa, MCA hata ileti görünümler ve kanalı kapatır. Benzer şekilde, veri aktarımı sırasında çıkış zincirindeki çıkışlar, ileti verileri için iletim arabelleğinde 1024 bayttan daha az bayt kalmaktan daha fazla kullanıcı alanı ayırırsa, MCA bir hata ileti görünümler ve kanalı kapatır. 1024 ' ün sınırı, kanalların kontrol ve idari akışlarının, akışların kesilmesine gerek kalmadan, gönderme çıkışları zinciri tarafından işlenmesine olanak sağlar.

Bu, *ExitReason* MQXR\_INIT ise çıkışa giriş/çıkış alanıdır ve diğer tüm durumlarda bir giriş alanıdır. *Version* değeri MQCXP\_VERSION\_5değerinden küçükse bu alan yoktur.

#### *SSLCertUserTanıtıcısı (MQCHAR12)*

Bu alan, uzak sertifikayla ilişkili UserId ' ne (UserId) ilişkin bilgi verir.

z/OSdışındaki tüm platformlarda boş

Bu, çıkışa giriş alanıdır. *Version* değeri MQCXP\_VERSION\_6değerinden küçükse bu alan yoktur.

#### *SSLRemCertIssNameUzunluğu (MQUZE)*

Bu alan, SSLCertRemoteIssuerNamePtr tarafından gösterilen uzak sertifikanın sertifika verenin tam ayırt edici adının bayt cinsinden uzunluğunu belirtir.

Bu, çıkışa giriş alanıdır. *Version* değeri MQCXP\_VERSION\_6değerinden küçükse bu alan yoktur. Bir TLS kanalı değilse, değer sıfırdır.

#### *SSLRemCertIssNamePtr (PMQVOID)*

Bu alan, uzak sertifikanın yayıncısının tam Ayırt Edici Adı 'nın adresini belirtir.

Bu değer, TLS kanalı değilse, boş değerli göstergeci.

Bu, çıkışa giriş alanıdır. *Version* değeri MQCXP\_VERSION\_6değerinden küçükse bu alan yoktur.

**Not:** Genel Ayırt Edici Adı ve Sertifika Veren Ayırt Edici Adı 'nın belirlenmesinde kanal güvenliği çıkışlarının davranışı IBM WebSphere MQ 7.1olarak değişir. Ek bilgi için bkz. [Kanal güvenlik çıkış programları](#).

#### *SecurityParms (PMQCSP)*

Bu alan, bir kullanıcı kimliği ve parola belirtmek için kullanılan MQCSP yapısının adresini belirtir.

Bu alanın başlangıç değeri, boş değerli göstergeci.

Bu, çıkışa giriş/çıkış alanıdır. *Version* değeri MQCXP\_VERSION\_6değerinden küçükse bu alan yoktur.

Çıkış tarafından döndürülen bu alandaki değer, MQXR\_TERM ' ye kadar IBM MQ tarafından kullanılabilir olmalıdır.

#### *CurHdrSıkıştırma (MQUZE)*

Bu alan, üstbilgi verilerini sıkıştırmak için kullanılmakta olan tekniğin kullanılacağını belirtir.

Aşağıdakilerden birine ayarlanır:

#### **MQCOMPRESS\_NONE**

Üstbilgi veri sıkıştırma işlemi gerçekleştirilmedi.

#### **MQCOMPRESS\_SYSTEM**

Üstbilgi veri sıkıştırması gerçekleştirilir.

Bu deęer, MQCD ' nin HdrCompList (Liste) alanından eriřilen, anlaşmalı desteklenen deęerlerden birine kanala gönderilen bir kanaldan çıkış yoluyla deęiřtirilebilir. Bu, iletinin içerięine dayalı olarak her ileti için seilebilecek üstbilgi verilerini sıkıřtırmak için kullanılan teknięin geerli olmasını saęlar. Deęiřtirilen deęer, yalnızca geerli ileti için kullanılır. Öznitelik desteklenmeyen bir deęere deęiřtirilirse kanal sona erer. Bir gönderme kanalının ileti çıkışı dışında deęiřtirilirse, deęer yoksayılır.

Bu, çıkışa giriř/çıkış alanıdır. *Version* deęeri MQCXP\_VERSION\_6deęerinden küçükse bu alan yoktur.

#### *CurMsgSıkıřtırma (MQUZE)*

Bu alan, ileti verilerini sıkıřtırmak için kullanılmakta olan teknięin kullanılacaęını belirtir.

Ařaęıdakilerden birine ayarlanır:

#### **MQCOMPRESS\_NONE**

Üstbilgi veri sıkıřtırma iřlemi gerekleřtirilmedi.

#### **MQCOMPRESS\_RLE**

İleti veri sıkıřtırması, alıřtırma uzunluęu kodlaması kullanılarak gerekleřtirilir.

#### **MQCOMPRESS\_ZLIBFAST**

İleti veri sıkıřtırması, zlib sıkıřtırma teknięi kullanılarak gerekleřtirilir. Hızlı sıkıřtırma süresi tercih edilir.

#### **MQCOMPRESS\_ZLIBHIGH**

İleti veri sıkıřtırması, zlib sıkıřtırma teknięi kullanılarak gerekleřtirilir. Yüksek düzeyde sıkıřtırma tercih edilir.

Bu deęer, MQCD ' nin MsgCompList (MsgComp) Listesi alanından eriřilen, anlaşmalı desteklenen deęerlerden birine kanala gönderilen bir kanaldan çıkış yoluyla deęiřtirilebilir. Bu, iletinin içerięine dayalı olarak her ileti için karar verilmek üzere ileti verilerini sıkıřtırmak için kullanılan teknięin kullanılmasını saęlar. Deęiřtirilen deęer, yalnızca geerli ileti için kullanılır. Öznitelik desteklenmeyen bir deęere deęiřtirilirse kanal sona erer. Bir gönderme kanalının ileti çıkışı dışında deęiřtirilirse, deęer yoksayılır.

Bu, çıkışa giriř/çıkış alanıdır. *Version* deęeri MQCXP\_VERSION\_6deęerinden küçükse bu alan yoktur.

#### *Hconn (MQHCONN)*

Bu alan, çıkışta herhangi bir MQI çağırısı yapması gerekiyorsa, çıkışa iliřkin baęlantı tanıtıcısını belirler.

Bu alan, istemci-baęlantı kanallarında alıřan MQHC\_UNUABLE\_HCONN (-1) deęerini içerdii için çıkışa uygun deęildir.

Bu, çıkışa giriř alanıdır. *Version* , MQCXP\_VERSION\_7' den küçükse bu alan yoktur.

#### *SharingConversations (MQBOOL)*

Bu alan, bu kanal yönetim ortamında alıřmakta olan tek bir etkileřim olup olmadıęını ya da bu kanal somut örneęinde birden ok konuřmanın alıřtırılıp alıřtırılmayacaęını belirtir.

Ayrıca, çıkış programının, aynı anda alıřan bařka bir çıkış programı tarafından deęiřtirilmekte olan MQCD ' nin riskine tabi olup olmadıęını da gösterir.

Bu alan yalnızca istemci-baęlantı ya da sunucu baęlantısı kanallarında alıřan çıkış programları için anlamlıdır.

Ařaęıdakilerden birine ayarlanır:

#### **YANLIř**

Çıkış eřęörünümü, bu kanal yönetim ortamında alıřmakta olan tek çıkış eřęörünümdür. Bu, dięer kanal örneklerinde alıřan dięer çıkışlardan ekiřme olmadan, MQCD alanlarını güvenli bir şekilde güncellemek için bu çıkışa olanak saęlar. MQCD alanlarına iliřkin deęiřikliklerin kanalın üzerinde iřlem yapılıp yapılamayacaęı, “Kanal çıkışındaki MQCD alanlarının deęiřtirilmesi” sayfa 1492iindeki MQCD alanları izelgesiyle tanımlanır.

#### **DOęRU**

Çıkış eřęörünümü, bu kanal yönetim ortamında alıřtırılmakta olan tek çıkış eřęörünümdü deęil. MQCD 'de yapılan deęiřiklikler, MQXR\_INIT dıřındaki Reasons 'tan çıkmak için “Kanal çıkışındaki MQCD alanlarının deęiřtirilmesi” sayfa 1492 iindeki MQCD alanları izelgesinde listelenen deęiřiklikler

dışında, bu MQCD' de yapılan değişiklikler kanal tarafından üzerinde işlem yapılmaz. Bu çıkış, MQCD alanlarını güncelliyorsa, bu kanal yönetim ortamında çalışan çıkışlar arasında diziselleştirme sağlayarak, diğer etkileşimlerden başka çıkışlar da devam eden başka bir çıkıştan söz sahibi olmamasını sağlar.

Bu, çıkışa giriş alanıdır. *Version* , MQCXP\_VERSION\_7' den küçükse bu alan yoktur.

*MCAUserSource (MQUZE)*

Bu alan, sağlanan MCA kullanıcı kimliğinin kaynağını belirtir.

Aşağıdaki değerlerden birini içerebilir:

#### **MQUSRC\_MAP**

Kullanıcı kimliği MCAUSER özniteisinde belirtilir.

#### **MQUSRC\_CHANNEL**

Kullanıcı kimliği, gelen ortaktan akılır ya da kanal nesnesinde tanımlanan MCAUSER alanında belirtilir.

Bu, çıkışa giriş alanıdır. Sürüm MQCXP\_VERSION\_8değerinden küçükse bu alan yoktur.

*pEntryPuanları (PMQIEP)*

Bu alan, MQI ya da DCI çağrısına ilişkin arabirim giriş noktasının adresini belirtir.

*Sürüm* , MQCXP\_VERSION\_8değerinden küçükse, alan yoktur.

*RemoteProduct (MQCHAR4)*

Bu alan, uzak ürün adını belirtir.

Bu alan, istemcinin uzak ürününü (örneğin, C ya da Java), DISPLAY CHSATUS' un **RPRODUCT** alanında görüntülendiği gibi tanımlar.

*Sürüm* , MQCXP\_VERSION\_9değerinden küçükse, alan yoktur.

*RemoteVersion (MQCHAR8)*

Bu alan, uzak sürümün adını belirtir.

This field identifies the version of the client libraries, as displayed in the **RVERSION** field of DURUMU GÖRÜNTÜLE.

*Sürüm* , MQCXP\_VERSION\_9değerinden küçükse, alan yoktur.

### **C bildirim**

Bu bildirim, MQCXP yapısına ilişkin C bildirimidir.

```
typedef struct tagMQCXP MQCXP;
struct tagMQCXP {
    MQCHAR4    StrucId;           /* Structure identifier */
    MQLONG     Version;          /* Structure version number */
    MQLONG     ExitId;           /* Type of exit */
    MQLONG     ExitReason;       /* Reason for invoking exit */
    MQLONG     ExitResponse;     /* Response from exit */
    MQLONG     ExitResponse2;    /* Secondary response from exit */
    MQLONG     Feedback;         /* Feedback code */
    MQLONG     MaxSegmentLength; /* Maximum segment length */
    MQBYTE16   ExitUserArea;     /* Exit user area */
    MQCHAR32   ExitData;         /* Exit data */
    MQLONG     MsgRetryCount;    /* Number of times the message has been
    retried */
    MQLONG     MsgRetryInterval; /* Minimum interval in milliseconds after
    which the put operation should be
    retried */
    MQLONG     MsgRetryReason;   /* Reason code from previous attempt to
    put the message */
    MQLONG     HeaderLength;     /* Length of header information */
    MQCHAR48   PartnerName;     /* Partner Name */
    MQLONG     FAPLevel;        /* Negotiated Formats and Protocols
    level */
    MQLONG     CapabilityFlags;  /* Capability flags */
    MQLONG     ExitNumber;       /* Exit number */
    /* Ver:3 */
    /* Ver:4 */
};
```

```

MQLONG    ExitSpace;          /* Number of bytes in transmission buffer
                               reserved for exit to use */
/* Ver:5 */
MQCHAR12  SSLCertUserid;     /* User identifier associated
                               with remote TLS certificate */
MQLONG    SSLRemCertIssNameLength; /* Length of
                               distinguished name of issuer
                               of remote TLS certificate */
MQPTR     SSLRemCertIssNamePtr; /* Address of
                               distinguished name of issuer
                               of remote TLS certificate */
PMQVOID   SecurityParms;     /* Security parameters */
MQLONG    CurHdrCompression; /* Header data compression
                               used for current message */
MQLONG    CurMsgCompression; /* Message data compression
                               used for current message */
/* Ver:6 */
MQHCONN   Hconn;            /* Connection handle */
MQBOOL    SharingConversations; /* Multiple conversations
                               possible on channel inst? */
/* Ver:7 */
MQLONG    MCAUserSource;     /* Source of the provided MCA user ID */
PMQIEP    pEntryPoints;     /* Address of the MQIEP structure */
/* Ver:8 */
MQCHAR4   RemoteProduct;     /* The identifier for the remote product */
MQCHAR8   RemoteVersion;     /* The version of the remote product */
/* Ver:9 */
};

```

## COBOL bildirim

Bu bildirim, MQCXP yapısına ilişkin COBOL bildirimidir.

```

** MQCXP structure
10 MQCXP.
** Structure identifier
15 MQCXP-STRUCID PIC X(4).
** Structure version number
15 MQCXP-VERSION PIC S9(9) BINARY.
** Type of exit
15 MQCXP-EXITID PIC S9(9) BINARY.
** Reason for invoking exit
15 MQCXP-EXITREASON PIC S9(9) BINARY.
** Response from exit
15 MQCXP-EXITRESPONSE PIC S9(9) BINARY.
** Secondary response from exit
15 MQCXP-EXITRESPONSE2 PIC S9(9) BINARY.
** Feedback code
15 MQCXP-FEEDBACK PIC S9(9) BINARY.
** Maximum segment length
15 MQCXP-MAXSEGMENTLENGTH PIC S9(9) BINARY.
** Exit user area
15 MQCXP-EXITUSERAREA PIC X(16).
** Exit data
15 MQCXP-EXITDATA PIC X(32).
** Number of times the message has been retried
15 MQCXP-MSGRETRYCOUNT PIC S9(9) BINARY.
** Minimum interval in milliseconds after which the put operation
** should be retried
15 MQCXP-MSGRETRYINTERVAL PIC S9(9) BINARY.
** Reason code from previous attempt to put the message
15 MQCXP-MSGRETRYREASON PIC S9(9) BINARY.
** Length of header information
15 MQCXP-HEADERLENGTH PIC S9(9) BINARY.
** Partner Name
15 MQCXP-PARTNERNAME PIC X(48).
** Negotiated Formats and Protocols level
15 MQCXP-FAPLEVEL PIC S9(9) BINARY.
** Capability flags
15 MQCXP-CAPABILITYFLAGS PIC S9(9) BINARY.
** Exit number
15 MQCXP-EXITNUMBER PIC S9(9) BINARY.
** Number of bytes in transmission buffer reserved for exit to use
15 MQCXP-EXITSPEACE PIC S9(9) BINARY.
** User Id associated with remote certificate
15 MQCXP-SSLCERTUSERID PIC X(12).
** Length of distinguished name of issuer of remote TLS
** certificate
15 MQCXP-SSLREMCERTISSNAMELENGTH PIC S9(9) BINARY.

```

```

** Address of distinguished name of issuer of remote TLS
** certificate
15 MQCXP-SSLREMCERTISSNAMEPTR    POINTER.
** Security parameters
15 MQCXP-SECURITYPARMS           PIC S9(18) BINARY.
** Header data compression used for current message
15 MQCXP-CURHDRCOMPRESSION       PIC S9(9) BINARY.
** Message data compression used for current message
15 MQCXP-CURMSGCOMPRESSION       PIC S9(9) BINARY.
** Connection handle
15 MQCXP-HCONN                   PIC S9(9) BINARY.
** Multiple conversations possible on channel instance?
15 MQCXP-SHARINGCONVERSATIONS    PIC S9(9) BINARY.
** Source of the provided MCA user ID
15 MQCXP-MCAUSERSOURCE           PIC S9(9) BINARY.
** Identifier of the remote product
15 MQCXP-RPRODUCT                PIC X(4).
** Identifier of the remote version
15 MQCXP-RVERSION                PIC X(8).

```

## RPG bildirim (ILE)

Bu bildirim, MQCXP yapısına ilişkin RPG bildirimidir.

```

D*.1.....2.....3.....4.....5.....6.....7..
D* MQCXP Structure
D*
D* Structure identifier
D  CXSID          1      4
D* Structure version number
D  CXVER          5      8I 0
D* Type of exit
D  CXXID          9      12I 0
D* Reason for invoking exit
D  CXREA         13      16I 0
D* Response from exit
D  CXRES         17      20I 0
D* Secondary response from exit
D  CXRE2         21      24I 0
D* Feedback code
D  CXFB          25      28I 0
D* Maximum segment length
D  CXMSL         29      32I 0
D* Exit user area
D  CXUA          33      48
D* Exit data
D  CXDAT         49      80
D* Number of times the message has been retried
D  CXMRC         81      84I 0
D* Minimum interval in milliseconds after which the put operation
D* should be retried
D  CXMRI         85      88I 0
D* Reason code from previous attempt to put the message
D  CXMRR         89      92I 0
D* Length of header information
D  CXHDL         93      96I 0
D* Partner Name
D  CXPNM         97      144
D* Negotiated Formats and Protocols level
D  CXFAP        145      148I 0
D* Capability flags
D  CXCAP        149      152I 0
D* Exit number
D  CXEXN        153      156I 0
D* Number of bytes in transmission buffer reserved for exit to use
D  CXHDL        157      160I 0
D* User identifier associated with remote TLS certificate
D  CXSSLCU       161      172
D* Length of distinguished name of issuer of remote TLS certificate
D  CXSRCINL      173      176I 0
D* Address of distinguished name of issuer of remote TLS certificate
D  CXSRCINP      177      192*
D* Security parameters
D  CXSECP        193      208*
D* Header data compression used for current message
D  CXCHC        209      212I 0
D* Message data compression used for current message
D  CXCMC        213      216I 0
D* Connection handle

```



D	CXHCONN	217	220I	0
D*	Multiple conversations possible on		channel instance?	
D	CXSHARECONV	221	224I	0
D*	Source of the provided MCA user ID			
D	MCAUSERSOURCE	225	228I	0
D*	Identifier of the remote product			
D	CXRPRO	229	232I	0
D*	Identifier of the remote version			
D	CXRVER	233	240I	0

## System/390 çevirici bildirimi

Bu bildirim, MQCXP yapısına ilişkin System/390 çevirici bildirimidir.

MQCXP	DSECT		
MQCXP_STRUCID	DS	CL4	Structure identifier
MQCXP_VERSION	DS	F	Structure version number
MQCXP_EXITID	DS	F	Type of exit
MQCXP_EXITREASON	DS	F	Reason for invoking exit
MQCXP_EXITRESPONSE	DS	F	Response from exit
MQCXP_EXITRESPONSE2	DS	F	Secondary response from exit
MQCXP_FEEDBACK	DS	F	Feedback code
MQCXP_MAXSEGMENTLENGTH	DS	F	Maximum segment length
MQCXP_EXITUSERAREA	DS	XL16	Exit user area
MQCXP_EXITDATA	DS	CL32	Exit data
MQCXP_MSGRETRYCOUNT	DS	F	Number of times the message has been retried
* MQCXP_MSGRETRYINTERVAL	DS	F	Minimum interval in milliseconds after which the put operation should be retried
* MQCXP_MSGRETRYREASON	DS	F	Reason code from previous attempt to put the message
* MQCXP_HEADERLENGTH	DS	F	Length of header information
MQCXP_PARTNERNAME	DS	CL48	Partner Name
MQCXP_FAPLEVEL	DS	F	Negotiated Formats and Protocols level
* MQCXP_CAPABILITYFLAGS	DS	F	Capability flags
MQCXP_EXITNUMBER	DS	F	Exit number
MQCXP_EXITSPACE	DS	F	Number of bytes in transmission buffer reserved for exit to use
* MQCXP_SSLCERTUSERID	DS	CL12	User identifier associated with remote TLS certificate
* MQCXP_SSLREMCERTISSNAMELENGTH	DS	F	Length of distinguished name of issuer of remote TLS certificate
* MQCXP_SSLREMCERTISSNAMEPTR	DS	F	Address of distinguished name of issuer of remote TLS certificate
* MQCXP_SECURITYPARMS	DS	F	Address of security parameters
MQCXP_CURHDRCOMPRESSION	DS	F	Header data compression used for current message
* MQCXP_CURMSGCOMPRESSION	DS	F	Message data compression used for current message
* MQCXP_HCONN	DS	F	Connection handle
MQCXP_SHARINGCONVERSATIONS	DS	F	Multiple conversations possible on channel inst?
* MQCXP_MCAUSERSOURCE	DS	F	Source of the provided MCA user ID
MQCXP_RPRODUCT	DS	CL4	Identifier of the remote product
MQCXP_RVERSION	DS	CL8	Identifier of the remote version
MQCXP_LENGTH	EQU	*-MQCXP	
	ORG	MQCXP	
MQCXP_AREA	DS	CL(MQCXP_LENGTH)	

## MQXWD-Çıkış bekleme tanımlayıcısı

MQXWD yapısı, MQXMK çağrısındaki bir giriş/çıkış değiştirgedir.

Bu yapı yalnızca z/OSüzerinde desteklenir.

### İlgili başvurular

“Alanlar” sayfa 1510

Bu konu, MQXWD yapısındaki tüm alanları listeler ve her bir alanı açıklar.

“C bildirimi” sayfa 1510

Bu bildirim, MQXWD yapısına ilişkin C bildirimidir.

“System/390 çevirici bildirimi” sayfa 1511

Bu bildirim, MQXWD yapısı için System/390 çevirici bildirimidir.

## **Alanlar**

Bu konu, MQXWD yapısındaki tüm alanları listeler ve her bir alanı açıklar.

*StrucId (MQCHAR4)*

Bu alan, yapı tanıtcısını belirtir.

Değer şu olmalıdır:

### **MQXWD\_STRUC\_ID**

Çıkış bekleme tanımlayıcı yapısıyla ilgili tanıtcı.

C programlama dili için, sabit MQXWD\_STRUC\_ID\_ARRAY değişmezi de tanımlanır; bu değişmez, MQXWD\_STRUC\_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisi olur.

Bu alanın ilk değeri MQXWD\_STRUC\_ID 'dir.

*Sürüm (MQUZE)*

Bu alan, yapı sürüm numarasını belirtir.

Değer şu olmalıdır:

### **MQXWD\_VERSION\_1**

Çıkış bekleme tanımlayıcı yapısı için sürüm numarası.

Bu alanın ilk değeri MQXWD\_VERSION\_1' dir.

*Reserved1 (MQUZE)*

Bu alan ayrılmıştır. Değeri sıfır olmalıdır.

Bu bir giriş alanıdır.

*Reserved2 (MQUZE)*

Bu alan ayrılmıştır. Değeri sıfır olmalıdır.

Bu bir giriş alanıdır.

*Reserved3 (MQUZE)*

Bu alan ayrılmıştır. Değeri sıfır olmalıdır.

Bu bir giriş alanıdır.

*ECB (MQUZA)*

Bu alan, beklenen olay denetim bloğunu belirtir.

Bu alan, beklenen olay denetim öbeğidir (ECB). MQXWEKE çağrısı yayınlanmadan önce sıfır değerine ayarlanmalıdır; başarıyla tamamlandığında, posta kodunu içerir.

Bu alan bir giriş/çıkış alanıdır.

## **C bildirimi**

Bu bildirim, MQXWD yapısına ilişkin C bildirimidir.

```
typedef struct tagMQXWD MQXWD;
struct tagMQXWD {
    MQCHAR4  StrucId;      /* Structure identifier */
    MQLONG   Version;     /* Structure version number */
    MQLONG   Reserved1;   /* Reserved */
    MQLONG   Reserved2;   /* Reserved */
    MQLONG   Reserved3;   /* Reserved */
    MQLONG   ECB;        /* Event control block to wait on */
};
```

## System/390 çevirici bildirimi

Bu bildirim, MQXWD yapısı için System/390 çevirici bildirimidir.

```
MQXWD          DSECT
MQXWD_STRUCID DS   CL4  Structure identifier
MQXWD_VERSION DS   F    Structure version number
MQXWD_RESERVED1 DS  F    Reserved
MQXWD_RESERVED2 DS  F    Reserved
MQXWD_RESERVED3 DS  F    Reserved
MQXWD_ECB      DS   F    Event control block to wait on
*
MQXWD_LENGTH   EQU   *-MQXWD
               ORG   MQXWD
MQXWD_AREA     DS    CL(MQXWD_LENGTH)
```

## Küme iş yükü çıkış çağrısı ve veri yapıları

Bu kısım, küme iş yükü çıkışa ve veri yapılarına ilişkin başvuru bilgileri sağlar. Bu, genel kullanıma açık programlama arabirimi bilgileridir.


Küme iş yükü çıkışlarını aşağıdaki programlama dillerinde yazabilirsiniz:

- C
- System/390 çevirici ( IBM MQ for z/OS )

Arama şu şekilde açıklanır:

- [“MQ\\_CLUSTER\\_WORKLOAD\\_EXIT -Arama açıklaması” sayfa 1512](#)

Çıkışta kullanılan yapı veri tipleri aşağıda açıklanmıştır:

- [“MQXCLWLN -Küme iş yükü kayıtlarında dolaşma” sayfa 1513](#)
- [“MQWXP -Küme iş yükü çıkış değiştirgesi yapısı” sayfa 1517](#)
- [“MQWDR-Küme iş yükü hedef kaydı yapısı” sayfa 1525](#)
- [“MQWQR -Küme iş yükü kuyruğu kayıt yapısı” sayfa 1529](#)
- [“MQWCR -Küme iş yükü kümesi kayıt yapısı” sayfa 1534](#)
-  z/OS üzerinde CLUSTER komutlarının zamanuyumsuz davranışı

Bu kısım boyunca, kuyruk yöneticisi öznitelikleri ve kuyruk öznitelikleri dolu olarak gösterilir. MQSC komutlarında kullanılan eşdeğer adlar aşağıda gösterilmiştir. MQSC komutlarının ayrıntıları için bkz. [MQSC komutları](#).

Çizelge 825. Kuyruk yöneticisi öznitelikleri	
Tam ad	MQSC ' de kullanılan ad
<i>ClusterWorkloadData</i>	CLWLDATA
<i>ClusterWorkloadExit</i>	CLAXLEXIT
<i>ClusterWorkloadLength</i>	CLWLEN

Çizelge 826. Kuyruk öznitelikleri	
Tam ad	MQSC ' de kullanılan ad
<i>DefBind</i>	DEFBIND
<i>DefPersistence</i>	DEĞERLERI
<i>DefPriority</i>	DEĞİŞMEZ
<i>InhibitPut</i>	PUT

Çizelge 826. Kuyruk öznitelikleri (devamı var)	
<b>Tam ad</b>	<b>MQSC ' de kullanılan ad</b>
QDesc	TASARIMLA

### İlgili görevler

Küme iş yükü çıkışlarının yazılması ve derlenmesi

## MQ\_CLUSTER\_WORKLOAD\_EXIT -Arama açıklaması

Küme iş yükü çıkışı, bir iletiyi kullanılabilir bir kuyruk yöneticisine yöneltmek için kuyruk yöneticisi tarafından çağrılır.

**Not:** Kuyruk yöneticisi, MQ\_CLUSTER\_WORKLOAD\_EXIT adlı bir giriş noktası sağlamıyor. Bunun yerine, küme iş yükü çıkışının adı, ClusterWorkloadExit kuyruk yöneticisi özniteliyerek tanımlanır.

MQ\_CLUSTER\_WORKLOAD\_EXIT çıkışı tüm altyapılarda desteklenir.

### Sözdizimi

MQ\_CLUSTER\_WORKLOAD\_EXIT (*ExitParms*)

### İlgili başvurular

MQXCLWLN -Küme iş yükü kayıtlarında dolaşma

The MQXCLWLN call is used to navigate through the chains of MQWDR, MQWQR, and MQWCR records stored in the cluster cache.

MQWXP -Küme iş yükü çıkış değiştirgesi yapısı

Aşağıdaki çizelge, MQWXP -Cluster iş yükü çıkış değiştirgesi yapısındaki alanları özetlemektedir.

MQWDR-Küme iş yükü hedef kaydı yapısı

Aşağıdaki çizelge, MQWDR -Küme iş yükü hedefi kayıt yapısındaki alanları özetler.

MQWQR -Küme iş yükü kuyruğu kayıt yapısı

Aşağıdaki çizelge, MQWQR -Küme iş yükü kuyruğu kayıt yapısındaki alanları özetlemektedir.

MQWCR -Küme iş yükü kümesi kayıt yapısı

Aşağıdaki çizelge, MQWCR küme iş yükü kaydı yapısındaki alanları özetlemektedir.

### MQ\_CLUSTER\_WORKLOAD\_EXIT için parametreler

MQ\_CLUSTER\_WORKLOAD\_EXIT çağrısındaki parametrelerin açıklaması.

#### **ExitParms ( MQWXP ) -giriş/çıkış**

Parametre blokundan çıkın.

- Çıkış, iş yükünün nasıl yönetileceğini belirtmek için MQWXP ' deki bilgileri ayarlar.

### İlgili başvurular

Kullanım notları

Küme iş yükü çıkışı tarafından gerçekleştirilen işlev, çıkışa ilişkin sağlayıcıya göre tanımlanır. The exit, however, must conform to the rules defined in the associated control block MQWXP.

MQ\_CLUSTER\_WORKLOAD\_EXIT için dil çağrıları

MQ\_CLUSTER\_WORKLOAD\_EXIT , iki dil, C ve High Level Assembler ürünlerini destekler.

### **Kullanım notları**

Küme iş yükü çıkışı tarafından gerçekleştirilen işlev, çıkışa ilişkin sağlayıcıya göre tanımlanır. The exit, however, must conform to the rules defined in the associated control block MQWXP.

Kuyruk yöneticisi, MQ\_CLUSTER\_WORKLOAD\_EXIT adlı bir giriş noktası sağlamıyor. Ancak, C programlama dilinde MQ\_CLUSTER\_WORKLOAD\_EXIT adı için bir typedef değeri sağlanmıştır. Parametrelerin doğru olduğundan emin olmak için, kullanıcı tarafından yazılan çıkışı bildirmek için typedef 'i kullanın.

### İlgili başvurular

MQ\_CLUSTER\_WORKLOAD\_EXIT için parametreler

MQ\_CLUSTER\_WORKLOAD\_EXIT çağrısındaki parametrelerin açıklaması.

MQ\_CLUSTER\_WORKLOAD\_EXIT için dil çağrıları

MQ\_CLUSTER\_WORKLOAD\_EXIT , iki dil, C ve High Level Assembler ürünlerini destekler.

### MQ\_CLUSTER\_WORKLOAD\_EXIT için dil çağrıları

MQ\_CLUSTER\_WORKLOAD\_EXIT , iki dil, C ve High Level Assembler ürünlerini destekler.

### C çağırma

```
MQ_CLUSTER_WORKLOAD_EXIT (&ExitParms);
```

MQ\_CLUSTER\_WORKLOAD\_EXIT yerine küme iş yükü çıkışı işlevinizin adını koyun.

MQ\_CLUSTER\_WORKLOAD\_EXIT parametrelerini aşağıdaki gibi bildirin:

```
MQWXP ExitParms; /* Exit parameter block */
```

### High Level Assembler çağırısı

```
CALL EXITNAME, (EXITPARMS)
```

Değiştiricileri aşağıdaki gibi bildirin:

```
EXITPARMS      CMQWXP      Exit parameter block
```

### İlgili başvurular

MQ\_CLUSTER\_WORKLOAD\_EXIT için parametreler

MQ\_CLUSTER\_WORKLOAD\_EXIT çağrısındaki parametrelerin açıklaması.

Kullanım notları

Küme iş yükü çıkışı tarafından gerçekleştirilen işlev, çıkışa ilişkin sağlayıcıya göre tanımlanır. The exit, however, must conform to the rules defined in the associated control block MQWXP.

### MQXCLWLN -Küme iş yükü kayıtlarında dolaşma

The MQXCLWLN call is used to navigate through the chains of MQWDR, MQWQR, and MQWCR records stored in the cluster cache.

Küme önbelleği, kümeyle ilgili bilgileri saklamak için kullanılan ana saklama alanı alanıdır.

Küme önbelleği durağansa, sabit bir boyutu vardır. Bu değeri dinamik olarak ayarlarsanız, küme önbelleği gerektiği gibi genişletilebilir.

Bir sistem parametresi ya da makrosu kullanarak küme önbelleği tipini STATIC ya da DYNAMIC olarak ayarlayın.

- **Multi** Multiplatforms üzerindeki ClusterCache Türü sistem parametresini kullanın.
- **z/OS** z/OS üzerindeki CSQ6SYSP makrosu içindeki CLCACHE parametresini kullanın.

## Sözdizimi

MQXCLWLN (*ExitParms*, *CurrentRecord*, *NextOffset*, *NextRecord*, *Compcode*, *Reason*)

### İlgili başvurular

MQ\_CLUSTER\_WORKLOAD\_EXIT -Arama açıklaması

Küme iş yükü çıkışı, bir iletiyi kullanılabilir bir kuyruk yöneticisine yöneltmek için kuyruk yöneticisi tarafından çağrılır.

MQWXP -Küme iş yükü çıkış değiştirgesi yapısı

Aşağıdaki çizelge, MQWXP -Cluster iş yükü çıkış değiştirgesi yapısındaki alanları özetlemektedir.

MQWDR-Küme iş yükü hedef kaydı yapısı

Aşağıdaki çizelge, MQWDR -Küme iş yükü hedefi kayıt yapısındaki alanları özetler.

MQWQR -Küme iş yükü kuyruğu kayıt yapısı

Aşağıdaki çizelge, MQWQR -Küme iş yükü kuyruğu kayıt yapısındaki alanları özetlemektedir.

MQWCR -Küme iş yükü kümesi kayıt yapısı

Aşağıdaki çizelge, MQWCR küme iş yükü kaydı yapısındaki alanları özetlemektedir.

### **MQXCLWLN için parametreler-Küme iş yükü kayıtlarına gidin**

MQXCLWLN çağrısındaki parametrelerin açıklaması.

#### **ExitParms (MQWXP) -giriş/çıkış**

Parametre blokundan çıkın.

Bu yapı, çıkışa ilişkin çağrıyla ilgili bilgileri içerir. Çıkış, iş yükünün nasıl yönetileceğini belirtmek için bu yapıdaki bilgileri ayarlar.

#### **CurrentRecord (MQPTR) -giriş**

Geçerli kaydın adresi.

Bu yapı, çıkışta şu anda incelenmekte olan kaydın adresiyle ilgili bilgileri içerir. Kayıt aşağıdaki tiplerden biri olmalıdır:

- Küme iş yükü hedef kaydı (MQWDR)
- Küme iş yükü kuyruğu kaydı (MQWQR)
- Küme iş yükü kümesi kaydı (MQWCR)

#### **NextOffset (MQLONG) -giriş**

Sonraki kaydın görelî konumu.

Bu yapı, bir sonraki kaydın ya da yapının görelî konularıyla ilgili bilgileri içerir. *NextOffset*, geçerli kayıttaki uygun görelî konum alanının değeridir ve aşağıdaki alanlardan biri olmalıdır:

- MQWDRiçindekiChannelDefGörelî Konumu alanı
- MQWDRiçindekiClusterRecGörelî Konumu alanı
- MQWQRiçindekiClusterRecGörelî Konumu alanı
- MQWCRiçindekiClusterRecGörelî Konumu alanı

#### **NextRecord (MQPTR) -çıkış**

Sonraki kaydın ya da yapının adresi.

Bu yapı, bir sonraki kaydın ya da yapının adresiyle ilgili bilgileri içerir. *CurrentRecord*, bir MQWDR' in adresiyse ve *NextOffset*, ChannelDefGörelî Konumu alanının değeri, *NextRecord* kanal tanımlama yapısının (MQCD) adresidir.

Sonraki kayıt ya da yapı yoksa, kuyruk yöneticisi *NextRecord* boş değerli işaretçiye ayarlar ve çağrı tamamlanma kodunu MQCC\_WARNING ve neden kodu MQRC\_NO\_RECORD\_AVAILABLEdöndürür.

#### **CompCode (MQHOT) -çıkış**

Tamamlanma kodu.

Tamamlanma kodu aşağıdaki değerlerden birine sahiptir:

**MQCC\_OK**

İşlem başarıyla tamamlandı.

**MQCC\_WARNING**

Uyarı (kısmi tamamlama).

**MQCC\_FAILED**

Arama başarısız oldu.

**Neden ( MQLONG ) -çıkış**

Neden kodu ön eleme CompCode

CompCode MQCC\_OKise:

**MQRC\_NONE**

( 0, X'0000' )

Rapor vermek için bir sebep yok.

CompCode MQCC\_WARNINGise:

**MQRC\_NO\_RECORD\_AVAILABLE**

( 2359, X'0937' )

Kayıt yok. Zincirdeki sonraki kaydın adresini elde etmek için bir küme iş yükü çıkışından bir MQXCLWLN çağrısı yayınlandı. Geçerli kayıt, zincirdeki son kayıttır. Düzeltici işlem: Yok.

CompCode MQCC\_FAILEDise:

**MQRC\_CURRENT\_RECORD\_ERROR**

( 2357, X'0935' )

**CurrentRecord** parametresi geçerli değil. Zincirdeki sonraki kaydın adresini elde etmek için bir küme iş yükü çıkışından bir MQXCLWLN çağrısı yayınlandı. **CurrentRecord** parametresi tarafından belirtilen adres, geçerli bir kaydın adresi değil.

**CurrentRecord** bir hedef kaydın adresi, MQWDR, kuyruk kaydı ( MQWQR ) ya da küme kaydı ( MQWCR ) olmalıdır küme önbelleğinde bulunuyor. Düzeltici işlem: Küme iş yükü çıkışınının, küme önbelleğinde bulunan geçerli bir kaydın adresini geçtiğinden emin olun.

**MQRC\_ENVIRONMENT\_ERROR**

( 2012, X'07DC' )

Arama, ortam içinde geçerli değil. Bir MQXCLWLN çağrısı yayınlandı, ancak küme iş yükü çıkıştan çıkılmadı.

**MQRC\_NEXT\_OFFSET\_ERROR**

( 2358, X'0936' )

**NextOffset** parametresi geçerli değil. Zincirdeki sonraki kaydın adresini elde etmek için bir küme iş yükü çıkışından bir MQXCLWLN çağrısı yayınlandı. **NextOffset** parametresiyle belirtilen görelî konum geçerli değil. **NextOffset** aşağıdaki alanlardan birinin değeri olmalıdır:

- MQWDRiçindekiChannelDefGörelî Konumu alanı
- MQWDRiçindekiClusterRecGörelî Konumu alanı
- MQWQRiçindekiClusterRecGörelî Konumu alanı
- MQWCRiçindekiClusterRecGörelî Konumu alanı

Düzeltici işlem: **NextOffset** parametresi için belirtilen değerin, daha önce listelenen alanlardan birinin değeri olduğundan emin olun.

**MQRC\_NEXT\_RECORD\_ERROR**

( 2361, X'0939' )

**NextRecord** parametresi geçerli değil.

**MQRC\_WXP\_ERROR**

( 2356, X'0934' )

İş yükü çıkış parametresi yapısı geçerli değil. Zincirdeki sonraki kaydın adresini elde etmek için bir küme iş yükü çıkışından bir MQXCLWLN çağrısı yayınlandı. The workload exit parameter structure **ExitParms** is not valid, for one of the following reasons:

- Parametre göstergesi geçerli değil. Geçerli olmayan parametre işaretlerinin saptanması her zaman mümkün değildir; algılanmazsa, önceden kestirilemeyen sonuçlar ortaya çıkar.
- StrucId alanı MQWXP\_STRUC\_ID değil.
- Sürüm alanı MQWXP\_VERSION\_2 değil.
- Bağlam alanı, kuyruk yöneticisi tarafından çıkışa geçirilen değeri içermiyor.

Düzeltilici işlem: **ExitParms** için belirtilen parametrenin, çıkış çağrıldığında çıkışa geçirilen MQWXP yapısı olduğunu doğrulayın.

### İlgili başvurular

MQXCLWLN kullanım notları-Küme iş yükü kayıtlarında dolaşma  
Önbellek statik olsa bile, küme kayıtlarında gezinmek için MQXCLWLN ' u kullanın.

MQXCLWLN' in dil çağrıları  
MQXCLWLN , iki dil, C ve High Level Assembler ürünlerini destekler.

### **MQXCLWLN kullanım notları-Küme iş yükü kayıtlarında dolaşma**

Önbellek statik olsa bile, küme kayıtlarında gezinmek için MQXCLWLN ' u kullanın.

Küme önbelleği dinamikse, kayıtlarda gezinmek için MQXCLWLN çağrısı kullanılmalıdır. Kayıtlar arasında gezinmek için basit işaretçi ve görelî konum aritmetiği kullanılırsa, çıkış olağan dışı bir şekilde sona erer.

Küme önbelleği durağan (static) ise, kayıtlarda gezinmek için MQXCLWLN kullanılmaması gerekir. Tipik olarak, önbellek durağan olduğunda bile MQXCLWLN kullanılır. Bundan sonra küme önbelleğini, iş yükü çıkışını değiştirmeye gerek kalmadan dinamik olarak değiştirebilirsiniz.

### İlgili başvurular

MQXCLWLN için parametreler-Küme iş yükü kayıtlarına gidin  
MQXCLWLN çağrısındaki parametrelerin açıklaması.

MQXCLWLN' in dil çağrıları  
MQXCLWLN , iki dil, C ve High Level Assembler ürünlerini destekler.

### **MQXCLWLN' in dil çağrıları**

MQXCLWLN , iki dil, C ve High Level Assembler ürünlerini destekler.

## C çağırma

```
MQXCLWLN (&ExitParms, CurrentRecord, NextOffset, &NextRecord, &CompCode, &Reason) ;
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
typedef struct tagMQXCLWLN {
MQWXP ExitParms; /* Exit parameter block */
MQPTR CurrentRecord; /* Address of current record*/
MQLONG NextOffset; /* Offset of next record */
MQPTR NextRecord; /* Address of next record or structure */
MQLONG CompCode; /* Completion code */
MQLONG Reason; /* Reason code qualifying CompCode */
}
```

## High Level Assembler çağırısı

```
CALL MQXCLWLN, (CLWLEXITPARMS, CURRENTRECORD, NEXTOFFSET, NEXTRECORD, COMPCODE, REASON)
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
CLWLEXITPARMS CMQWXP, Cluster workload exit parameter block
CURRENTRECORD CMQWDRA, Current record
NEXTOFFSET DS F Next offset
NEXTRECORD DS F Next record
```



COMPCODE DS F Completion code  
REASON DS F Reason code qualifying COMPCODE

## İlgili başvurular

[MQXCLWLN için parametreler-Küme iş yükü kayıtlarına gidin](#)  
[MQXCLWLN çağrısındaki parametrelerin açıklaması.](#)

[MQXCLWLN kullanım notları-Küme iş yükü kayıtlarında dolaşma](#)  
Önbellek statik olsa bile, küme kayıtlarında gezinmek için MQXCLWLN ' u kullanın.

## MQWXP -Küme iş yükü çıkış değiştirgesi yapısı

Aşağıdaki çizelge, MQWXP -Cluster iş yükü çıkış değiştirgesi yapısındaki alanları özetlemektedir.

Çizelge 827. MQWXP' deki alanlar		
Alan	Tanım	Sayfa
<i>StrucId</i>	Yapı tanıtcısı	<a href="#">StrucId</a>
<i>Version</i>	Yapı sürüm numarası	<a href="#">S\u00fcr\u00fcm</a>
<i>ExitId</i>	Çıkış tipi	<a href="#">ExitId</a>
<i>ExitReason</i>	Çıkışa çağırılması nedeni	<a href="#">ExitReason</a>
<i>ExitResponse</i>	Çıkıştan yanıt	<a href="#">ExitResponse</a>
<i>ExitResponse2</i>	Çıkıştan ikincil yanıt	<a href="#">ExitResponse2</a>
<i>Feedback</i>	Feedback kodu	<a href="#">Geribildirim</a>
<i>Flags</i>	İşaretler değerleri. Bu bit işaretleri, konulmakta olan iletiye ilişkin bilgileri belirtmek için kullanılır	<a href="#">İşaretler</a>
<i>ExitUserArea</i>	Kullanıcı alanından çık	<a href="#">ExitUserAlanı</a>
<i>ExitData</i>	Verilerden çık	<a href="#">ExitData</a>
<i>MsgDescPtr</i>	İleti tanımlayıcısının adresi ( MQMD )	<a href="#">MsgDescPtr 'si</a>
<i>MsgBufferPtr</i>	İleti verilerinin bazılarını ya da tümünü içeren arabelleğin adresi	<a href="#">MsgBufferPtr</a>
<i>MsgBufferLength</i>	İleti verilerini içeren arabelleğin uzunluğu	<a href="#">MsgBufferUzunluğu</a>
<i>MsgLength</i>	İletin tam uzunluğu	<a href="#">MsgLength</a>
<i>QName</i>	Kuyruğun adı	<a href="#">QName</a>
<i>QMgrName</i>	Yerel kuyruk yöneticisinin adı	<a href="#">QMgrName</a>
<i>DestinationCount</i>	Olası varış noktalarının sayısı	<a href="#">DestinationCount</a>
<i>DestinationChosen</i>	Hedef seçildi	<a href="#">DestinationChosen</a>
<i>DestinationArrayPtr</i>	Hedef kayıtlarla ilgili bir dizi işaretçi adresi ( MQWDR )	<a href="#">DestinationArrayPtr 'si</a>
<i>QArrayPtr</i>	Kuyruk kayıtlarıyla ilgili bir dizi işaretinin adresi ( MQWQR )	<a href="#">QArrayPtr</a>
<b>Not:</b> Sürüm MQWXP_VERSION_2değerinden küçükse, geri kalan alanlar yoksayılr.		
<i>CacheContext</i>	Bağlam Bilgileri	<a href="#">CacheContext</a>
<i>CacheType</i>	Küme önbelleğinin tipi	<a href="#">CacheType</a>

Çizelge 827. MQWXP' deki alanlar (devamı var)		
Alan	Tanım	Sayfa
<b>Not:</b> Sürüm MQWXP_VERSION_3değerinden küçükse, geri kalan alanlar yoksayılr.		
<i>CLWLMRUChannels</i>	İzin verilen etkin giden küme kanalı sayısı üst sınırı	<a href="#">CLWLMRUChannels</a>
<b>Not:</b> Sürüm MQWXP_VERSION_4değerinden küçükse, geri kalan alanlar yoksayılr.		
<i>pEntryPoints</i>	MQI ve DCI çağrılarının yapılmasına izin vermek için MQIEP yapısının adresi	<a href="#">pEntryNoktaları</a>

Küme iş yükü çıkış parametresi yapısı, küme iş yükü çıkışa geçirilen bilgileri açıklamalıdır.

Küme iş yükü çıkış değiştirgesi yapısı tüm altyapılarda desteklenir

Buna ek olarak, MQWXP1, MQWXP2 ve MQWXP3 yapıları geriye dönük uyumluluk için kullanılabilir.

### İlgili başvurular

[MQ\\_CLUSTER\\_WORKLOAD\\_EXIT](#) -Arama açıklaması

Küme iş yükü çıkışı, bir iletiyi kullanılabilir bir kuyruk yöneticisine yönelmek için kuyruk yöneticisi tarafından çağrılır.

[MQXCLWLN](#) -Küme iş yükü kayıtlarında dolaşma

The MQXCLWLN call is used to navigate through the chains of MQWDR, MQWQR, and MQWCR records stored in the cluster cache.

[MQWDR](#)-Küme iş yükü hedef kaydı yapısı

Aşağıdaki çizelge, MQWDR -Küme iş yükü hedefi kayıt yapısındaki alanları özetler.

[MQWQR](#) -Küme iş yükü kuyruğu kayıt yapısı

Aşağıdaki çizelge, MQWQR -Küme iş yükü kuyruğu kayıt yapısındaki alanları özetlemektedir.

[MQWCR](#) -Küme iş yükü kümesi kayıt yapısı

Aşağıdaki çizelge, MQWCR küme iş yükü kaydı yapısındaki alanları özetlemektedir.

### **MQWXP içindeki alanlar-Küme iş yükü çıkış değiştirgesi yapısı**

MQWXP -Cluster iş yükü çıkış değiştirgesi yapısındaki alanların tanımı

#### **StrucId (MQCHAR4)-giriş**

Küme iş yükü çıkış parametresi yapısıyla ilgili yapı tanıtıcısı.

- StrucId değeri MQWXP\_STRUC\_IDolur.
- For the C programming language, the constant MQWXP\_STRUC\_ID\_ARRAY is also defined. Bu değer, MQWXP\_STRUC\_IDile aynı değere sahiptir. Bu, bir dizgi yerine bir karakter dizisidir.

#### **Sürüm (MQHOT)-giriş**

Yapı sürüm numarasını belirtir. Sürüm , aşağıdaki değerlerden birini alır:

##### **MQWXP\_VERSION\_1**

Version-1 kümesi iş yükü çıkış parametre yapısı.

MQWXP\_VERSION\_1 , tüm ortamlarda desteklenir.

##### **MQWXP\_VERSION\_2**

Version-2 küme iş yükü çıkış parametre yapısı.

MQWXP\_VERSION\_2 aşağıdaki ortamlarda desteklenir:

-  AIX
-  IBM i
-  Linux

- **Windows** Windows

### **MQWXP\_VERSION\_3**

Version-3 küme iş yükü çıkış parametre yapısı.

MQWXP\_VERSION\_3 aşağıdaki ortamlarda desteklenir:

- **AIX** AIX
- **IBM i** IBM i
- **Linux** Linux
- **Windows** Windows

### **MQWXP\_VERSION\_4**

Version-4 kümesi iş yükü çıkış parametre yapısı.

MQWXP\_VERSION\_4 aşağıdaki ortamlarda desteklenir:

- **AIX** AIX
- **IBM i** IBM i
- **Linux** Linux
- **Windows** Windows

### **MQWXP\_CURRENT\_VERSION**

Küme iş yükü çıkış parametresi yapısının yürürlükteki sürümü.

### **ExitId (MQLONG)-giriş**

Çağrılmakta olan çıkışa ilişkin tipi gösterir. Desteklenen tek çıkış, küme iş yükü çıkışıdır.

- ExitId değeri MQXT\_CLUSTER\_WORKLOAD\_EXIT olmalıdır.

### **ExitReason (MQUZZ)-giriş**

Küme iş yükü çıkışının çağrılmasına ilişkin nedeni belirtir. ExitReason , aşağıdaki değerlerden birini alır:

#### **MQXR\_INIT**

Çıkışa ilk kez çağrılmakta olduğunu gösterir.

Çıkışa gereksinim olabileceği kaynakları (ana saklama alanı gibi) edinin ve kullanıma hazırlayın.

#### **MQXR\_TERM**

Çıkışa son verilmek üzere olduğunu gösterir.

Çıkış, ana saklama alanı gibi ilk kullanıma hazırlanandan itibaren edinmiş olabileceği tüm kaynakları serbest.

#### **MQXR\_CLWL\_OPEN**

MQOPEN tarafından çağrılan.

#### **MQXR\_CLWL\_PUT**

Called by MQPUT or MQPUT1.

#### **MQXR\_CLWL\_MOVE**

Kanal durumu değişince MCA tarafından çağrılıyor.

#### **MQXR\_CLWL\_REPOS**

Havuz yöneticisi PCF iletisi için MQPUT ya da MQPUT1 tarafından çağrılan.

#### **MQXR\_CLWL\_REPOS\_MOVE**

Kanal durumu değiştiyse, bir havuz yöneticisi PCF iletisi için MCA tarafından çağrılıyor.

### **ExitResponse (MQHOT)-çıkış**

İletinin işlenmesinin devam edip etmeyeceğini belirtmek için ExitResponse seçeneğini belirleyin. Bu değer aşağıdaki değerlerden biri olmalıdır:

## MQXCC\_OK

İletiyi olağan şekilde işlemeye devam edin.

- DestinationChosen , iletinin gönderileceği hedefi tanımlar.

## MQXCC\_SUPPRESS\_FUNCTION

İletiyi işlemeyi durdurun.

- Kuyruk yöneticisi tarafından alınan işlemler, çıkışa neden çağrıldığı için bağlıdır:

Çizelge 828. Kuyruk yöneticisi tarafından yapılan işlemler	
ExitReason	Uygulanan İşlem
<ul style="list-style-type: none"><li>– MQXR_CLWL_OPEN</li><li>– MQXR_CLWL_REPOS</li><li>– MQXR_CLWL_PUT</li></ul>	MQOPEN, MQPUTya da MQPUT1 çağrısı, tamamlanma kodu MQCC_FAILED ve neden kodu MQRC_STOPPED_BY_CLUSTER_EXITile başarısız olur.
<ul style="list-style-type: none"><li>– MQXR_CLWL_MOVE</li><li>– MQXR_CLWL_REPOS_MOVE</li></ul>	İleti, ölü-harf kuyruğunda yer alıyor.

## MQXCC\_SUPPRESS\_EXIT

Geçerli iletiyi olağan şekilde işlemeye devam edin. Kuyruk yöneticisi sona erdirilinceye kadar çıkışı yeniden çağırmayın.

The queue manager processes subsequent messages as if the ClusterWorkloadÇık queue manager attribute is blank. DestinationChosen , geçerli iletinin gönderildiği hedefi tanımlar.

## Diğer herhangi bir değer

Process the message as if MQXCC\_SUPPRESS\_FUNCTION is specified.

## ExitResponse2 (MQLONG)-giriş/çıkış

Daha fazla bilgi içeren kuyruk yöneticisini sağlamak için ExitResponse2 seçeneğini belirleyin.

- MQXR2\_STATIC\_CACHE is the default value, and is set on entry to the exit.
- ExitReason , MQXR\_INITdeğerine sahip olduğunda, çıkış ExitResponse2içinde aşağıdaki değerlerden birini ayarlayabilir:

### MQXR2\_STATIC\_CACHE

Çıkış, statik bir küme önbelleği gerektirir.

- Küme önbelleği statik ise, çıkış, küme önbelleğindeki kayıtların zincirinde gezinmek için MQXCLWLN çağrısını kullanmamaya gerek yoktur.
- Küme önbelleği dinamikse, çıkış önbellekteki kayıtlarda düzgün bir şekilde gidemez.

**Not:** Kuyruk yöneticisi, çıkış ExitResponse alanında MQXCC\_SUPPRESS\_EXIT değerini döndürdüğü halde MQXR\_INIT çağrısından geri dönüşü işler.

### MQXR2\_DYNAMIC\_CACHE

Çıkış, statik ya da dinamik bir önbellek ile çalışabilir.

- Çıkış bu değeri döndürürse, küme önbelleğindeki kayıtların zincirinde gezinmek için çıkışın MQXCLWLN çağrısını kullanması gerekir.

## Geribildirim (MQHOT)-giriş

Ayrılmış bir alan. Değer sıfır.

## İşaretler (MQHOT)-giriş

Yerleştirmekte olan iletiyle ilgili bilgileri gösterir.

- İşaretler değeri MQWXP\_PUT\_BY\_CLUSTER\_CHL' dir. İleti, yerel olarak ya da kümeli olmayan bir kanaldan değil, bir küme kanalından kaynaklanır. Diğer bir deyişle, ileti başka bir küme kuyruk yöneticisinden gelmiş olabilir.

**Reserved (MQHOT)-giriş**

Ayrılmış bir alan. Değer sıfır.

**ExitUserArea (MQBYTE16)-giriş/çıkış**

Çıkışa çağrılar arasında iletişim kurmak için ExitUserArea seçeneğini belirleyin.

- ExitUserArea , çıkışa ilişkin ilk çağrıdan önce ikili sıfır olarak ilk kullanıma hazırlandı. Çıkışta bu alanda yapılan değişiklikler, MQCONN çağrısı ile eşleşen MQDISC çağrısı arasında oluşan çıkışa ilişkin çağrılar arasında korunur. MQDISC çağrısı gerçekleştiğinde, alan ikili sıfıra sıfırlanır.
- The first invocation of the exit is indicated by the ExitReason field having the value MQXR\_INIT.
- Aşağıdaki sabitler tanımlanır:

**MQXUA\_NONE -dizgi**

**MQXUA\_NONE\_ARRAY -karakter dizisi**

Kullanıcı bilgisi yok. Her iki sabit değer de alanın uzunluğuna ilişkin ikili sıfırdır.

**MQ\_EXIT\_USER\_AREA\_LENGTH**

ExitUserAreauzunluğu.

**ExitData (MQCHAR32)-giriş**

ClusterWorkloadData kuyruk yöneticisi özniteliğinin değeri. Bu öznitelik için herhangi bir değer tanımlanmadıysa, bu alan tamamen boşluktur.

- ExitData uzunluğu MQ\_EXIT\_DATA\_LENGTHtarafından verilir.

**MsgDescPtr (PMQMD)-giriş**

İşlenmekte olan iletiye ilişkin ileti tanımlayıcısının (MQMD) kopyasının adresi.

- Çıkışa göre ileti tanımlayıcısında yapılan değişiklikler kuyruk yöneticisi tarafından yoksayılır.
- ExitReason , aşağıdaki değerlerden birine sahipse MsgDescPtr boş değerli işaretçiye ayarlanır ve çıkışa ileti tanımlayıcısı iletilmezse:
  - MQXR\_INIT
  - MQXR\_TERM
  - MQXR\_CLWL\_OPEN

**MsgBufferPtr (PMQVOID)-giriş**

İleti verilerinin ilk MsgBufferUzunluğu byte 'ının bir kopyasını içeren bir arabelleğin adresi.

- Çıkışa göre ileti verilerinde yapılan değişiklikler kuyruk yöneticisi tarafından yoksayılır.
- Şu durumda çıkışa ileti verisi iletilmedi:
  - MsgDescPtr boş değerli işaretçi.
  - İletinin veri yok.
  - ClusterWorkloadLength kuyruk yöneticisi özniteliği sıfır.

Bu durumda, MsgBufferPtr boş değerli göstergedir.

**MsgBufferUzunluğu (MQUZE)-giriş**

Çıkışa aktarılan ileti verilerini içeren arabelleğin uzunluğu.

- Uzunluk, ClusterWorkloadLength kuyruk yöneticisi özniteliği tarafından denetlenir.
- Uzunluk, iletinin tam uzunluğundan az olabilir, bkz. MsgLength.

**MsgLength (MQHOT)-giriş**

Çıkışa geçirilen tam iletinin uzunluğu.

- MsgBufferUzunluğu , iletinin tam uzunluğundan az olabilir.
- MsgLength is zero if ExitReason is MQXR\_INIT, MQXR\_TERM, or MQXR\_CLWL\_OPEN.

**QName (MQCHAR48)-giriş**

Hedef kuyruğun adı. Kuyruk, bir küme kuyruğudur.

- QName , MQ\_Q\_NAME\_LENGTHuzunluğudur.

**QMgrName (MQCHAR48)-giriş**

Küme iş yükü çıkışını çağırılan yerel kuyruk yöneticisinin adı.

- QMgrName , MQ\_Q\_MGR\_NAME\_LENGTH uzunluğundadır.

**DestinationCount (MQUZE)-giriş**

Olası varış noktalarının sayısı. Hedefler, hedef kuyruğun eşgörünümleridir ve hedef kayıtlar tarafından açıklanmıştır.

- Hedef kayıt, MQWDR yapısıdır. Her bir olası rota için, kuyruğun her bir örneğine yönelik bir yapı vardır.
- MQWDR yapıları bir işaretçiler dizisi tarafından ele alınmakta, bkz. DestinationArrayPtr.

**DestinationChosen (MQUZE)-giriş/çıkış**

Seçilen istikamet.

- İletinin gönderileceği rota ve kuyruk örneğini tanımlayan MQWDR yapısının numarası.
- Değer, 1- DestinationCount aralığında yer alıyor.
- Çıkışa giriş sırasında, DestinationChosen , kuyruk yöneticisinin seçtiği rota ve kuyruk örneğini gösterir. Çıkış bu seçeneği kabul edebilir ya da farklı bir rota ve kuyruk örneği seçebilir.
- Çıkışa göre ayarlanan değer, 1- DestinationCount aralığında olmalıdır. Başka bir değer döndürülürse, kuyruk yöneticisi çıkışa giriş sırasında DestinationChosen değerini kullanır.

**DestinationArrayPtr (PPMQWDR)-giriş**

Hedef kayıtları gösteren bir dizi işaretinin adresi (MQWDR).

- DestinationCount hedef kayıtları vardır.

**QArrayPtr (PPMQWR)-giriş**

Kuyruk kayıtlarına ilişkin bir dizi işaretçi dizisi (MQWR).

- Kuyruk kayıtları kullanılabilirse, bunların DestinationCount tanesi vardır.
- Kullanılabilir kuyruk kaydı yoksa, QArrayPtr boş değerli göstergeci.

**Not:** QArrayPtr can be the null pointer even when DestinationCount is greater than zero.

**CacheContext (MQPTR): Sürüm 2-giriş**

The CacheContext field is reserved for use by the queue manager. Çıkış, bu alanın değerini değiştirmemelidir.

**CacheType (MQLONG): Sürüm 2-giriş**

Küme önbelleğinde aşağıdaki tiplerden biri vardır:

**MQCLCT\_STATIC**

Önbellek durağan (static).

- Önbelleğin boyutu sabittir ve kuyruk yöneticisi çalıştıkça büyümemez.
- Bu önbellekteki kayıtlarda dolaşmak için MQXCLWLN çağrısını kullanmanız gerekmez.

**MQCLCT\_DYNAMIC**

Önbellek dinamik.

- Değişen küme bilgilerini barındırmak için önbelleğin boyutu artırılabilir.
- Bu önbellekteki kayıtlarda dolaşmak için MQXCLWLN çağrısını kullanmanız gerekir.

**CLWLMRChannels (MQWWT): Sürüm 3-giriş**

Küme iş yükü seçimi algoritması tarafından kullanılmak üzere değerlendirilmek üzere etkin giden küme kanalı sayısı üst sınırını belirtir.

- CLWLMRChannels , 1-999 999 999 numaradır.

**pEntryPuanları (PMQIEP): Sürüm 4**

MQI ve DCI çağrılarının yapılabileceği MQIEP yapısının adresi.

**İlgili başvurular**

[MQWXPiçin ilk değerler ve dil bildirimleri](#)

İlk değerler ve C ve High Level Assembler Language bildirimleri for MQWXP -Cluster workload exit parametre yapısı.

### **MQWXP için ilk değerler ve dil bildirimleri**

İlk değerler ve C ve High Level Assembler Language bildirimleri for MQWXP -Cluster workload exit parametre yapısı.

Çizelge 829. MQWXP' deki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
<i>StrucId</i>	MQWXP_STRUC_ID	'WXP↵'
<i>Version</i>	MQWXP_VERSION_2	2
<i>ExitId</i>	Yok	0
<i>ExitReason</i>	MQXCC_OK	0
<i>ExitResponse</i>	Yok	0
<i>ExitResponse2</i>	Yok	0
<i>Flags</i>	Yok	0
<i>ExitUserArea</i>	{MQXUA_NONE_ARRAY}	0
<i>ExitData</i>	Yok	""
<i>MsgDescPtr</i>	Yok	NULL
<i>MsgBufferPtr</i>	Yok	NULL
<i>MsgBufferLength</i>	Yok	0
<i>MsgBufferPtr</i>	Yok	0
<i>QName</i>	Yok	""
<i>QMgrName</i>	Yok	""
<i>DestinationCount</i>	Yok	0
<i>DestinationChosen</i>	Yok	0
<i>DestinationArrayPtr</i>	Yok	NULL
<i>QArrayPtr</i>	Yok	NULL
<i>CacheContext</i>	Yok	NULL
<i>CacheType</i>	MQCLCT_DYNAMIC	1
<i>CLWLMRUChannels</i>	Yok	0
<i>pEntryPoints</i>	Yok	NULL

**Notlar:**

- ↵ simgesi tek bir boş karakteri temsil eder.
- C programlama dilinde, MQWXP\_DEFAULT makro değişkeni varsayılan değerleri içerir. Yapıdaki alanlara ilişkin ilk değerleri sağlamak için bunu aşağıdaki şekilde kullanın:

```
MQWDR MyWXP = {MQWXP_DEFAULT};
```

## C bildirimi

```

typedef struct tagMQWXP {
    MQCHAR4   StrucId;           /* Structure identifier */
    MQLONG    Version;          /* Structure version number */
    MQLONG    ExitId;           /* Type of exit */
    MQLONG    ExitReason;       /* Reason for invoking exit */
    MQLONG    ExitResponse;     /* Response from exit */
    MQLONG    ExitResponse2;    /* Reserved */
    MQLONG    Feedback;         /* Reserved */
    MQLONG    Flags;            /* Flags */
    MQBYTE16  ExitUserArea;     /* Exit user area */
    MQCHAR32  ExitData;         /* Exit data */
    PMQMD     MsgDescPtr;       /* Address of message descriptor */
    PMQVOID   MsgBufferPtr;     /* Address of buffer containing some
    or all of the message data */

    MQLONG    MsgBufferLength;  /* Length of buffer containing message
    data */

    MQLONG    MsgLength;        /* Length of complete message */
    MQCHAR48  QName;            /* Queue name */
    MQCHAR48  QMgrName;         /* Name of local queue manager */
    MQLONG    DestinationCount; /* Number of possible destinations */
    MQLONG    DestinationChosen; /* Destination chosen */
    PPMQWDR   DestinationArrayPtr; /* Address of an array of pointers to
    destination records */

    PPMQWQR   QArrayPtr;        /* Address of an array of pointers to
    queue records */

    /* version 1 */
    MQPTR     CacheContext;     /* Context information */
    MQLONG    CacheType;        /* Type of cluster cache */
    /* version 2 */
    MQLONG    CLWLMRUChannels;  /* Maximum number of most recently
    used cluster channels */

    /* version 3 */
    PMQIEP    pEntryPoints;     /* Address of the MQIEP structure */
    /* version 4 */
};

```

## High Level Assembler

MQWXP	DSECT		
MQWXP_STRUCID	DS	CL4	Structure identifier
MQWXP_VERSION	DS	F	Structure version number
MQWXP_EXITID	DS	F	Type of exit
MQWXP_EXITREASON	DS	F	Reason for invoking exit
MQWXP_EXITRESPONSE	DS	F	Response from exit
MQWXP_EXITRESPONSE2	DS	F	Reserved
MQWXP_FEEDBACK	DS	F	Reserved
MQWXP_RESERVED	DS	F	Reserved
MQWXP_EXITUSERAREA	DS	XL16	Exit user area
MQWXP_EXITDATA	DS	CL32	Exit data
MQWXP_MSGDESCPTR	DS	F	Address of message
*			descriptor
MQWXP_MSGBUFFERPTR	DS	F	Address of buffer containing
*			some or all of the message
*			data
MQWXP_MSGBUFFERLENGTH	DS	F	Length of buffer containing
*			message data
MQWXP_MSGLENGTH	DS	F	Length of complete message
MQWXP_QNAME	DS	CL48	Queue name
MQWXP_QMGRNAME	DS	CL48	Name of local queue manager
MQWXP_DESTINATIONCOUNT	DS	F	Number of possible
*			destinations
MQWXP_DESTINATIONCHOSEN	DS	F	Destination chosen
MQWXP_DESTINATIONARRAYPTR	DS	F	Address of an array of
*			pointers to destination
*			records
MQWXP_QARRAYPTR	DS	F	Address of an array of
*			pointers to queue records
MQWXP_CACHECONTEXT	DS	F	Context information
MQWXP_CACHETYPE	DS	F	Type of cluster cache
MQWXP_CLWLMRUCHANNELS	DS	F	Number of most recently used
*			channels for workload balancing
MQWXP_LENGTH	EQU	*-MQWXP	Length of structure



**İlgili başvurular**

[MQWXP içindeki alanlar-Küme iş yükü çıkış değiştirgesi yapısı](#)  
[MQWXP -Cluster iş yükü çıkış değiştirgesi yapısındaki alanların tanımı](#)

**MQWDR-Küme iş yükü hedef kaydı yapısı**

Aşağıdaki çizelge, MQWDR -Küme iş yükü hedefi kayıt yapısındaki alanları özetler.

<i>Çizelge 830. MQWDR' daki alanlar</i>		
Alan	Tanım	Sayfa
<i>StrucId</i>	Yapı tanıtıcısı	<a href="#">StrucId</a>
<i>Version</i>	Yapı sürüm numarası	<a href="#">S\u00fcr\u00fcm</a>
<i>StrucLength</i>	MQWDR yapısının uzunluğu	<a href="#">StrucLength</a>
<i>QMgrFlags</i>	Kuyruk yöneticisi işaretleri	<a href="#">QMgrFlags</a>
<i>QMgrIdentifier</i>	Kuyruk yöneticisi tanıtıcısı	<a href="#">QMgrIdentifier</a>
<i>QMgrName</i>	Kuyruk yöneticisi adı	<a href="#">QMgrName</a>
<i>ClusterRecOffset</i>	İlk küme kaydının mantıksal görel konumu ( MQWCR )	<a href="#">ClusterRecKayması</a>
<i>ChannelState</i>	Kanal durumu	<a href="#">ChannelState</a>
<i>ChannelDefOffset</i>	Kanal tanımlama yapısının mantıksal görel konumu ( MQCD )	<a href="#">ChannelDefGörel Konumu</a>
<b>Not:</b> Sürüm MQWDR_VERSION_2değerinden küçükse, geri kalan alanlar yoksayılr.		
<i>DestSeqNumber</i>	Kanal hedef sıra numarası	<a href="#">DestSeqNumarası</a>
<i>DestSeqFactor</i>	Ağırlıklandırma için kanal hedef sıra katsayısı	<a href="#">DestSeqKatsayısı</a>

Küme iş yükü hedef kayıt yapısı, iletiye ilişkin olası hedeflerden biriyle ilgili bilgileri içerir. Hedef kuyruğun her bir eşgörünümü için bir küme iş yükü hedef kaydı yapısı vardır.

Küme iş yükü hedef kaydı yapısı tüm ortamlarda desteklenir.

Buna ek olarak, MQWDR1 ve MQWDR2 yapıları geriye dönük uyumluluk için kullanılabilir.

**İlgili başvurular**

[MQ\\_CLUSTER\\_WORKLOAD\\_EXIT -Arama açıklaması](#)

Küme iş yükü çıkışı, bir iletiyi kullanılabilir bir kuyruk yöneticisine yönelmek için kuyruk yöneticisi tarafından çağrılır.

[MQXCLWLN -Küme iş yükü kayıtlarında dolaşma](#)

The MQXCLWLN call is used to navigate through the chains of MQWDR, MQWQR, and MQWCR records stored in the cluster cache.

[MQWXP -Küme iş yükü çıkış değiştirgesi yapısı](#)

Aşağıdaki çizelge, MQWXP -Cluster iş yükü çıkış değiştirgesi yapısındaki alanları özetlemektedir.

[MQWQR -Küme iş yükü kuyruğu kayıt yapısı](#)

Aşağıdaki çizelge, MQWQR -Küme iş yükü kuyruğu kayıt yapısındaki alanları özetlemektedir.

[MQWCR -Küme iş yükü kümesi kayıt yapısı](#)

Aşağıdaki çizelge, MQWCR küme iş yükü kaydı yapısındaki alanları özetlemektedir.

**MQWDR-Küme iş yükü hedefi kayıt yapısındaki alanlar**

MQWDR -Cluster iş yükü hedef kaydı yapısındaki parametrelerin açıklaması.

### **StrucId ( MQCHAR4 ) -giriş**

Küme iş yükü hedef kaydı yapısına ilişkin yapı tanıtıcısı.

- StrucId değeri MQWDR\_STRUC\_ID olur.
- For the C programming language, the constant MQWDR\_STRUC\_ID\_ARRAY is also defined. Bu değer, MQWDR\_STRUC\_ID ile aynı değere sahiptir. Bu, bir dizgi yerine bir karakter dizisidir.

### **Sürüm ( MQLONG ) -giriş**

Yapı sürüm numarası. Sürüm , aşağıdaki değerlerden birini alır:

#### **MQWDR\_VERSION\_1**

Version-1 küme iş yükü hedef kaydı.

#### **MQWDR\_VERSION\_2**

Version-2 küme iş yükü hedef kaydı.

#### **MQWDR\_CURRENT\_VERSION**

Küme iş yükü hedef kaydının yürürlükteki sürümü.

### **StrucLength ( MQLONG ) -giriş**

MQWDR yapısının uzunluğu. StrucLength , aşağıdaki değerlerden birini alır:

#### **MQWDR\_LENGTH\_1**

version-1 küme iş yükü hedef kaydının uzunluğu.

#### **MQWDR\_LENGTH\_2**

version-2 küme iş yükü hedef kaydının uzunluğu.

#### **MQWDR\_CURRENT\_LENGTH**

Küme iş yükü hedef kaydının geçerli sürümü uzunluğu.

### **QMgrFlags ( MQLONG ) -giriş**

Kuyruk yöneticisi işaretleri, MQWDR yapısı tarafından tanımlanan hedef kuyruğu örneğini barındıran kuyruk yöneticisinin özelliklerini belirtir. Aşağıdaki işaretler tanımlanır:

#### **MQQMF\_REPOSITORY\_Q\_MGR**

Hedef, tam bir havuz kuyruk yöneticidir.

#### **MQQMF\_CLUSSDR\_USER\_DEFINED**

Küme-gönderici kanalı el ile tanımlandı.

#### **MQQMF\_CLUSSDR\_AUTO\_DEFINED**

Küme-gönderici kanalı otomatik olarak tanımlandı.

#### **MQQMF\_AVAILABLE**

İleti almak için hedef kuyruk yöneticisi kullanılabilir.

#### **Diğer değerler**

Bu alandaki diğer işaretler, kuyruk yöneticisi tarafından iç amaçlara göre ayarlanabilir.

### **QMgrIdentifier ( MQCHAR48 ) -giriş**

The queue manager identifier is a unique identifier for the queue manager that hosts the instance of the destination queue described by the MQWDR structure.

- Tanıtıcı, kuyruk yöneticisi tarafından oluşturulur.
- QMgrIdentifier , MQ\_Q\_MGR\_IDENTIFIER\_LENGTH uzunluğundadır.

### **QMgrName ( MQCHAR48 ) -giriş**

MQWDR yapısı tarafından tanımlanan hedef kuyruğu örneğini barındıran kuyruk yöneticisinin adı.

- QMgrName , yerel kuyruk yöneticisinin adı olabilir ve kümedeki başka bir kuyruk yöneticisi olarak da olabilir.
- QMgrName , MQ\_Q\_MGR\_NAME\_LENGTH uzunluğundadır.

### **ClusterRecKayma ( MQLONG ) -giriş**

MQWDR yapısına ait olan ilk MQWCR yapısının mantıksal görelî konumu.

- Statik önbellekler için, ClusterRecKayması , MQWDR yapısına ait olan ilk MQWCR yapısının görelî konutdur.

- Kayma, MQWDR yapısının başlangıcındaki bayt cinsinden ölçülür.
- Dinamik önbellekler içeren işaretçi aritmetiği için mantıksal kayma değerini kullanmayın. Sonraki kaydın adresini elde etmek için MQXCLWLN çağrısı kullanılmalıdır.

### **ChannelState ( MQLONG ) -giriş**

Yerel kuyruk yöneticisini MQWDR yapısı ile tanıtilan kuyruk yöneticisine bağlayan kanaldaki durumu. Aşağıdaki değerler olanaklı:

#### **MQCHS\_BINDING**

Kanal ortakla pazarlık yapıyor.

#### **MQCHS\_INACTIVE**

Kanal etkin değil.

#### **MQCHS\_INITIALIZING**

Kanal başlatılıyor.

#### **MQCHS\_PAUSED**

Kanal duraklatıldı.

#### **MQCHS\_REQUESTING**

İstekte bulunanın kanalı bağlantı istiyor.

#### **MQCHS\_RETRYING**

Kanal bağlantı kurma girişiminde bulunuyor.

#### **MQCHS\_RUNNING**

Kanal veri aktarıyor ya da ileti bekliyor.

#### **MQCHS\_STARTING**

Kanal aktif hale gelmek için bekliyor.

#### **MQCHS\_STOPPING**

Kanal durduruluyor.

#### **MQCHS\_STOPPED**

Kanal durdu.

### **ChannelDefKayma ( MQLONG ) -giriş**

The logical offset of the channel definition ( MQCD ) for the channel that links the local queue manager to the queue manager identified by the MQWDR structure.

- ChannelDefGörelili Konumu , ClusterRecKayması gibidir
- İşaretçi aritmetikte mantıksal görelili konum kullanılamaz. Sonraki kaydın adresini elde etmek için, MQXCLWLN çağrısı kullanılmalıdır.

### **DestSeqFactor ( MQLONG ) -giriş**

Kanal seçimini ağırlığa dayalı olarak seçmenize olanak tanıyan hedef sıra katsayısı.

- Kuyruk yöneticisi bunu değiştirmeden önce DestSeqFactor kullanılır.
- İş yükü yöneticisi, iletilerin ağırlıklarına göre kanallara dağıtılmasını sağlayan bir şekilde DestSeqKatsayısı ' i artırır.

### **DestSeqSayı ( MQLONG ) -giriş**

Kuyruk yöneticisi tarafından değişiklik yapmadan önce, küme kanalı hedef değeri.

- İş yükü yöneticisi, her ileti o kanala her yerleştirdiğinde DestSeqNumber (DestSeq) sayısını artırır.
- İş yükü çıkışı, iletiyi hangi kanalda indireceğine karar vermek için DestSeqNumber (DestSeq) numarasını kullanabilir.

### **İlgili başvurular**

MQWDR için ilk değerler ve dil bildirimleri

İlk değerler ve C ve High Level Assembler Dil bildirimleri MQWDR -Küme iş yükü hedef kaydı için.

### **MQWDR için ilk değerler ve dil bildirimleri**

İlk değerler ve C ve High Level Assembler Dil bildirimleri MQWDR -Küme iş yükü hedef kaydı için.

Çizelge 831. MQWDR' da alanların ilk değerleri

Alan adı	Değişmezin adı	Değişmezin değeri
<i>StrucId</i>	MQWDR_STRUC_ID	'WDR↵'
<i>Version</i>	MQWDR_VERSION_1	1
<i>StrucLength</i>	MQWDR_CURRENT_LENGTH <sup>3</sup>	136
<i>QMgrFlags</i>	MQWDR_NONE	0
<i>QMgrIdentifier</i>	Yok	" "
<i>QMgrName</i>	Yok	" "
<i>ClusterRecOffset</i>	Yok	0
<i>ChannelState</i>	Yok	0
<i>ChannelDefOffset</i>	Yok	0
<i>DestSeqNumber</i>	Yok	0
<i>DestSeqFactor</i>	Yok	0

**Notlar:**

1. ↵ simgesi tek bir boş karakteri temsil eder.
2. C programlama dilinde, MQWDR\_DEFAULT makro değişkeni varsayılan değerleri içerir. Yapıdaki alanlara ilişkin ilk değerleri sağlamak için bunu aşağıdaki şekilde kullanın:

```
MQWDR MyWDR = {MQWDR_DEFAULT};
```

3. Başlangıçtaki değerler, yapının uzunluğunu yürürlükteki sürümün uzunluğuna ayarlar ve yapının sürüm 1 'i değil.

## High Level Assembler

```
MQWDR          DSECT
MQWDR_STRUCID  DS   CL4      Structure identifier
MQWDR_VERSION  DS   F        Structure version number
MQWDR_STRUCLNGTH DS   F      Length of MQWDR structure
MQWDR_QMGRFLAGS DS   F      Queue manager flags
MQWDR_QMGRIDENTIFIER DS CL48  Queue manager identifier
MQWDR_QMGRNAME DS   CL48    Queue manager name
MQWDR_CLUSTERRECOFFSET DS   F  Offset of first cluster
*              record
MQWDR_CHANNELSTATE DS   F    Channel state
MQWDR_CHANNELDEFOFFSET DS   F  Offset of channel definition
*              structure
MQWDR_LENGTH    EQU  *-MQWDR Length of structure
MQWDR_AREA      ORG  MQWDR
                DS   CL(MQWDR_LENGTH)
```

## C bildirimi

```
typedef struct tagMQWDR {
    MQCHAR4   StrucId;          /* Structure identifier */
    MQLONG    Version;         /* Structure version number */
    MQLONG    StrucLength;     /* Length of MQWDR structure */
    MQLONG    QMgrFlags;      /* Queue manager flags */
    MQCHAR48  QMgrIdentifier;  /* Queue manager identifier */
    MQCHAR48  QMgrName;       /* Queue manager name */
    MQLONG    ClusterRecOffset; /* Offset of first cluster record */
    MQLONG    ChannelState;   /* Channel state */
    MQLONG    ChannelDefOffset; /* Offset of channel definition structure */
};
```

```

/* Ver:1 */
MQLONG   DestSeqNumber;    /* Cluster channel destination sequence number */
MQINT64  DestSeqFactor;    /* Cluster channel factor sequence number */
/* Ver:2 */
};

```

### İlgili başvurular

MQWDR-Küme iş yükü hedefi kayıt yapısındaki alanlar  
MQWDR -Cluster iş yükü hedef kaydı yapısındaki parametrelerin açıklaması.

### MQWQR -Küme iş yükü kuyruğu kayıt yapısı

Aşağıdaki çizelge, MQWQR -Küme iş yükü kuyruğu kayıt yapısındaki alanları özetlemektedir.

Çizelge 832. MQWQRiçindeki alanlar		
Alan	Tanım	Sayfa
<i>StrucId</i>	Yapı tanıtıcısı	<a href="#">StrucId</a>
<i>Version</i>	Yapı sürüm numarası	<a href="#">S\u00fcr\u00fcm</a>
<i>StrucLength</i>	MQWQR yapısının uzunluğu	<a href="#">StrucLength</a>
<i>QFlags</i>	Kuyruk işaretleri	<a href="#">QFlags</a>
<i>QName</i>	Kuyruk adı	<a href="#">QName</a>
<i>QMgrIdentifier</i>	Kuyruk yöneticisi tanıtıcısı	<a href="#">QMgrIdentifier</a>
<i>ClusterRecOffset</i>	İlk küme kaydının kayması (MQWCR)	<a href="#">ClusterRecKayması</a>
<i>QType</i>	Kuyruk tipi	<a href="#">QType</a>
<i>QDesc</i>	Kuyruk tanımlaması	<a href="#">QDesc</a>
<i>DefBind</i>	Varsayılan bağ tanımı	<a href="#">DefBind</a>
<i>DefPersistence</i>	Varsayılan ileti kalıcılığı	<a href="#">DefPersistence</a>
<i>DefPriority</i>	Varsayılan ileti önceliği	<a href="#">DefPriority</a>
<i>InhibitPut</i>	Kuyruğa ilişkin işlemlere izin verilip verilmeyeceğini	<a href="#">InhibitPut</a>
<b>Not:</b> Sürüm MQWQR_VERSION_2değerinden küçükse, geri kalan alanlar yoksayılr.		
<i>CWLQueuePriority</i>	Kuyruğun önceliğini gösteren 0-9 arasında bir değer	<a href="#">CWLQueuePriority</a>
<i>CLWLQueueRank</i>	Kuyruğun sırasını simgeleyen 0-9 arasında bir değer	<a href="#">CLWLQueueRank</a>
<b>Not:</b> Sürüm MQWQR_VERSION_3değerinden küçükse, geri kalan alanlar yoksayılr.		
<i>DefPutResponse</i>	Varsayılan yanıt koyma yanıtı	<a href="#">DefPutYanıtı</a>

Küme iş yükü kuyruğu kaydı yapısı, iletiye ilişkin olası hedeflerden biriyle ilgili bilgileri içerir. Hedef kuyruğun her bir eşgörünümü için bir küme iş yükü kuyruğu kaydı yapısı vardır.

Küme iş yükü kuyruğu kaydı yapısı tüm ortamlarda desteklenir.

Buna ek olarak, MQWQR1 ve MQWQR2 yapıları geriye dönük uyumluluk için kullanılabilir.

### İlgili başvurular

[MQ\\_CLUSTER\\_WORKLOAD\\_EXIT](#) -Arama açıklaması

Küme iş yükü çıkışı, bir iletiyi kullanılabilir bir kuyruk yöneticisine yönelmek için kuyruk yöneticisi tarafından çağrılır.

[MQXCLWLN](#) -Küme iş yükü kayıtlarında dolaşma

The MQXCLWLN call is used to navigate through the chains of MQWDR, MQWQR, and MQWCR records stored in the cluster cache.

#### MQWXP -Küme iş yükü çıkış değiştirgesi yapısı

Aşağıdaki çizelge, MQWXP -Cluster iş yükü çıkış değiştirgesi yapısındaki alanları özetlemektedir.

#### MQWDR-Küme iş yükü hedef kaydı yapısı

Aşağıdaki çizelge, MQWDR -Küme iş yükü hedefi kayıt yapısındaki alanları özetler.

#### MQWCR -Küme iş yükü kümesi kayıt yapısı

Aşağıdaki çizelge, MQWCR küme iş yükü kaydı yapısındaki alanları özetlemektedir.

### **MQWQR içindeki alanlar-Küme iş yükü kuyruğu kaydı yapısı**

MQWQR -Küme iş yükü kuyruğu kaydı yapısındaki alanların açıklaması.

#### **StrucId ( MQCHAR4 ) -giriş**

Küme iş yükü kuyruğu kaydı yapısına ilişkin yapı tanıtıcısı.

- StrucId değeri MQWQR\_STRUC\_ID olur.
- For the C programming language, the constant MQWQR\_STRUC\_ID\_ARRAY is also defined. Bu değer, MQWQR\_STRUC\_ID ile aynı değere sahiptir. Bu, bir dizgi yerine bir karakter dizisidir.

#### **Sürüm ( MQLONG ) -giriş**

Yapı sürüm numarası. Sürüm , aşağıdaki değerlerden birini alır:

##### **MQWQR\_VERSION\_1**

Version-1 kümesi iş yükü kuyruğu kaydı.

##### **MQWQR\_VERSION\_2**

Version-2 küme iş yükü kuyruk kaydı.

##### **MQWQR\_VERSION\_3**

Version-3 küme iş yükü kuyruğu kaydı.

##### **MQWQR\_CURRENT\_VERSION**

Küme iş yükü kuyruğu kaydının yürürlükteki sürümü.

#### **StrucLength ( MQLONG ) -giriş**

MQWQR yapısının uzunluğu. StrucLength , aşağıdaki değerlerden birini alır:

##### **MQWQR\_LENGTH\_1**

version-1 küme iş yükü kuyruğu kaydının uzunluğu.

##### **MQWQR\_LENGTH\_2**

version-2 küme iş yükü kuyruğu kaydının uzunluğu.

##### **MQWQR\_LENGTH\_3**

version-3 küme iş yükü kuyruğu kaydının uzunluğu.

##### **MQWQR\_CURRENT\_LENGTH**

Küme iş yükü kuyruğu kaydının geçerli sürümü uzunluğu.

#### **QFlags ( MQLONG ) -giriş**

Kuyruk işaretleri, kuyruğun özelliklerini belirtir. Aşağıdaki işaretler tanımlanır:

##### **MQQF\_LOCAL\_Q**

Hedef, yerel bir kuyruğdur.

##### **MQQF\_CLWL\_USEQ\_ANY**

Yerel ve uzak kuyrukların kullanımında kullanılmasına izin verir.

##### **MQQF\_CLWL\_USEQ\_LOCAL**

Yalnızca yerel kuyruğun yerleştirilmesine izin verir.

#### **Diğer değerler**

Bu alandaki diğer işaretler, kuyruk yöneticisi tarafından iç amaçlara göre ayarlanabilir.

#### **QName ( MQCHAR48 ) -giriş**

İletinin olası hedeflerinden biri olan kuyruğun adı.

- QName , MQ\_Q\_NAME\_LENGTHuzunluğudur.

#### **QMgrIdentifier ( MQCHAR48 ) -giriş**

Kuyruk yöneticisi tanıtıcısı, kuyruk yöneticisi için, MQWQR yapısı tarafından tanımlanan kuyruğu barındıran benzersiz bir tanıtıcıdır.

- Tanıtıcı, kuyruk yöneticisi tarafından oluşturulur.
- QMgrIdentifier , MQ\_Q\_MGR\_IDENTIFIER\_LENGTHuzunluğudur.

#### **ClusterRecKayma ( MQLONG ) -giriş**

MQWQR yapısına ait olan ilk MQWCR yapısının mantıksal görelî konumu.

- Statik önbellekler için, ClusterRecGörelî Konumu , MQWQR yapısına ait olan ilk MQWCR yapısının görelî konudur.
- Görelî konum, MQWQR yapısının başlangıcındaki bayt cinsinden ölçülür.
- Dinamik önbellekler içeren işaretçi aritmetiği için mantıksal kayma değerini kullanmayın. Sonraki kaydın adresini elde etmek için MQXCLWLN çağrısı kullanılmalıdır.

#### **QType ( MQLONG ) -giriş**

Hedef kuyruğun kuyruk tipi. Aşağıdaki değerler olanaklı:

##### **MQCQT\_LOCAL\_Q**

Yerel kuyruk.

##### **MQCQT\_ALIAS\_Q**

Diğer ad kuyruğu.

##### **MQCQT\_REMOTE\_Q**

Uzak kuyruk.

##### **MQCQT\_Q\_MGR\_ALIAS**

Kuyruk yöneticisi diğer adı.

#### **QDesc ( MQCHAR64 ) -giriş**

Kuyruk yöneticinde tanımlanan kuyruk tanımlama kuyruğu özniteliği, MQWQR yapısı tarafından tanımlanan hedef kuyruğunun yönetim ortamını barındırır.

- QDesc , MQ\_Q\_DESC\_LENGTHuzunluğudur.

#### **DefBind ( MQLONG ) -giriş**

Kuyruk yöneticinde tanımlı olan varsayılan bağ tanımlama kuyruğu özniteliği, MQWQR yapısı tarafından tanımlanan hedef kuyruğunun eşgörünümünü barındırır. Kümeler içeren gruplar kullanılırken MQBND\_BIND\_ON\_OPEN ya da MQBND\_BIND\_ON\_GROUP belirtilmelidir. Aşağıdaki değerler olanaklı:

##### **MQBND\_BIND\_ON\_OPEN**

Binding fixed by MQOPEN call.

##### **MQBND\_BIND\_NOT\_FIXED**

Bağ tanımı düzeltilmedi.

##### **MQBND\_BIND\_ON\_GROUP**

Bir uygulamanın, bir ileti grubunun tümünün aynı hedef eşgörünüme ayrılmasını istemesine olanak tanır.

#### **DefPersistence ( MQLONG ) -giriş**

Kuyruk yöneticinde tanımlı olan varsayılan ileti kalıcı saklama kuyruğu özniteliği, MQWQR yapısı tarafından tanımlanan hedef kuyruğunun eşgörünümünü barındıran bir kuyruk öznitelidir. Aşağıdaki değerler olanaklı:

##### **MQPER\_PERSISTENT**

İleti kalıcı.

##### **MQPER\_NOT\_PERSISTENT**

İleti kalıcı değil.

### DefPriority ( MQLONG ) -giriş

Kuyruk yöneticinde tanımlı olan varsayılan ileti önceliği kuyruğu özniteliği, MQWQR yapısı tarafından tanımlanan hedef kuyruğu eşgörünümlerini içerir. Öncelik aralığı 0- MaxPriority' dir.

- 0 en düşük önceliğe sahip.
- MaxPriority , hedef kuyruğun bu örneğini barındıran kuyruk yöneticisinin kuyruk yöneticisi öznesidir.

### InhibitPut ( MQUZE ) -giriş

The put inhibited queue attribute defined on the queue manager that hosts the instance of the destination queue described by the MQWQR structure. Aşağıdaki değerler olanaklı:

#### MQQA\_PUT\_INHIBITED

Put operasyonları engellenir.

#### MQQA\_PUT\_ALLOWED

Put işlemlerine izin verilir.

### CLWLQueuePriority ( MQLONG ) -giriş

MQWQR yapısı tarafından açıklanan hedef kuyruğunun eşgörünümünü barındıran kuyruk yöneticinde tanımlı küme iş yükü kuyruğu önceliği özniteliği.

### CLWLQueueRank ( MQLONG ) -giriş

MQWQR yapısı tarafından açıklanan hedef kuyruğu örneğini barındıran kuyruk yöneticinde tanımlanan küme iş yükü kuyruğu sırası.

### DefPutYanıtı ( MQHOT ) -giriş

The default put response queue attribute defined on the queue manager that hosts the instance of the destination queue described by the MQWQR structure. Aşağıdaki değerler olanaklı:

#### MQPRT\_SYNC\_RESPONSE

MQPUT ya da MQPUT1 çağrılarına zamanuyumlu yanıt.

#### MQPRT\_ASYNC\_RESPONSE

MQPUT ya da MQPUT1 çağrılarına zamanuyumsuz yanıt.

### İlgili başvurular

MQWQR için ilk değerler ve dil bildirimleri

İlk değerler ve C ve High Level Assembler Dil bildirimleri MQWQR -Küme iş yükü kuyruğu kaydı için.

### MQWQR için ilk değerler ve dil bildirimleri

İlk değerler ve C ve High Level Assembler Dil bildirimleri MQWQR -Küme iş yükü kuyruğu kaydı için.

Çizelge 833. MQWQRiçindeki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
<i>StrucId</i>	MQWQR_STRUC_ID_ARRAY	'WQR→'
<i>Version</i>	MQWQR_VERSION_1	1
<i>StrucLength</i>	MQWQR_CURRENT_LENGTH <sup>3</sup>	212
<i>QFlags</i>	Yok	0
<i>QName</i>	Yok	" "
<i>QMgrIdentifier</i>	Yok	" "
<i>ClusterRecOffset</i>	Yok	0
<i>QType</i>	Yok	0
<i>QDesc</i>	Yok	" "
<i>DefBind</i>	Yok	0



Çizelge 833. MQWQRiçindeki alanların ilk değerleri (devamı var)

Alan adı	Değişmezin adı	Değişmezin değeri
<i>DefPersistence</i>	Yok	0
<i>DefPriority</i>	Yok	0
<i>InhibitPut</i>	Yok	0
<i>CLWLQueuePriority</i>	Yok	0
<i>CLWLQueueRank</i>	Yok	0
<i>DefPutResponse</i>	Yok	1

**Notlar:**

1. ~ simgesi tek bir boş karakteri temsil eder.
2. C programlama dilinde, MQWQR\_DEFAULT makro değişkeni varsayılan değerleri içerir. Yapıdaki alanlara ilişkin ilk değerleri sağlamak için bunu aşağıdaki şekilde kullanın:

```
MQWQR MyWQR = {MQWQR_DEFAULT};
```

3. Başlangıçtaki değerler, yapının uzunluğunu yürürlükteki sürümün uzunluğuna ayarlar ve yapının sürüm 1 'i değil.

**C bildiriimi**

```
typedef struct tagMQWQR {
    MQCHAR4   StrucId;           /* Structure identifier */
    MQLONG    Version;          /* Structure version number */
    MQLONG    StrucLength;      /* Length of MQWQR structure */
    MQLONG    QFlags;           /* Queue flags */
    MQCHAR48  QName;           /* Queue name */
    MQCHAR48  QMgrIdentifier;    /* Queue manager identifier */
    MQLONG    ClusterRecOffset; /* Offset of first cluster record */
    MQLONG    QType;           /* Queue type */
    MQCHAR64  QDesc;           /* Queue description */
    MQLONG    DefBind;         /* Default binding */
    MQLONG    DefPersistence;   /* Default message persistence */
    MQLONG    DefPriority;      /* Default message priority */
    MQLONG    InhibitPut;      /* Whether put operations on the queue
                               are allowed */

    /* version 2 */
    MQLONG    CLWLQueuePriority; /* Queue priority */
    MQLONG    CLWLQueueRank;    /* Queue rank */
    /* version 3 */
    MQLONG    DefPutResponse;   /* Default put response */
};
```

**High Level Assembler**

MQWQR	DSECT	
MQWQR_STRUCID	DS	CL4 Structure identifier
MQWQR_VERSION	DS	F Structure version number
MQWQR_STRUCLNGTH	DS	F Length of MQWQR structure
MQWQR_QFLAGS	DS	F Queue flags
MQWQR_QNAME	DS	CL48 Queue name
MQWQR_QMGRIDENTIFIER	DS	CL48 Queue manager identifier
MQWQR_CLUSTERRECOFFSET	DS	F Offset of first cluster record
*		
MQWQR_QTYPE	DS	F Queue type
MQWQR_QDESC	DS	CL64 Queue description
MQWQR_DEFBIND	DS	F Default binding
MQWQR_DEFPERSISTENCE	DS	F Default message persistence
MQWQR_DEFPRIORITY	DS	F Default message priority
MQWQR_INHIBITPUT	DS	F Whether put operations on

*				the queue are allowed
MQWQR_DEFPUTRESPONSE	DS	F		Default put response
MQWQR_LENGTH	EQU	*-MQWQR		Length of structure
	ORG	MQWQR		
MQWQR_AREA	DS	CL(MQWQR_LENGTH)		

### İlgili başvurular

MQWQR içindeki alanlar-Küme iş yükü kuyruğu kaydı yapısı

MQWQR -Küme iş yükü kuyruğu kaydı yapısındaki alanların açıklaması.

### MQWCR -Küme iş yükü kümesi kayıt yapısı

Aşağıdaki çizelge, MQWCR küme iş yükü kaydı yapısındaki alanları özetlemektedir.

Çizelge 834. MQWCR' deki alanlar		
Alan	Tanım	Sayfa
<i>ClusterName</i>	Kümenin adı	<a href="#">ClusterName</a>
<i>ClusterRecOffset</i>	Sonraki küme kaydının kayması ( MQWCR )	<a href="#">ClusterRecKayması</a>
<i>ClusterFlags</i>	Küme işaretleri	<a href="#">ClusterFlags</a>

Küme iş yükü kümesi kayıt yapısı, bir kümeye ilişkin bilgileri içerir. Hedef kuyruğun ait olduğu her küme için, bir küme iş yükü kümesi kayıt yapısı vardır.

Küme iş yükü kümesi kayıt yapısı tüm ortamlarda desteklenir.

### İlgili başvurular

[MQ\\_CLUSTER\\_WORKLOAD\\_EXIT](#) -Arama açıklaması

Küme iş yükü çıkışı, bir iletiyi kullanılabilir bir kuyruk yöneticisine yöneltmek için kuyruk yöneticisi tarafından çağrılır.

[MQXCLWLN](#) -Küme iş yükü kayıtlarında dolaşma

The MQXCLWLN call is used to navigate through the chains of MQWDR, MQWQR, and MQWCR records stored in the cluster cache.

[MQWXP](#) -Küme iş yükü çıkış değiştirgesi yapısı

Aşağıdaki çizelge, MQWXP -Cluster iş yükü çıkış değiştirgesi yapısındaki alanları özetlemektedir.

[MQWDR](#)-Küme iş yükü hedef kaydı yapısı

Aşağıdaki çizelge, MQWDR -Küme iş yükü hedefi kayıt yapısındaki alanları özetler.

[MQWQR](#) -Küme iş yükü kuyruğu kayıt yapısı

Aşağıdaki çizelge, MQWQR -Küme iş yükü kuyruğu kayıt yapısındaki alanları özetlemektedir.

### **MQWCR -Küme iş yükü kümesi kayıt yapısındaki alanlar.**

MQWCR -Küme iş yükü kümesi kayıt yapısındaki alanların açıklaması.

#### **ClusterName ( MQCHAR48 ) -giriş**

MQWCR yapısının sahibi olan hedef kuyruk örneğinin ait olduğu kümenin adı. Hedef kuyruk örneği, bir MQWDR yapısı tarafından tanımlanır.

- ClusterName , MQ\_CLUSTER\_NAME\_LENGTHuzunluğudur.

#### **ClusterRecKayma ( MQLONG ) -giriş**

Sonraki MQWCR yapısının mantıksal görelî konumu.

- Başka bir MQWCR yapısı yoksa, ClusterRecKayma değeri sıfırdır.
- Görelî konum, MQWCR yapısının başlangıcındaki bayt cinsinden ölçülür.

#### **ClusterFlags ( MQLONG ) -giriş**

Küme işaretleri, MQWCR yapısıyla tanımlanan kuyruk yöneticisinin özelliklerini gösterir. Aşağıdaki işaretler tanımlanır:

### **MQQMF\_REPOSITORY\_Q\_MGR**

Hedef, tam bir havuz kuyruk yöneticidir.

### **MQQMF\_CLUSSDR\_USER\_DEFINED**

Küme-gönderici kanalı el ile tanımlandı.

### **MQQMF\_CLUSSDR\_AUTO\_DEFINED**

Küme-gönderici kanalı otomatik olarak tanımlandı.

### **MQQMF\_AVAILABLE**

İleti almak için hedef kuyruk yöneticisi kullanılabilir.

### **Diğer değerler**

Bu alandaki diğer işaretler, kuyruk yöneticisi tarafından iç amaçlara göre ayarlanabilir.

## **İlgili başvurular**

MQWCR için ilk değerler ve dil bildirimleri

İlk değerler ve C ve High Level Assembler Dil bildirimleri MQWCR -Küme iş yükü kümesi kaydı yapısı için.

## **MQWCR için ilk değerler ve dil bildirimleri**

İlk değerler ve C ve High Level Assembler Dil bildirimleri MQWCR -Küme iş yükü kümesi kaydı yapısı için.

Çizelge 835. MQWCR' deki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
<i>ClusterName</i>	Yok	" "
<i>ClusterRecOffset</i>	Yok	0
<i>ClusterFlags</i>	Yok	0

## **C bildirimi**

```
typedef struct tagMQWCR {
    MQCHAR48 ClusterName; /* Cluster name */
    MQLONG ClusterRecOffset; /* Offset of next cluster record */
    MQLONG ClusterFlags; /* Cluster flags */
};
```

## **High Level Assembler**

```
MQWCR DSECT
MQWCR_CLUSTERNAME DS CL48 Cluster name
MQWCR_CLUSTERRECOFFSET DS F Offset of next cluster
* record
MQWCR_CLUSTERFLAGS DS F Cluster flags
MQWCR_LENGTH EQU *-MQWCR Length of structure
ORG MQWCR
MQWCR_AREA DS CL(MQWCR_LENGTH)
```

## **İlgili başvurular**

MQWCR -Küme iş yükü kümesi kayıt yapısındaki alanlar.

MQWCR -Küme iş yükü kümesi kayıt yapısındaki alanların açıklaması.

## **API çıkış başvurusu**

Bu kısım, bir programcı yazma API 'si çıkışlarına yönelik olarak daha çok ilgi çeken başvuru bilgileri sağlar.

## **Genel kullanım notları**

notlar:

1. Tüm çıkış işlevleri MQXEP çağrısını yayınlayabilir; bu çağrı, özel olarak API çıkış işlevlerinden kullanılmak üzere tasarlanmıştır.
2. MQ\_INIT\_EXIT işlevi, MQXEP dışında bir MQ çağrısı yayınlayamaz.
3. Yürürlükteki bağlantı için MQDISC çağrısını yayınlayamazsınız.
4. Bir çıkış işlevi MQCONN çağrısını ya da MQCNO\_HANDLE\_SHARE\_NONE seçeneğiyle MQCONNX çağrısını yayınlarsa, çağrı, MQRC\_ALREADY\_CONNECTED neden koduyla tamamlanır ve döndürülen tanıtıcı, bir parametre olarak çıkışa geçenlerle aynıdır.
5. Genel olarak, bir API çıkışı işlevinde bir MQI çağrısı olduğunda, API çıkışları özyinelemeli olarak çağrılmaz. Ancak, bir çıkış işlevi MQCNO\_HANDLE\_SHARE\_BLOCK ya da MQCNO\_HANDLE\_SHARE\_NO\_BLOCK seçenekleriyle MQCONNX çağrısını yayınlarsa, arama yeni bir paylaşılan tanıtıcı döndürür. Bu, çıkış takımını kendi bağlantı tanıtıcısı ile sağlar ve dolayısıyla uygulamanın iş biriminden bağımsız bir iş birimi sağlar. Çıkış takımı, kendi iş birimi içindeki iletileri koymak ve almak için bu tanıtıcıyı kullanabilir ve bu iş birimini kesinleştirebilir ya da yedekleyebilir; bu işlem, uygulamanın çalışma birimini herhangi bir şekilde etkilemeden yapılabilir.

Çıkış işlevi, uygulama tarafından kullanılan tanıtıcıdan farklı bir bağlantı tanıtıcısı kullandığından, çıkış işlevi tarafından verilen MQ çağrıları, çağrılmakta olan ilgili API çıkış işlevleriyle sonuçlanır. Bu nedenle, çıkış işlevleri yinelemeli olarak çağrılabilir. MQAXP ' de hem *ExitUserArea* alanında, hem de çıkış zinciri alanında bağlantı tanıtıcısı kapsamı olduğunu unutmayın. Sonuç olarak, bir çıkış işlevi bu alanları, etkin bir şekilde çağrılan başka bir yönetim ortamına sinyal vermek için kullanamıyor.

6. Ayrıca, çıkış işlevleri uygulamanın çalışma birimi içinde de ileti alabilir ve ileti alabilir. Uygulama iş birimini kesinleştirirken ya da yedeklediğinde, iş birimi içindeki tüm iletiler, çalışma birimine kimin yerleştirdiği (uygulama ya da çıkış işlevi) ne olursa olsun, birlikte kesinleştirilir ya da yedeklenir. Ancak, çıkış, uygulamanın tersi durumda (örneğin, bir iş birimindeki kesinleştirilmemiş ileti sayısı üst sınırını aşarak) sistem sınırlarını aşmasına neden olabilir.

Bir çıkış işlevi uygulamanın bu şekilde çalışma birimini kullandığında, çıkış işlevi genellikle MQCMIT çağrısının yayınlanmamasını önlemelidir; bu işlem, uygulamanın iş birimini kesinleştirir ve uygulamanın doğru işleyişini çözebilir. Ancak, çıkış işlevi bazen MQBACK çağrısını yayınlamaya gerek olabilir; çıkış işlevi, işlenmekte olan iş birimini engelleyen ciddi bir hatayla karşılaşırsa (örneğin, bir iletiyi uygulamanın iş biriminin bir parçası olarak koymakta). MQBACK çağrıldığında, iş sınırlarının uygulama biriminin değiştirilmediğinden emin olun. Bu durumda, uygulama, tamamlanma kodu MQCC\_UYARI ve neden kodu MQRC\_BACKED\_OUT ' un uygulamaya geri döndürülmesini sağlamak için, çıkış işlevinin uygun değerleri ayarlaması gerekir. Böylece, uygulama, iş biriminin geriletildiğinden emin olabilir.

If an exit function uses the application's connection handle to issue MQ calls, those calls do not themselves result in further invocations of API exit functions.

7. Bir MQXR\_BEFORE çıkış işlevi olağan dışı bir şekilde sona ererse, kuyruk yöneticisi hatadan kurtulabilir. Bu durumda, kuyruk yöneticisi, çıkış işlevinde MQXCC\_FAILED döndürdüğü halde işlemeyi sürdürür. Kuyruk yöneticisi kurtarılamazsa, uygulama sonlandırılır.
8. Bir MQXR\_AFTER çıkış işlevi olağan dışı bir şekilde sona ererse, kuyruk yöneticisi hatadan kurtulabilir. Bu durumda, kuyruk yöneticisi, çıkış işlevinde MQXCC\_FAILED döndürdüğü halde işlemeyi sürdürür. Kuyruk yöneticisi kurtarılamazsa, uygulama sonlandırılır. İkinci durumda, bir iş birimi dışında alınan iletiler kaybedilir (bu, kuyruktan ileti kaldırıldıktan hemen sonra başarısız olan uygulamayla aynı durumdur).
9. MCA işlemi, iki aşamalı kesinleştirme gerçekleştirir.

Bir API çıkışı, hazırlanmış bir MCA sürecinden MQCMIT ' i ele geçirirse ve iş birimi içinde bir işlem gerçekleştirmeye çalışırsa, bu işlem başarısız olur; neden kodu MQRC\_UOW\_NOT\_AVAS neden koduyla başarısız olur.

10. For a multi-installation environment, the only way to have an exit that works with both IBM WebSphere MQ 7.0 and IBM WebSphere MQ 7.1 is to write the exit in a way that links at IBM WebSphere MQ 7.0 with mqm.Lib and, for non-primary or relocated exits, to ensure that the application finds the correct mqm.Lib for the installation with which the queue manager is currently associated, prior to the application launch. (Örneğin, kuyruk yöneticisi bir IBM WebSphere MQ 7.0 kuruluşuna aitse bile, uygulamayı başlatmadan önce **setmqenv -m QM** komutunu çalıştırın.)

11. Where multiple installations of IBM MQ are available, use the exits written for an earlier version of IBM MQ, as new functionality added in the later version might not work with earlier versions. Yayın düzeyleri arasındaki değişimler hakkında daha fazla bilgi için bkz. [IBM MQ 8.0' da değişen özellikler.](#)

## IBM MQ API çıkış parametresi yapısı (MQAXP)

Bir dış denetim bloğu olan MQAXP yapısı, API çıkışına giriş ya da çıkış değiştirgesi olarak kullanılır. Bu konuda, kuyruk yöneticilerinin çıkış işlevlerinin nasıl işlediğini de içeren bilgiler de verilir.

MQAXP aşağıdaki C bildirimine sahiptir:

```
typedef struct tagMQAXP {
    MQCHAR4   StrucId;           /* Structure identifier */
    MQLONG    Version;          /* Structure version number */
    MQLONG    ExitId;           /* Exit Identifier */
    MQLONG    ExitReason;       /* Exit invocation reason */
    MQLONG    ExitResponse;     /* Response code from exit */
    MQLONG    ExitResponse2;    /* Secondary response code from exit */
    MQLONG    Feedback;        /* Feedback code from exit */
    MQLONG    APICallerType;     /* MQSeries API caller type */
    MQBYTE16  ExitUserArea;     /* User area for use by exit */
    MQCHAR32  ExitData;         /* Exit data area */
    MQCHAR48  ExitInfoName;     /* Exit information name */
    MQBYTE48  ExitPDArea;      /* Problem determination area */
    MQCHAR48  QMgrName;        /* Name of local queue manager */
    PMQACH    ExitChainAreaPtr; /* Inter exit communication area */
    MQHCONFIG Hconfig;         /* Configuration handle */
    MQLONG    Function;        /* Function Identifier */
    /* Ver:1 */
    MQHMSG    ExitMsgHandle     /* Exit message handle
    /* Ver:2 */
};
```

Bir API çıkışındaki işlevler çağrıldığında aşağıdaki parametre listesi iletilir:

### StrucId (MQCHAR4)-giriş

Aşağıdaki değer içeren çıkış değiştirgesi yapısı tanıtıcısı:

```
MQAXP_STRUC_ID.
```

Çıkış işleyici, bu alanı her çıkış işlevine girişte ayarlar.

### Sürüm (MQUZE)-giriş

Yapı sürüm numarası, şu değeri kullanarak:

#### MQAXP\_VERSION\_1

Sürüm 1 API çıkış parametresi yapısı.

#### MQAXP\_VERSION\_2

Sürüm 2 API çıkış parametresi yapısı.

#### MQAXP\_CURRENT\_VERSION

API çıkış parametresi yapısı için geçerli sürüm numarası.

Çıkış işleyici, bu alanı her çıkış işlevine girişte ayarlar.

### ExitId (MQHOT)-giriş

Çıkış tanıtıcısı, çıkış yordama girişini gösterir; çıkış tipini gösterir:

#### MQXT\_API\_EXIT

API çıkışı.

### ExitReason (MQHOT)-giriş

Çıkışı başlatmanın nedeni, her çıkış işlevine giriş için ayarlandır:

#### MQXR\_BAĞLANTI

Bir MQCONN ya da MQCONNX çağrısından önce kendini kullanıma hazırlamak ya da bir MQDISC çağrısından sonra sona erdirmek için çıkış çağrılır.

## **MQXR\_BEDEN**

Bir API çağrısı yürütülmeden önce ya da bir MQGET ' ye veri dönüştürmeden önce çıkış çağrılıyor.

## **MQXR\_AFTER**

Bir API çağrısı yürütüldükten sonra çıkış çağrılıyor.

## **ExitResponse (MQUZE)-çıkış**

Çıkıştan gelen yanıt, her çıkış işlevine girişte ilk kullanıma hazırlandı:

## **MQXCC\_OK**

Normal olarak devam edin.

Çıkış işlevi tarafından, kuyruk yöneticisiyle iletişim kurmak için, çıkış işlevinin yürütülmesi sonucu bu alanın ayarlanması gerekir. Değer aşağıdakilerden biri olmalıdır:

## **MQXCC\_OK**

Çıkış işlevi başarıyla tamamlandı. Normal olarak devam edin.

Bu değer, tüm MQXR\_ \* çıkış işlevleri tarafından ayarlanabilir. ExitResponse2 , daha sonra zincirin çıkış işlevlerinin çağrılıp çağrılmayacağına karar vermek için kullanılır.

## **MQXCC\_FAILED**

Bir hata nedeniyle çıkış işlevi başarısız oldu.

Bu değer, tüm MQXR\_ \* çıkış işlevleri tarafından ayarlanabilir. Kuyruk yöneticisi CompCode ' u MQCC\_FAILED olarak ayarlar ve şu nedenlere neden olur:

- İşlev MQ\_INIT\_EXIT ise MQRC\_API\_EXIT\_INIT\_ERROR
- İşlev MQ\_TERM\_EXIT işleviyse MQRC\_API\_EXIT\_TERM\_ERROR
- Diğer tüm çıkış işlevleri için MQRC\_API\_EXIT\_ERROR

Değer kümesi, daha sonra zincirin sonraki bir çıkış işleviyle değiştirilebilir.

ExitResponse2 yoksayıldı; kuyruk yöneticisi, MQXR2\_SUPPRESS\_CHAIN döndürüldüğü halde işlemeye devam ediyor.

## **MQXCC\_SUPPRESS\_FUNCTION**

IBM MQ API işlevini engelle.

Bu değer yalnızca bir MQXR\_BEFORE çıkış işleviyle ayarlanabilir. API çağrısını atlıyor.

Bu, MQ\_DATA\_CONV\_ON\_GET\_EXIT tarafından döndürülürse, veri dönüştürme atlanır. Kuyruk yöneticisi CompCode 'u MQCC\_FAILED ve MQRC\_SUPPRESSED\_BY\_EXIT' e neden olur, ancak değer kümesi, daha sonra zincirin çıkış işleviyle değiştirilebilir. Çıkışa ilişkin diğer parametreler de bunlardan sol çıkış olarak kalır. ExitResponse2 , daha sonra zincirin çıkış işlevlerinin çağrılıp çağrılmayacağına karar vermek için kullanılır.

Bu değer bir MQXR\_AFTER ya da MQXR\_CONNECTION çıkış işlevi tarafından ayarlandıysa, kuyruk yöneticisi MQXcc\_failed değerinin döndürülmesine karşın işlemeyi sürdürür.

## **MQXCC\_SKIP\_FUNCTION**

IBM MQ API işlevini atla.

Bu değer yalnızca bir MQXR\_BEFORE çıkış işleviyle ayarlanabilir. API çağrısını atlıyor. Bu, MQ\_DATA\_CONV\_ON\_GET\_EXIT tarafından döndürülürse, veri dönüştürme atlanır. Çıkış işlevi, CompCode değerini ve uygulamaya döndürülebilecek değerlerin nedenini belirlemelidir; ancak, ayarlanan değerler daha sonra zincirin çıkış işleviyle değiştirilebilir. Çıkışa ilişkin diğer parametreler de bunlardan sol çıkış olarak kalır. ExitResponse2 , daha sonra zincirin çıkış işlevlerinin çağrılıp çağrılmayacağına karar vermek için kullanılır.

Bu değer bir MQXR\_AFTER ya da MQXR\_CONNECTION çıkış işlevi tarafından ayarlandıysa, kuyruk yöneticisi MQXcc\_failed değerinin döndürülmesine karşın işlemeyi sürdürür.

## **MQXCC\_SUPPRESS\_EXIT**

Çıkış kümesine ait olan tüm çıkış işlevlerini engelle.

Bu değer yalnızca MQXR\_BEFORE ve MQXR\_AFTER çıkış işlevleriyle ayarlanabilir. It bypasses *Tümü* subsequent invocations of exit functions belonging to this set of exits for this logical

connection. Bu atlama işlemi, MQXR\_CONNECTION için bir ExitReason ile MQ\_TERM\_EXIT işlevi çağrıldığında, mantıksal bağlantı kesme isteği gerçekleşinceye kadar devam eder.

Çıkış işlevi, CompCode değerini ve uygulamaya döndürülebilecek değerlerin nedenini belirlemelidir; ancak, ayarlanan değerler daha sonra zincirin çıkış işleviyle değiştirilebilir. Çıkışa ilişkin diğer parametreler de bunlardan sol çıkış olarak kalır. ExitResponse2 yoksayıldı.

Bu değer bir MQXR\_CONNECTION çıkış işlevi tarafından ayarlandıysa, kuyruk yöneticisi MQXcc\_failed değerinin döndürülmesine karşın işlemeyi sürdürür.

ExitResponse ve ExitResponse2 arasındaki etkileşme ve çıkış işlemindeki etkiyle ilgili bilgi için bkz. [“Kuyruk yöneticilerinin çıkış işlevlerinin nasıl işlendiğini” sayfa 1541.](#)

### **ExitResponse2 (MQHOT)-çıkış**

Bu, MQXR\_BEFORE çıkış işlevlerine ilişkin birincil çıkış yanıt kodunu niteleyen ikincil çıkış yanıt kodudur. Şu şekilde kullanıma hazırlandı:

```
MQXR2_DEFAULT_CONTINUATION
```

IBM MQ API çağrısı çıkış işlevine ilişkin giriş. Daha sonra, aşağıdaki değerlerden birine ayarlanabilir:

#### **MQXR2\_DEFAULT\_CONTINUATION**

Whether to continue with the next exit in the chain, depending on the value of ExitResponse.

ExitResponse MQXCC\_SUPPRESS\_FUNCTION ya da MQXCC\_SKIP\_FUNCTION ise, MQXR\_BEFORE zincirinde ve MQXR\_AFTER zincirindeki eşleşen çıkış işlevlerinde daha sonra çıkış işlevleri atlanır. MQXR\_BEFORE zincirinde daha önce çıkış işlevleriyle eşleşen MQXR\_AFTER zincirindeki çıkış işlevlerini çağırın.

Tersi durumda, zincirin sonraki çıkışını çağırın.

#### **MQXR2\_SUPPRESS\_CHAIN**

Zinciri bastır.

MQXR\_BEFORE zincirinde daha sonra çıkış işlevlerini ve bu API çağrısı çağrısına ilişkin MQXR\_AFTER zincirindeki eşleşen çıkış işlevlerini atlayın. MQXR\_BEFORE zincirinde daha önce çıkış işlevleriyle eşleşen MQXR\_AFTER zincirindeki çıkış işlevlerini çağırın.

#### **MQXR2\_CONTINUE\_CHAIN**

Zincirdeki bir sonraki çıkışa devam edin.

ExitResponse ve ExitResponse2 arasındaki etkileşme ve çıkış işlemindeki etkiyle ilgili bilgi için bkz. [“Kuyruk yöneticilerinin çıkış işlevlerinin nasıl işlendiğini” sayfa 1541.](#)

### **Feedback (MQUZE)-giriş/çıkış**

Çıkış işlevi çağrılarında geribildirim kodlarını iletin. Bu, şu şekilde kullanıma hazırlandı:

```
MQFB_NONE (0)
```

bir zincirdeki ilk çıkışa ait ilk işlevi çağırılmadan önce.

Çıkışlar, geçerli MQFB\_\* ya da MQRC\_\* değeri de içinde olmak üzere, bu alanı herhangi bir değere ayarlayabilirler. Bu alanı, bu alanı MQFB\_APPL\_FIRST ' nin MQFB\_APPL\_SON aralığında kullanıcı tanımlı bir geribildirim değerine de ayarlayabilir.

### **APICallerType (MQUZE)-giriş**

IBM MQ API çağırınının kuyruk yöneticisi için dış mu, yoksa iç mi olduğunu gösteren API çağırın tipi: MQXACT\_EXTERNAL ya da MQXACT\_INTERAL.

### **ExitUser(ExitUser) Alanı (MQBYTE16)-giriş/çıkış**

Belirli bir ExitInfonesesiyle ilişkili tüm çıkışlar için kullanılabilir olan bir kullanıcı alanı. Hconn için ilk çıkış işlevini (MQ\_INIT\_EXIT) çağırılmadan önce, MQXUA\_NONE ( ExitUserArea uzunluğu için ikili sıfırlar) olarak kullanıma hazırlandı. Bundan sonra, bu alanda bir çıkış işlevi tarafından yapılan tüm değişiklikler, aynı çıkışa ilişkin işlevlere ilişkin çağrılar boyunca korunur.

Bu alan, birden çok 4 MQLONG ' ye hizalanmış olabilir.

Çıkışlar, bu alandan ayırdıkları herhangi bir depolamayı da tutturabilir.

Her bir hconn için, çıkışlar zincirindeki her çıkışa farklı bir ExitUserAlanı vardır. The ExitUserArea cannot be shared by exits in a chain, and the contents of the ExitUserArea for one exit are not available to another exit in a chain.

C programları için, sabit MQXUA\_NONE\_ARRAY, MQXUA\_NONE ile aynı değerle, ancak bir dizgi yerine bir karakter dizisi olarak tanımlanır.

Bu alanın uzunluğu MQ\_EXIT\_USER\_AREA\_LENGTH tarafından verilir.

#### **ExitData (MQCHAR32)-giriş**

Çıkış verileri, çıkışta sağlanan çıkışa özgü verilerin 32 karakterine her çıkış işlevine ilişkin giriş olarak ayarlanır. Çıkışta hiçbir değer tanımlamadıysanız, bu alanın tamamı boşluk olur.

Bu alanın uzunluğu MQ\_EXIT\_DATA\_LENGTH tarafından verilir.

#### **ExitInfoAd (MQCHAR48)-giriş**

Çıkış bilgileri adı, her bir çıkış işlevine ilişkin girişin, stanzas içindeki çıkış tanımlarında belirtilen ApiExit\_name adlı işleve ilişkin olarak ayarlanmasını sağlar.

#### **ExitPDArea (MQBYTE48)-giriş/çıkış**

Bir çıkış işlevinin her çağrısına ilişkin olarak, MQXPDA\_NONE (alan uzunluğu için ikili sıfır) ile ilgili bir sorun saptama alanı kullanıma hazırlandı.

C programları için, MQXPDA\_NONE\_ARRAY değişmezi, MQXPDA\_NONE ile aynı değerle tanımlanır, ancak dizgi yerine bir karakter dizisi olarak tanımlanır.

Çıkış işleyicisi bu alanı her zaman, işlev başarılı olduğunda bile, bir çıkışta sonunda IBM MQ izlemesine yazar.

Bu alanın uzunluğu MQ\_EXIT\_PD\_AREA\_LENGTH tarafından verilir.

#### **QMgrName (MQCHAR48)-giriş**

Uygulamanın bağlandığı kuyruk yöneticisinin adı, IBM MQ API çağrısının işlenmesi sonucunda bir çıkış çağırıldı.

MQCONN ya da MQCONNX çağrılarında sağlanan bir kuyruk yöneticisinin adı boş bırakılırsa, bu alan uygulamanın bağlı olduğu kuyruk yöneticisi adına, uygulamanın sunucu ya da istemci olup olmadığını belirlemeye devam eder.

Çıkış işleyicisi, bu alanı her çıkış işlevine girişte ayarlar.

Bu alanın uzunluğu MQ\_Q\_MGR\_NAME\_LENGTH değeri tarafından verilir.

#### **ExitChainAreaPtr (PMQACH)-giriş/çıkış**

Bu, bir zincirdeki farklı çıkışlara ilişkin çağrılar arasında veri iletmek için kullanılır. Bir çıkış zincirindeki ilk çıkışa ilişkin ilk işlevi (MQ\_INIT\_EXIT ile ExitReason MQXR\_CONNECTION ile) çağırılmadan önce BOŞ (NULL) işaretçisi olarak ayarlanır. Bir çağrıda bulunan çıkışta döndürülen değer, bir sonraki çağrıya geçirilir.

Çıkış zinciri alanının nasıl kullanılacağı hakkında daha ayrıntılı bilgi için [“Çıkış zinciri alanı ve çıkış zinciri alanı üstbilgisi \(MQACH\)” sayfa 1544](#) dosyasına bakın.

#### **Hconfig (MQHCONFIG)-giriş**

Başlatılmakta olan işlevlerin kümesini temsil eden yapılandırma tanıtıcısı. Bu değer, MQ\_INIT\_EXIT işlevinde kuyruk yöneticisi tarafından oluşturulur ve daha sonra API çıkış işlevine geçirilir. Bu, her çıkış işlevine giriş için ayarlanır.

MQI ve DCI çağrılarını yapmak için MQIEP yapısının bir göstergesi olarak Hconfig 'i kullanabilirsiniz. HConfig 'in, HConfig değıştirgesini MQIEP yapısına gösterge olarak kullanmadan önce, MQIEP yapısının StrucId olan ilk 4 baytından (HConfig) eşleşmesini denetlemeniz gerekir.

#### **İşlev (MQUZE)-giriş**

İşlev tanıtıcısı, [“Dış sabitler” sayfa 1546](#) içinde açıklanan MQXF\_ \* değışmezleri olan geçerli değerler.

Çıkış işleyicisi bu alanı, çağrılmakta olan çıkışa neden olan IBM MQ API çağrısına bağlı olarak, her bir çıkış işlevine giriş sırasında doğru değere ayarlar.



## ExitMsgHandle (MQHMSG)-giriş/çıkış

İşlev MQXF\_GET ve ExitReason ise, MQXR\_AFTER olduğunda, bu alanda geçerli bir ileti tanıtıcısı döndürülür. Bu alanda, API çıkışı kaydedilirken, MQXEPO yapısında belirtilen ExitProperties dizisiyle eşleşen diğer tüm özellikler, ileti tanımlayıcı alanlarına ve diğer tüm özelliklere ilişkin tüm özellikler için döndürülür.

ExitMsgHandle içinde döndürülen ileti dışı tanımlayıcı özellikleri, belirtilmişse, MQGMO yapısındaki MsgHandle ' den ya da ileti verilerinde kullanılamaz.

İşlev MQXF\_GET ve ExitReason ise, MQXR\_BEFORE ise, çıkış programı bu alanı MQHM\_NONE olarak ayarlarsa, ExitMsgtanıtıcı özelliklerinin doldurulması engellenir.

Sürüm daha düşükse bu alan ayarlanmazsa MQAXP\_VERSION\_2' dir.

## Kuyruk yöneticilerinin çıkış işlevlerinin nasıl işlendiğini

Bir çıkış işlevinin döndürmesiyle ilgili kuyruk yöneticisi tarafından gerçekleştirilen işleme hem ExitResponse , hem de ExitResponse2göre değişir.

Çizelge 836 sayfa 1541 , olası birleşimleri ve MQXR\_BEFORE çıkış işleviyle ilgili etkileri özetlemektedir:

- API çağrısının CompCode ve Neden parametrelerini kim belirler
- MQXR\_BEFORE zincirindeki geri kalan çıkış işlevlerinin ve MQXR\_AFTER zincirindeki eşleşen çıkış işlevlerinin çağrılıp çağrılmayacağı
- API çağrısının çağrılıp çağrılmayacağı

MQXR\_AFTER çıkış işlevi için:

- CompCode ve Reason, MQXR\_BEFORE ile aynı şekilde ayarlanır.
- ExitResponse2 yoksayıldı (MQXR\_AFTER zincirindeki kalan çıkış işlevleri her zaman çağrılır)
- MQXCC\_SUPPRESS\_FUNCTION ve MQXCC\_SKIP\_FUNCTION geçerli değil

Bir MQXR\_CONNECTION çıkış işlevi için:

- CompCode ve Reason, MQXR\_BEFORE ile aynı şekilde ayarlanır.
- ExitResponse2 yoksayıldı
- MQXCC\_SUPPRESS\_FUNCTION, MQXCC\_SKIP\_FUNCTION, MQXCC\_SUPPRESS\_EXIT geçerli değil

Bir çıkışa ya da kuyruk yöneticisinin CompCode ve Reason olarak ayarlandığı tüm durumlarda, değer kümesinin daha sonra çağrılan bir çıkış ya da API çağrısı tarafından (API çağrısı daha sonra çağrılırsa) değiştirilebilir.

Çizelge 836. MQXR_BEFORE çıkış işlemi			
ExitResponsedeğeri	CompCode (Derleyici Kodu) ve Neden	ExitResponse2 değeri (varsayılan devamı) Zinciri	ExitResponse2 değeri (varsayılan devam) API
MQXCC_OK	exit	Y	Y
MQXCC_SUPPRESS_EXIT	exit	Y	Y
MQXCC_SUPPRESS_FUNCTION	Kuyruk Yöneticisi	N	N
MQXCC_SKIP İŞLEVI	exit	N	N
MQXCC_FAILED	Kuyruk Yöneticisi	N	N

## İstemcilerin çıkış işlevlerini nasıl işlediğini

Genel olarak, istemciler, çıkış işlevlerini sunucu uygulamalarının yaptığı şekilde işler ve bu yapıdaki QMgrName özniteliği, işlevin bir sunucuda mı, yoksa bir istemcide mi geçerli olduğunu uygular.

Ancak, istemcide *mqs.ini* dosyası kavramı yoktur; bu nedenle *ApiExitCommon* ve *APIExitTemplate* kısmı geçerli değildir. Yalnızca *ApiExitLocal* standı uygulanır ve bu stanza *mqclient.ini* dosyasında yapılandırılır.

## IBM MQ API çıkışı bağlam yapısı (MQAXC)

Bir dış denetim öbeği olan MQAXC yapısı, API çıkışa giriş parametresi olarak kullanılır.

MQAXC aşağıdaki C bildirimine sahiptir:

```
typedef struct tagMQAXC {
    MQCHAR4   StrucId;           /* Structure identifier */
    MQLONG    Version;          /* Structure version number */
    MQLONG    Environment;      /* Environment */
    MQCHAR12  UserId;           /* UserId associated with appl */
    MQBYTE40  SecurityId        /* Extension to UserId running appl */
    MQCHAR264 ConnectionName;   /* Connection name */
    MQLONG    LongMCAUserIdLength; /* long MCA user identifier length */
    MQLONG    LongRemoteUserIdLength; /* long remote user identifier length */
    MQPTR     LongMCAUserIdPtr;  /* long MCA user identifier address */
    MQPTR     LongRemoteUserIdPtr; /* long remote user identifier address */
    MQCHAR28  ApplName;         /* Application name */
    MQLONG    ApplType;         /* Application type */
    MQPID     ProcessId;        /* Process identifier */
    MQTID     ThreadId;         /* Thread identifier */

    /* Ver:1 */
    MQCHAR    ChannelName[20]   /* Channel Name */
    MQBYTE4   Reserved1;       /* Reserved */
    PMQCD     pChannelDefinition; /* Channel Definition pointer */
};
```

MQAXC ' ye ilişkin deęiřtirgeler řunlardır:

### StrucId (MQCHAR4)-giriř

MQAXC\_STRUC\_ID deęerine sahip çıkıř baęlamı yapısı tanıtıcısı. C programları için, MQAXC\_STRUC\_ID\_ARRAY deęiřmezi, MQAXC\_STRUCT\_ID deęeriyle aynı deęeri, ancak dizgi yerine bir karakter dizisi olarak tanımlanır.

Çıkıř iřleyici, bu alanı her çıkıř iřlevine giriřte ayarlar.

### Sürüm (MQUZE)-giriř

Yapı sürüm numarası, řu deęeri kullanarak:

#### MQAXC\_VERSION\_2

Çıkıř baęlamı yapısına iliřkin sürüm numarası.

#### MQAXC\_CURRENT\_VERSION

Çıkıř baęlamı yapısına iliřkin yürürlükteki sürüm numarası.

Çıkıř iřleyici, bu alanı her çıkıř iřlevine giriřte ayarlar.

### Ortam (MQUZE)-giriř

Bir çıkıř iřleviyle sonuçlanan bir IBM MQ API çağrısının yayınlandıęı ortam. Bu alan için geçerli deęerler řunlardır:

#### MQXE\_DIęER

Bu deęer, bir sunucu uygulamasından çıkıřa çağrılıp çağrılmıyorsa, bir API çıkıęı çağrılarla tutarlı olur. Yani, bir istemcide bir API çıkıřı deęiřmeden çalışır ve farklı bir řey görmeyen bir çıkıř olur.

Çıkıřta istemcinin çalışıp çalışmadıęını belirlemesi gerekiyorsa, çıkıřta *ChannelName* ve *ChannelDefinition* alanlarına bakılarak çıkılabilir.

#### MQXE\_MCA

İleti kanalı aracısı

#### MQXE\_MCA\_SVRCONN

İstemci adına hareket eden bir ileti kanalı aracısı

#### MQXE\_COMMAND\_SERVER

Komut sunucusu

## **MQXE\_MQSC**

runmqsc komutu yorumlayıcısı

Çıkış işleyici, bu alanı her çıkış işlevine girişte ayarlar.

### **UserId (MQCHAR12)-giriş**

Uygulamayla ilişkili kullanıcı kimliği. Özellikle, istemci bağlantılarında, bu alan, kanal kodunun altında çalıştığı kullanıcı kimliğinin yerine, kabul edilen kullanıcının kullanıcı kimliğini içerir. İstemciden boş bir kullanıcı kimliği akışı varsa, önceden kullanılmakta olan kullanıcı kimliği üzerinde değişiklik yapılmamaktadır. Yani, yeni kullanıcı kimliği kabul edilmez.

Çıkış işleyici, bu alanı her çıkış işlevine girişte ayarlar. Bu alanın uzunluğu MQ\_USER\_ID\_LENGTH tarafından verilir.

Bir istemci durumunda, istemciden sunucuya gönderilen kullanıcı kimliği bu tanıtıcıdır. Bu, kullanıcı kimliğini değiştiren bir MCAUser ya da CHLAUTH yapılandırması olabileceği için, istemcinin kuyruk yöneticisinde karşı çalıştırıldığı etkin kullanıcı kimliği olmayabilir.

### **SecurityId (MQBYTE40)-giriş**

Uygulamayı çalıştıran kullanıcı kimliğine bir uzantıdır. Uzunluğu MQ\_SECURITY\_ID\_LENGTH tarafından verilir.

Bir istemci durumunda, istemciden sunucuya gönderilen kullanıcı kimliği bu tanıtıcıdır. Bu, kullanıcı kimliğini değiştiren bir MCAUser ya da CHLAUTH yapılandırması olabileceği için, istemcinin kuyruk yöneticisinde karşı çalıştırıldığı etkin kullanıcı kimliği olmayabilir.

### **ConnectionName (MQCHAR264)-giriş**

Bağlantı adı alanı, istemcinin adresine ayarlanır. Örneğin, TCP/IP için istemci IP adresi olur.

Bu alanın uzunluğu MQ\_CONN\_NAME\_LENGTH değeri tarafından verilir.

Bir istemci durumunda, bu, kuyruk yöneticisinin iş ortağı adresidir.

### **LongMCAUserIdLength (MQUZE)-giriş**

Uzun MCA kullanıcı kimliğinin uzunluğu.

MCA kuyruk yöneticisine bağlandığında, bu alan uzun MCA kullanıcı kimliğinin uzunluğuna ayarlanır (ya da bu tip bir tanıtıcı yoksa sıfır).

Bir istemci durumunda bu, istemci uzun kullanıcı kimliğidir.

### **LongRemoteUserIdUzunluğu (MQUZE)-giriş**

Uzun uzak kullanıcı kimliğinin uzunluğu.

MCA kuyruk yöneticisine bağlandığında, bu alan uzun uzak kullanıcı kimliğinin uzunluğuna ayarlanır. Ters durumda, bu alan sıfır olarak ayarlanır.

Bir istemci durumunda, bu alanı sıfır olarak ayarlayın.

### **LongMCAUserIdPtr (MQPTR)-giriş**

Uzun MCA kullanıcı kimliğinin adresi.

MCA kuyruk yöneticisine bağlandığında, bu alan uzun MCA kullanıcı tanıtıcısının adresine (ya da böyle bir tanıtıcı yoksa, boş değerli bir işaretçiye) ayarlıdır.

Bir istemci durumunda bu, istemci uzun kullanıcı kimliğidir.

### **LongRemoteUserIdPtr (MQPTR)-giriş**

Uzun uzak kullanıcı kimliğinin adresi.

MCA kuyruk yöneticisine bağlandığında bu alan, uzun uzak kullanıcı kimliğinin adresi olarak ayarlanır (ya da böyle bir tanıtıcı yoksa, boş değerli bir işaretçiye).

Bir istemci durumunda, bu alanı sıfır olarak ayarlayın.

### **ApplName (MQCHAR28)-giriş**

IBM MQ API çağırısını yayınlayan uygulamanın ya da bileşenin adı.

ApplName ' nın oluşturulmasına ilişkin kurallar, bir MQPUT için varsayılan adın oluşturulmasına ilişkin kurallardır.

Bu alanın değeri, program adı için işletim sistemi sorgulanarak bulunur. Uzunluğu, MQ\_APPL\_NAME\_LENGTH tarafından verilir.

### **ApplType (MQHOT)-giriş**

IBM MQ API çağırısını yayınlayan uygulama ya da bileşen tipi.

Bu değer, uygulamanın derlendiği altyapı için MQAT\_DEFAULT ' dur ya da tanımlı MQAT\_ \* değerlerinden birine denk gelir.

Çıkış işleyici, bu alanı her çıkış işlevine girişte ayarlar.

### **ProcessId (MQPID)-giriş**

İşletim sistemi işlem tanıtıcısı.

Uygun olduğunda, çıkış işleyicisi bu alanı her bir çıkış işlevine girişte ayarlar.

### **ThreadId (MQTID)-giriş**

MQ iş parçacığı tanıtıcısı. Bu tanıtıcı, MQ izleme ve FFST dökümlerinde kullanılan tanıtıcıyla aynıdır, ancak işletim sistemi iş parçacığı tanıtıcısından farklı olabilir.

Uygun olduğunda, çıkış işleyicisi bu alanı her bir çıkış işlevine girişte ayarlar.

### **ChannelName (MQCHAR)-giriş**

Varsa ve biliniyorsa, kanalın adı boşluklarla dolduruldu.

Geçerli değilse, bu alan NULL (boş değerli) karakterler olarak ayarlanır.

### **Reserved1 (MQBYTE4)-giriş**

Bu alan ayrılmıştır.

### **ChanneDefinition (PMQCD)-giriş**

Uygulanabiliyorsa ve biliniyorsa, kullanılan kanal tanımlamasına ilişkin gösterge.

Geçerli değilse, bu alan NULL (boş değerli) karakterler olarak ayarlanır.

İşaretçinin yalnızca, bağlantı bir IBM MQ kanalı adına işlem gerçekleştiriyorsa ve kanal tanımlaması okunduysa tamamlanır.

Kanal için ilk MQCONN çağırısı yapıldığında, sunucu üzerinde özellikle kanal tanımı verilmez. Ayrıca, işaretçi doldurulsa, işaretçinin gösterdiği yapı (ve herhangi bir alt yapı) yalnızca okunur olarak değerlendirilmelidir; yapının herhangi bir güncellemesi önceden kestirilemeyen sonuçlara yol açabilir ve desteklenmez.

Bir istemci durumunda, istemci için belirlenmiş bir değere sahip olanlar dışındaki alanlar, bir istemci uygulaması için uygun olan değerleri içerir.

## **Çıkış zinciri alanı ve çıkış zinciri alanı üstbilgisi (MQACH)**

Gerekirse, bir çıkış zinciri bir çıkış zinciri alanı için depolama alanı edinebilir ve bu depolama alanını göstermek için MQAXP 'de ExitChainAreaPtr ' yi ayarlayabilir.

Çıkışlar (aynı ya da farklı çıkış işlevlerinden biri) birden çok çıkış zinciri alanı edinebilir ve bunları birbirine bağlayabilir. Çıkış zinciri alanları, çıkış işleyicisinden çağrılırken yalnızca bu listeye eklenmeli ya da bu listeden kaldırılmalıdır. Bu, farklı iş parçacıklarının aynı anda listeye alan ekleme ya da listeden alan kaldırma işlemi nedeniyle diziselleştirme (serileştirme) sorunu olmamasını sağlar.

Bir çıkış zinciri alanı, bir MQACH üstbilgi yapısıyla başlamalı, bunun için C bildirimi aşağıdaki gibi olur:

```
typedef struct tagMQACH {
    MQCHAR4   StructId;           /* Structure identifier */
    MQLONG    Version;           /* Structure version number */
    MQLONG    StructLength;      /* Length of the MQACH structure */
    MQLONG    ChainAreaLength;   /* Exit chain area length */
    MQCHAR48  ExitInfoName      /* Exit information name */
}
```

```

    PMQACH    NextChainAreaPtr;    /* Pointer to next exit chain area */
};

```

Çıkış zinciri alanı üstbilgisindeki alanlar şunlardır:

### StrucId (MQCHAR4)-giriş

Çıkış zinciri alanı yapısı tanıtıcısı, MQACH\_STRUCT\_DEFAULT MQACH\_XX\_ENCODE\_CASE\_ONE default ile tanımlanan başlangıç değeri ile tanımlanır.

C programları için, constant MQACH\_STRUC\_ID\_ARRAY de tanımlanır; bu değer, MQACH\_STRUCT\_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisi olarak tanımlanır.

### Sürüm (MQUZE)-giriş

Yapı sürüm numarası, aşağıdaki gibi:

#### MQACH\_VERSION\_1

Çıkış parametresi yapısına ilişkin sürüm numarası.

#### MQACH\_CURRENT\_VERSION

Çıkış bağlamı yapısına ilişkin yürürlükteki sürüm numarası.

MQACH\_XX\_ENCODE\_CASE\_ONE default ile tanımlanan bu alanın başlangıç değeri MQACH\_CURRENT\_VERSION.

**Not:** Bu yapının yeni bir sürümünü tanıtırırsanız, var olan parçanın yerleşim düzeni değişmez. Çıkış işlevleri, sürüm numarasının, çıkış işlevinin kullanması gereken alanları içeren en düşük sürüme eşit ya da bu sürümden daha büyük olduğunu denetlemelidir.

### StrucLength (MQBOLE)-giriş

MQACH yapısının uzunluğu. Çıkışlar, çıkış verilerinin başlangıcını belirlemek için bu alanı kullanabilir ve çıkışı, çıkış tarafından yaratılan yapının uzunluğuna göre ayarlayabilir.

MQACH\_DEFAULT ile tanımlanan bu alanın başlangıç değeri MQACH\_CURRENT\_LENGTH olur.

### ChainAreaUzunluğu (MQUZE)-giriş

Çıkış zinciri alanı uzunluğu, MQACH üstbilgisi de içinde olmak üzere, yürürlükteki çıkış zinciri alanının genel uzunluğuna ayarlanır.

MQACH\_XX\_ENCODE\_CASE\_ONE default ile tanımlanan bu alanın ilk değeri sıfır.

### ExitInfoAd (MQCHAR48)-giriş

Çıkış bilgisi adı.

Bir çıkış MQACH yapısı yarattığında, bu alanı kendi ExitInfoadıyla kullanıma hazırlamalıdır; böylece, daha sonra bu MQACH yapısı bu çıkışa ilişkin başka bir yönetim ortamı tarafından ya da bir işbirliği çıkışa göre bulunabilir.

MQACH\_DEFAULT ile tanımlanan bu alanın başlangıç değeri sıfır uzunluklu bir dizgidir ({}).

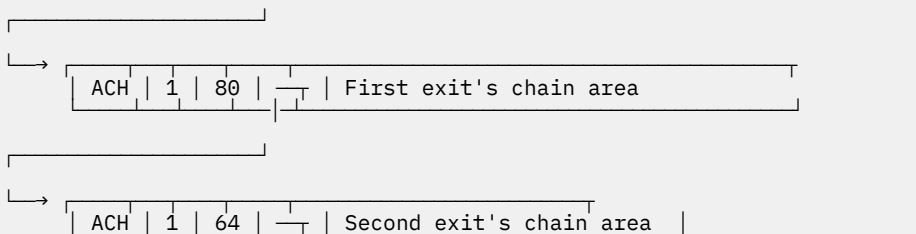
### NextChainAreaPtr (PMQACH)-giriş

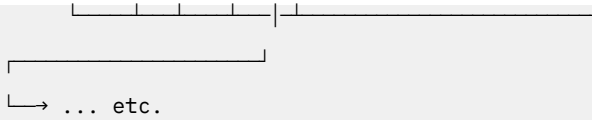
Boş değerli gösterge (NULL) olan MQACH\_XX\_ENCODE\_CASE\_ONE default tarafından tanımlanan bir başlangıç değeri olan bir sonraki çıkış zinciri alanına bir işaretçi.

Çıkış işlevleri, edindikleri çıkış zinciri alanları için depolama alanını serbest bırakmalı ve zincir işaretçileri, çıkış zinciri alanlarını listeden çıkarmak için yönlendirmelidir.

Çıkış zinciri alanı aşağıdaki gibi oluşturulabilir:

MQAXP.ExitChainAreaPtr —





## Dış sabitler

API için kullanılabilir dış sabitler için başvuru bilgileri olarak bu konuyu kullanın.

API çıkışları için aşağıdaki dış sabitler kullanılabilir:

### MQXF\_ \* (çıkış işlevi tanıtıcıları)

MQXF_INIT	1	X'00000001'
MQXF_TERM	2	X'00000002'
MQXF_CONN	3	X'00000003'
MQXF_CONNX	4	X'00000004'
MQXF_DISC	5	X'00000005'
MQXF_OPEN	6	X'00000006'
MQXF_CLOSE	7	X'00000007'
MQXF_PUT1	8	X'00000008'
MQXF_PUT	9	X'00000009'
MQXF_GET	10	X'0000000A'
MQXF_DATA_CONV_ON_GET	11	X'0000000B'
MQXF_INQ	12	X'0000000C'
MQXF_SET	13	X'0000000D'
MQXF_BEGIN	14	X'0000000E'
MQXF_CMIT	15	X'0000000F'
MQXF_BACK	16	X'00000010'
MQXF_STAT	18	X'00000012'
MQXF_CB	19	X'00000013'
MQXF_CTL	20	X'00000014'
MQXF_CALLBACK	21	X'00000015'
MQXF_SUB	22	X'00000016'
MQXF_SUBRQ	23	X'00000017'
MQXF_XACLOSE	24	X'00000018'
MQXF_XACOMMIT	25	X'00000019'
MQXF_XACOMplete	26	X'0000001A'
MQXF_XAEND	27	X'0000001B'
MQXF_XAFORGET	28	X'0000001C'
MQXF_XAOPEN	29	X'0000001D'
MQXF_XAPREPARE	30	X'0000001E'
MQXF_XARECOVER	31	X'0000001F'
MQXF_XAROLLBACK	32	X'00000020'
MQXF_XASTART	33	X'00000021'
MQXF_AXREG	34	X'00000022'
MQXF_AXUNREG	35	X'00000023'

### MQXR\_ \* (çıkış nedenleri)

MQXR_BEFORE	1	X'00000001'
MQXR_AFTER	2	X'00000002'
MQXR_CONNECTION	3	X'00000003'

### MQXE\_ \* (ortamlar)

MQXE_OTHER	0	X'00000000'
MQXE_MCA	1	X'00000001'
MQXE_MCA_SVRCONN	2	X'00000002'
MQXE_COMMAND_SERVER	3	X'00000003'
MQXE_MQSC	4	X'00000004'

### MQ\* \_ \* (ek sabitler)

MQAXP_VERSION_1	1
MQAXP_VERSION_2	2
MQAXC_VERSION_1	1
MQACH_VERSION_1	1
MQAXP_CURRENT_VERSION	1
MQAXC_CURRENT_VERSION	1

MQACH_CURRENT_VERSION	1
MQXACT_EXTERNAL	1
MQXACT_INTERNAL	2
MQXT_API_EXIT	2
MQACH_LENGTH_1	68 (32-bit platforms) 72 (64-bit platforms) 80 (128-bit platforms)
MQACH_CURRENT_LENGTH	68 (32-bit platforms) 72 (64-bit platforms) 80 (128-bit platforms)

### MQ\*\_\* (boş değer değişmezleri)

MQXPDA_NONE	X'00...00' (48 nulls)
MQXPDA_NONE_ARRAY	'\0', '\0', ..., '\0', '\0'

### MQXCC\_\* (tamamlanma kodları)

MQXCC_FAILED	-8
--------------	----

### MQRC\_\* (neden kodları)

#### MQRC\_API\_EXIT\_ERROR 2374 X'00000946'

Bir çıkış işlevi çağrısı geçersiz bir yanıt kodu döndürdü ya da bir şekilde başarısız oldu ve kuyruk yöneticisi yapılacak sonraki işlemi saptayamıyor.

Hatalı yanıt kodunu belirlemek için MQAXP 'nin ExitResponse ve ExitResponse2 alanlarını inceleyin ve çıkışı geçerli bir yanıt kodu döndürecek şekilde değiştirin.

#### MQRC\_API\_EXIT\_INIT\_ERROR 2375 X'00000947'

Kuyruk yöneticisi bir API çıkış işlevi için yürütme ortamını kullanıma hazırlarken bir hata saptadı.

#### MQRC\_API\_EXIT\_TERM\_ERROR 2376 X'00000948'

Kuyruk yöneticisi bir API çıkış işlemine ilişkin yürütme ortamını kapatırken bir hata saptadı.

#### MQRC\_EXIT\_REASON\_ERROR 2377 X'00000949'

Bir çıkış giriş noktası kayıt çağrısı (MQXEP) çağrısında sağlanan ExitReason alanının değeri hatalı.

Hatalı çıkış nedeni değerini saptamak ve düzeltmek için ExitReason alanının değerini inceleyin.

#### MQRC\_RESERVED\_VALUE\_ERROR 2378 X'0000094A'

Ayrılmış alanın değeri hatalı.

Ayrılmış değeri belirlemek ve düzeltmek için Ayrılmış alanın değerini inceleyin.

## C dili typedefs

Bu konuda, C dilinde kullanılabilen API çıkışlarıyla ilişkili typedefler hakkında bilgi sağlanır.

Aşağıda, API çıkışlarıyla ilişkili C dili yazı tipi tipleri yer alıyor:

```
typedef PMQLONG MQPOINTER PPMQLONG;
typedef PMQBYTE MQPOINTER PPMQBYTE;
typedef PMQHOBX MQPOINTER PPMQHOBX;
typedef PMQOD MQPOINTER PPMQOD;
typedef PMQMD MQPOINTER PPMQMD;
typedef PMQPMO MQPOINTER PPMQPMO;
typedef PMQGMO MQPOINTER PPMQGMO;
typedef PMQCNO MQPOINTER PPMQCNO;
typedef PMQBO MQPOINTER PPMQBO;

typedef MQAXP MQPOINTER PMQAXP;
typedef MQACH MQPOINTER PMQACH;
typedef MQAXC MQPOINTER PMQAXC;

typedef MQCHAR MQCHAR16[16];
typedef MQCHAR16 MQPOINTER PMQCHAR16;
```

```
typedef MQLONG MQPID;  
typedef MQLONG MQTID;
```

## Çıkış giriş noktası kayıt çağrısı (MQXEP)

MQXEP, MQXEP C dili çağrısı ve MQXEP C işlev prototipi hakkında bilgi edinmek için bu bilgileri kullanın.

MQXEP çağrısını şu şekilde kullanın:

1. Register the before and after IBM MQ API exit invocation points at which to invoke exit functions
2. Çıkış işlevi giriş noktalarını belirle
3. Çıkış işlevi giriş noktalarının kaydını sil

MQXEP çağrılarını genellikle MQ\_INIT\_EXIT çıkış işlevinde kodlayabilirsiniz, ancak sonraki çıkış işlevlerinde bunları belirtebilirsiniz.

Önceden kayıtlı bir çıkış işlevini kaydettirmek için bir MQXEP çağrısı kullanırsanız, kayıtlı çıkış işlevini yerine ikinci MQXEP çağrısı başarıyla tamamlanır.

Bir NULL çıkış işlevini kaydetmek için bir MQXEP çağrısı kullanırsanız, MQXEP çağrısı başarıyla tamamlanır ve çıkış işlevi kayıt dışı olur.

MQXEP çağrıları, bir bağlantı isteğinin ömrü sırasında belirli bir çıkış işlevini kaydetmek, kayıttan kaldırmak ve yeniden kaydettirmek için kullanılırsa, önceden kayıtlı çıkış işlevi yeniden etkinleştirilir. Hala ayrılmış ve bu çıkış işlevi eşgörünümüyle ilişkilendirilmiş herhangi bir saklama alanı, çıkışa ilişkin işlevler tarafından kullanılabilir. (Bu saklama alanı genellikle sonlandırma çıkış işlevinin çağrılması sırasında serbest bırakılır.)

MQXEP arabirimi aşağıdaki gibi olur:

```
MQXEP (Hconfig, ExitReason, Function, EntryPoint, &ExitOpts, &CompCode, &Reason)
```

Burada:

### Hconfig (MQHCONFIG)-giriş

Başlatılmakta olan işlevlerin kümesini içeren API çıkışını temsil eden yapılandırma tanıtıcısı. Bu değer, MQ\_INIT\_EXIT işlevini çağırılmadan hemen önce kuyruk yöneticisi tarafından oluşturulur ve MQXEP ' de her bir API çıkış işlevine geçilir.

### ExitReason (MQHOT)-giriş

Giriş noktasının kaydedilmesinin nedeni, aşağıdaki nedenlerden.

- Bağlantı düzeyinde kullanıma hazırlama ya da sonlandırma (MQXR\_CONNECTION)
- Bir IBM MQ API çağrısından önce (MQXR\_BEFORE)
- IBM MQ API çağrısından sonra (MQXR\_AFTER)

### İşlev (MQUZE)-giriş

İşlev tanıtıcısı, MQXF\_ \* değişmezleri olan geçerli değerler (bkz. [“Dış sabitler” sayfa 1546](#)).

### EntryPoint (PMQFUNC)-giriş

Kaydolmak için çıkış işlevine ilişkin giriş noktasının adresi. NULL değeri, çıkış işlevinin sağlanmadığını ya da çıkış işlevinin önceki bir kaydının kayıttan kaldırılmakta olduğunu gösterir.

### ExitOpts(MQXEPO)

API çıkışları, API çıkışlarının kaydedilme şeklini denetleyen seçenekleri belirleyebilir. Bu alan için boş bir gösterge belirtilirse, MQXEPO yapısının varsayılan değerleri varsayılır.

### CompCode (MQHOT)-çıkış

Tamamlanma kodu, geçerli değerler şunlardır:

#### MQCC\_OK

İşlem başarıyla tamamlandı.

#### MQCC\_FAILED

Arama başarısız oldu.



**Neden (MQUZE)-çıkış**

Tamamlanma kodunu niteleyen neden kodu.

Tamamlanma kodu MQCC\_OK ise:

**MQRC\_NONE**

(0, X'000 ') Raporlamak için bir neden yok.

Tamamlanma kodu MQCC\_FAILED ise:

**MQRC\_HCONFIL\_ERROR**

(2280, X'8E8') Belirtilen yapılandırma tanıtıcısı geçerli değil. MQAXP ' deki yapılanış tanıtıcısını kullanın.

**MQRC\_EXIT\_REASON\_ERROR**

(2377, X' 949 ') Belirtilen çıkış işlevi çağırma nedeni geçerli değil ya da sağlanan çıkış işlevi tanıtıcısı için geçerli değil.

Geçerli çıkış işlevi çağırma nedenlerinden birini (MQXR\_ \* değeri) kullanın ya da geçerli bir işlev tanıtıcısı ve çıkış nedeni birleşimi kullanın. (Bkz. Çizelge 837 sayfa 1549.)

**MQRC\_FUNCTION\_HATASI**

(2281, X'8E9') Belirtilen işlev tanıtıcısı, API çıkış nedeni için geçerli değil. Aşağıdaki çizelge, işlev tanıtıcılarının ve ExitReasons' in geçerli birleşimlerini göstermektedir.

<i>Çizelge 837. İşlev tanıtıcılarının ve ExitReasons birleşimlerinin geçerli birleşimleri</i>	
<b>İşlev</b>	<b>ExitReason</b>
MQXF_INIT MQXF_TERM	MQXR_BAĞLANTI
MQXF_CONN MQXF_CONNX MQXF_DISC MQXF_Aç MQXF_CLOSE MQXF_PUT1 MQXF_PUT MQXF_GET MQXF_INQ MQXF_SET MQXF_BEGIN MQXF_CMIT MQXF_BACK MQXF_STAT MQXF_CB MQXF_CTL MQXF_CALLIK MQXF_SUB MQXF_SUBRQ	MQXR_BEDEN MQXR_AFTER
MQXF_DATA_CONV_ON_GET	MQXR_BEDEN

**MQRC\_RESOURCE\_SORUNU**

(2102, X'836 ') Bir çıkış işlevini kaydetme ya da kayıttan kaldırma girişimi, bir kaynak sorunu nedeniyle başarısız oldu.

**MQRC\_UNEXPECTED\_ERROR**

(2195, X'893 ') Çıkış işlevini kaydetme ya da kayıttan kaldırma girişimi beklenmeyen bir şekilde başarısız oldu.

## MQRC\_ÖZELLİĞİ \_

(2442, X'098A') ExitProperties adı geçersiz.

## MQRC\_XEPO\_ERROR

(2507, X'09CB') Çıkış seçenekleri yapısı geçerli değil.

## MQXEP C dili çağırısı

```
MQXEP (Hconfig, ExitReason, Function, EntryPoint, &ExitOpts, &CompCode, &Reason);
```

Değiştirge listesine ilişkin bildirim:

```
MQHCONFIG      Hconfig;          /* Configuration handle */
MQLONG         ExitReason;      /* Exit reason */
MQLONG         Function;        /* Function identifier */
PMQFUNC       EntryPoint;       /* Function entry point */
MQXEPO         ExitOpts;        /* Options that control the action of MQXEP */
MQLONG         CompCode;        /* Completion code */
MQLONG         Reason;          /* Reason code qualifying completion
                                code */
```

## MQXEP C işlevi prototipi

```
void MQXEP (
MQHCONFIG      Hconfig,          /* Configuration handle */
MQLONG         ExitReason,       /* Exit reason */
MQLONG         Function,        /* Function identifier */
PMQFUNC       EntryPoint,       /* Function entry point */
MQXEPO         pExitOpts;       /* Options that control the action of MQXEP */
PMQLONG        pCompCode,       /* Address of completion code */
PMQLONG        pReason);        /* Address of reason code qualifying completion
                                code */
```

## Çıkış işlevleri

Bu kısım, işlev çağrılarını kullanırken size yardımcı olacak bazı genel bilgileri ve tek tek çıkış işlevlerinin nasıl çağrılacağını açıklar.

API çıkış yordamlarına ilişkin genel kuralları anlamak ve çıkış yürütme ortamını ayarlamak ve temizlemek için bu bilgileri kullanın.

## API çıkış yordamlarıyla ilgili genel kurallar

API çıkış yordamları çağrılırken aşağıdaki genel kurallar geçerlidir:

- Tüm durumlarda, API çağrı parametreleri doğrulanmadan önce API çıkış işlevleri yönlendirilir ve herhangi bir güvenlik denetiminden önce (MQCONN, MQCONNX ya da MQOPEN durumunda).
- Çıkış yordamlarından girilen ve çıkışa girilen alanların değerleri şunlardır:
  - Bir önce IBM MQ API çıkış işlevinde, bir alanın değeri uygulama programı tarafından ya da önceki bir çıkış işlevi çağrısıyla ayarlanabilir.
  - Bir önceki IBM MQ API çıkış işlevinden çıktıda, bir alanın değeri değişmeden bırakılabilir ya da çıkış işlevi tarafından başka bir değere ayarlanabilmektedir.
  - On input to an *bundan sonra* IBM MQ API exit function, the value of a field can be the value set by the queue manager after processing the IBM MQ API call, or can be set to a value by a previous exit function invocation in the chain of exit functions.
  - Bir sonraki IBM MQ API çağrısı çıkış işlevinden çıktıda, bir alanın değeri değişmeden bırakılabilir ya da çıkış işlevi tarafından başka bir değere ayarlanabilmektedir.
- Çıkış işlevleri, ExitResponse ve ExitResponse2 alanlarını kullanarak kuyruk yöneticisiyle iletişim kurmalıdır.

- CompCode ve Neden kodu alanları, uygulamaya geri iletişim kurar. Kuyruk yöneticisi ve çıkış işlevleri CompCode ögesini ve neden kodu alanlarını ayarlayabilir.
- MQXEP çağrısı, MQXEP ' yi (MQXEP) çağırma işlevleriyle ilgili yeni neden kodlarını döndürür. Ancak, çıkış işlevleri bu yeni neden kodlarını varolan ve yeni uygulamaların anlayabileceği herhangi bir neden kodlarına çevirebilir.
- Her bir çıkış işlevi prototipi, CompCode ve Neden dışında, fazladan bir yönelim düzeyi olan API işlevine benzer deęiřtirgeleri içerir.
- API çıkışları, MQI çağrılarını yayınlayabilir (MQDISC dışında), ancak bu MQI çağrılarını kendilerine API çıkışları çağırılmaz.

Uygulamanın bir sunucuda mı, yoksa bir istemcide mi olduğunu, API çıkış çağrılarının sıralamasını tahmin edemediđinizi unutmayın. An API exit BEFORE call might not be followed immediately by an AFTER call.

BEFORE çağrısı, başka bir BEFORE çağrısı tarafından izlenebilir. Örneđin:

```
MQCTL ' DEN
Geri Çađırmadan önce
MQPUT ' DEN
MQPUT ' DEN
Geri Çađırma SONR
MQCTL ' DEN
```

ya da

```
XAXX_ENCODE_CASE_ONE open ' DAN
MQCONNX ' DEN
MQCONNX ' DEN
XAXX_ENCODE_CASE_ONE open ' DAN
```

On the client, there is an exit that can modify the behavior of the MQCONN or MQCONNX call, called the PreConnect exit. PreConnect çıkışı, kuyruk yöneticisi adı da içinde olmak üzere, MQCONN ya da MQCONNX çağrısındaki deęiřtirgelerden herhangi birini deęiřtirebilir. İstemci önce bu çıkışı çağırır, sonra MQCONN ya da MQCONNX çağrısını çağırır. Yalnızca ilk MQCONN ya da MQCONNX çağrısının API çıkışını çağırıldığını unutmayın; sonraki yeniden bağlanma çağrılarının herhangi bir etkisi yoktur.

## Yürütme ortamı

Genel olarak, çıkış işlevlerinden gelen tüm hatalar, MQAXP ' deki ExitResponse ve ExitResponse2 alanlarını kullanarak çıkış işleyicisine geri iletilir.

Bu hatalar, MQCC\_ \* ve MQRC\_ \* deđerlerine dönüřtürülür ve CompCode ve Reason alanlarında uygulamaya geri iletilir. Ancak, çıkış işleyici mantığında saptanan hatalar, CompCode ve neden alanlarında MQCC\_ \* ve MQRC\_ \* deđerleri olarak uygulamaya geri iletilir.

Bir MQ\_TERM\_EXIT işlevi bir hata döndürürse:

- MQDISC çağrısı zaten alındı
- *bundan sonra* MQ\_TERM\_EXIT çıkış işlevini kullanmak için başka bir fırsat yoktur (ve dolayısıyla çıkış yürütme ortamı temizlemesi gerçekleştirilir)
- Yürütme ortamı temizlemesi gerçekleştirilmedi

Hala kullanımda olabileceđi için çıkış boşaltılmıyor. Ayrıca, *önce* çıkışı başarılı olan çıkış zincirinde diđer kayıtlı çıkışlar da ters sırayla yönlendirilecek.

## Çıkış yürütme ortamı ayarlanıyor

Belirtik bir MQCONN ya da MQCONNX çağrısı işlenirken, çıkış yürütme ortamı, çıkış başlatma işlevini (MQ\_INIT\_EXIT) çağırılmadan önce, çıkış yürütme ortamını ayarlar. Çıkış yürütme ortamı ayarı, çıkışa yüklenmeyi, depolama alanı edinilmesini ve çıkış parametresi yapılarını kullanıma hazırlanmasını içerir. Çıkış yapılandırma tanıtıcısı da ayrıdır.

Bu aşama sırasında hatalar oluşursa, MQCONN ya da MQCONNX çağrısı CompCode MQCC\_FAILED ile başarısız olur ve aşağıdaki neden kodlarından biridir:

#### **MQRC\_API\_EXIT\_LOAD\_ERROR**

Bir API çıkış modülünü yükleme girişimi başarısız oldu.

#### **MQRC\_API\_EXIT\_NOT\_FOUND**

API çıkış modülünde bir API çıkış işlevi bulunamadı.

#### **MQRC\_STORAGE\_NOT\_AVAM**

Kullanılabilir bellek yetersiz olduğundan, bir API çıkış işlevi için yürütme ortamını kullanıma hazırlama girişimi başarısız oldu.

#### **MQRC\_API\_EXIT\_INIT\_ERROR**

Bir API çıkış işlevi için yürütme ortamı kullanıma hazırlanırken bir hata saptandı.

### **Çıkış yürütme ortamı temizleniyor**

Belirtik bir MQDISC çağrısı işlenirken ya da uygulamanın sona erdirilmesi sonucunda örtük bir bağlantı kesme isteği işlenirken, çıkış işleme mantığının, kayıtlı olduğu durumlarda, çıkış sonlandırma işlevini (MQ\_TERM\_EXIT) çağırıdıktan sonra çıkış yürütme ortamını temizlemesi gerekebilir.

Çıkış yürütme ortamının temizlenmesi, çıkış değiştirgesi yapılarına ilişkin saklama alanının serbest bırakılmasını, daha önce belleğe yüklenmiş olan modülleri silmenin büyük olasılıkla gerçekleştirilmesini içerir.

If errors occur during this phase, an explicit MQDISC call fails with CompCode MQCC\_FAILED and the following reason code (errors are not highlighted on implicit disconnect requests):

#### **MQRC\_API\_EXIT\_TERM\_ERROR**

Bir API çıkış işlevine ilişkin yürütme ortamı kapatılırken bir hata saptandı. Çıkışta, MQDISC ' den önce ya da MQ\_TERM\* API çıkış işlevi çağrılarında sonra hata döndürmemelidir.

### **İstemcilerde API çıkışları**

Bir istemci, MQCONN ve MQCONNX çağrılarının davranışını değiştirmek ve API çıkış özelliklerini desteklememek için PreConnect çıkışını kullanır.

### **PreConnect çıkışı**

Bir istemcide, kanal tanımlamasını LDAP sunucusu gibi merkezi bir havuzdan aramak için PreConnect çıkışıdır.

PreConnect çıkışı, bir MQCONN ya da MQCONNX çağrısındaki herhangi bir değiştirgeyi ya da tüm değiştirgeyi değiştirebilir; örneğin, kuyruk yöneticisi adı gibi.

İstemci uygulamaları durumunda, MQCONN ya da MQCONNX API çıkışı yalnızca kuyruk yöneticisinin adı bilindiğinde ve bu ad PreConnect çıkışıyla değiştirilebileceğinden, API çıkıştan önce PreConnect çıkışa çağırılması gerekir.

Yalnızca başlangıçtaki MQCONN ya da MQCONNX çağrısının çıkışa çağırıldığını unutmayın.

### **API çıkış özellikleri**

Bir sunucuda, API çıkışları kullanıma hazırlama sırasında bir MQXEPO yapısını kaydedebilir. MQXEPO yapısı, çıkışa ilişkin özellikler grubunun ayrıntılarını içeren ExitProperties alanını içerir. Bu, çıkışta herhangi bir uygulama iletilisi özellik tanıtıcısından ayrı olarak işlem yapabildiği ayrı bir ileti özelliği tanıtıcısı oluşturmada etkiye sahiptir.

Bir istemcide API çıkış özellikleri desteklenmez. İstemcide bir özellik grubu adı kaydettirmek için bir girişimde bulunulursa, bu işlev bir MQRC\_EXIT\_PROPS\_NOT\_DESPORTED neden koduyla başarısız olur.

## Geriletme-MQ\_BACK\_EXIT

MQ\_BACK\_EXIT, *önce ve bundan sonra* geriletme işlemini gerçekleştirmek için geriletme çıkış işlevi sağlar. MQXR\_BACK ve MQXR\_AFTER işlev tanıtıcısını kullanarak, MQXR\_BEFORE ve MQXR\_AFTER çıkış nedenleriyle *önce ve bundan sonra* arka çıkış arama çıkış işlevlerini kaydettikten sonra.

Bu işleve ilişkin arabirim şöyledir:

```
MQ_BACK_EXIT (&ExitParms, &ExitContext, &Hconn, &CompCode, &Reason)
```

parametrelerin bulunduğu yer:

### ExitParms (MQAXP)-giriş/çıkış

Parametre yapısından çıkın.

### ExitContext (MQAXC)-giriş/çıkış

Bağlam yapısından çıkın.

### Hconn (MQHCONN)-giriş

Bağlantı tanıtıcısı.

### CompCode (MQHOT)-giriş/çıkış

Tamamlanma kodu, geçerli değerler şunlardır:

#### MQCC\_OK

İşlem başarıyla tamamlandı.

#### MQCC\_UYARI

Kısmi tamamlama.

#### MQCC\_FAILED

Arama başarısız oldu

### Neden (MQlong)-giriş/çıkış

Neden kodu, tamamlanma kodunu nitelemektedir.

Tamamlanma kodu MQCC\_OK ise, tek geçerli değer şöyledir:

#### MQRC\_NONE

(0, x '000') Raporlamak için bir neden yok.

Tamamlanma kodu MQCC\_FAILED ya da MQCC\_UYARI ise, çıkış işlevi neden kodu alanını geçerli bir MQRC\_ \* değerine ayarlayabiliyor.

## C dili çağırma

Kuyruk yöneticisi, aşağıdaki değişkenleri mantıksal olarak tanımlar:

```
MQAXP    ExitParms;        /* Exit parameter structure */
MQAXC    ExitContext;      /* Exit context structure */
MQHCONN  Hconn;           /* Connection handle */
MQLONG   CompCode;        /* Completion code */
MQLONG   Reason;          /* Reason code qualifying completion code */
```

Kuyruk yöneticisi, çıkışı mantıksal olarak aşağıdaki gibi çağırır:

```
MQ_BACK_EXIT (&ExitParms, &ExitContext, &Hconn, &CompCode, &Reason);
```

Çıkışınız, aşağıdaki C işlevi prototipini eşleştirmelidir:

```
void MQENTRY MQ_BACK_EXIT (
PMQAXP    pExitParms,      /* Address of exit parameter structure */
PMQAXC    pExitContext,    /* Address of exit context structure */
PMQHCONN  pHconn,         /* Address of connection handle */
PMQLONG   pCompCode,      /* Address of completion code */
PMQLONG   pReason);       /* Address of reason code qualifying completion
                           code */
```

## **Başlat-MQ\_BEGIN\_EXIT**

MQ\_BEGIN\_EXIT, *önce ve bundan sonra* MQBEGIN çağrı işlemlerini gerçekleştirmek için bir başlangıç çıkış işlevi sağlar. MQXR\_BEFORE ve MQXR\_BEDEN çıkış nedenleriyle *önce ve sonra* MQBEGIN çağrı çıkış işlevlerinden sonra MQX\_BEGIN işlev tanıtıcısını kullanın.

Bu işleve ilişkin arabirim şöyledir:

```
MQ_BEGIN_EXIT (&ExitParms, &ExitContext, &Hconn, &pBeginOptions, &CompCode,
               &Reason)
```

parametrelerin bulunduğu yer:

### **ExitParms (MQAXP)-giriş/çıkış**

Parametre yapısından çıkın.

### **ExitContext (MQAXC)-giriş/çıkış**

Bağlam yapısından çıkın.

### **Hconn (MQHCONN)-giriş**

Bağlantı tanıtıcısı.

### **pBeginOptions (PMQBO)-giriş/çıkış**

Başlangıç seçenekleri işaretçisi.

### **CompCode (MQHOT)-giriş/çıkış**

Tamamlanma kodu, geçerli değerler şunlardır:

#### **MQCC\_OK**

İşlem başarıyla tamamlandı.

#### **MQCC\_UYARI**

Kısmi tamamlama.

#### **MQCC\_FAILED**

Arama başarısız oldu

### **Neden (MQlong)-giriş/çıkış**

Neden kodu, tamamlanma kodunu nitelemektedir.

Tamamlanma kodu MQCC\_OK ise, tek geçerli değer şöyledir:

#### **MQRC\_NONE**

(0, x '000') Raporlamak için bir neden yok.

Tamamlanma kodu MQCC\_FAILED ya da MQCC\_UYARI ise, çıkış işlevi neden kodu alanını geçerli bir MQRC\_\* değerine ayarlayabiliyor.

## **C dili çağırma**

Kuyruk yöneticisi, aşağıdaki değişkenleri mantıksal olarak tanımlar:

```
MQAXP    ExitParms;      /* Exit parameter structure */
MQAXC    ExitContext;    /* Exit context structure */
MQHCONN  Hconn;          /* Connection handle */
PMQBO    pBeginOptions;  /* Ptr to begin options */
MQLONG   CompCode;      /* Completion code */
MQLONG   Reason;        /* Reason code qualifying completion code */
```

Kuyruk yöneticisi, çıkışı mantıksal olarak aşağıdaki gibi çağırır:

```
MQ_BEGIN_EXIT (&ExitParms, &ExitContext, &Hconn, &pBeginOptions, &CompCode,
               &Reason);
```

Çıkışınız, aşağıdaki C işlevi prototipini eşleştirmelidir:

```
void MQENTRY MQ_BEGIN_EXIT (
```

```

PMQAXP    pExitParms,      /* Address of exit parameter structure */
PMQAXC    pExitContext,   /* Address of exit context structure */
PMQHCONN  pHconn,        /* Address of connection handle */
PPMQBO    ppBeginOptions, /* Address of ptr to begin options */
PMQLONG   pCompCode,     /* Address of completion code */
PMQLONG   pReason);      /* Address of reason code qualifying completion
                           code */

```

## Geri Çağırım-MQ\_CALLBACK\_EXIT

MQ\_CALLBACK\_EXIT, *önce* ve *bundan sonra* geri bildirme işlemlerini gerçekleştirmek için bir çıkış işlevi sağlar. *önce* ve *bundan sonra* geri bildirme çağrısı çıkış işlevlerini kaydettirmek için MQXR\_AFTER ve MQXR\_AFTER çıkış nedenleriyle MQXF\_CALLBACK işlev tanıtıcısını kullanın.

Bu işleve ilişkin arabirim şöyledir:

```

MQ_CALLBACK_EXIT (&ExitParms, &ExitContext, &Hconn, &pMsgDesc, &pGetMsgOpts,
                  &pBuffer, &PMQCBCContext)

```

parametrelerin bulunduğu yer:

### ExitParms (MQAXP)-giriş/çıkış

Parametre yapısından çık

### ExitContext (MQAXC)-giriş/çıkış

Bağlam yapısından çık

### Hconn (MQHCONN)-giriş/çıkış

Bağlantı tanıtıcı

### pMsgAçıklaması

İleti tanımlayıcısı

### pGetMsgOpts

MQGET işlemini denetleyen seçenekler

### pBuffer

İleti verilerinin içereceği alan

### PMQCBCContext

Geri bildirme için bağlam verileri

## C dili çağırma

Kuyruk yöneticisi, aşağıdaki değişkenleri mantıksal olarak tanımlar:

```

MQAXP    ExitParms;      /* Exit parameter structure */
MQAXC    ExitContext;   /* Exit context structure */
MQHCONN  Hconn;         /* Connection handle */
PMQMD    pMsgDesc;      /* Message descriptor */
PMQGM0   pGetMsgOpts;   /* Options that define the operation of the consumer */
PMQVOID   pBuffer;      /* Area to contain the message data */
PMQCBC    pContext;     /* Context data for the callback */

```

Kuyruk yöneticisi, çıkışı mantıksal olarak aşağıdaki gibi çağırır:

```

MQ_SUBRQ_EXIT (&ExitParms, &ExitContext, &Hconn, &pMsgDesc, &pGetMsgOpts, &pBuffer,
               &pContext);

```

Çıkışınız, aşağıdaki C işlevi prototipini eşleştirmelidir:

```

void MQENTRY MQ_CALLBACK_EXIT (
PMQAXP    pExitParms;   /* Exit parameter structure */
PMQAXC    pExitContext; /* Exit context structure */
PMQHCONN  pHconn;       /* Connection handle */
PPMQMD    ppMsgDesc;    /* Message descriptor */
PPMQGM0   ppGetMsgOpts; /* Options that define the operation of the consumer */

```

```
PPMQVOID ppBuffer; /* Area to contain the message data */
PPMQCBC ppContext; /* Context data for the callback */
```

## Kullanım notları

1. Geri Çağırım çıkışı, tüketici çağrılmadan önce ve tüketicinin tüketici işlevinin tamamlanmasından sonra çağrılır. MQMD ve MQGMO yapıları değiştirilebilir de, önceki çıkıştaki değerleri değiştirerek, ileti kuyruktan tüketici işlevine teslim edilecek şekilde kaldırıldığı için, kuyruktan bir iletinin alınmasını yeniden kullanmaz.

### **Geri bildirme işlevlerini yönet-MQ\_CB\_EXIT**

MQ\_CB\_EXIT, *önce* ve *bundan sonra* MQCB çağrısını gerçekleştirmek için bir çıkış işlevi sağlar. *önce* ve *bundan sonra* MQCB arama çıkış işlevlerini kaydettirmek için MQXR\_BEFORE ve MQXR\_AFTER çıkış nedenleriyle MQXF\_CB işlev tanıtıcısını kullanın.

Bu işleve ilişkin arabirim şöyledir:

```
MQ_CB_EXIT (&ExitParms, &ExitContext, &Hconn, &Operation, &pCallbackDesc,
            &Hobj, &pMsgDesc, &pGetMsgOpts, &CompCode, &Reason)
```

parametrelerin bulunduğu yer:

#### **ExitParms (MQAXP)-giriş/çıkış**

Parametre yapısından çık

#### **ExitContext (MQAXC)-giriş/çıkış**

Bağlam yapısından çık

#### **Hconn (MQHCONN)-giriş/çıkış**

Bağlantı tanıtıcı

#### **Operation (MQUZE)-giriş/çıkış**

İşlem değeri

#### **pCallbackDesc (PMQCBD)-giriş/çıkış**

Geri Çağırma

#### **Hobj (MQHOBJ)-giriş/çıkış**

Nesne tanıtıcısı

#### **pMsgDesc (PMQMD)-giriş/çıkış**

İleti tanımlayıcısı

#### **pGetMsgOpts (PMQGMO)-giriş/çıkış**

MQCB 'nin işlemini denetleyen seçenekler

#### **CompCode (MQHOT)-giriş/çıkış**

Tamamlanma kodu

#### **Neden (MQLONG)-giriş/çıkış**

Neden kodu ön eleme CompCode

## C dili çağırma

Kuyruk yöneticisi, aşağıdaki değişkenleri mantıksal olarak tanımlar:

```
MQAXP ExitParms; /* Exit parameter structure */
MQAXC ExitContext; /* Exit context structure */
MQHCONN Hconn; /* Connection handle */
MQLONG Operation; /* Operation value. */
MQCBD pMsgDesc; /* Callback descriptor. */
MQHOBJ Hobj; /* Object handle. */
PMQMD pMsgDesc; /* Message descriptor */
PMQGMO pGetMsgOpts; /* Options that define the operation of the consumer */
PMQLONG CompCode; /* Completion code. */
PMQLONG Reason; /* Reason code qualifying CompCode. */
```



Kuyruk yöneticisi, çıkışı mantıksal olarak aşağıdaki gibi çağırır:

```
MQ_CB_EXIT (&ExitParms, &ExitContext, &Hconn, &Operation, &Hobj, &pMsgDesc,  
&pGetMsgOpts, &CompCode, &Reason);
```

Çıkışınız, aşağıdaki C işlevi prototipini eşleştirmelidir:

```
void MQENTRY MQ_CB_EXIT (  
PMQAXP    pExitParms;    /* Exit parameter structure */  
PMQAXC    pExitContext;  /* Exit context structure */  
PMQHCONN  pHconn;        /* Connection handle */  
PMQLONG   pOperation;    /* Callback operation */  
PMQHOBJS  pHobj;         /* Object handle */  
PPMQMD    ppMsgDesc;     /* Message descriptor */  
PPMQGMO   ppGetMsgOpts;  /* Options that control the action of MQCB */  
PMQLONG   pCompCode;     /* Completion code */  
PMQLONG   pReason;       /* Reason code qualifying CompCode */
```

### **Kapat-MQ\_CLOSE\_EXIT**

MQ\_CLOSE\_EXIT, *önce ve bundan sonra* MQCLOSE çağrılarının gerçekleştirilmesini gerçekleştirmek için bir kapatma işlevi sağlar. *önce ve bundan sonra* MQCLOSE çağrı çıkış işlevlerini kaydettirmek için MQXR\_BEFORE ve MQXR\_AFTER çıkış nedenleriyle MQXF\_CLOSE işleviyle işlem tanımlamayı kullanın.

Bu işleve ilişkin arabirim şöyledir:

```
MQ_CLOSE_EXIT (&ExitParms, &ExitContext, &Hconn, &pHobj,  
&Options, &CompCode, &Reason)
```

parametrelerin bulunduğu yer:

#### **ExitParms (MQAXP)-giriş/çıkış**

Parametre yapısından çıkın.

#### **ExitContext (MQAXC)-giriş/çıkış**

Bağlam yapısından çıkın.

#### **Hconn (MQHCONN)-giriş**

Bağlantı tanıtıcısı.

#### **pHobj (PMQHOBJS)-giriş**

Nesne tanıtıcısı işaretçisi.

#### **Seçenekler (MQUZE)-giriş/çıkış**

Seçenekleri kapatın.

#### **CompCode (MQHOT)-giriş/çıkış**

Tamamlanma kodu, geçerli değerler şunlardır:

##### **MQCC\_OK**

İşlem başarıyla tamamlandı.

##### **MQCC\_FAILED**

Arama başarısız oldu

#### **Neden (MQlong)-giriş/çıkış**

Neden kodu, tamamlanma kodunu nitelemektedir.

Tamamlanma kodu MQCC\_OK ise, tek geçerli değer şöyledir:

##### **MQRC\_NONE**

(0, x '000') Raporlamak için bir neden yok.

Tamamlanma kodu MQCC\_FAILED ise, çıkış işlevi neden kodu alanını geçerli bir MQRC\_ \* değerine ayarlayabilir.

## C dili çağırma

Kuyruk yöneticisi, aşağıdaki değişkenleri mantıksal olarak tanımlar:

```
MQAXP      ExitParms;      /* Exit parameter structure */
MQAXC      ExitContext;   /* Exit context structure */
MQHCONN    Hconn;        /* Connection handle */
PMQHOBJS   pHobj;        /* Ptr to object handle */
MQLONG     Options;      /* Close options */
MQLONG     CompCode;     /* Completion code */
MQLONG     Reason;       /* Reason code */
```

Kuyruk yöneticisi, çıkışı mantıksal olarak aşağıdaki gibi çağırır:

```
MQ_CLOSE_EXIT (&ExitParms, &ExitContext,&Hconn, &pHobj, &Options,
               &CompCode, &Reason);
```

Çıkışınız, aşağıdaki C işlevi prototipini eşleştirmelidir:

```
void MQENTRY MQ_CLOSE_EXIT (
PMQAXP      pExitParms,   /* Address of exit parameter structure */
PMQAXC      pExitContext, /* Address of exit context structure */
PMQHCONN    pHconn,      /* Address of connection handle */
PMQHOBJS    ppHobj,      /* Address of ptr to object handle */
PMQLONG     pOptions,     /* Address of close options */
PMQLONG     pCompCode,    /* Address of completion code */
PMQLONG     pReason);    /* Address of reason code qualifying
                           completion code */
```

### **Kesinleştir-MQ\_CMITS\_EXIT**

MQ\_CMITS\_EXIT, *önce* ve *bundan sonra* kesinleştirme işlemlerini gerçekleştirmek için bir kesinleştirme çıkış işlevi sağlar. *önce* ve *bundan sonra* kesinleştirme çağırısı çıkış işlevlerini kaydettirmek için MQXR\_BEFORE ve MQXR\_AFTER çıkış nedenleriyle MQXF\_CMITS işlev tanıtıcısını kullanın.

Kesinleştirme işlemi başarısız olursa ve hareket geriletirse, MQCMITS çağırısı MQCC\_UYARI ve MQRC\_BACKED\_OUT ile başarısız olur. Bu dönüş ve neden kodları, çıkışı vermek için herhangi bir *bundan sonra* MQCMITS çıkış işlevine geçirilir ve iş biriminin yedeklendiğine ilişkin bir gösterge sağlar.

Bu işleve ilişkin arabirim şöyledir:

```
MQ_CMITS_EXIT (&ExitParms, &ExitContext, &Hconn, &CompCode, &Reason)
```

parametrelerin bulunduğu yer:

#### **ExitParms (MQAXP)-giriş/çıkış**

Parametre yapısından çıkın.

#### **ExitContext (MQAXC)-giriş/çıkış**

Bağlam yapısından çıkın.

#### **Hconn (MQHCONN)-giriş**

Bağlantı tanıtıcısı.

#### **CompCode (MQHOT)-giriş/çıkış**

Tamamlanma kodu, geçerli değerler şunlardır:

##### **MQCC\_OK**

İşlem başarıyla tamamlandı.

##### **MQCC\_UYARI**

Kısmi tamamlama.

##### **MQCC\_FAILED**

Arama başarısız oldu

## Neden (MQlong)-giriş/çıkış

Neden kodu, tamamlanma kodunu nitelemektedir.

Tamamlanma kodu MQCC\_OK ise, tek geçerli değer şöyledir:

### MQRC\_NONE

(0, x '000') Raporlamak için bir neden yok.

Tamamlanma kodu MQCC\_FAILED ya da MQCC\_UYARI ise, çıkış işlevi neden kodu alanını geçerli bir MQRC\_\* değerine ayarlayabiliyor.

## C dili çağırma

Kuyruk yöneticisi, aşağıdaki değişkenleri mantıksal olarak tanımlar:

```
MQAXP    ExitParms;      /* Exit parameter structure */
MQAXC    ExitContext;    /* Exit context structure */
MQHCONN  Hconn;         /* Connection handle */
MQLONG   CompCode;      /* Completion code */
MQLONG   Reason;        /* Reason code qualifying completion code */
```

Kuyruk yöneticisi, çıkışı mantıksal olarak aşağıdaki gibi çağırır:

```
MQ_CMITY_EXIT (&ExitParms, &ExitContext,&Hconn, &CompCode, &Reason);
```

Çıkışınız, aşağıdaki C işlevi prototipini eşleştirmelidir:

```
void MQENTRY MQ_CMITY_EXIT (
PMQAXP    pExitParms,      /* Address of exit parameter structure */
PMQAXC    pExitContext,    /* Address of exit context structure */
PMQHCONN  pHconn,         /* Address of connection handle */
PMQLONG   pCompCode,      /* Address of completion code */
PMQLONG   pReason);      /* Address of reason code qualifying completion
                           code */
```

## Kullanım notları

1. Burada açıklanan MQ\_GET\_EXIT işlev arabirimi, hem MQXF\_GET çıkış işlevi, hem de [“MQXF\\_DATA\\_CONV\\_ON\\_GET” sayfa 1565 çıkış işlevi için kullanılır.](#)

Bu iki çıkış işlevi için ayrı giriş noktaları tanımlıdır, bu nedenle *her ikisi* ' i engellemek için MQXEP çağırısının iki kez kullanılması gerekir; bu çağrı için MQXF\_GET işlev tanıtıcısı kullanın.

MQ\_GET\_EXIT arabirimi MQXF\_GET ve MQXF\_DATA\_CONV\_ON\_GET için aynı olduğu için, her ikisi için de tek bir çıkış işlevi kullanılabilir; MQAXP yapısındaki *Function* alanı, hangi çıkış işlevinin çağrıldığını gösterir. Diğer bir seçenek olarak, MQXEP çağırısı, iki vaka için farklı çıkış işlevlerini kaydetmek için kullanılabilir.

### **Bağlan ve uzantı bağlan-MQ\_CONNX\_EXIT**

MQ\_CONNX\_EXIT provides connection exit function to perform *önce* and *bundan sonra* MQCONN processing, and connection extension exit function to perform *önce* and *bundan sonra* MQCONNX processing.

Burada açıklandığı gibi, aynı arabirim, hem MQCONN hem de MQCONNX çağrı çıkış işlevleri için çağrılır.

Message Channel Agent (MCA) gelen istemci bağlantısına yanıt verdiğinde, MCA, istemci durumu tam olarak bilinmeden önce bir dizi IBM MQ API çağırısını bağlanabilir ve bu çağrılara bir sayı yapabilir. Bu API çağrıları, MCA programının kendisine dayalı olarak MQAXC ile API çıkış işlevlerini çağırır (örneğin, MQAXC ' nin UserId ve ConnectionName alanlarında).

MCA, sonraki gelen istemci API çağrılarında yanıt verdiğinde, MQAXC yapısı gelen istemciye dayalıdır; UserId ve ConnectionName alanlarını uygun bir şekilde ayarlıdır.

Uygulama tarafından MQCONN ya da MQCONNX çağrısında belirlenen kuyruk yöneticisi adı, temeldeki bağlantı çağrısına iletilir. Bir önce MQ\_CONN\_EXIT tarafından kuyruk yöneticisinin adının değiştirilmeye çalışılması herhangi bir etkiye sahip değildir.

önce ve bundan sonra MQCONN ve MQCONNX çağrı çıkış işlevlerini kaydettirmek için MQXR\_CONN ve MQXR\_AFTER çıkış nedenleriyle MQXR\_CONN ve MQXF\_CONN işlev tanıtıcılarını kullanın.

An MQ\_CONN\_EXIT exit called for reason MQXR\_BEFORE *bunu yapmak için* issue any IBM MQ API calls, as the correct environment has not been set up at this time.

Bir MQ\_CONN\_EXIT, çağrılmakta olduğu bağlantı için bir API çıkış çağrısından MQDISC ' yi çağırılmaz. Bu kısıtlama, hem istemci hem de sunucu API çıkışları için geçerlidir.

MQCONN ve MQCONNX için arabirim aynı:

```
MQ_CONN_EXIT (&ExitParms, &ExitContext, &pQMgrName, &pConnectOpts,  
&pHconn, &CompCode, &Reason);
```

parametrelerin bulunduğu yer:

#### **ExitParms (MQAXP)-giriş/çıkış**

Parametre yapısından çıkın.

#### **ExitContext (MQAXC)-giriş/çıkış**

Bağlam yapısından çıkın.

#### **pQMgrAd (PMQCHAR)-giriş**

MQCONNX çağrısında belirtilen kuyruk yöneticisi adına işaret eden gösterge. Çıkış, MQCONN ya da MQCONNX çağrısında bu adı değiştirmemelidir.

#### **pConnectOpts (PMQCNO)-giriş/çıkış**

MQCONNX çağrısının işlemini denetleyen seçeneklere işaret eden gösterge.

Ayrıntılar için bkz. “MQCNO-Bağlantı seçenekleri” sayfa 314.

MQXF\_CONN çıkış işlevi için, pConnectOpts varsayılan bağlantı seçenekleri yapısını gösterir (MQCNO\_VARSAYILANDANI).

#### **pHconn (PMQHCONN)-giriş**

Bağlantı tanıtıcısı işaretçisi.

#### **CompCode (MQHOT)-giriş/çıkış**

Tamamlanma kodu, geçerli değerler şunlardır:

##### **MQCC\_OK**

İşlem başarıyla tamamlandı.

##### **MQCC\_UYARI**

Uyarı (kısmi tamamlama)

##### **MQCC\_FAILED**

Arama başarısız oldu

#### **Neden (MQlong)-giriş/çıkış**

Neden kodu, tamamlanma kodunu nitelemektedir.

Tamamlanma kodu MQCC\_OK ise, tek geçerli değer şöyledir:

##### **MQRC\_NONE**

(0, x '000') Raporlamak için bir neden yok.

Tamamlanma kodu MQCC\_FAILED ya da MQCC\_UYARI ise, çıkış işlevi neden kodu alanını geçerli bir MQRC\_\* değerine ayarlayabiliyor.

## **C dili çağırma**

Kuyruk yöneticisi, aşağıdaki değişkenleri mantıksal olarak tanımlar:

```

MQAXP      ExitParms;      /* Exit parameter structure */
MQAXC      ExitContext; /* Exit context structure */
PMQCHAR    pQMgrName;   /* Ptr to Queue manager name */
PMQCNO     pConnectOpts; /* Ptr to Connection options */
PMQHCONN   pHconn;     /* Ptr to Connection handle */
MQLONG     CompCode;    /* Completion code */
MQLONG     Reason;      /* Reason code */

```

Kuyruk yöneticisi, çıkışı mantıksal olarak aşağıdaki gibi çağırır:

```

MQ_CONNX_EXIT (&ExitParms, &ExitContext, &pQMgrName, &pConnectOpts,
               &pHconn, &CompCode, &Reason);

```

Çıkışınız, aşağıdaki C işlevi prototipini eşleştirmelidir:

```

void MQENTRY MQ_CONNX_EXIT (
PMQAXP      pExitParms,      /* Address of exit parameter structure */
PMQAXC      pExitContext,    /* Address of exit context structure */
PPMQCHAR    ppQMgrName,     /* Address of ptr to queue manager name */
PPMQCNO     ppConnectOpts,   /* Address of ptr to connection options */
PPMQHCONN   ppHconn,        /* Address of ptr to connection handle */
PMQLONG     pCompCode,       /* Address of completion code */
PMQLONG     pReason);        /* Address of reason code qualifying
                               completion code */

```

## Kullanım notları

1. Burada açıklanan MQ\_CONNX\_EXIT işlev arabirimi, hem MQCONN çağırısı, hem de MQCONNX çağırısı için kullanılır. Ancak, bu iki arama için ayrı giriş noktaları tanımlanır. *her ikisi* çağrılarını engellemek için, MQXEP çağırısının, MQXF\_CONN işlev tanıtıcısı ile en az iki kez ve MQXF\_CONNX ile birlikte kullanılması gerekir.

MQ\_CONNX\_EXIT arabirimi, MQCONN ve MQCONNX için aynı olduğu için, her iki çağrı için de tek bir çıkış işlevi kullanılabilir; MQAXP yapısındaki *Function* alanı, devam etmekte olan çağrıyı gösterir. Diğer bir seçenek olarak, MQXEP çağırısı iki çağrıya ilişkin farklı çıkış işlevlerini kaydetmek için kullanılabilir.

2. Bir ileti kanalı aracısı (MCA) gelen istemci bağlantısına yanıt verdiğinde, MCA, istemci durumu tam olarak bilinmeden önce bir sayıda MQ çağırısını yayımlayabilir. Bu MQ çağrıları, API çıkış işlevlerinin MCA ile ilişkili verileri içeren MQAXC yapısıyla çağrılmasına ve istemciye (örneğin, kullanıcı kimliği ve bağlantı adı gibi) çağrılmasına neden olur. Ancak, istemci durumu tam olarak bilindikten sonra, sonraki MQ çağrıları, API çıkış işlevlerinin MQAXC yapısındaki uygun istemci verileriyle çağrılmaya neden olur.
3. Herhangi bir parametre geçerlilik denetimi kuyruk yöneticisi tarafından gerçekleştirilmeden önce, tüm MQXR\_BEFORE çıkış işlevleri çağrılır. Bu nedenle deęiřtirgeler geersiz olabilir (parametrelerin adresleri için geersiz iřaretiler de iinde olmak üzere).

MQ\_CONNX\_EXIT işlevi, kuyruk yöneticisi tarafından herhangi bir yetki denetimi gerekleřtirmeden önce çağrılır.

4. Çıkış işlevi, MQCONN ya da MQCONNX çağırısında belirtilen kuyruk yöneticisinin adını deęiřtirmemelidir. Çıkış, çıkış işleviyle deęiřtirilirse, sonuçlar tanımsız olur.
5. MQ\_CONNX\_EXIT için bir MQXR\_BEFORE çıkış işlevi, MQXEP dışında MQ çağrıları yayımlayamaz.

## Geri çağırma denetimi-MQ\_CTL\_EXIT

MQ\_CTL\_EXIT, *önce* ve *bundan sonra* denetim geri bildirme işlemlerini gerekleřtirmek için bir abonelik isteęi çıkış işlevi saęlar. *önce* ve *bundan sonra* denetim geri bildirme çağırısı çıkış işlevlerini kaydettirmek için MQXR\_BEFORE ve MQXR\_AFTER çıkış nedenleriyle MQXF\_CTL işlev tanıtıcısını kullanın.

Bu işleve ilişkin arabirim şöyledir:

```

MQ_CTL_EXIT (&Hconn, &Operation, &ControlOpts, &CompCode, &Reason)

```

parametrelerin bulunduğu yer:

### **Hconn (MQHCONN)-giriş/çıkış**

Bağlantı tanıtıcısı.

### **İşlem (MQUZE) giriş/çıkış**

Belirtilen nesne tanıtıcısı için tanımlanan geri bildirme üzerinde işlenmekte olan işlem

### **ControlOpts (MQCTLO) giriş/çıkış**

MQCTL ' nin işlemini denetleyen seçenekler

### **CompCode (MQHOT)-giriş/çıkış**

Tamamlanma kodu, geçerli değerler şunlardır:

#### **MQCC\_OK**

İşlem başarıyla tamamlandı.

#### **MQCC\_UYARI**

Kısmi tamamlama.

#### **MQCC\_FAILED**

Arama başarısız oldu

### **Neden (MQLONG)-giriş/çıkış**

Neden kodu, tamamlanma kodunu nitelemektedir.

Tamamlanma kodu MQCC\_OK ise, tek geçerli değer şöyledir:

#### **MQRC\_NONE**

(0, x '000') Raporlamak için bir neden yok.

Tamamlanma kodu MQCC\_FAILED ya da MQCC\_UYARI ise, çıkış işlevi neden kodu alanını geçerli bir MQRC\_ \* değerine ayarlayabiliyor.

## **C dili çağırma**

Kuyruk yöneticisi, aşağıdaki değişkenleri mantıksal olarak tanımlar:

```
MQHCONN  Hconn;          /* Connection handle */
MQLONG   Operation;     /* Operation being processed */
MQCTLO   ControlOpts;  /* Options that control the action of MQCTL */
MQLONG   CompCode;     /* Completion code */
MQLONG   Reason;       /* Reason code qualifying completion code */
```

Kuyruk yöneticisi, çıkışı mantıksal olarak aşağıdaki gibi çağırır:

```
MQ_CTL_EXIT (&Hconn, &Operation, &ControlOpts, &CompCode, &Reason);
```

Çıkışınız, aşağıdaki C işlevi prototipini eşleştirmelidir:

```
void MQENTRY MQ_CTL_EXIT (
PMQHCONN pHconn;          /* Address of connection handle */
PMQLONG  pOperation;     /* Address of operation being processed */
PMQCTLO  pControlOpts;  /* Address of options that control the action of MQCTL */
PMQLONG  pCompCode;     /* Address of completion code */
PMQLONG  pReason;       /* Address of reason code qualifying completion code */
```

### **Bağlantıyı Kes-MQ\_DISC\_EXIT**

MQ\_DISC\_EXIT, *önce* ve *bundan sonra* MQDISC çıkış işlemlerini gerçekleştirmek için bir bağlantı kesme işlevi sağlar. *önce* ve *bundan sonra* MQDISC arama çıkış işlevlerini kaydettirmek için MQXR\_AFTER ve MQXR\_AFTER çıkış nedenleriyle birlikte MQXF\_DISC işlev tanıtıcısını kullanın.

Bu işleve ilişkin arabirim

```
MQ_DISC_EXIT (&ExitParms, &ExitContext, &pHconn,  
&CompCode, &Reason);
```

parametrelerin bulunduğu yer:

### **ExitParms (MQAXP)-giriş/çıkış**

Parametre yapısından çıkın.

### **ExitContext (MQAXC)-giriş/çıkış**

Bağlam yapısından çıkın.

### **pHconn (PMQHCONN)-giriş**

Bağlantı tanıtıcısı işaretçisi.

*MQDISC çağrısı öncesi için*, bu alanın değeri aşağıdakilerden biridir:

- MQCONN ya da MQCONNX çağrısına döndürülen bağlantı tanıtıcısı
- Sıfır, ortama özgü bir bağdaştırıcının kuyruk yöneticisine bağlı olduğu ortamlar için
- Önceki bir çıkış işlevi çağrısına göre ayarlanan bir değer

*MQDISC çağrısı sonrası*, bu alanın değeri sıfır ya da önceki bir çıkış işlevi çağırımı tarafından ayarlanan bir değer.

### **CompCode (MQHOT)-giriş/çıkış**

Tamamlanma kodu, geçerli değerler şunlardır:

#### **MQCC\_OK**

İşlem başarıyla tamamlandı.

#### **MQCC\_UYARI**

Kısmi tamamlanma

#### **MQCC\_FAILED**

Arama başarısız oldu

### **Neden (MQLONG)-giriş/çıkış**

Neden kodu, tamamlanma kodunu nitelemektedir.

Tamamlanma kodu MQCC\_OK ise, tek geçerli değer şöyledir:

#### **MQRC\_NONE**

(0, x '000') Raporlamak için bir neden yok.

Tamamlanma kodu MQCC\_FAILED ya da MQCC\_UYARI ise, çıkış işlevi neden kodu alanını geçerli bir MQRC\_\* değerine ayarlayabiliyor.

## **C dili çağırma**

Kuyruk yöneticisi, aşağıdaki değişkenleri mantıksal olarak tanımlar:

```
MQAXP      ExitParms;      /* Exit parameter structure */  
MQAXC      ExitContext;    /* Exit context structure */  
PMQHCONN   pHconn;        /* Ptr to Connection handle */  
MQLONG     CompCode;      /* Completion code */  
MQLONG     Reason;        /* Reason code */
```

Kuyruk yöneticisi, çıkışı mantıksal olarak aşağıdaki gibi çağırır:

```
MQ_DISC_EXIT (&ExitParms, &ExitContext, &pHconn,  
&CompCode, &Reason);
```

Çıkışınız, aşağıdaki C işlevi prototipini eşleştirmelidir:

```
void MQENTRY MQ_DISC_EXIT (  
PMQAXP      pExitParms,    /* Address of exit parameter structure */  
PMQAXC      pExitContext,  /* Address of exit context structure */
```

PPMQHCONN	ppHconn,	/* Address of ptr to connection handle */
PMQLONG	pCompCode,	/* Address of completion code */
PMQLONG	pReason);	/* Address of reason code qualifying completion code */

## AI-MQ\_GET\_EXIT

MQ\_GET\_EXIT, *önce* ve *bundan sonra* MQGET çağrılarının işlenmesini gerçekleştirmek için bir alma çıkış işlevi sağlar.

İki işlev tanıtıcısı vardır:

1. MQXR\_BEFORE ve MQXR\_AFTER çıkış nedenleriyle *önce* ve *sonra* MQGET çağrı çıkış işlevlerinden çıkış nedenleriyle MQXF\_GET ögesini kullanın.
2. MQXF\_DATA\_CONV\_ON\_GET işlev tanıtıcısını kullanmaya ilişkin bilgi için [“MQXF\\_DATA\\_CONV\\_ON\\_GET” sayfa 1565](#) konusuna bakın.

Bu işleve ilişkin arabirim şöyledir:

```
MQ_GET_EXIT (&ExitParms, &ExitContext, &Hconn, &Hobj, &pMsgDesc,
             &pGetMsgOpts, &BufferLength, &pBuffer, &pDataLength,
             &CompCode, &Reason)
```

parametrelerin bulunduğu yer:

### ExitParms (MQAXP)-giriş/çıkış

Parametre yapısından çıkın.

### ExitContext (MQAXC)-giriş/çıkış

Bağlam yapısından çıkın.

### Hconn (MQHCONN)-giriş

Bağlantı tanıtıcısı.

### Hobj (MQHOBJ)-giriş/çıkış

Nesne tanıtıcısı.

### pMsgDesc (PMQMD)-giriş/çıkış

İşaretçiyi ileti tanımlayıcısı.

### pGetMsgOpts (PMQGMO)-giriş/çıkış

İleti seçeneklerini almak için işaretçi.

### BufferLength (MQUZN)-giriş/çıkış

İleti arabelleği uzunluğu.

### pBuffer (PMQBYTE)-giriş/çıkış

İşaretçi ileti arabelleği.

### pDataUzunluk (PMQHOT)-giriş/çıkış

Veri uzunluğu alanına işaretçi.

### CompCode (MQHOT)-giriş/çıkış

Tamamlanma kodu, geçerli değerler şunlardır:

#### MQCC\_OK

İşlem başarıyla tamamlandı.

#### MQCC\_UYARI

Kısmi tamamlama.

#### MQCC\_FAILED

Arama başarısız oldu

### Neden (MQlong)-giriş/çıkış

Neden kodu, tamamlanma kodunu nitelemektedir.

Tamamlanma kodu MQCC\_OK ise, tek geçerli değer şöyledir:

#### MQRC\_NONE

(0, x '000') Raporlamak için bir neden yok.



Tamamlanma kodu MQCC\_FAILED ya da MQCC\_UYARI ise, çıkış işlevi neden kodu alanını geçerli bir MQRC\_ \* değerine ayarlayabiliyor.

## C dili çağırma

Kuyruk yöneticisi, aşağıdaki değişkenleri mantıksal olarak tanımlar:

```
MQAXP      ExitParms;      /* Exit parameter structure */
MQAXC      ExitContext;   /* Exit context structure */
MQHCONN    Hconn;        /* Connection handle */
MQHOBJ     Hobj;         /* Object handle */
PMQMD      pMsgDesc;     /* Ptr to message descriptor */
PMQPMO     pGetMsgOpts;  /* Ptr to get message options */
MQLONG     BufferLength;  /* Message buffer length */
PMQBYTE    pBuffer;     /* Ptr to message buffer */
PMQLONG    pDataLength;  /* Ptr to data length field */
MQLONG     CompCode;    /* Completion code */
MQLONG     Reason;      /* Reason code */
```

Kuyruk yöneticisi, çıkışı mantıksal olarak aşağıdaki gibi çağırır:

```
MQ_GET_EXIT (&ExitParms, &ExitContext, &Hconn, &Hobj, &pMsgDesc,
             &pGetMsgOpts, &BufferLength, &pBuffer, &pDataLength,
             &CompCode, &Reason)
```

Çıkışınız, aşağıdaki C işlevi prototipini eşleştirmelidir:

```
void MQENTRY MQ_GET_EXIT (
PMQAXP      pExitParms,   /* Address of exit parameter structure */
PMQAXC      pExitContext, /* Address of exit context structure */
PMQHCONN    pHconn,      /* Address of connection handle */
PMQHOBJ     pHobj,       /* Address of object handle */
PPMQMD      ppMsgDesc,   /* Address of ptr to message descriptor */
PPMQGMO     ppGetMsgOpts, /* Address of ptr to get message options */
PMQLONG     pBufferLength, /* Address of message buffer length */
PPMQBYTE    ppBuffer,    /* Address of ptr to message buffer */
PPMQLONG    ppDataLength, /* Address of ptr to data length field */
PMQLONG     pCompCode,   /* Address of completion code */
PMQLONG     pReason);    /* Address of reason code qualifying
                           completion code */
```

## Kullanım notları

1. Burada açıklanan MQ\_GET\_EXIT işlev arabirimi, hem MQXF\_GET çıkış işlevi, hem de “MQXF\_DATA\_CONV\_ON\_GET” sayfa 1565 çıkış işlevi için kullanılır.

Bu iki çıkış işlevi için ayrı giriş noktaları tanımlıdır, bu nedenle *her ikisi* ' i engellemek için MQXEP çağrısının iki kez kullanılması gerekir; bu çağrı için MQXF\_GET işlev tanıtıcısı kullanın.

MQ\_GET\_EXIT arabirimi MQXF\_GET ve MQXF\_DATA\_CONV\_ON\_GET için aynı olduğu için, her ikisi için de tek bir çıkış işlevi kullanılabilir; MQAXP yapısındaki *Function* alanı, hangi çıkış işlevinin çağrıldığını gösterir. Diğer bir seçenek olarak, MQXEP çağrısı, iki vaka için farklı çıkış işlevlerini kaydetmek için kullanılabilir.

## MQXF\_DATA\_CONV\_ON\_GET

MQXF\_DATA\_CONV\_ON\_GET işlev tanıtıcısı MQ\_GET\_EXIT ile birlikte kullanılır.

Bu çağrıya ilişkin arabirimle ilgili bilgi edinmek için [MQ\\_GET\\_EXIT](#) başlıklı konuya ve örnek bir C dili bildirimine bakın.

## Kullanım notları

Kaydedilmişse, bu giriş noktası, uygulamaya gelen iletiler, ancak herhangi bir veri dönüştürmesi gerçekleşmeden önce çağrıldığında çağrılır. İleti, veri dönüştürme işlemi iletilmeden önce, API çıkışı, şifre çözme ya da açma işlemi gibi işlemleri gerçekleştirmesi gerekirse yararlı olabilir. Çıkış, gerekirse, veri

dönüştürmenin MQXCC\_SUPRESS\_FUNCTION; döndürerek atlanmasına neden olabilir; daha fazla bilgi için bakınız: MQAXP yapısı.

İstemcide bu giriş noktasına kaydolmak, veri dönüştürme işleminin istemci makinesinde yerel olarak gerçekleştirilmesine neden olur. Bu nedenle, doğru işlem için istemciye uygulama dönüştürme çıkışlarını kurmanız gerekebilir. Zamanuyumsuz olarak tüketmek için MQXF\_DATA\_CONV\_ON\_GET ' nin de kullanıldığını unutmayın.

MQ\_GET\_EXIT çağrısı kullanıldığında, bir önce MQGET veri dönüştürme çıkış işlevini kaydetmek için MQXR\_BEFORE çıkış nedeni ile MQXF\_DATA\_CONV\_ON\_GET komutunu kullanın.

MQXF\_DATA\_CONV\_ON\_GET için MQXR\_AFTER çıkış işlevi yok; MQXF\_GET için MQXR\_GET çıkış işlevi, veri dönüştürme işleminden sonra çıkış işlemleri için gereken yeteneği sağlar.

MQ\_GET\_EXIT çağrısı için ayrı giriş noktaları tanımlıdır, bu nedenle her ikisi çıkış işlevlerini engellemek için, MQXEP çağrısının iki kez kullanılması gerekir; bu çağrıya ilişkin MQXF\_DATA\_CONV\_ON\_GET işlev tanıtıcısı kullanılır.

MQ\_GET\_EXIT arabirimi MQXF\_GET ve MQXF\_DATA\_CONV\_ON\_GET için aynı olduğu için, her ikisi için de tek bir çıkış işlevi kullanılabilir; MQAXP yapısındaki *Function* alanı, hangi çıkış işlevinin çağrıldığını gösterir. Diğer bir seçenek olarak, MQXEP çağrısı, iki vaka için farklı çıkış işlevlerini kaydetmek için kullanılabilir.

### ***Kullanıma hazırlama-MQ\_INIT\_EXIT***

MQ\_INIT\_EXIT, MQAXP ' de ExitReason ayarı tarafından MQXR\_CONNECTION değerine ayarlanarak belirtilen bağlantı düzeyinde kullanıma hazırlama sağlar.

Kullanıma hazırlama sırasında aşağıdakine dikkat edin:

- The MQ\_INIT\_EXIT function calls MQXEP to register the IBM MQ API verbs and the ENTRY and EXIT points in which it is interested.
- Çıkışlar, tüm IBM MQ API fiillerinin kesişmesine gerek yoktur. Çıkış işlevleri yalnızca bir ilgi kaydedildiyse çağrılır.
- Çıkış tarafından kullanılacak saklama alanı, kullanıma hazırlanırken edinilebilir.
- Bu işleve yapılan bir çağrı başarısız olursa, bunu çağırın MQCONN ya da MQCONNX çağrısı da CompCode ile başarısız olur ve MQAXP ' deki ExitResponse alanının değerine bağlı olan neden olur.
- Şu anda doğru ortam ayarlanmamış olduğundan, bir MQ\_INIT\_EXIT çıkışı IBM MQ API çağrıları yayınlamamalıdır.
- MQXCC\_FAILED ile bir MQ\_INIT\_EXIT başarısız olursa, kuyruk yöneticisi MQCONN ya da MQCONNX çağrısından MQCC\_FAILED ve MQRC\_API\_EXIT\_ERROR ile çağrılan çağrıyı döndürür.
- Kuyruk yöneticisi, ilk MQ\_INIT\_EXIT çağrılmadan önce API çıkış işlevi yürütme ortamını kullanıma hazırlarken bir hatayla karşılaşırsa, kuyruk yöneticisi MQCONN ya da MQCONN\_EXIT çağrısını MQCC\_FAILED ve MQRC\_API\_EXIT\_INIT\_ERROR ile çağırın MQCONNX ya da MQCONNX çağrısından geri döner.

Arabirimin MQ\_INIT\_EXIT değeri:

```
MQ_INIT_EXIT (&ExitParms, &ExitContext, &CompCode, &Reason)
```

parametrelerin bulunduğu yer:

#### **ExitParms (MQAXP)-giriş/çıkış**

Parametre yapısından çıkın.

#### **ExitContext (MQAXC)-giriş/çıkış**

Bağlam yapısından çıkın.

#### **CompCode (MQHOT)-giriş/çıkış**

Tamamlanma koduna ilişkin gösterge, geçerli değerler şunlardır:

#### **MQCC\_OK**

İşlem başarıyla tamamlandı.

## **MQCC\_UYARI**

Kısmi tamamlama.

## **MQCC\_FAILED**

Arama başarısız oldu

## **Neden (MQlong)-giriş/çıkış**

Neden kodunun tamamlanma kodunu niteleyen işaretçi.

Tamamlanma kodu MQCC\_OK ise, tek geçerli değer şöyledir:

## **MQRC\_NONE**

(0, x '000') Raporlamak için bir neden yok.

Tamamlanma kodu MQCC\_FAILED ya da MQCC\_UYARI ise, çıkış işlevi neden kodu alanını geçerli bir MQRC\_\* değerine ayarlayabiliyor.

Uygulamaya döndürülen CompCode ve Neden, MQAXP ' de ExitResponse alanının değerine bağlıdır.

## **C dili çağırma**

Kuyruk yöneticisi, aşağıdaki değişkenleri mantıksal olarak tanımlar:

```
MQAXP      ExitParms;      /* Exit parameter structure */
MQAXC      ExitContext;    /* Exit context structure */
MQLONG     CompCode;      /* Completion code */
MQLONG     Reason;        /* Reason code */
```

Kuyruk yöneticisi, çıkışı mantıksal olarak aşağıdaki gibi çağırır:

```
MQ_INIT_EXIT (&ExitParms, &ExitContext, &CompCode, &Reason)
```

Çıkışınız, aşağıdaki C işlevi prototipini eşleştirmelidir:

```
void MQENTRY MQ_INIT_EXIT (
PMQAXP     pExitParms,     /* Address of exit parameter structure */
PMQAXC     pExitContext,   /* Address of exit context structure */
PMQLONG    pCompCode,     /* Address of completion code */
PMQLONG    pReason);      /* Address of reason code qualifying
                             completion code */
```

## **Kullanım notları**

1. MQ\_INIT\_EXIT işlevi, durdurulacak belirli MQ çağrılarına ilişkin çıkış işlevlerinin adreslerini kaydettirmek için MQXEP çağrısını yayınlatabilir. Tüm MQ çağrılarının önünü kesmek ya da hem MQXR\_BEFORE hem de MQXR\_AFTER çağrıları durdurmak için gerekli değildir. Örneğin, bir çıkış takımı MQPUT ' un yalnızca MQXR\_BEFORE çağrısını engellemeye karar verebilirdi.
2. Çıkış takımındaki çıkış işlevleri tarafından kullanılacak saklama alanı, MQ\_INIT\_EXIT işlevi tarafından edinilebilir. Diğer bir seçenek olarak, çıkış işlevleri, gerektiğinde ve gerektiğinde depolama alanı edinebilir. Ancak, çıkış takımı sonlandırılmadan önce tüm saklama alanı serbest bırakılmalıdır; MQ\_TERM\_EXIT işlevi depolama alanını serbest bırakmaz ya da daha önce çağrılan bir çıkış işlevi olabilir.
3. MQ\_INIT\_EXIT, MQAXP ' nin ExitResponse alanında MQXCC\_FAILED değerini döndürürse ya da başka bir şekilde başarısız olursa, **CompCode** ve **Reason** değiştirgelerinin uygun değerlere ayarlanmasına neden olan MQCONN ya da MQCONNX çağrısının çağrılmasına neden olan MQCONN ya da MQCONNX çağrısı da başarısız olur.
4. An MQ\_INIT\_EXIT function cannot issue MQ calls other than MQXEP.

## **Sorgu-MQ\_INQ\_EXIT**

MQ\_INQ\_EXIT, *önce* ve *bundan sonra* MQINQ çağrılarının işlenmesini gerçekleştirmek için bir sorgu çıkış işlevi sağlar. MQXR\_INQ işlev tanıtıcısını kullanarak, MQXR\_BEFORE ve MQXR\_AFTER çıkış nedenlerini kullanarak *önce* ve *sonra* MQINQ çağrı çıkış işlevlerinden birini kaydettirin.

Bu işleve ilişkin arabirim şöyledir:

```
MQ_INQ_EXIT (&ExitParms, &ExitContext, &Hconn, &Hobj, &SelectorCount,  
&pSelectors, &IntAttrCount, &pIntAttrs, &CharAttrLength,  
&pCharAttrs, &CompCode, &Reason)
```

parametrelerin bulunduğu yer:

### **ExitParms (MQAXP)-giriş/çıkış**

Parametre yapısından çıkın.

### **ExitContext (MQAXC)-giriş/çıkış**

Bağlam yapısından çıkın.

### **Hconn (MQHCONN)-giriş**

Bağlantı tanıtıcısı.

### **Hobj (MQHOBJ)-giriş**

Nesne tanıtıcısı.

### **SelectorCount (MQUZE)-giriş**

Seçicilerin sayısı

### **pSelectors (PMQLONG)-giriş/çıkış**

Seçici değerleri dizisine işaretçi.

### **IntAttrSayı (MQUZE)-giriş**

Tamsayı özniteliklerinin sayısı.

### **pIntAttrs (PMQLong)-giriş/çıkış**

Tamsayı öznitelik değerleri dizisine işaretçi.

### **CharAttrUzunluk (MQUZE)-giriş/çıkış**

Karakter öznitelikleri dizisi uzunluğu.

### **pCharAttrs (PMQCHAR)-giriş/çıkış**

İşaretçi karakter öznitelikleri dizisine.

### **CompCode (MQHOT)-giriş/çıkış**

Tamamlanma kodu, geçerli değerler şunlardır:

#### **MQCC\_OK**

İşlem başarıyla tamamlandı.

#### **MQCC\_UYARI**

Kısmi tamamlama.

#### **MQCC\_FAILED**

Arama başarısız oldu

### **Neden (MQlong)-giriş/çıkış**

Neden kodu, tamamlanma kodunu nitelemektedir.

Tamamlanma kodu MQCC\_OK ise, tek geçerli değer şöyledir:

#### **MQRC\_NONE**

(0, x '000') Raporlamak için bir neden yok.

Tamamlanma kodu MQCC\_FAILED ya da MQCC\_UYARI ise, çıkış işlevi neden kodu alanını geçerli bir MQRC\_\* değerine ayarlayabiliyor.

## **C dili çağırma**

Kuyruk yöneticisi, aşağıdaki değişkenleri mantıksal olarak tanımlar:

```

MQAXP    ExitParms;        /* Exit parameter structure */
MQAXC    ExitContext;     /* Exit context structure */
MQHCONN  Hconn;          /* Connection handle */
MQHOBJ   Hobj;           /* Object handle */
MQLONG   SelectorCount;  /* Count of selectors */
PMQLONG  pSelectors;     /* Ptr to array of attribute selectors */
MQLONG   IntAttrCount;   /* Count of integer attributes */
PMQLONG  pIntAttrs;     /* Ptr to array of integer attributes */
MQLONG   CharAttrLength; /* Length of char attributes array */
PMQCHAR  pCharAttrs;    /* Ptr to character attributes */
MQLONG   CompCode;      /* Completion code */
MQLONG   Reason;        /* Reason code qualifying completion code */

```

Kuyruk yöneticisi, çıkışı mantıksal olarak aşağıdaki gibi çağırır:

```

MQ_INQ_EXIT (&ExitParms, &ExitContext, &Hconn, &Hobj, &SelectorCount,
             &pSelectors, &IntAttrCount, &pIntAttrs, &CharAttrLength,
             &pCharAttrs, &CompCode, &Reason)

```

Çıkışınız, aşağıdaki C işlevi prototipini eşleştirmelidir:

```

void MQENTRY MQ_INQ_EXIT (
PMQAXP    pExitParms,        /* Address of exit parameter structure */
PMQAXC    pExitContext,     /* Address of exit context structure */
PMQHCONN  pHconn,          /* Address of connection handle */
PMQHOBJ   pHobj,           /* Address of object handle */
PMQLONG   pSelectorCount,   /* Address of selector count */
PPMQLONG  ppSelectors,     /* Address of ptr to array of selectors */
PMQLONG   pIntAttrCount;   /* Address of count of integer attributes */
PPMQLONG  ppIntAttrs,     /* Address of ptr to array of integer attributes */
PMQLONG   pCharAttrLength, /* Address of character attribute length */
PPMQCHAR  ppCharAttrs,    /* Address of ptr to character attributes array */
PMQLONG   pCompCode,      /* Address of completion code */
PMQLONG   pReason);       /* Address of reason code qualifying completion
                           code */

```

### **Aç-MQ\_OPEN\_EXIT**

MQ\_OPEN\_EXIT, *önce* ve *bundan sonra* MQOPEN çağrı işlemlerini gerçekleştirmek için açık bir çıkış işlevi sağlar. *önce* ve *bundan sonra* MQOPEN çağrı çıkış işlevlerini kaydettirmek için MQXR\_AFTER ve MQXR\_AFTER çıkış nedenleriyle MQXF\_OPEN işlev tanıtıcısını kullanın.

Bu işleve ilişkin arabirim

```

MQ_OPEN_EXIT (&ExitParms, &ExitContext, &Hconn, &pObjDesc, &Options,
             &pHobj, &CompCode, &Reason)

```

parametrelerin bulunduğu yer:

#### **ExitParms (MQAXP)-giriş/çıkış**

Parametre yapısından çıkın.

#### **ExitContext (MQAXC)-giriş/çıkış**

Bağlam yapısından çıkın.

#### **Hconn (MQHCONN)-giriş**

Bağlantı tanıtıcısı.

#### **pObjDesc (PMQOD)-giriş/çıkış**

Nesne tanımlayıcısına ilişkin gösterge.

#### **Seçenekler (MQUZE)-giriş/çıkış**

Seçenekleri açın.

#### **pHobj (PMQHOBj)-giriş**

Nesne tanıtıcısı işaretçisi.

#### **CompCode (MQHOT)-giriş/çıkış**

Tamamlanma kodu, geçerli değerler şunlardır:

## **MQCC\_OK**

İşlem başarıyla tamamlandı.

## **MQCC\_UYARI**

Kısmi tamamlanma

## **MQCC\_FAILED**

Arama başarısız oldu

## **Neden (MQlong)-giriş/çıkış**

Neden kodu, tamamlanma kodunu nitelemektedir.

Tamamlanma kodu MQCC\_OK ise, tek geçerli değer şöyledir:

## **MQRC\_NONE**

(0, x '000') Raporlamak için bir neden yok.

Tamamlanma kodu MQCC\_FAILED ya da MQCC\_UYARI ise, çıkış işlevi neden kodu alanını geçerli bir MQRC\_\* değerine ayarlayabiliyor.

## **C dili çağırma**

Kuyruk yöneticisi, aşağıdaki değişkenleri mantıksal olarak tanımlar:

```
MQAXP      ExitParms;      /* Exit parameter structure */
MQAXC      ExitContext;    /* Exit context structure */
MQHCONN     Hconn;         /* Connection handle */
PMQOD      pObjDesc;      /* Ptr to object descriptor */
MQLONG     Options;        /* Open options */
PMQHOBJS   pHobj;         /* Ptr to object handle */
MQLONG     CompCode;      /* Completion code */
MQLONG     Reason;        /* Reason code */
```

Kuyruk yöneticisi, çıkışı mantıksal olarak aşağıdaki gibi çağırır:

```
MQ_OPEN_EXIT (&ExitParms, &ExitContext, &Hconn, &pObjDesc, &Options,
              &pHobj, &CompCode, &Reason);
```

Çıkışınız, aşağıdaki C işlevi prototipini eşleştirmelidir:

```
void MQENTRY MQ_OPEN_EXIT (
PMQAXP      pExitParms,    /* Address of exit parameter structure */
PMQAXC      pExitContext,  /* Address of exit context structure */
PMQHCONN     pHconn,       /* Address of connection handle */
PPMQOD      ppObjDesc,    /* Address of ptr to object descriptor */
PMQLONG     pOptions,      /* Address of open options */
PPMHOBJS    ppHobj,        /* Address of ptr to object handle */
PMQLONG     pCompCode,     /* Address of completion code */
PMQLONG     pReason);     /* Address of reason code qualifying
                             completion code */
```

## **Put-MQ\_PUT\_EXIT**

MQ\_PUT\_EXIT, *önce ve bundan sonra* MQPUT çağrı işlemlerini gerçekleştirmek için bir put exit işlevi sağlar. *önce ve bundan sonra* MQPUT çağrı çıkış işlevlerini kaydettirmeden önce MQXR\_PUT ve MQXR\_AFTER çıkış nedenleriyle MQXF\_PUT işlev tanıtıcısını kullanın.

Bu işleve ilişkin arabirim şöyledir:

```
MQ_PUT_EXIT (&ExitParms, &ExitContext, &Hconn, &Hobj, &pMsgDesc,
            &pPutMsgOpts, &BufferLength, &pBuffer, &CompCode, &Reason)
```

parametrelerin bulunduğu yer:

## **ExitParms (MQAXP)-giriş/çıkış**

Parametre yapısından çıkın.

### **ExitContext (MQAXC)-giriş/çıkış**

Bağlam yapısından çıkın.

### **Hconn (MQHCONN)-giriş**

Bağlantı tanıtıcısı.

### **Hobj (MQHOBJ)-giriş/çıkış**

Nesne tanıtıcısı.

### **pMsgDesc (PMQMD)-giriş/çıkış**

İşaretçiyi ileti tanımlayıcısı.

### **pPutMsgOpts (PMQPMO)-giriş/çıkış**

İleti seçeneklerini işaretlemek için işaretçi.

### **BufferLength (MQUZN)-giriş/çıkış**

İleti arabelleği uzunluğu.

### **pBuffer (PMQBYTE)-giriş/çıkış**

İşaretçi ileti arabelleği.

### **CompCode (MQHOT)-giriş/çıkış**

Tamamlanma kodu, geçerli değerler şunlardır:

#### **MQCC\_OK**

İşlem başarıyla tamamlandı.

#### **MQCC\_UYARI**

Kısmi tamamlama.

#### **MQCC\_FAILED**

Arama başarısız oldu

### **Neden (MQlong)-giriş/çıkış**

Neden kodu, tamamlanma kodunu nitelemektedir.

Tamamlanma kodu MQCC\_OK ise, tek geçerli değer şöyledir:

#### **MQRC\_NONE**

(0, x '000') Raporlamak için bir neden yok.

Tamamlanma kodu MQCC\_FAILED ya da MQCC\_UYARI ise, çıkış işlevi neden kodu alanını geçerli bir MQRC\_ \* değerine ayarlayabiliyor.

## **C dili çağırma**

Kuyruk yöneticisi, aşağıdaki değişkenleri mantıksal olarak tanımlar:

```
MQAXP      ExitParms;      /* Exit parameter structure */
MQAXC      ExitContext;    /* Exit context structure */
MQHCONN    Hconn;         /* Connection handle */
MQHOBJ     Hobj;          /* Object handle */
PMQMD      pMsgDesc;      /* Ptr to message descriptor */
PMQPMO     pPutMsgOpts;   /* Ptr to put message options */
MQLONG     BufferLength;   /* Message buffer length */
PMQBYTE    pBuffer;       /* Ptr to message data */
MQLONG     CompCode;      /* Completion code */
MQLONG     Reason;        /* Reason code */
```

Kuyruk yöneticisi, çıkışı mantıksal olarak aşağıdaki gibi çağırır:

```
MQ_PUT_EXIT (&ExitParms, &ExitContext, &Hconn, &Hobj, &pMsgDesc,
             &pPutMsgOpts, &BufferLength, &pBuffer, &CompCode, &Reason)
```

Çıkışınız, aşağıdaki C işlevi prototipini eşleştirmelidir:

```
void MQENTRY MQ_PUT_EXIT (
MQAXP      pExitParms,     /* Address of exit parameter structure */
MQAXC      pExitContext,  /* Address of exit context structure */
```

```

PMQHCONN      pHConn,          /* Address of connection handle */
PMQHOBJS      pHObj,          /* Address of object handle */
PPMQMD        pMsgDesc,      /* Address of ptr to message descriptor */
PPMQPMO       ppPutMsgOpts,  /* Address of ptr to put message options */
PMQLONG       pBufferLength, /* Address of message buffer length */
PMQBYTE       pBuffer,      /* Address of ptr to message buffer */
PMQLONG       pCompCode,     /* Address of completion code */
PMQLONG       pReason);     /* Address of reason code qualifying
                             completion code */

```

## Kullanım notları

- Kuyruk yöneticisi tarafından oluşturulan rapor iletileri olağan çağrı işlemini atlıyor. Sonuç olarak, bu tür iletiler MQ\_PUT\_EXIT işlevi ya da MQPUT1 işlevi tarafından engellenemez. Ancak, ileti kanalı aracı tarafından oluşturulan rapor iletileri olağan bir şekilde işlenir ve bu nedenle MQ\_PUT\_EXIT işlevi ya da MQ\_PUT1\_EXIT işlevi tarafından kesilebilir. MCA tarafından oluşturulan tüm rapor iletilerini ele geçirdiğinizden emin olmak için, hem MQ\_PUT\_EXIT, hem de MQ\_PUT1\_EXIT MQ\_PUT1\_EXITusedkullanılmalıdır.

### Put1 - MQ\_PUT1\_EXIT

MQ\_PUT1\_EXIT , *önce* ve *sonra* MQPUT1 arama işlemini gerçekleştirmek için bir *tek ileti koyma* çıkış işlevini sağlar. Use function identifier MQXF\_PUT1 with exit reasons MQXR\_BEFORE and MQXR\_AFTER to register *önce* and *bundan sonra* MQPUT1 call exit functions.

Bu işleve ilişkin arabirim şöyledir:

```

MQ_PUT1_EXIT (&ExitParms, &ExitContext, &Hconn, &pObjDesc, &pMsgDesc,
&pPutMsgOpts, &BufferLength, &pBuffer, &CompCode, &Reason)

```

parametrelerin bulunduğu yer:

#### ExitParms (MQAXP)-giriş/çıkış

Parametre yapısından çıkın.

#### ExitContext (MQAXC)-giriş/çıkış

Bağlam yapısından çıkın.

#### Hconn (MQHCONN)-giriş

Bağlantı tanıtıcısı.

#### pObjDesc (PMQOD)-giriş/çıkış

Nesne tanımlayıcısına ilişkin gösterge.

#### pMsgDesc (PMQMD)-giriş/çıkış

İşaretçiyi ileti tanımlayıcısı.

#### pPutMsgOpts (PMQPMO)-giriş/çıkış

İleti seçeneklerini işaretlemek için işaretçi.

#### BufferLength (MQUZN)-giriş/çıkış

İleti arabelleği uzunluğu.

#### pBuffer (PMQBYTE)-giriş/çıkış

İşaretçi ileti arabelleği.

#### CompCode (MQHOT)-giriş/çıkış

Tamamlanma kodu, geçerli değerler şunlardır:

##### MQCC\_OK

İşlem başarıyla tamamlandı.

##### MQCC\_UYARI

Kısmi tamamlama.

##### MQCC\_FAILED

Arama başarısız oldu



## Neden (MQlong)-giriş/çıkış

Neden kodu, tamamlanma kodunu nitelemektedir.

Tamamlanma kodu MQCC\_OK ise, tek geçerli değer şöyledir:

### MQRC\_NONE

(0, x '000') Raporlamak için bir neden yok.

Tamamlanma kodu MQCC\_FAILED ya da MQCC\_UYARI ise, çıkış işlevi neden kodu alanını geçerli bir MQRC\_\* değerine ayarlayabiliyor.

## C dili çağırma

Kuyruk yöneticisi, aşağıdaki değişkenleri mantıksal olarak tanımlar:

```
MQAXP      ExitParms;      /* Exit parameter structure */
MQAXC      ExitContext;   /* Exit context structure */
MQHCONN    Hconn;        /* Connection handle */
PMQOD      pObjDesc;     /* Ptr to object descriptor */
PMQMD      pMsgDesc;     /* Ptr to message descriptor */
PMQPMO     pPutMsgOpts;  /* Ptr to put message options */
MQLONG     BufferLength;  /* Message buffer length */
PMQBYTE    pBuffer;     /* Ptr to message data */
MQLONG     CompCode;     /* Completion code */
MQLONG     Reason;      /* Reason code */
```

Kuyruk yöneticisi, çıkışı mantıksal olarak aşağıdaki gibi çağırır:

```
MQ_PUT1_EXIT (&ExitParms, &ExitContext, &Hconn, &pObjDesc, &pMsgDesc,
              &pPutMsgOpts, &BufferLength, &pBuffer, &CompCode, &Reason)
```

Çıkışınız, aşağıdaki C işlevi prototipini eşleştirmelidir:

```
void MQENTRY MQ_PUT1_EXIT (
PMQAXP      pExitParms,      /* Address of exit parameter structure */
PMQAXC      pExitContext,   /* Address of exit context structure */
PMQHCONN    pHconn,        /* Address of connection handle */
PPMQOD      ppObjDesc,     /* Address of ptr to object descriptor */
PPMQMD      ppMsgDesc,     /* Address of ptr to message descriptor */
PPMQPMO     ppPutMsgOpts,  /* Address of ptr to put message options */
PMQLONG     pBufferLength,  /* Address of message buffer length */
PPMQBYTE    ppBuffer,     /* Address of ptr to message buffer */
PMQLONG     pCompCode,     /* Address of completion code */
PMQLONG     pReason);     /* Address of reason code qualifying
                           completion code */
```

## Ayarla-MQ\_SET\_EXIT

MQ\_SET\_EXIT, önce ve bundan sonra MQSET arama işlemlerini gerçekleştirmek için bir set çıkış işlevi sağlar. önce ve bundan sonra MQSET çağrı çıkış işlevlerini kaydettirmek için MQXR\_BEFORE ve MQXR\_AFTER çıkış nedenleriyle MQXF\_SET işlev tanıtıcısını kullanın.

Bu işleve ilişkin arabirim şöyledir:

```
MQ_SET_EXIT (&ExitParms, &ExitContext, &Hconn, &Hobj, &SelectorCount,
             &pSelectors, &IntAttrCount, &pIntAttrs, &CharAttrLength,
             &pCharAttr, &CompCode, &Reason)
```

parametrelerin bulunduğu yer:

### ExitParms (MQAXP)-giriş/çıkış

Parametre yapısından çıkın.

### ExitContext (MQAXC)-giriş/çıkış

Bağlam yapısından çıkın.

**Hconn (MQHCONN)-giriş**

Bağlantı tanıtıcısı.

**Hobj (MQHOBJ)-giriş**

Nesne tanıtıcısı.

**SelectorCount (MQUZE)-giriş**

Seçicilerin sayısı

**pSelectors (PMQLONG)-giriş/çıkış**

Seçici değerleri dizisine işaretçi.

**IntAttrSayı (MQUZE)-giriş**

Tamsayı özniteliklerinin sayısı.

**pIntAttr (PMQLong)-giriş/çıkış**

Tamsayı öznitelik değerleri dizisine işaretçi.

**CharAttrUzunluk (MQUZE)-giriş/çıkış**

Karakter öznitelikleri dizisi uzunluğu.

**pCharAttr (PMQCHAR)-giriş/çıkış**

Karakter özniteliği değerlerini gösteren gösterge.

**CompCode (MQHOT)-giriş/çıkış**

Tamamlanma kodu, geçerli değerler şunlardır:

**MQCC\_OK**

İşlem başarıyla tamamlandı.

**MQCC\_UYARI**

Kısmi tamamlama.

**MQCC\_FAILED**

Arama başarısız oldu

**Neden (MQlong)-giriş/çıkış**

Neden kodu, tamamlanma kodunu nitelemektedir.

Tamamlanma kodu MQCC\_OK ise, tek geçerli değer şöyledir:

**MQRC\_NONE**

(0, x '000') Raporlamak için bir neden yok.

Tamamlanma kodu MQCC\_FAILED ya da MQCC\_UYARI ise, çıkış işlevi neden kodu alanını geçerli bir MQRC\_\* değerine ayarlayabiliyor.

**C dili çağırma**

Kuyruk yöneticisi, aşağıdaki değişkenleri mantıksal olarak tanımlar:

```
MQAXP      ExitParms;          /* Exit parameter structure */
MQAXC      ExitContext;       /* Exit context structure */
MQHCONN    Hconn;            /* Connection handle */
MQHOBJ     Hobj;             /* Object handle */
MQLONG     SelectorCount;    /* Count of selectors */
PMQLONG    pSelectors;       /* Ptr to array of attribute selectors */
MQLONG     IntAttrCount;     /* Count of integer attributes */
PMQLONG    pIntAttr;        /* Ptr to array of integer attributes */
MQLONG     CharAttrLength;   /* Length of char attributes array */
PMQCHAR    pCharAttr;       /* Ptr to character attributes */
MQLONG     CompCode;        /* Completion code */
MQLONG     Reason;          /* Reason code qualifying completion code */
```

Kuyruk yöneticisi, çıkışı mantıksal olarak aşağıdaki gibi çağırır:

```
MQ_SET_EXIT (&ExitParms, &ExitContext, &Hconn, &Hobj, &SelectorCount,
             &pSelectors, &IntAttrCount, &pIntAttr, &CharAttrLength,
             &pCharAttr, &CompCode, &Reason)
```

Çıkışınız, aşağıdaki C işlevi prototipini eşleştirmelidir:

```
void MQENTRY MQ_SET_EXIT (
PMQAXP    pExitParms,        /* Address of exit parameter structure */
PMQAXC    pExitContext,     /* Address of exit context structure */
PMQHCONN  pHconn,          /* Address of connection handle */
PMQHOBJS  pHobj,           /* Address of object handle */
PMLONG    pSelectorCount,   /* Address of selector count */
PMLONG    ppSelectors,      /* Address of ptr to array of selectors */
PMLONG    pIntAttrCount;    /* Address of count of integer attributes */
PMLONG    ppIntAttrs,       /* Address of ptr to array of integer attributes */
PMLONG    pCharAttrLength,  /* Address of character attribute length */
PPMCHAR   ppCharAttrs,     /* Address of ptr to character attributes array */
PMLONG    pCompCode,        /* Address of completion code */
PMLONG    pReason);        /* Address of reason code qualifying completion
                             code */
```

### **Durum-MQ\_STAT\_EXIT**

MQ\_STAT\_EXIT, *önce* ve *bundan sonra* MQSTAT çağrı işlemlerini gerçekleştirmek için bir durum çıkışı işlevi sağlar. *önce* ve *bundan sonra* MQSTAT çağrı çıkış işlevlerini kaydettirmek için MQXR\_STAT ve MQXR\_AFTER çıkış nedenlerini kullanarak MQXF\_STAT işlev tanıtıcısını kullanın.

Bu işleve ilişkin arabirim şöyledir:

```
MQ_STAT_EXIT (&ExitParms, &ExitContext, &Hconn, &Type, &pStatus
              &CompCode, &Reason)
```

parametrelerin bulunduğu yer:

#### **ExitParms (MQAXP)-giriş/çıkış**

Parametre yapısından çıkın.

#### **ExitContext (MQAXC)-giriş/çıkış**

Bağlam yapısından çıkın.

#### **Hconn (MQHCONN)-giriş**

Bağlantı tanıtıcısı.

#### **Tip (MQHOT)-giriş**

Alınacak durum bilgileri tipi.

#### **pStatus (PMQSTS)-çıkış**

İşaretçiyi durum arabelleği.

#### **CompCode (MQHOT)-giriş/çıkış**

Tamamlanma kodu, geçerli değerler şunlardır:

##### **MQCC\_OK**

İşlem başarıyla tamamlandı.

##### **MQCC\_UYARI**

Kısmi tamamlama.

##### **MQCC\_FAILED**

Arama başarısız oldu

#### **Neden (MQlong)-giriş/çıkış**

Neden kodu, tamamlanma kodunu nitelemektedir.

Tamamlanma kodu MQCC\_OK ise, tek geçerli değer şöyledir:

##### **MQRC\_NONE**

(0, x '000') Raporlamak için bir neden yok.

Tamamlanma kodu MQCC\_FAILED ya da MQCC\_UYARI ise, çıkış işlevi neden kodu alanını geçerli bir MQRC\_\* değerine ayarlayabiliyor.

## C dili çağırma

Çıkışınız, aşağıdaki C işlevi prototipini eşleştirmelidir:

```
void MQENTRY MQ_STAT_EXIT (
PMQAXP    pExitParms,        /* Address of exit parameter structure */
PMQAXC    pExitContext,     /* Address of exit context structure */
PMQHCONN  pHconn,          /* Address of connection handle */
PMQLONG   pType,            /* Address of status type */
PPMQSTS   ppStatus,        /* Address of status buffer */
PMQLONG   pCompCode,       /* Address of completion code */
PMQLONG   pReason);       /* Address of reason code qualifying completion
                           code */
```

### **Sonlandırma-MQ\_TERM\_EXIT**

MQ\_TERM\_EXIT, MQXF\_TERM ve ExitReason MQXR\_CONNECTION işlev tanıtıcısına sahip bir bağlantı düzeyi sonlandırıcı sağlar. Kaydedilmişse, her bağlantı kesme isteği için MQ\_TERM\_EXIT bir kez çağrılır.

Sona erdirme işleminin bir parçası olarak, çıkış için artık gerekli olmayan saklama alanı serbest bırakılabilir ve gereken tüm temizleme işlemleri gerçekleştirilebilir.

Bir MQ\_TERM\_EXIT, MQXCC\_FAILED ile başarısız olursa, kuyruk yöneticisi bunu MQDISC 'den (MQCC\_FAILED ve MQRC\_API\_EXIT\_ERROR) çağrılan MQDISC' den geri döndürür.

Kuyruk yöneticisi, son MQ\_TERM\_EXIT ' i çağırdıktan sonra API çıkış işlevi yürütme ortamını sonlandırırken bir hatayla karşılaşır, MQDIS\_EXIT çağrılan MQDISC çağrısından MQCC\_FAILED ve MQRC\_API\_EXIT\_TERM\_ERROR içeren MQDISC çağrısından geri döner.

Bu işleve ilişkin arabirim şöyledir:

```
MQ_TERM_EXIT (&ExitParms, &ExitContext, &CompCode, &Reason)
```

parametrelerin bulunduğu yer:

#### **ExitParms (MQAXP)-giriş/çıkış**

Parametre yapısından çıkın.

#### **ExitContext (MQAXC)-giriş/çıkış**

Bağlam yapısından çıkın.

#### **CompCode (MQHOT)-giriş/çıkış**

Tamamlanma kodu, geçerli değerler şunlardır:

##### **MQCC\_OK**

İşlem başarıyla tamamlandı.

##### **MQCC\_FAILED**

Arama başarısız oldu

#### **Neden (MQlong)-giriş/çıkış**

Neden kodu, tamamlanma kodunu nitelemektedir.

Tamamlanma kodu MQCC\_OK ise, tek geçerli değer şöyledir:

##### **MQRC\_NONE**

(0, x '000') Raporlamak için bir neden yok.

Tamamlanma kodu MQCC\_FAILED ise, çıkış işlevi neden kodu alanını geçerli bir MQRC\_ \* değerine ayarlayabilir.

Uygulamaya döndürülen CompCode ve Neden, MQAXP ' de ExitResponse alanının değerine bağlıdır.

## C dili çağırma

Kuyruk yöneticisi, aşağıdaki değişkenleri mantıksal olarak tanımlar:

```
MQAXP      ExitParms;      /* Exit parameter structure */
MQAXC      ExitContext;  /* Exit context structure */
MQQLONG    CompCode;     /* Completion code */
MQQLONG    Reason;       /* Reason code */
```

Kuyruk yöneticisi, çıkışı mantıksal olarak aşağıdaki gibi çağırır:

```
MQ_TERM_EXIT (&ExitParms, &ExitContext, &CompCode, &Reason)
```

Çıkışınız, aşağıdaki C işlevi prototipini eşleştirmelidir:

```
void MQENTRY MQ_TERM_EXIT (
  PMQAXP      pExitParms,      /* Address of exit parameter structure */
  PMQAXC      pExitContext,    /* Address of exit context structure */
  PMQQLONG    pCompCode,       /* Address of completion code */
  PMQQLONG    pReason);        /* Address of reason code qualifying
                                completion code */
```

## Kullanım notları

1. MQ\_TERM\_EXIT işlevi isteğe bağlıdır. Yapılacak bir sonlandırma işlemi yoksa, bir çıkış takımının sonlandırma çıkışını kaydettirmesi gerekli değildir.

Bağlantı sırasında çıkış takımına ait işlevler bağlantı sırasında kaynakları edinirse, MQ\_TERM\_EXIT işlevi, bu kaynakların serbest bırakılacağı, örneğin, depolamanın dinamik olarak elde edilen saklama alanı için uygun olduğu bir noktadır.

2. MQDISC çağrısı yayınlandığında bir MQ\_TERM\_EXIT işlevi kaydedildiyse, tüm MQDISC çıkış işlevleri çağrıldıktan sonra çıkış işlevi çağrılır.
3. MQ\_TERM\_EXIT, MQAXP 'nin ExitResponse alanında MQXCC\_FAILED değerini döndürürse ya da başka bir şekilde başarısız olursa, **CompCode** ve **Reason** değıştirgelerinin uygun değerlere ayarlanmasına neden olan MQ\_TERM\_EXIT' in çağrılmasına neden olan MQDISC çağrısı da başarısız olur.

## Aboneliği kaydet-MQ\_SUB\_EXIT

MQ\_SUB\_EXIT, *önce* ve *bundan sonra* aboneliği yeniden kayıt işlemlerini gerçekleştirmek için bir çıkış işlevi sağlar. *önce* ve *bundan sonra* abonelik kayıt çağrısı çıkış işlevlerini kaydettirmek için MQXR\_AFTER ve MQXR\_AFTER çıkış nedenleriyle MQXF\_SUB işlev tanıtıcısını kullanın.

Bu işleve ilişkin arabirim şöyledir:

```
MQ_SUB_EXIT (&ExitParms, &ExitContext, &Hconn, &pSubDesc, &pHobj, &pHsub, &CompCode, &Reason)
```

parametrelerin bulunduğu yer:

### ExitParms (MQAXP)-giriş/çıkış

Parametre yapısından çıkın.

### ExitContext (MQAXC)-giriş/çıkış

Bağlam yapısından çıkın.

### Hconn (MQHCONN)-giriş/çıkış

Bağlantı tanıtıcısı.

### pSubDesc-giriş/çıkış

Öznitelik seçicilerinin dizisi.

### pHobj -giriş/çıkış

Nesne tanıtıcısı

### pHsub (MQHOBJ) giriş/çıkış

Abonelik tanıtıcısı

### CompCode (MQHOT)-giriş/çıkış

Tamamlanma kodu, geçerli değerler şunlardır:

#### MQCC\_OK

İşlem başarıyla tamamlandı.

#### MQCC\_UYARI

Kısmi tamamlama.

#### MQCC\_FAILED

Arama başarısız oldu

### Neden (MQlong)-giriş/çıkış

Neden kodu, tamamlanma kodunu nitelemektedir.

Tamamlanma kodu MQCC\_OK ise, tek geçerli değer şöyledir:

#### MQRC\_NONE

(0, x '000') Raporlamak için bir neden yok.

Tamamlanma kodu MQCC\_FAILED ya da MQCC\_UYARI ise, çıkış işlevi neden kodu alanını geçerli bir MQRC\_\* değerine ayarlayabiliyor.

## C dili çağırma

Kuyruk yöneticisi, aşağıdaki değişkenleri mantıksal olarak tanımlar:

```
MQAXP    ExitParms;      /* Exit parameter structure */
MQAXC    ExitContext;   /* Exit context structure */
MQHCONN  Hconn;         /* Connection handle */
PMQSD    pSubDesc;     /* Subscription descriptor */
PMQHOBJS pHobj;        /* Object Handle */
PMQHOBJS pHsub;        /* Subscription handle */
MQLONG   CompCode;     /* Completion code */
MQLONG   Reason;       /* Reason code qualifying completion code */
```

Kuyruk yöneticisi, çıkışı mantıksal olarak aşağıdaki gibi çağırır:

```
MQ_SUB_EXIT (&ExitParms, &ExitContext, &Hconn, &pSubDesc, &pHobj, &pHsub,
             &CompCode, &Reason);
```

Çıkışınız, aşağıdaki C işlevi prototipini eşleştirmelidir:

```
PMQAXP    pExitParms;    /* Exit parameter structure */
PMQAXC    pExitContext;  /* Exit context structure */
PMQHCONN  pHconn;       /* Connection handle */
PPMQSD    ppSubDesc;    /* Subscription descriptor */
PPMQHOBJS ppHobj;       /* Object Handle */
PPMQHOBJS ppHsub;       /* Subscription handle */
PMQLONG   pCompCode;    /* Completion code */
PMQLONG   pReason;      /* Reason code qualifying completion code */
```

### Abonelik isteği-MQ\_SUBRQ\_EXIT

MQ\_SUBRQ\_EXIT, *önce ve bundan sonra* abonelik isteği işlemlerini gerçekleştirmek için bir abonelik isteği çıkış işlevi sağlar. *önce ve bundan sonra* abonelik isteği çağırısı çıkış işlevlerini kaydettirmek için MQXR\_BEFORE ve MQXR\_AFTER çıkış nedenleriyle MQXF\_SUBRQ işlev tanıtıcısını kullanın.

Bu işleve ilişkin arabirim şöyledir:

```
MQ_SUBRQ_EXIT (&ExitParms, &ExitContext, &Hconn, &pHsub, &Action, &pSubRqOpts,
              &CompCode, &Reason)
```

parametrelerin bulunduğu yer:

### ExitParms (MQAXP)-giriş/çıkış

Parametre yapısından çıkın.

### **ExitContext (MQAXC)-giriş/çıkış**

Bağlam yapısından çıkın.

### **Hconn (MQHCONN)-giriş/çıkış**

Bağlantı tanıtıcısı.

### **pHsub (MQHOBJ) giriş/çıkış**

Abonelik tanıtıcısı

### **İşlem (MQUZE) giriş/çıkış**

İşlem

### **pSubRqOpts (MQSRO) giriş/çıkış**

### **CompCode (MQHOT)-giriş/çıkış**

Tamamlanma kodu, geçerli değerler şunlardır:

#### **MQCC\_OK**

İşlem başarıyla tamamlandı.

#### **MQCC\_UYARI**

Kısmi tamamlama.

#### **MQCC\_FAILED**

Arama başarısız oldu

### **Neden (MQlong)-giriş/çıkış**

Neden kodu, tamamlanma kodunu nitelemektedir.

Tamamlanma kodu MQCC\_OK ise, tek geçerli değer şöyledir:

#### **MQRC\_NONE**

(0, x '000') Raporlamak için bir neden yok.

Tamamlanma kodu MQCC\_FAILED ya da MQCC\_UYARI ise, çıkış işlevi neden kodu alanını geçerli bir MQRC\_\* değerine ayarlayabiliyor.

## **C dili çağırma**

Kuyruk yöneticisi, aşağıdaki değişkenleri mantıksal olarak tanımlar:

```
MQAXP    ExitParms;        /* Exit parameter structure */
MQAXC    ExitContext;     /* Exit context structure */
MQHCONN  Hconn;           /* Connection handle */
PMQLONG  pHsub;           /* Subscription handle */
MQLONG   Action;          /* Action */
PMQSRO   pSubRqOpts;     /* Subscription Request Options */
MQLONG   CompCode;       /* Completion code */
MQLONG   Reason;         /* Reason code qualifying completion code */
```

Kuyruk yöneticisi, çıkışı mantıksal olarak aşağıdaki gibi çağırır:

```
MQ_SUBRQ_EXIT (&ExitParms, &ExitContext, &Hconn, &pHsub, &Action, &pSubRqOpts,
               &CompCode, &Reason);
```

Çıkışınız, aşağıdaki C işlevi prototipini eşleştirmelidir:

```
void MQENTRY MQ_SUBRQ_EXIT (
PMQAXP    pExitParms,      /* Address of exit parameter structure */
PMQAXC    pExitContext,    /* Address of exit context structure */
PMQHCONN  pHconn,         /* Address of connection handle */
PPMHOBJS  ppHsub,         /* Address of Subscription handle */
PMQLONG   pAction;        /* Address of Action */
PPMSRO    ppSubRqOpts;    /* Address of Subscription Request Options */
PMQLONG   pCompCode,      /* Address of completion code */
PMQLONG   pReason;        /* Address of reason code qualifying completion
                           code */
```

## ***xa\_close-XA\_CLOSE\_EXIT***

XA\_CLOSE\_EXIT, xa\_close işlemenden önce ve sonra gerçekleştirilmek üzere bir xa\_close çıkış işlevi sağlar. xa\_close çağrı çıkış işlevlerinden önce ve sonra kaydedildikten sonra MQXR\_BEFORE ve MQXR\_AFTER çıkış nedenleriyle MQXF\_XACLOSE işlev tanıtıcısını kullanın.

Bu işleve ilişkin arabirim şöyledir:

```
XA_CLOSE_EXIT (&ExitParms, &ExitContext, &Hconn, &pXa_info, &Rmid, &Flags, &XARetCode)
```

parametrelerin bulunduğu yer:

### **ExitParms (MQAXP)-giriş/çıkış**

Parametre yapısından çıkın.

### **ExitContext (MQAXC)-giriş/çıkış**

Bağlam yapısından çıkın.

### **Hconn (MQHCONN)-giriş**

Bağlantı tanıtıcısı.

### **pXa\_info (PMQCHAR)-giriş/çıkış**

Eşgörünümüne özgü kaynak yöneticisi bilgileri.

### **Rmid (MQHOT)-giriş/çıkış**

Kaynak yöneticisi tanıtıcısı.

### **İşaretler (MQUZE)-giriş/çıkış**

Kaynak yöneticisi seçenekleri.

### **XARetCode (MQHOT)-giriş/çıkış**

XA çağrısından yanıt.

## **C dili çağırma**

Kuyruk yöneticisi, aşağıdaki değişkenleri mantıksal olarak tanımlar:

```
MQAXP      ExitParms;      /* Exit parameter structure */
MQAXC      ExitContext;    /* Exit context structure */
MQHCONN     Hconn;         /* Connection handle */
PMQCHAR     pXa_info;      /* Instance-specific RM info */
MQLONG      Rmid;         /* Resource manager identifier */
MQLONG      Flags;        /* Resource manager options*/
MQLONG      XARetCode;    /* Response from XA call */
```

Kuyruk yöneticisi, çıkışı mantıksal olarak aşağıdaki gibi çağırır:

```
XA_CLOSE_EXIT (&ExitParms, &ExitContext, &Hconn, &pXa_info, &Rmid, &Flags, &XARetCode);
```

Çıkışınız, aşağıdaki C işlevi prototipini eşleştirmelidir:

```
typedef void MQENTRY XA_CLOSE_EXIT (
    PMQAXP      pExitParms,    /* Address of exit parameter structure */
    PMQAXC      pExitContext,  /* Address of exit context structure */
    PMQHCONN     pHconn,       /* Address of connection handle */
    PPMQCHAR     ppXa_info,    /* Address of instance-specific RM info */
    PMQLONG      pRmid,        /* Address of resource manager identifier */
    PMQLONG      pFlags,       /* Address of resource manager options*/
    PMQLONG      pXARetCode);  /* Address of response from XA call */
```

## ***xa\_commit-XA\_COMMIT\_EXIT***

XA\_COMMIT\_EXIT, xa\_commit işlemenden önce ve sonra gerçekleştirilmek üzere bir xa\_commit çıkış işlevi sağlar. Önceki ve xa\_commit çağrı çıkış işlevlerinin ardından MQXR\_BEFORE ve MQXR\_AFTER çıkış nedenleriyle birlikte MQXF\_XACOMMIT işlev tanıtıcısını kullanın.

Bu işleve ilişkin arabirim şöyledir:



```
XA_COMMIT_EXIT (&ExitParms, &ExitContext, &Hconn, &pXID, &Rmid, &Flags, &XARetCode)
```

parametrelerin bulunduğu yer:

### **ExitParms (MQAXP)-giriş/çıkış**

Parametre yapısından çıkın.

### **ExitContext (MQAXC)-giriş/çıkış**

Bağlam yapısından çıkın.

### **Hconn (MQHCONN)-giriş**

Bağlantı tanıtıcısı.

### **pXID (MQPTR)-giriş/çıkış**

İşlem dalı tanıtıcısı.

### **Rmid (MQHOT)-giriş/çıkış**

Kaynak yöneticisi tanıtıcısı.

### **İşaretler (MQUZE)-giriş/çıkış**

Kaynak yöneticisi seçenekleri.

### **XARetCode (MQHOT)-giriş/çıkış**

XA çağrısından yanıt.

## **C dili çağırma**

Kuyruk yöneticisi, aşağıdaki değişkenleri mantıksal olarak tanımlar:

```
MQAXP      ExitParms;    /* Exit parameter structure */
MQAXC      ExitContext; /* Exit context structure */
MQHCONN    Hconn;       /* Connection handle */
MQPTR      pXID;        /* Transaction branch ID */
MQLONG     Rmid;        /* Resource manager identifier */
MQLONG     Flags;       /* Resource manager options*/
MQLONG     XARetCode;   /* Response from XA call */
```

Kuyruk yöneticisi, çıkışı mantıksal olarak aşağıdaki gibi çağırır:

```
XA_COMMIT_EXIT (&ExitParms, &ExitContext, &Hconn, &pXID, &Rmid, &Flags, &XARetCode);
```

Çıkışınız, aşağıdaki C işlevi prototipini eşleştirmelidir:

```
typedef void MQENTRY XA_COMMIT_EXIT (
    PMQAXP      pExitParms,    /* Address of exit parameter structure */
    PMQAXC      pExitContext, /* Address of exit context structure */
    PMQHCONN    pHconn,       /* Address of connection handle */
    PMQPTR      ppXID,        /* Address of transaction branch ID */
    PMQLONG     pRmid,        /* Address of resource manager identifier */
    PMQLONG     pFlags,       /* Address of resource manager options*/
    PMQLONG     pXARetCode); /* Address of response from XA call */
```

## **xa\_complete-XA\_COMPLETE\_EXIT**

X\_COMPLETE\_EXIT, xa\_complete işleminden önce ve sonra gerçekleştirilecek bir xa\_complete çıkış işlevi sağlar. MQXF\_XACOMPLETE işlev tanıtıcısını kullanarak, önce MQXR\_BEFORE ve MQXR\_AFTER çıkış nedenlerini kullanarak, xa\_complete çağrı çıkış işlevlerinden önce ve sonra kayıt yaptır.

Bu işleve ilişkin arabirim şöyledir:

```
XA_COMPLETE_EXIT (&ExitParms, &ExitContext, &Hconn, &pHandle, &pRetval, &Rmid, &Flags, &XARetCode)
```

parametrelerin bulunduğu yer:

### **ExitParms (MQAXP)-giriş/çıkış**

Parametre yapısından çıkın.

### **ExitContext (MQAXC)-giriş/çıkış**

Bağlam yapısından çıkın.

### **Hconn (MQHCONN)-giriş**

Bağlantı tanıtıcısı.

### **pHandle (PMQUWN)-giriş/çıkış**

Zamanuyumsuz işleme ilişkin gösterge.

### **pRetVal (PMQLong)-giriş/çıkış**

Zamanuyumsuz işlemin dönüş değeri.

### **Rmid (MQHOT)-giriş/çıkış**

Kaynak yöneticisi tanıtıcısı.

### **İşaretler (MQUZE)-giriş/çıkış**

Kaynak yöneticisi seçenekleri.

### **XARetCode (MQHOT)-giriş/çıkış**

XA çağrısından yanıt.

## **C dili çağırma**

Kuyruk yöneticisi, aşağıdaki değişkenleri mantıksal olarak tanımlar:

```
MQAXP  ExitParms;    /* Exit parameter structure */
MQAXC  ExitContext; /* Exit context structure */
MQHCONN Hconn;      /* Connection handle */
PMQLONG pHandle;    /* Ptr to asynchronous op */
PMQLONG pRetVal;    /* Return value of async op */
MQLONG  Rmid;       /* Resource manager identifier */
MQLONG  Flags;      /* Resource manager options*/
MQLONG  XARetCode; /* Response from XA call */
```

Kuyruk yöneticisi, çıkışı mantıksal olarak aşağıdaki gibi çağırır:

```
XA_COMPLETE_EXIT (&ExitParms, &ExitContext, &Hconn, &pHandle, &pRetVal, &Rmid, &Flags,
&XARetCode);
```

Çıkışınız, aşağıdaki C işlevi prototipini eşleştirmelidir:

```
typedef void MQENTRY XA_COMPLETE_EXIT (
PMQAXP  pExitParms, /* Address of exit parameter structure */
PMQAXC  pExitContext, /* Address of exit context structure */
PMQHCONN pHconn, /* Address of connection handle */
PPMQLONG ppHandle, /* Address of ptr to asynchronous op */
PPMQLONG ppRetVal, /* Address of return value of async op */
PMQLONG  pRmid, /* Address of resource manager identifier */
PMQLONG  pFlags, /* Address of resource manager options*/
PMQLONG  pXARetCode); /* Address of response from XA call */
```

### ***xa\_end-XA\_END\_EXIT***

XA\_END\_EXIT, xa\_end işlemiden önce ve sonra gerçekleştirilecek bir xa\_end çıkış işlevi sağlar.

MQXF\_XAEND işlev tanıtıcısını kullanarak, önce MQXR\_BEFORE ve MQXR\_AFTER çıkış nedenleriyle, xa\_end çağrı çıkış işlevlerinden önce ve sonra kayıt yaptır.

Bu işleve ilişkin arabirim şöyledir:

```
XA_END_EXIT (&ExitParms, &ExitContext, &Hconn, &pXID, &Rmid, &Flags, &XARetCode)
```

parametrelerin bulunduğu yer:

### **ExitParms (MQAXP)-giriş/çıkış**

Parametre yapısından çıkın.

### **ExitContext (MQAXC)-giriş/çıkış**

Bağlam yapısından çıkın.

### **Hconn (MQHCONN)-giriş**

Bağlantı tanıtıcısı.

### **pXID (MQPTR)-giriş/çıkış**

İşlem dalı tanıtıcısı.

### **Rmid (MQHOT)-giriş/çıkış**

Kaynak yöneticisi tanıtıcısı.

### **İşaretler (MQUZE)-giriş/çıkış**

Kaynak yöneticisi seçenekleri.

### **XARetCode (MQHOT)-giriş/çıkış**

XA çağrısından yanıt.

## **C dili çağırma**

Kuyruk yöneticisi, aşağıdaki değişkenleri mantıksal olarak tanımlar:

```
MQAXP  ExitParms;    /* Exit parameter structure */
MQAXC  ExitContext; /* Exit context structure */
MQHCONN Hconn;      /* Connection handle */
MQPTR  pXID;        /* Transaction branch ID */
MQLONG Rmid;        /* Resource manager identifier */
MQLONG Flags;       /* Resource manager options*/
MQLONG XARetCode;  /* Response from XA call */
```

Kuyruk yöneticisi, çıkışı mantıksal olarak aşağıdaki gibi çağırır:

```
XA_END_EXIT (&ExitParms, &ExitContext, &Hconn, &pXID, &Rmid, &Flags, &XARetCode);
```

Çıkışınız, aşağıdaki C işlevi prototipini eşleştirmelidir:

```
typedef void MQENTRY XA_END_EXIT (
    PMQAXP  pExitParms, /* Address of exit parameter structure */
    PMQAXC  pExitContext, /* Address of exit context structure */
    PMQHCONN pHconn, /* Address of connection handle */
    PMQPTR  ppXID, /* Address of transaction branch ID */
    PMQLONG pRmid, /* Address of resource manager identifier */
    PMQLONG pFlags, /* Address of resource manager options*/
    PMQLONG pXARetCode); /* Address of response from XA call */
```

### ***xa\_forget-XA\_Unutkan\_çıkışı***

XA\_FORGET\_EXIT, xa\_forget işlemeden önce ve sonra gerçekleştirmesi için bir xa\_forget çıkış işlevi sağlar. MQXF\_XAFORGET işlev tanıtıcısını kullanarak, MQXR\_BEFORE ve MQXR\_AFTER çıkış nedenleriyle, xa\_forget çağrı çıkış işlevinden önce ve sonra kayıt yaptıktan sonra kullanın.

Bu işleve ilişkin arabirim şöyledir:

```
XA_FORGET_EXIT (&ExitParms, &ExitContext, &Hconn, &pXID, &Rmid, &Flags, &XARetCode)
```

parametrelerin bulunduğu yer:

### **ExitParms (MQAXP)-giriş/çıkış**

Parametre yapısından çıkın.

### **ExitContext (MQAXC)-giriş/çıkış**

Bağlam yapısından çıkın.

### **Hconn (MQHCONN)-giriş**

Bağlantı tanıtıcısı.

### **pXID (MQPTR)-giriş/çıkış**

İşlem dalı tanıtıcısı.

### **Rmid (MQHOT)-giriş/çıkış**

Kaynak yöneticisi tanıtıcısı.

### **İşaretler (MQUZE)-giriş/çıkış**

Kaynak yöneticisi seçenekleri.

### **XARetCode (MQHOT)-giriş/çıkış**

XA çağrısından yanıt.

## **C dili çağırma**

Kuyruk yöneticisi, aşağıdaki değişkenleri mantıksal olarak tanımlar:

```
MQAXP  ExitParms;    /* Exit parameter structure */
MQAXC  ExitContext; /* Exit context structure */
MQHCONN Hconn;      /* Connection handle */
MQPTR  pXID;        /* Transaction branch ID */
MQLONG Rmid;        /* Resource manager identifier */
MQLONG Flags;       /* Resource manager options*/
MQLONG XARetCode;   /* Response from XA call */
```

Kuyruk yöneticisi, çıkışı mantıksal olarak aşağıdaki gibi çağırır:

```
XA_FORGET_EXIT (&ExitParms, &ExitContext, &Hconn, &pXID, &Rmid, &Flags, &XARetCode);
```

Çıkışınız, aşağıdaki C işlevi prototipini eşleştirmelidir:

```
typedef void MQENTRY XA_FORGET_EXIT (
    PMQAXP  pExitParms, /* Address of exit parameter structure */
    PMQAXC  pExitContext, /* Address of exit context structure */
    PMQHCONN pHconn, /* Address of connection handle */
    PMQPTR  ppXID, /* Address of transaction branch ID */
    PMQLONG pRmid, /* Address of resource manager identifier */
    PMQLONG pFlags, /* Address of resource manager options*/
    PMQLONG pXARetCode); /* Address of response from XA call */
```

### **xa\_open-XA\_OPEN\_EXIT**

XA\_OPEN\_EXIT, xa\_open işlemeden önce ve sonra gerçekleştirilmek üzere bir xa\_open çıkış işlevi sağlar. xa\_open çağrı çıkış işlevlerinden önce ve sonra kaydedildikten sonra MQXR\_BEFORE ve MQXR\_AFTER çıkış nedenleriyle MQXF\_XAOL işlev tanıtıcısını kullanın.

Bu işleve ilişkin arabirim şöyledir:

```
XA_OPEN_EXIT (&ExitParms, &ExitContext, &Hconn, &pXa_info, &Rmid, &Flags, &XARetCode)
```

parametrelerin bulunduğu yer:

### **ExitParms (MQAXP)-giriş/çıkış**

Parametre yapısından çıkın.

### **ExitContext (MQAXC)-giriş/çıkış**

Bağlam yapısından çıkın.

### **Hconn (MQHCONN)-giriş**

Bağlantı tanıtıcısı.

### **pXa\_info (PMQCHAR)-giriş/çıkış**

Eşgörünüme özgü kaynak yöneticisi bilgileri.

### **Rmid (MQHOT)-giriş/çıkış**

Kaynak yöneticisi tanıtıcısı.

## İşaretler (MQUZE)-giriş/çıkış

Kaynak yöneticisi seçenekleri.

## XARetCode (MQHOT)-giriş/çıkış

XA çağrısından yanıt.

## C dili çağırma

Kuyruk yöneticisi, aşağıdaki değişkenleri mantıksal olarak tanımlar:

```
MQAXP   ExitParms;    /* Exit parameter structure */
MQAXC   ExitContext; /* Exit context structure */
MQHCONN Hconn;       /* Connection handle */
PMQCHAR pXa_info;    /* Instance-specific RM info */
MQLONG  Rmid;        /* Resource manager identifier */
MQLONG  Flags;       /* Resource manager options */
MQLONG  XARetCode;   /* Response from XA call */
```

Kuyruk yöneticisi, çıkışı mantıksal olarak aşağıdaki gibi çağırır:

```
XA_OPEN_EXIT (&ExitParms, &ExitContext, &Hconn, &pXa_info, &Rmid, &Flags, &XARetCode);
```

Çıkışınız, aşağıdaki C işlevi prototipini eşleştirmelidir:

```
typedef void MQENTRY XA_OPEN_EXIT (
    PMQAXP   pExitParms, /* Address of exit parameter structure */
    PMQAXC   pExitContext, /* Address of exit context structure */
    PMQHCONN pHconn, /* Address of connection handle */
    PPMQCHAR ppXa_info, /* Address of instance-specific RM info */
    PMQLONG  pRmid, /* Address of resource manager identifier */
    PMQLONG  pFlags, /* Address of resource manager options */
    PMQLONG  pXARetCode); /* Address of response from XA call */
```

## ***xa\_prepare-XA\_PREPARE\_EXIT***

XA\_PREPARE\_EXIT, xa\_prepare işleminden önce ve sonra gerçekleştirilecek bir xa\_prepare çıkış işlevi sağlar. MQXF\_XAPREPARE işlev tanıtıcısını, önce MQXR\_BEFORE ve MQXR\_AFTER çıkış nedenleriyle kullanarak, xa\_prepare çağrı çıkış işlevlerinin önüne ve sonrasına kayıt yaptır.

Bu işleve ilişkin arabirim şöyledir:

```
XA_PREPARE_EXIT (&ExitParms, &ExitContext, &Hconn, &pXID, &Rmid, &Flags, &XARetCode)
```

parametrelerin bulunduğu yer:

### **ExitParms (MQAXP)-giriş/çıkış**

Parametre yapısından çıkın.

### **ExitContext (MQAXC)-giriş/çıkış**

Bağlam yapısından çıkın.

### **Hconn (MQHCONN)-giriş**

Bağlantı tanıtıcısı.

### **pXID (MQPTR)-giriş/çıkış**

İşlem dalı tanıtıcısı.

### **Rmid (MQHOT)-giriş/çıkış**

Kaynak yöneticisi tanıtıcısı.

### **İşaretler (MQUZE)-giriş/çıkış**

Kaynak yöneticisi seçenekleri.

### **XARetCode (MQHOT)-giriş/çıkış**

XA çağrısından yanıt.

## C dili çağırma

Kuyruk yöneticisi, aşağıdaki değişkenleri mantıksal olarak tanımlar:

```
MQAXP  ExitParms; /* Exit parameter structure */
MQAXC  ExitContext; /* Exit context structure */
MQHCONN Hconn; /* Connection handle */
MQPTR  pXID; /* Transaction branch ID */
MQLONG Rmid; /* Resource manager identifier */
MQLONG Flags; /* Resource manager options*/
MQLONG XARetCode; /* Response from XA call */
```

Kuyruk yöneticisi, çıkışı mantıksal olarak aşağıdaki gibi çağırır:

```
XA_PREPARE_EXIT (&ExitParms, &ExitContext, &Hconn, &pXID, &Rmid, &Flags, &XARetCode);
```

Çıkışınız, aşağıdaki C işlevi prototipini eşleştirmelidir:

```
typedef void MQENTRY XA_PREPARE_EXIT (
    PMQAXP  pExitParms, /* Address of exit parameter structure */
    PMQAXC  pExitContext, /* Address of exit context structure */
    PMQHCONN pHconn, /* Address of connection handle */
    PMQPTR  ppXID, /* Address of transaction branch ID */
    PMQLONG pRmid, /* Address of resource manager identifier */
    PMQLONG pFlags, /* Address of resource manager options*/
    PMQLONG pXARetCode); /* Address of response from XA call */
```

### ***xa\_recover-XA\_RECOVER\_EXIT***

XA\_RECOVER\_EXIT, xa\_recover işleminden önce ve sonra gerçekleştirilmek üzere bir xa\_recover çıkış işlevi sağlar. xa\_recover çağrı çıkış işlevlerinden önce ve sonra, MQXR\_BEFORE ve MQXR\_AFTER çıkış nedenleriyle MQXF\_XARECOVER işlev tanıtıcısını kullanın.

Bu işleve ilişkin arabirim şöyledir:

```
XA_RECOVER_EXIT (&ExitParms, &ExitContext, &Hconn, &pXID, &Count, &Rmid, &Flags, &XARetCode)
```

parametrelerin bulunduğu yer:

#### **ExitParms (MQAXP)-giriş/çıkış**

Parametre yapısından çıkın.

#### **ExitContext (MQAXC)-giriş/çıkış**

Bağlam yapısından çıkın.

#### **Hconn (MQHCONN)-giriş**

Bağlantı tanıtıcısı.

#### **pXID (MQPTR)-giriş/çıkış**

İşlem dalı tanıtıcısı.

#### **Count (MQUZE)-giriş/çıkış**

XID dizisinde en çok XID sayısı

#### **Rmid (MQHOT)-giriş/çıkış**

Kaynak yöneticisi tanıtıcısı.

#### **İşaretler (MQUZE)-giriş/çıkış**

Kaynak yöneticisi seçenekleri.

#### **XARetCode (MQHOT)-giriş/çıkış**

XA çağrısından yanıt.

## C dili çağırma

Kuyruk yöneticisi, aşağıdaki değişkenleri mantıksal olarak tanımlar:

```

MQAXP  ExitParms;    /* Exit parameter structure */
MQAXC  ExitContext; /* Exit context structure */
MQHCONN Hconn;      /* Connection handle */
MQPTR  pXID;        /* Transaction branch ID */
MQLONG Count;      /* Max XIDs in XID array */
MQLONG Rmid;       /* Resource manager identifier */
MQLONG Flags;      /* Resource manager options*/
MQLONG XARetCode;  /* Response from XA call */

```

Kuyruk yöneticisi, çıkışı mantıksal olarak aşağıdaki gibi çağırır:

```
XA_RECOVER_EXIT (&ExitParms, &ExitContext, &Hconn, &pXID, &Count, &Rmid, &Flags, &XARetCode);
```

Çıkışınız, aşağıdaki C işlevi prototipini eşleştirmelidir:

```

typedef void MQENTRY XA_RECOVER_EXIT (
    PMQAXP  pExitParms,    /* Address of exit parameter structure */
    PMQAXC  pExitContext, /* Address of exit context structure */
    PMQHCONN pHconn,      /* Address of connection handle */
    MQPTR   ppXID,        /* Address of transaction branch ID */
    PMQLONG pCount,       /* Address of max XIDs in XID array */
    PMQLONG pRmid,        /* Address of resource manager identifier */
    PMQLONG pFlags,       /* Address of resource manager options*/
    PMQLONG pXARetCode); /* Address of response from XA call */

```

### **xa\_geridönüş-XA\_ROLLBACK\_EXIT**

XA\_ROLLBACK\_EXIT, xa\_rollback işlemeden önce ve sonra gerçekleştirilmek üzere bir xa\_rollback çıkış işlevi sağlar. Önceki ve xa\_geridönüş çağırısı çıkış işlevlerinden önce ve sonra MQXR\_AFTER ve MQXR\_AFTER çıkış nedenleriyle MQXF\_XAROLLBACK işlev tanıtıcısını kullanın.

Bu işleve ilişkin arabirim şöyledir:

```
XA_ROLLBACK_EXIT (&ExitParms, &ExitContext, &Hconn, &pXID, &Rmid, &Flags, &XARetCode)
```

parametrelerin bulunduğu yer:

#### **ExitParms (MQAXP)-giriş/çıkış**

Parametre yapısından çıkın.

#### **ExitContext (MQAXC)-giriş/çıkış**

Bağlam yapısından çıkın.

#### **Hconn (MQHCONN)-giriş**

Bağlantı tanıtıcısı.

#### **pXID (MQPTR)-giriş/çıkış**

İşlem dalı tanıtıcısı.

#### **Rmid (MQHOT)-giriş/çıkış**

Kaynak yöneticisi tanıtıcısı.

#### **İşaretler (MQUZE)-giriş/çıkış**

Kaynak yöneticisi seçenekleri.

#### **XARetCode (MQHOT)-giriş/çıkış**

XA çağırısından yanıt.

## **C dili çağırma**

Kuyruk yöneticisi, aşağıdaki değişkenleri mantıksal olarak tanımlar:

```

MQAXP  ExitParms;    /* Exit parameter structure */
MQAXC  ExitContext; /* Exit context structure */
MQHCONN Hconn;      /* Connection handle */
MQPTR  pXID;        /* Transaction branch ID */
MQLONG Rmid;       /* Resource manager identifier */

```

```
MQLONG Flags; /* Resource manager options*/
MQLONG XARetCode; /* Response from XA call */
```

Kuyruk yöneticisi, çıkışı mantıksal olarak aşağıdaki gibi çağırır:

```
XA_ROLLBACK_EXIT (&ExitParms, &ExitContext, &Hconn, &pXID, &Rmid, &Flags, &XARetCode);
```

Çıkışınız, aşağıdaki C işlevi prototipini eşleştirmelidir:

```
typedef void MQENTRY XA_ROLLBACK_EXIT (
    PMQAXP pExitParms, /* Address of exit parameter structure */
    PMQAXC pExitContext, /* Address of exit context structure */
    PMQHCONN pHconn, /* Address of connection handle */
    PMQPTR ppXID, /* Address of transaction branch ID */
    PMQLONG pRmid, /* Address of resource manager identifier */
    PMQLONG pFlags, /* Address of resource manager options*/
    PMQLONG pXARetCode); /* Address of response from XA call */
```

### ***xa\_start-XA\_START\_EXIT***

XA\_START\_EXIT, xa\_start işleminden önce ve sonra gerçekleştirilecek bir xa\_start çıkış işlevi sağlar. MQXR\_BEFORE ve MQXR\_AFTER çıkış nedenleriyle MQXF\_XASTART işlev tanıtıcısını kullanarak, Xa\_start çağrı çıkış işlevlerinden önce ve sonra kayıt yaptır.

Bu işleve ilişkin arabirim şöyledir:

```
XA_START_EXIT (&ExitParms, &ExitContext, &Hconn, &pXID, &Rmid, &Flags, &XARetCode)
```

parametrelerin bulunduğu yer:

#### **ExitParms (MQAXP)-giriş/çıkış**

Parametre yapısından çıkın.

#### **ExitContext (MQAXC)-giriş/çıkış**

Bağlam yapısından çıkın.

#### **Hconn (MQHCONN)-giriş**

Bağlantı tanıtıcısı.

#### **pXID (MQPTR)-giriş/çıkış**

İşlem dalı tanıtıcısı.

#### **Rmid (MQHOT)-giriş/çıkış**

Kaynak yöneticisi tanıtıcısı.

#### **İşaretler (MQUZE)-giriş/çıkış**

Kaynak yöneticisi seçenekleri.

#### **XARetCode (MQHOT)-giriş/çıkış**

XA çağrısından yanıt.

## **C dili çağırma**

Kuyruk yöneticisi, aşağıdaki değişkenleri mantıksal olarak tanımlar:

```
MQAXP ExitParms; /* Exit parameter structure */
MQAXC ExitContext; /* Exit context structure */
MQHCONN Hconn; /* Connection handle */
MQPTR pXID; /* Transaction branch ID */
MQLONG Rmid; /* Resource manager identifier */
MQLONG Flags; /* Resource manager options*/
MQLONG XARetCode; /* Response from XA call */
```

Kuyruk yöneticisi, çıkışı mantıksal olarak aşağıdaki gibi çağırır:

```
XA_START_EXIT (&ExitParms, &ExitContext, &Hconn, &pXID, &Rmid, &Flags, &XARetCode);
```



Çıkışınız, aşağıdaki C işlevi prototipini eşleştirmelidir:

```
typedef void MQENTRY XA_START_EXIT (  
    PMQAXP  pExitParms, /* Address of exit parameter structure */  
    PMQAXC  pExitContext, /* Address of exit context structure */  
    PMQHCONN pHconn, /* Address of connection handle */  
    PMQPTR  ppXID, /* Address of transaction branch ID */  
    PMQLONG pRmid, /* Address of resource manager identifier */  
    PMQLONG pFlags, /* Address of resource manager options*/  
    PMQLONG pXARetCode); /* Address of response from XA call */
```

### **ax\_reg-AX\_REG\_EXIT**

AX\_REG\_EXIT, ax\_reg işleminden önce ve sonra gerçekleştirilmek üzere bir ax\_reg çıkış işlevi sağlar. MQXR\_AXREG ve MQXR\_AFTER çıkış nedenleriyle MQXR\_AXREG işlev tanıtıcısını kullanarak, önce ve sonra Ax\_reg çağrısı çıkış işlevlerinden sonra MQXR\_BEFORE ve MQXR\_AFTER kullanın.

Bu işleve ilişkin arabirim şöyledir:

```
AX_REG_EXIT (&ExitParms, &ExitContext, &pXID, &Rmid, &Flags, &XARetCode)
```

parametrelerin bulunduğu yer:

#### **ExitParms (MQAXP)-giriş/çıkış**

Parametre yapısından çıkın.

#### **ExitContext (MQAXC)-giriş/çıkış**

Bağlam yapısından çıkın.

#### **Hconn (MQHCONN)-giriş**

Bağlantı tanıtıcısı.

#### **pXID (MQPTR)-giriş/çıkış**

İşlem dalı tanıtıcısı.

#### **Rmid (MQHOT)-giriş/çıkış**

Kaynak yöneticisi tanıtıcısı.

#### **İşaretler (MQUZE)-giriş/çıkış**

Kaynak yöneticisi seçenekleri.

#### **XARetCode (MQHOT)-giriş/çıkış**

XA çağrısından yanıt.

## **C dili çağırma**

Kuyruk yöneticisi, aşağıdaki değişkenleri mantıksal olarak tanımlar:

```
MQAXP  ExitParms; /* Exit parameter structure */  
MQAXC  ExitContext; /* Exit context structure */  
MQPTR  pXID; /* Transaction branch ID */  
MQLONG Rmid; /* Resource manager identifier */  
MQLONG Flags; /* Resource manager options*/  
MQLONG XARetCode; /* Response from XA call */
```

Kuyruk yöneticisi, çıkışı mantıksal olarak aşağıdaki gibi çağırır:

```
AX_REG_EXIT (&ExitParms, &ExitContext, &pXID, &Rmid, &Flags, &XARetCode);
```

Çıkışınız, aşağıdaki C işlevi prototipini eşleştirmelidir:

```
typedef void MQENTRY AX_REG_EXIT (  
    PMQAXP  pExitParms, /* Address of exit parameter structure */  
    PMQAXC  pExitContext, /* Address of exit context structure */  
    PMQPTR  ppXID, /* Address of transaction branch ID */  
    PMQLONG pRmid, /* Address of resource manager identifier */
```

```
PMQLONG pFlags, /* Address of resource manager options*/
PMQLONG pXARetCode); /* Address of response from XA call */
```

### **ax\_unreg-AX\_UNREG\_EXIT**

AX\_UNREG\_EXIT, ax\_unreg işlemeden önce ve sonra gerçekleştirilecek bir ax\_unreg çıkış işlevi sağlar. MQXR\_AFTER ve MQXR\_AFTER çıkış nedenlerini içeren MQXF\_AXUNREG işlev tanıtıcısını, ax\_unreg çağrı çıkış işlevlerinden önce ve sonra kaydettirmek için kullanın.

Bu işleve ilişkin arabirim şöyledir:

```
AX_UNREG_EXIT (&ExitParms, &ExitContext, &Rmid, &Flags, &XARetCode);
```

parametrelerin bulunduğu yer:

#### **ExitParms (MQAXP)-giriş/çıkış**

Parametre yapısından çıkın.

#### **ExitContext (MQAXC)-giriş/çıkış**

Bağlam yapısından çıkın.

#### **Rmid (MQHOT)-giriş/çıkış**

Kaynak yöneticisi tanıtıcısı.

#### **İşaretler (MQUZE)-giriş/çıkış**

Kaynak yöneticisi seçenekleri.

#### **XARetCode (MQHOT)-giriş/çıkış**

XA çağrısından yanıt.

## **C dili çağırma**

Kuyruk yöneticisi, aşağıdaki değişkenleri mantıksal olarak tanımlar:

```
MQAXP ExitParms; /* Exit parameter structure */
MQAXC ExitContext; /* Exit context structure */
MQLONG Rmid; /* Resource manager identifier */
MQLONG Flags; /* Resource manager options*/
MQLONG XARetCode; /* Response from XA call */
```

Kuyruk yöneticisi, çıkışı mantıksal olarak aşağıdaki gibi çağırır:

```
AX_UNREG_EXIT (&ExitParms, &ExitContext, &Rmid, &Flags, &XARetCode);
```

Çıkışınız, aşağıdaki C işlevi prototipini eşleştirmelidir:

```
typedef void MQENTRY AX_UNREG_EXIT (
    PMQAXP pExitParms, /* Address of exit parameter structure */
    PMQAXC pExitContext, /* Address of exit context structure */
    PMQLONG pRmid, /* Address of resource manager identifier */
    PMQLONG pFlags, /* Address of resource manager options*/
    PMQLONG pXARetCode); /* Address of response from XA call */
```

## **Çıkış işlevlerinin çağırılmasıyla ilgili genel bilgiler**

Bu konuda, özellikle hatalar ve beklenmeyen olaylarla ilgili olarak, çıkışlarınızı planlamanıza yardımcı olacak bazı genel yönergeler sağlanmaktadır.

### **Çıkış hatası**

Bir çıkış işlevi yıkıcı, uyumsuz, MQGET çağrısından sonra olağandışı bir şekilde sonlandırılırsa, ancak iletiye iletilmeden önce, çıkış işleyicisi hatadan kurtarılabilir ve denetimden uygulamaya geçilebilir.

Bu durumda, ileti kaybolabilir. Bu durum, bir uygulama kuyruktan ileti aldıktan hemen sonra başarısız olduğunda ortaya çıktı.

MQGET çağrısı MQCC\_FAILED ve MQRC\_API\_EXIT\_ERROR ile tamamlanabilir.

Bir önce API çağrısı çıkış işlevi olağan dışı bir şekilde sonlandırılırsa, çıkış işleyicisi başarısızlıktan kurtulabilir ve API çağrısını işlemeyen uygulamaya denetim geçirebilir. Bu durumda, çıkış işlevinin sahip olduğu kaynakları kurtarması gerekir.

Zincirleme çıkışlar kullanılırsa, başarıyla yönlendirilen herhangi bir önce API çağrısı çıkışı için bundan sonra API çağrısı çıkışlar sürülebilirler. API çağrısı, MQCC\_FAILED ve MQRC\_API\_EXIT\_ERROR ile başarısız olabilir.

#### Çıkış işlevleri için örnek hata işleme

Aşağıdaki çizge, noktaları göstermektedir (e N) Bu hatalar oluşabilir. Bu, çıkışların aşağıdaki çizelgeyle nasıl birlikte okunacağını ve nasıl birlikte okunacağını göstermek için bir örnektir. Bu örnekte, her API çağrısından önce ve sonra zincirleme çıkışlar içeren davranışı göstermek için iki çıkış işlevi çağrılır.

Application	ErrPt	Exit function	API call
Start			
MQCONN	-->		
	e1		
		MQ_INIT_EXIT	
	e2	before MQ_CONN_EXIT	1
	e3	before MQ_CONN_EXIT	2
	e4		--> MQCONN
	e5	after MQ_CONN_EXIT	2
	e6	after MQ_CONN_EXIT	1
	e7		
MQOPEN	-->		
	e8	before MQ_OPEN_EXIT	1
	e9	before MQ_OPEN_EXIT	2
	e10		--> MQOPEN
	e11	after MQ_OPEN_EXIT	2
	e12	after MQ_OPEN_EXIT	1
	e12		
MQPUT	-->		
	e13	before MQ_PUT_EXIT	1
	e14	before MQ_PUT_EXIT	2
	e15		--> MQPUT
	e16	after MQ_PUT_EXIT	2
	e17	after MQ_PUT_EXIT	1
	e17		
MQCLOSE	-->		
	e18	before MQ_CLOSE_EXIT	1
	e19	before MQ_CLOSE_EXIT	2
	e20		--> MQCLOSE
	e21	after MQ_CLOSE_EXIT	2
	e22	after MQ_CLOSE_EXIT	1
	e22		
MQDISC	-->		
	e23	before MQ_DISC_EXIT	1
		before MQ_DISC_EXIT	2

```

e24
                                --> MQDISC
e25  after MQ_DISC_EXIT 2
e26  after MQ_DISC_EXIT 1
e27

<--

end

```

Aşağıdaki çizelge, her hata noktasında gerçekleştirilecek işlemleri listeler. Burada gösterilen kurallar tüm diğerleri için geçerli olabildiği için, yalnızca hata noktalarının bir alt kümesi kapsam dahilinde olmuştur. Her durumda, amaçlanan davranışı belirten işlemdir.

<i>Çizelge 838. API çıkış hataları ve yapılması gereken uygun işlemler</i>		
<b>Err Pt</b>	<b>Tanım</b>	<b>İşlemler</b>
e1	Ortam ayarları ayarlanırken hata oluştu.	<ol style="list-style-type: none"> <li>1. Ortam kurulumunu gereken şekilde geri al</li> <li>2. Çıkış işlevsiz sürücü</li> <li>3. MQCC_FAILED ile MQCONN başarısız oldu, MQRC_API_EXIT_LOAD_ERROR</li> </ol>
e2	MQ_INIT_EXIT işlevi şu şekilde tamamlanır: <ul style="list-style-type: none"> <li>• MQXCC_FAILED</li> <li>• MQXCC_*</li> </ul>	<ul style="list-style-type: none"> <li>• MQXCC_FAILED için:               <ol style="list-style-type: none"> <li>1. Ortamı temizle</li> <li>2. MQCC_FAILED ile MQCONN başarısız oldu, MQRC_API_EXIT_INIT_ERROR</li> </ol> </li> <li>• MQXCC_* için *               <ol style="list-style-type: none"> <li>1. MQXCC_* ve MQXR2_*<sup>1</sup>değerleri için hareket edin.</li> <li>2. Ortamı temizle</li> </ol> </li> </ul>
e3	Önce MQ_CONNX_EXIT 1 işlevi aşağıdaki gibi tamamlanır: <ul style="list-style-type: none"> <li>• MQXCC_FAILED</li> <li>• MQXCC_*</li> </ul>	<ul style="list-style-type: none"> <li>• MQXCC_FAILED için:               <ol style="list-style-type: none"> <li>1. Sürücü MQ_TERM_EXIT işlevi</li> <li>2. Ortamı temizle</li> <li>3. MQCC_FAILED ile MQCONN çağrısı başarısız oldu, MQRC_API_EXIT_ERROR</li> </ol> </li> <li>• MQXCC_* için *               <ol style="list-style-type: none"> <li>1. MQXCC_* ve MQXR2_*<sup>1</sup>değerleri için hareket edin.</li> <li>2. Gerekliyse MQ_TERM_EXIT işlevini kullan</li> <li>3. Gerekirse ortamı temizle</li> </ol> </li> </ul>

Çizelge 838. API çıkış hataları ve yapılması gereken uygun işlemler (devamı var)

Err Pt	Tanım	İşlemler
e4	<p>Önce MQ_CONNX_EXIT 2 işlevi aşağıdaki gibi tamamlanır:</p> <ul style="list-style-type: none"> <li>• MQXCC_FAILED</li> <li>• MQXCC_*</li> </ul>	<ul style="list-style-type: none"> <li>• MQXCC_FAILED için: <ol style="list-style-type: none"> <li>1. Sürücü <i>bundan sonra</i> MQ_CONNX_EXIT 1 işlevi</li> <li>2. Sürücü MQ_TERM_EXIT işlevi</li> <li>3. Ortamı temizle</li> <li>4. MQCC_FAILED ile MQCONN çağrısı başarısız oldu, MQRC_API_EXIT_ERROR</li> </ol> </li> <li>• MQXCC_* için * <ol style="list-style-type: none"> <li>1. MQXCC_* ve MQXR2_*<sup>1</sup>değerleri için hareket edin.</li> <li>2. Drive <i>bundan sonra</i> MQ_CONNX_EXIT 1 function if exit not suppressed</li> <li>3. Gerekirse MQ_TERM_EXIT işlevini kullan</li> <li>4. Gerekirse ortamı temizle</li> </ol> </li> </ul>
e5	<p>MQCONN çağrısı başarısız oldu.</p>	<ol style="list-style-type: none"> <li>1. Geçiş MQCONN CompCode ve Reason</li> <li>2. <i>önce</i> MQ_CONNX_EXIT 2 başarılı olursa ve çıkış engellenmezse, <i>bundan sonra</i> MQ_CONNX_EXIT 2 işlevini kullanın.</li> <li>3. <i>önce</i> MQ_CONNX_EXIT 1 başarılı olursa ve çıkış engellenmezse, <i>bundan sonra</i> MQ_CONNX_EXIT 1 işlevini kullanın.</li> <li>4. Sürücü MQ_TERM_EXIT işlevi</li> <li>5. Ortamı temizle</li> </ol>
e6	<p>Sonra MQ_CONNX_EXIT 2 işlevi aşağıdaki gibi tamamlanır:</p> <ul style="list-style-type: none"> <li>• MQXCC_FAILED</li> <li>• MQXCC_*</li> </ul>	<ul style="list-style-type: none"> <li>• MQXCC_FAILED için: <ol style="list-style-type: none"> <li>1. Sürücü <i>bundan sonra</i> MQ_CONNX_EXIT 1 işlevi</li> <li>2. MQCC_FAILED ile MQCONN çağrısını tamamla, MQRC_API_EXIT_ERROR</li> </ol> </li> <li>• MQXCC_* için * <ol style="list-style-type: none"> <li>1. MQXCC_* ve MQXR2_*<sup>1</sup>değerleri için hareket edin.</li> <li>2. Gerekirse <i>bundan sonra</i> MQ_CONNX_EXIT 1 işlevini kullanın</li> </ol> </li> </ul>
e7	<p>Sonra MQ_CONNX_EXIT 1 işlevi aşağıdaki gibi tamamlanır:</p> <ul style="list-style-type: none"> <li>• MQXCC_FAILED</li> <li>• MQXCC_*</li> </ul>	<ul style="list-style-type: none"> <li>• MQXCC_FAILED için, MQCONN çağrısını MQCC_FAILED ile tamamla, MQRC_API_EXIT_ERROR</li> <li>• MQXCC_* için, MQXCC_* ve MQXR2_*<sup>1</sup>değerlerine göre hareket edin.</li> </ul>
e8	<p>Önce MQ_OPEN_EXIT 1 işlevi şu şekilde tamamlanır:</p> <ul style="list-style-type: none"> <li>• MQXCC_FAILED</li> <li>• MQXCC_*</li> </ul>	<ul style="list-style-type: none"> <li>• MQXCC_FAILED için, MQCC_FAILED ile MQXX_ENCODE_CASE_ONE open çağrısını tamamla, MQRC_API_EXIT_ERROR</li> <li>• MQXCC_* için, MQXCC_* ve MQXR2_*<sup>1</sup>değerlerine göre hareket edin.</li> </ul>

Çizelge 838. API çıkış hataları ve yapılması gereken uygun işlemler (devamı var)

Err Pt	Tanım	İşlemler
e9	Önce MQ_OPEN_EXIT 2 işlevi şu şekilde tamamlanır: <ul style="list-style-type: none"><li>• MQXCC_FAILED</li><li>• MQXCC_*</li></ul>	<ul style="list-style-type: none"><li>• MQXCC_FAILED için:<ol style="list-style-type: none"><li>1. Sürücü <i>bundan sonra</i> MQ_OPEN_EXIT 1 işlevi</li><li>2. MQCC_FAILED ile MQXX_ENCODE_CASE_ONE open çağrısını tamamla, MQRC_API_EXIT_ERROR</li></ol></li><li>• MQXCC_* için, MQXCC_* ve MQXR2_*<sup>1</sup>değerlerine göre hareket edin.</li></ul>
e10	MQOPEN çağrısı başarısız oldu	<ol style="list-style-type: none"><li>1. MQOPEN CompCode ögesini geçti ve neden</li><li>2. Çıkış engellenmezse <i>bundan sonra</i> MQ_OPEN_EXIT 2 işlevini kullan</li><li>3. Çıkış engellenmezse ve zincirleme çıkışlar engellenmezse, <i>bundan sonra</i> MQ_OPEN_EXIT 1 işlevini kullanın</li></ol>
e11	Sonra MQ_OPEN_EXIT 2 işlevi şu şekilde tamamlanır: <ul style="list-style-type: none"><li>• MQXCC_FAILED</li><li>• MQXCC_*</li></ul>	<ul style="list-style-type: none"><li>• MQXCC_FAILED için:<ol style="list-style-type: none"><li>1. Sürücü <i>bundan sonra</i> MQ_OPEN_EXIT 1 işlevi</li><li>2. MQCC_FAILED ile MQXX_ENCODE_CASE_ONE open çağrısını tamamla, MQRC_API_EXIT_ERROR</li></ol></li><li>• MQXCC_* için *<ol style="list-style-type: none"><li>1. MQXCC_* ve MQXR2_*<sup>1</sup>değerleri için hareket edin.</li><li>2. Çıkış engellenmezse <i>bundan sonra</i> MQ_OPEN_EXIT 1 işlevini kullan</li></ol></li></ul>
e25	Sonra MQ_DISC_EXIT 2 işlevi aşağıdaki gibi tamamlanır: <ul style="list-style-type: none"><li>• MQXCC_FAILED</li><li>• MQXCC_*</li></ul>	<ul style="list-style-type: none"><li>• MQXCC_FAILED için:<ol style="list-style-type: none"><li>1. Sürücü <i>bundan sonra</i> MQ_DISC_EXIT 1 işlevi</li><li>2. Sürücü MQ_TERM_EXIT işlevi</li><li>3. Çıkış yürütme ortamını temizle</li><li>4. MQCC_FAILED ile MQDISC çağrısını tamamla, MQRC_API_EXIT_ERROR</li></ol></li><li>• MQXCC_* için *<ol style="list-style-type: none"><li>1. MQXCC_* ve MQXR2_*<sup>1</sup>değerleri için hareket edin.</li><li>2. Sürücü MQ_TERM_EXIT işlevi</li><li>3. Çıkış yürütme ortamını temizle</li></ol></li></ul>

**Not:**

1. MQXCC\_\* ve MQXR2\_\* değerlerinin ve bunların karşılık gelen işlemlerinin değerleri, [Kuyruk yöneticilerinin çıkış işlevlerinin nasıl işlendiği](#) konusunda tanımlanır.

**ExitResponse alanları yanlış ayarlandı**

Bu konuda, ExitResponse alanı, desteklenen değerler dışında herhangi bir değere ayarlandığında neler olacağı hakkında bilgi verilir.

ExitResponse alanı, desteklenen değerlerden biri dışında bir değere ayarlıysa, aşağıdaki işlemler geçerli olur:

- Bir önce MQCONN ya da MQDISC API çıkış işlevi için:

- ExitResponse2 değeri yok sayılır.
- Çıkış zincirinde (varsa) başka *önce* çıkış işlevi çağrılmaz; API çağrısının kendisi yayınlanmaz.
- Başarılı bir şekilde çağrılan *önce* çıkışları için, *bundan sonra* çıkışları ters sırada çağrılır.
- Kaydedilmişse, başarıyla çağrılan zincirdeki *önce* MQCONN ya da MQDISC çıkış işlevlerinin sonlandırma çıkış işlevleri, bu çıkış işlevlerinin ardından temizlenir.
- MQCONN ya da MQDISC çağrısı MQRC\_API\_EXIT\_ERROR ile başarısız olur.
- MQCONN ya da MQDISC dışında bir *önce* IBM MQ API çıkış işlevi için:
  - ExitResponse2 değeri yok sayılır.
  - Çıkış zincirinde (varsa) başka *önce* ya da *bundan sonra* veri dönüştürme işlevi çağrılır.
  - Başarılı bir şekilde çağrılan *önce* çıkışları için, *bundan sonra* çıkışları ters sırada çağrılır.
  - IBM MQ API çağrısının kendisi yayınlanmaz.
  - IBM MQ API çağrısı MQRC\_API\_EXIT\_ERROR ile başarısız olur.
- Bir *bundan sonra* MQCONN ya da MQDISC API çıkış işlevi için:
  - ExitResponse2 değeri yok sayılır.
  - API çağrısından önce başarıyla çağrılan kalan çıkış işlevleri, ters sırada çağrılır.
  - Kayıtlıysa, başarıyla çağrılan zincirdeki *önce* ya da *bundan sonra* MQCONN ya da MQDISC çıkış işlevleri için sonlandırma çıkış işlevleri, çıkıştan sonra temizlemek için yönlendirilir.
  - MQCC\_UYARI için daha şiddetli bir CompCode ve çıkış tarafından döndürülen CompCode , uygulamaya geri döndürülür.
  - Uygulamaya MQRC\_API\_EXIT\_ERROR değeri döndürülmesi nedeniyle bir neden döndürülebilir.
  - IBM MQ API çağrısı başarıyla yayınlandı.
- MQCONN ya da MQDISC dışında bir *sonraki* IBM MQ API çağrısı çıkış işlevi için:
  - ExitResponse2 değeri yok sayılır.
  - API çağrısından önce başarıyla çağrılan kalan çıkış işlevleri, ters sırada çağrılır.
  - MQCC\_UYARI için daha şiddetli bir CompCode ve çıkış tarafından döndürülen CompCode , uygulamaya geri döndürülür.
  - Uygulamaya MQRC\_API\_EXIT\_ERROR değeri döndürülmesi nedeniyle bir neden döndürülebilir.
  - IBM MQ API çağrısı başarıyla yayınlandı.
- Çıkış alma işlevinde *önce* veri dönüştürme işlemi için:
  - ExitResponse2 değeri yok sayılır.
  - API çağrısından önce başarıyla çağrılan kalan çıkış işlevleri, ters sırada çağrılır.
  - İleti dönüştürülmedi ve dönüştürülemez ileti uygulamaya geri döndürüldü.
  - MQCC\_UYARI için daha şiddetli bir CompCode ve çıkış tarafından döndürülen CompCode , uygulamaya geri döndürülür.
  - Uygulamaya MQRC\_API\_EXIT\_ERROR değeri döndürülmesi nedeniyle bir neden döndürülebilir.
  - IBM MQ API çağrısı başarıyla yayınlandı.

**Not:** Hata çıkışta olduğu için, MQRC\_API\_EXIT\_ERROR döndürülmesi, MQRC\_NOT\_CONVERTED döndürülmekten daha iyi olur.

Bir çıkış işlevi, ExitResponse2 alanını desteklenen değerlerden biri dışında bir değere ayarlarsa, bunun yerine MQXR2\_DEFAULT\_CONTINUATION değeri varsayılır.

## Kurulabilir hizmetler arabirimi başvuru bilgileri

Bu konu derlemi, kurulabilir hizmetlere ilişkin başvuru bilgileri sağlar.

İşlevler ve veri tipleri, her hizmet tipi için grup içinde alfabetik sırayla listelenir.

## İlgili görevler

Kuyruk yöneticisi olanaklarının genişletmesi

**ALW** Kurulabilir hizmetlerin yapılandırılması

## İlgili başvurular

**ALW** UNIX, Linux ve Windows için kurulabilir hizmetler ve bileşenler

**IBM i** IBM için kurulabilir hizmetler ve bileşenler

**IBM i** IBM için kurulabilir hizmetler arabirimi başvuru bilgileri

Use this information to understand the reference information for the installable services for IBM i.

## İşlevlerin nasıl gösterildiğini

Kurulabilir hizmet işlevlerinin belgelenmesi.

Her işlev için, işlev tanıtıcısı (MQZEP için) de içinde olmak üzere, bir tanımlama vardır.

*Parametreler* , ortaya çıkmaları gereken sırayla gösterilir. Hepsi hazır olmalı.

Her parametre adı, onun veri tipi tarafından takip edilir. Bunlar, "[Temel veri tipleri](#)" sayfa 235' de açıklanan temel veri tipleridir.

Parametrelerin açıklamasından sonra, C dili çağrısına da verilir.

## MQZ\_AUTHENTICATE\_USER-Kullanıcı kimliği doğrulamalı kullanıcı

Bu işlev bir MQZAS\_VERSION\_5 yetkilendirme hizmeti bileşeni tarafından sağlanır ve kuyruk yöneticisi tarafından bir kullanıcının kimliğini doğrulamak ya da kimlik bağlamı alanlarını ayarlamak için kuyruk yöneticisi tarafından çağrılır. Bu, IBM MQ kullanıcı uygulaması bağlamı oluşturulduğunda çağrılır.

Uygulama bağlamı, bağlantı çağruları sırasında uygulamanın kullanıcı bağlamının kullanıma hazırlandığı noktada ve her bir noktada, uygulamanın kullanıcı bağlamının değiştirildiği noktada kurulur. Bir bağlantı çağrısı yapıldığında, uygulamanın kullanıcı bağlamı bilgileri *IdentityContext* alanında yeniden edinilir.

Bu işleve ilişkin işlev tanıtıcısı (MQZEP için) MQZID\_AUTHENTICATE\_USER ' tır.

## Sözdizimi

MQZ\_AUTHENTICATE\_USER ( *QMgrName* , *SecurityParms* , *ApplicationContext* , *IdentityContext* , *CorrelationPtr* , *ComponentData* , *Continuation* , *CompCode* , *Reason* )

## Parametreler

### QMgrName

Tip: MQCHAR48 -giriş

Kuyruk yöneticisi adı. Bileşeni çağırın kuyruk yöneticisinin adı. Bu ad, değiştirgenin tam uzunluğuna kadar boşluklarla dolduruldu; ad boş değerli bir karakterle sonlandırılmaz.

Kuyruk yöneticisi adı, bileşene bilgi için iletilir; yetki hizmeti arabirimi, bileşenin bu bileşeni tanımlı bir şekilde kullanmasını gerektirmez.

### SecurityParms

Tip: MQCSP-input

Güvenlik parametreleri. Kullanıcı kimliği, parola ve kimlik doğrulama türleriyle ilgili veriler. MQCSP yapısının AuthenticationType özniteliği MQCSP\_AUTH\_USER\_ID\_AND\_PWD olarak belirtilirse, hem kullanıcı kimliği, hem de parola, IdentityContext (MQZIC) parametresindeki eşdeğer alanlarla karşılaştırılıp eşleşmediklerini saptarlar. Daha fazla bilgi için bkz "[MQCSP-Güvenlik değiştirgeleri](#)" sayfa 335.

Bir MQCONN MQI çağrısı sırasında bu değiştirge boş değerli (null) ya da varsayılan değerler içeriyor.



## ApplicationContext

Tip: MQZAC-input

Uygulama bağlamı. Çağrılan uygulamayla ilgili veriler. Ayrıntılar için bkz. [MQZAC-Application context](#).

Tüm MQCONN ya da MQCONNX MQI çağrısı sırasında, MQZAC yapısındaki kullanıcı bağlamı bilgileri yeniden alınır.

## IdentityContext

Tip: MQZIC-giriş/çıkış

Kimlik bağlamı. Kullanıcı kimliği doğrulamasına giriş sırasında, bu değer yürürlükteki kimlik bağlamını tanımlar. Kullanıcı kimliği doğrulama işlevi, kuyruk yöneticisinin yeni kimlik bağlamını benimsediği noktada bu değişikliği değiştirebilir. MQZIC yapısıyla ilgili ek ayrıntılar için [MQZIC-Identity context](#) başlıklı konuya bakın.

## CorrelationPtr

Tip: MQPTR-output

İlinti göstergesi. Herhangi bir ilinti verisinin adresini belirtir. Bu gösterge daha sonra diğer OAM çağrılarına geçirilir.

## ComponentData

Tip: MQBYTE x ComponentDataUzunluk-giriş/çıkış

Bileşen verileri. Bu veriler, bu bileşen adına kuyruk yöneticisi tarafından tutulur; bu bileşen tarafından sağlanan işlevlerden herhangi biri tarafından yapılan değişiklikler korunur ve bu bileşenin sonraki işlevlerinden biri çağrıldığında sunulur.

Bu veri alanının uzunluğu, MQZ\_INIT\_AUTHORITY çağrısının ComponentDataLength parametresindeki kuyruk yöneticisi tarafından geçirilir.

## Devam

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Devam işareti. Aşağıdaki değerleri belirtebilirsiniz:

### MQZCI\_VARSAYILAN

Diğer bileşenlere bağlı olarak devam edin.

### MQZCI\_STOP

Sonraki bileşenle devam ettirmeyin.

## CompCode

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Tamamlanma kodu. Bu değer aşağıdaki değerlerden biri olmalıdır:

### MQCC\_OK

İşlem başarıyla tamamlandı.

### MQCC\_FAILED

Arama başarısız oldu.

## Neden

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Neden kodu ön eleme *CompCode*.

*CompCode* , MQCC\_OK ise:

### MQRC\_NONE

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_FAILED ise:

### MQRC\_SERVICE\_ERROR

(2289, X'8F1') Hizmet erişimi beklenmeyen bir hata oluştu.

Bu neden kodlarına ilişkin ek bilgi için [İletiler ve neden kodları](#) başlıklı konuya bakın.

## C çağırma

```
MQZ_AUTHENTICATE_USER (QMgrName, SecurityParms, ApplicationContext,  
IdentityContext, &CorrelationPtr, ComponentData,  
&Continuation, &CompCode, &Reason);
```

Hizmete geçirilen deęiřtirgeleri řu řekilde bildirin:

```
MQCHAR48 QMgrName; /* Queue manager name */  
MQCSP SecurityParms; /* Security parameters */  
MQZAC ApplicationContext; /* Application context */  
MQZIC IdentityContext; /* Identity context */  
MQPTR CorrelationPtr; /* Correlation pointer */  
MQBYTE ComponentData[n]; /* Component data */  
MQLONG Continuation; /* Continuation indicator set by  
component */  
MQLONG CompCode; /* Completion code */  
MQLONG Reason; /* Reason code qualifying CompCode */
```

### MQZ\_CHECK\_AUTHORITY-Denetim yetkisi

Bu iřlev bir MQZAS\_VERSION\_1 yetki hizmeti bileřeni tarafından saęlanır ve kuyruk yneticisi tarafından, belirli bir nesne zerinde belirli bir eylemi ya da eylemleri gerekleřtirme yetkisinin olup olmadıęını denetlemek iin kuyruk yneticisi tarafından bařlatılır.

Bu iřleve iliřkin iřlev tanıtıcısı (MQZEP iin) MQZID\_CHECK\_AUTHORITY olur.

### Szdizimi

```
MQZ_CHECK_AUTHORITY( QMgrName , EntityName , EntityType , ObjectName ,  
ObjectType , Authority , ComponentData , Continuation , CompCode , Reason )
```

### Parametreler

#### QMgrName

Tip: MQCHAR48 -giriř

Kuyruk yneticisi adı. Bileřeni aęıran kuyruk yneticisinin adı. Bu ad, deęiřtirgenin tam uzunluęuna kadar bořluklarla dolduruldu; ad boř deęerli bir karakterle sonlandırılmaz.

Kuyruk yneticisi adı, bileřene bilgi iin iletilir; yetki hizmeti arabirimi, bileřenin bu bileřeni tanımlı bir řekilde kullanmasını gerektirmez.

#### EntityName

Tip: MQCHAR12 -giriř

Varlık adı. Nesneye iliřkin yetkisi denetlenmek zere olan varlıęın adı. Dizilimin uzunluk st sınırı 12 karakterdir; bořlukla doldurulsa bořluk karakterinden daha kısıdır. Ad boř deęerli bir karakter tarafından sonlandırılmadı.

Bu varlıęın temel gvenlik hizmetiyle tanınması řart deęil. Bilinmiyorsa, denetim iin zel **kimse** grubunun (tm varlıkların ait olduęu varsayıldıęı) yetkilendirmeleri kullanılır. Tm boř ad geerli ve bu řekilde kullanılabilir.

#### EntityType

Tip: MQHOME-giriři

Varlık tipi. EntityName ile belirtilen varlık tipi. Bu deęer ařaęıdaki deęerlerden biri olmalıdır:

#### MQZAET\_PRINCIPAL

Mdr.

#### MQZAET\_GRP

Grup.

**ObjectName**

Tip: MQCHAR48 -giriş

Nesne adı. Erişimin gerekli olduğu nesnenin adı. Dizilimin uzunluk üst sınırı 48 karakterdir; boşlukla sağa yaslanmış ise, bu karakter uzunluğuna göre daha kısadır. Ad boş değerli bir karakter tarafından sonlandırılmadı.

*ObjectType* MQOT\_Q\_MGR ise, bu ad *QMgrName* ile aynıdır.

**ObjectType**

Tip: MQHOME-girişi

Nesne tipi. *ObjectName* tarafından belirtilen varlık tipi. Bu değer aşağıdaki değerlerden biri olmalıdır:

**MQOT\_AUTH\_INFO**

Kimlik doğrulama bilgileri.

**MQOT\_KANAL**

Kanal.

**MQOT\_CLNTCONN\_CHANNEL**

İstemci bağlantı kanalı.

**MQOT\_LISTENER**

Dinleyici.

**MQOT\_NAMELIST**

İsim listesi.

**MQOT\_PROCESS**

Süreç tanımlaması.

**MQOT\_Q**

Kuyruk.

**MQOT\_Q\_MGR**

Kuyruk yöneticisi.

**MQOT\_SERVICE**

Servis.

**Yetki**

Tip: MQHOME-girişi

Denetlenecek yetki. Bir yetki imleniyorsa, bu alan uygun yetki işlemine eşittir (MQZAO\_ \* sabiti). Birden fazla yetki denetleniyorsa, bu değer, ilgili MQZAO\_ \* değişmezlerinin bit ya da ya da alt öğelerinden biri olur.

MQI çağrılarının kullanılması için aşağıdaki yetkiler geçerlidir:

**MQZAO\_CONNECT**

MQCONN çağrısını kullanma yeteneği.

**MQZAO\_GÖZAT**

Bir göz atma seçeneği ile MQGET çağrısını kullanma yeteneği.

Bu, MQGET çağrısında MQGMO\_BROWSE\_FIRST, MQGMO\_BROWSE\_MSG\_UNDER\_CURSOR ya da MQGMO\_BROWSE\_NEXT seçeneğinin belirtilmesine olanak sağlar.

**MQZAO\_INPUT**

Müdür. MQGET çağrısını bir giriş seçeneği ile kullanma yeteneği.

Bu, MQOO\_INPUT\_SHARED, MQOO\_INPUT\_EXCLUSIVE YA DA MQOPEN çağrısında MQOO\_INPUT\_AS\_Q\_DEF seçeneğinin belirtilmesini sağlar.

**MQZAO\_OUTPUT**

MQPUT çağrısını kullanma yeteneği.

Bu, MQOPEN çağrısında MQOO\_OUTPUT seçeneğinin belirtilmesini sağlar.

**MQZAO\_SORGULAMA**

MQINQ çağrısını kullanma yeteneği.

Bu, MQOPEN çağrısında MQOO\_ENSORE seçeneğinin belirtilmesini sağlar.

**MQZAO\_SET**

MQSET çağrısını kullanma yeteneği.

Bu, MQOO\_SET seçeneğinin MQOPEN çağrısında belirtilmesini sağlar.

**MQZAO\_PASS\_IDENTITY\_CONTEXT**

Kimlik bağlamını geçirebilme yeteneği.

Bu, MQPUT çağrısında MQOO\_PASS\_IDENTITY\_CONTEXT seçeneğinin belirtilmesini ve MQPUT ve MQPUT1 çağrılarında MQPMO\_PASS\_IDENTITY\_CONTEXT seçeneğinin belirtilmesini sağlar.

**MQZAO\_PASS\_ALL\_CONTEXT**

Tüm bağlamı geçirebilme yeteneği.

Bu, MQPUT çağrısında MQOO\_PASS\_ALL\_CONTEXT seçeneğinin belirtilmesine ve MQPUT ve MQPUT1 çağrılarında belirtilecek MQPMO\_PASS\_ALL\_CONTEXT seçeneğinin geçerli olduğunu sağlar.

**MQZAO\_SET\_IDENTITY\_CONTEXT**

Kimlik bağlamını ayarlama yeteneği.

Bu, MQPUT çağrısında MQOO\_SET\_IDENTITY\_CONTEXT seçeneğinin belirtilmesini ve MQPUT ve MQPUT1 çağrılarında MQPMO\_SET\_IDENTITY\_CONTEXT seçeneğinin belirtilmesini sağlar.

**MQZAO\_SET\_ALL\_CONTEXT**

Tüm bağlamı ayarlayabilme yeteneği.

Bu, MQPUT çağrısında MQOO\_SET\_ALL\_CONTEXT seçeneğinin belirtilmesine ve MQPUT ve MQPUT1 çağrılarında belirtilecek MQPMO\_SET\_ALL\_CONTEXT seçeneğinin belirlenmesine olanak sağlar.

**MQZAO\_ALTERNATE\_USER\_AUTHORITY**

Diğer kullanıcı yetkisini kullanma yeteneği.

Bu, MQOPEN çağrısında MQOO\_ALTERNATE\_USER\_AUTHORITY seçeneğinin belirtilmesine ve MQPUT1 çağrısında MQPMO\_ALTERNATE\_USER\_AUTHORITY seçeneğinin belirtilmesine olanak tanır.

**MQZAO\_ALL\_MQI**

MQI yetkilerinin tümü.

Bu, tüm yetkilerin geçerli olduğunu sağlar.

Kuyruk yöneticisinin denetimi için aşağıdaki yetkiler geçerlidir:

**MQZAO\_CREATE**

Belirli bir tipte nesnelere oluşturma yeteneği.

**MQZAO\_DELETE**

Belirli bir nesneyi silme yeteneği.

**MQZAO\_GÖRÜNTÜLE**

Belirli bir nesnenin özniteliklerini görüntüleme yeteneği.

**MQZAO\_CHANGE**

Belirlenen bir nesnenin özniteliklerini değiştirme yeteneği.

**MQZAO\_CLEAR**

Belirtilen bir kuyruktan tüm iletileri silme yeteneği.

**MQZAO\_YETKİLİVER**

Belirli bir nesne için diğer kullanıcılara yetki verme yeteneği.

**MQZAO\_CONTROL**

Bir dinleyici, hizmet ya da istemci dışı kanal nesnesini başlatma ya da durdurma yeteneği ve istemci olmayan bir kanal nesnesini ping etme yeteneği.

**MQZAO\_CONTROL\_EXTENDED**

Bir sıra numarasını sıfırlamak ya da istemci dışındaki bir kanal nesnesindeki belirsiz bir iletiyi çözebilme yeteneği.

**MQZAO\_ALL\_ADMIN**

Kimlik bağlamını ayarlama yeteneği.

MQZAO\_CREATE dışında tüm denetim yetkileri.

Aşağıdaki yetkiler, MQI ' nin her iki kullanımı ve bir kuyruk yöneticisinin denetlenmesine uygulanır:

**MQZAO\_ALL**

Tüm yetkiler, MQZAO\_CREATE dışındaki tüm yetkiler.

**MQZAO\_NONE**

Yetki yok.

**ComponentData**

Tip: MQBYTE x ComponentDataUzunluk-giriş/çıkış

Bileşen verileri. Bu veriler, bu bileşen adına kuyruk yöneticisi tarafından tutulur; bu bileşen tarafından sağlanan işlevlerden herhangi biri tarafından yapılan değişiklikler korunur ve bu bileşen işlevlerinden biri bir sonraki defa çağrılır.

Bu veri alanının uzunluğu, MQZ\_INIT\_AUTHORITY çağrısının **ComponentDataLength** parametresindeki kuyruk yöneticisi tarafından geçirilir.

**Devam**

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Bileşene göre devam göstergesi kümesi. Aşağıdaki değerler belirlenebilir:

**MQZCI\_VARSAYILAN**

Devam eden kuyruk yöneticisine bağımlı.

MQZ\_CHECK\_AUTHORITY için bu, MQZCI\_STOP ile aynı etkiye sahiptir.

**MQZCI\_CONTINY**

Sonraki bileşenle devam edin.

**MQZCI\_STOP**

Sonraki bileşenle devam ettirmeyin.

Bir bileşene çağrı başarısız olursa (yani, *CompCode* , MQCC\_FAILED değerini döndürür) ve *Continuation* değıştirgesi MQZCI\_default ya da MQZCI\_CONTINUE ise, kuyruk yöneticisi varsa diğeri bileşenleri çağrırmaya devam eder.

Çağrı başarılı olursa (yani, *CompCode* , MQCC\_OK değerini döndürür), *Devam* ayarı ne olursa olsun, başka bileşen çağrılanmaz.

Çağrı başarısız olursa ve *Continuation* parametresi MQZCI\_STOP ise, başka bileşen çağrılamaz ve hata kuyruk yöneticisine döndürülür. Components have no knowledge of previous calls, so the *Devam* parameter is always set to MQZCI\_DEFAULT before the call.

**CompCode**

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Tamamlanma kodu. Bu değeri aşağıdaki değerilerden biri olmalıdır:

**MQCC\_OK**

İşlem başarıyla tamamlandı.

**MQCC\_FAILED**

Arama başarısız oldu.

## Neden

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Neden kodu ön eleme *CompCode*.

*CompCode* , MQCC\_OK ise:

### **MQRC\_NONE**

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_FAILED ise:

### **MQRC\_NOT\_YETKILI**

(2035, X'7F3') Erişim için yetkili değil.

### **MQRC\_SERVICE\_ERROR**

(2289, X'8F1') Hizmet erişimi beklenmeyen bir hata oluştu.

### **MQRC\_SERVICE\_NOT\_VAR**

(2285, X'8ED') Udering hizmeti kullanılmıyor.

Bu neden kodlarına ilişkin ek bilgi için [API tamamlama ve neden kodları](#) başlıklı konuya bakın.

## C çağırma

```
MQZ_CHECK_AUTHORITY (QMgrName, EntityName, EntityType, ObjectName,  
Objectype, Authority, ComponentData,  
&Continuation, &CompCode, &Reason);
```

Hizmete geçirilen parametreler aşağıdaki gibi bildirilir:

```
MQCHAR48  QMgrName;          /* Queue manager name */  
MQCHAR12  EntityName;       /* Entity name */  
MQLONG    EntityType;       /* Entity type */  
MQCHAR48  ObjectName;       /* Object name */  
MQLONG    Objectype;        /* Object type */  
MQLONG    Authority;        /* Authority to be checked */  
MQBYTE    ComponentData[n]; /* Component data */  
MQLONG    Continuation;     /* Continuation indicator set by  
                             component */  
MQLONG    CompCode;         /* Completion code */  
MQLONG    Reason;          /* Reason code qualifying CompCode */
```

## **MQZ\_CHECK\_AUTHORITY\_2 -Denetim yetkisi (genişletilmiş)**

Bu işlev bir MQZAS\_VERSION\_2 yetki hizmeti bileşeni tarafından sağlanır ve kuyruk yöneticisi tarafından, belirli bir nesne üzerinde belirli bir eylemi ya da eylemleri gerçekleştirme yetkisinin olup olmadığı denetlemek için kuyruk yöneticisi tarafından başlatılır.

Bu işleve ilişkin işlev tanıtıcısı (MQZEP için) MQZID\_CHECK\_AUTHORITY olur.

MQZ\_CHECK\_AUTHORITY\_2 , MQZ\_CHECK\_AUTHORITY gibi, ancak **EntityName** parametresi **EntityData** parametresiyle değiştiriliyor.

### **Sözdizimi**

```
MQZ_CHECK_AUTHORITY_2( QMgrName , EntityData , EntityType , ObjectName ,  
Objectype , Authority , ComponentData , Continuation , CompCode , Reason )
```

### **Parametreler**

#### **QMgrName**

Tip: MQCHAR48 -giriş

Kuyruk yöneticisi adı. Bileşeni çağıran kuyruk yöneticisinin adı. Bu ad, değiştirgenin tam uzunluğuna kadar boşluklarla dolduruldu; ad boş değerli bir karakterle sonlandırılmaz.

Kuyruk yöneticisi adı, bileşene bilgi için iletilir; yetki hizmeti arabirimi, bileşenin bu bileşeni tanımlı bir şekilde kullanmasını gerektirmez.

### EntityData

Tip: MQZED-giriş

Varlık verileri. Denetlenecek nesneye ilişkin yetkilendirmeye varlıkla ilgili veriler. Ayrıntılar için bkz. [“MQZED-Varlık tanımlayıcısı” sayfa 1653.](#)

Bu varlığın temel güvenlik hizmetiyle tanınması şart değil. Bilinmiyorsa, denetim için özel **kimse** grubunun (tüm varlıkların ait olduğu varsayıldığı) yetkilendirmeleri kullanılır. Tüm boş ad geçerli ve bu şekilde kullanılabilir.

### EntityType

Tip: MQHOME-girişi

Varlık tipi. *EntityData* tarafından belirtilen varlık tipi. Bu değer aşağıdaki değerlerden biri olmalıdır:

#### **MQZAET\_PRINCIPAL**

Müdür.

#### **MQZAET\_GRUP**

Grup.

### ObjectName

Tip: MQCHAR48 -giriş

Nesne adı. Erişimin gerekli olduğu nesnenin adı. Dizilimin uzunluk üst sınırı 48 karakterdir; boşlukla sağa yaslanmış ise, bu karakter uzunluğuna göre daha kısadır. Ad boş değerli bir karakter tarafından sonlandırılmadı.

*ObjectType* MQOT\_Q\_MGR ise, bu ad *QMgrName* ile aynıdır.

### ObjectType

Tip: MQHOME-girişi

Nesne tipi. *ObjectName* tarafından belirtilen varlık tipi. Bu değer aşağıdaki değerlerden biri olmalıdır:

#### **MQOT\_AUTH\_INFO**

Kimlik doğrulama bilgileri.

#### **MQOT\_KANAL**

Kanal.

#### **MQOT\_CLNTCONN\_CHANNEL**

İstemci bağlantı kanalı.

#### **MQOT\_LISTENER**

Dinleyici.

#### **MQOT\_NAMELIST**

İsim listesi.

#### **MQOT\_PROCESS**

Süreç tanımlaması.

#### **MQOT\_Q**

Kuyruk.

#### **MQOT\_Q\_MGR**

Kuyruk yöneticisi.

#### **MQOT\_SERVICE**

Servis.

#### **MQOT\_KONUSU**

Konu.

### Yetki

Tip: MQHOME-girişi

Denetlenecek yetki. Bir yetki imleniyorsa, bu alan uygun yetki işlemine eşittir (MQZAO\_ \* sabiti). Birden fazla yetki denetleniyorsa, bu değer, ilgili MQZAO\_ \* değişmezlerinin bit ya da ya da alt öğelerinden biri olur.

MQI çağrılarının kullanılması için aşağıdaki yetkiler geçerlidir:

#### **MQZAO\_CONNECT**

MQCONN çağrısını kullanma yeteneği.

#### **MQZAO\_GÖZAT**

Bir göz atma seçeneği ile MQGET çağrısını kullanma yeteneği.

Bu, MQGET çağrısında MQGMO\_BROWSE\_FIRST, MQGMO\_BROWSE\_MSG\_UNDER\_CURSOR ya da MQGMO\_BROWSE\_NEXT seçeneğinin belirtilmesine olanak sağlar.

#### **MQZAO\_INPUT**

Müdür. MQGET çağrısını bir giriş seçeneği ile kullanma yeteneği.

Bu, MQOO\_INPUT\_SHARED, MQOO\_INPUT\_EXCLUSIVE YA DA MQOPEN çağrısında MQOO\_INPUT\_AS\_Q\_DEF seçeneğinin belirtilmesini sağlar.

#### **MQZAO\_OUTPUT**

MQPUT çağrısını kullanma yeteneği.

Bu, MQOPEN çağrısında MQOO\_OUTPUT seçeneğinin belirtilmesini sağlar.

#### **MQZAO\_SORGULAMA**

MQINQ çağrısını kullanma yeteneği.

Bu, MQOPEN çağrısında MQOO\_ENSORE seçeneğinin belirtilmesini sağlar.

#### **MQZAO\_SET**

MQSET çağrısını kullanma yeteneği.

Bu, MQOO\_SET seçeneğinin MQOPEN çağrısında belirtilmesini sağlar.

#### **MQZAO\_PASS\_IDENTITY\_CONTEXT**

Kimlik bağlamını geçirebilme yeteneği.

Bu, MQPUT çağrısında MQOO\_PASS\_IDENTITY\_CONTEXT seçeneğinin belirtilmesini ve MQPUT ve MQPUT1 çağrılarında MQPMO\_PASS\_IDENTITY\_CONTEXT seçeneğinin belirtilmesini sağlar.

#### **MQZAO\_PASS\_ALL\_CONTEXT**

Tüm bağlamı geçirebilme yeteneği.

Bu, MQPUT çağrısında MQOO\_PASS\_ALL\_CONTEXT seçeneğinin belirtilmesine ve MQPUT ve MQPUT1 çağrılarında belirtilecek MQPMO\_PASS\_ALL\_CONTEXT seçeneğinin geçerli olduğunu sağlar.

#### **MQZAO\_SET\_IDENTITY\_CONTEXT**

Kimlik bağlamını ayarlama yeteneği.

Bu, MQPUT çağrısında MQOO\_SET\_IDENTITY\_CONTEXT seçeneğinin belirtilmesini ve MQPUT ve MQPUT1 çağrılarında MQPMO\_SET\_IDENTITY\_CONTEXT seçeneğinin belirtilmesini sağlar.

#### **MQZAO\_SET\_ALL\_CONTEXT**

Tüm bağlamı ayarlayabilme yeteneği.

Bu, MQPUT çağrısında MQOO\_SET\_ALL\_CONTEXT seçeneğinin belirtilmesine ve MQPUT ve MQPUT1 çağrılarında belirtilecek MQPMO\_SET\_ALL\_CONTEXT seçeneğinin belirlenmesine olanak sağlar.

#### **MQZAO\_ALTERNATE\_USER\_AUTHORITY**

Diğer kullanıcı yetkisini kullanma yeteneği.

Bu, MQOPEN çağrısında MQOO\_ALTERNATE\_USER\_AUTHORITY seçeneğinin belirtilmesine ve MQPUT1 çağrısında MQPMO\_ALTERNATE\_USER\_AUTHORITY seçeneğinin belirtilmesine olanak tanır.



**MQZAO\_ALL\_MQI**

MQI yetkilerinin tümü.

Bu, tüm yetkilerin geçerli olduğunu sağlar.

Kuyruk yöneticisinin denetimi için aşağıdaki yetkiler geçerlidir:

**MQZAO\_CREATE**

Belirli bir tipte nesnelere oluşturma yeteneği.

**MQZAO\_DELETE**

Belirli bir nesneyi silme yeteneği.

**MQZAO\_GÖRÜNTÜLE**

Belirli bir nesnenin özniteliklerini görüntüleme yeteneği.

**MQZAO\_CHANGE**

Belirlenen bir nesnenin özniteliklerini değiştirme yeteneği.

**MQZAO\_CLEAR**

Belirtilen bir kuyruktan tüm iletileri silme yeteneği.

**MQZAO\_YETKİLİVER**

Belirli bir nesne için diğer kullanıcılara yetki verme yeteneği.

**MQZAO\_CONTROL**

Bir dinleyici, hizmet ya da istemci dışı kanal nesnesini başlatma ya da durdurma yeteneği ve istemci olmayan bir kanal nesnesini ping etme yeteneği.

**MQZAO\_CONTROL\_EXTENDED**

Bir sıra numarasını sıfırlamak ya da istemci dışındaki bir kanal nesnesindeki belirsiz bir iletiyi çözebilme yeteneği.

**MQZAO\_ALL\_ADMIN**

Kimlik bağlamını ayarlama yeteneği.

MQZAO\_CREATE dışında tüm denetim yetkileri.

Aşağıdaki yetkiler, MQI ' nin her iki kullanımı ve bir kuyruk yöneticisinin denetlenmesine uygulanır:

**MQZAO\_ALL**

Tüm yetkiler, MQZAO\_CREATE dışındaki tüm yetkiler.

**MQZAO\_NONE**

Yetki yok.

**ComponentData**

Tip: MQBYTE x ComponentDataUzunluk-giriş/çıkış

Bileşen verileri. Bu veriler, bu bileşen adına kuyruk yöneticisi tarafından tutulur; bu bileşen tarafından sağlanan işlemlerden herhangi biri tarafından yapılan değişiklikler korunur ve bu bileşen işlemlerinden biri bir sonraki defa çağrılır.

Bu veri alanının uzunluğu, MQZ\_INIT\_AUTHORITY çağrısının **ComponentDataLength** parametresindeki kuyruk yöneticisi tarafından geçirilir.

**Devam**

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Bileşene göre devam göstergesi kümesi. Aşağıdaki değerler belirlenebilir:

**MQZCI\_VARSAYILAN**

Devam eden kuyruk yöneticisine bağımlı.

MQZ\_CHECK\_AUTHORITY için bu, MQZCI\_STOP ile aynı etkiye sahiptir.

**MQZCI\_CONTINY**

Sonraki bileşenle devam edin.

**MQZCI\_STOP**

Sonraki bileşenle devam ettirmeyin.

## CompCode

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Tamamlanma kodu. Bu deęer ařaęıdaki deęerlerden biri olmalıdır:

### **MQCC\_OK**

İřlem bařarıyla tamamlandı.

### **MQCC\_FAILED**

Arama bařarisız oldu.

## Neden

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Neden kodu ön eleme *CompCode*.

*CompCode* , MQCC\_OK ise:

### **MQRC\_NONE**

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_FAILED ise:

### **MQRC\_NOT\_YETKILI**

(2035, X'7F3') Eriřim için yetkili deęil.

### **MQRC\_SERVICE\_ERROR**

(2289, X'8F1') Hizmet eriřimi beklenmeyen bir hata oluřtu.

### **MQRC\_SERVICE\_NOT\_VAR**

(2285, X'8ED') Undering hizmeti kullanılmıyor.

Bu neden kodlarına iliřkin ek bilgi için [API tamamlama ve neden kodları](#)bařlıklı konuya bakın.

## C çağırma

```
MQZ_CHECK_AUTHORITY_2 (QMgrName, &EntityData, EntityType,  
ObjectName, ObjectType, Authority, ComponentData,  
&Continuation, &CompCode, &Reason);
```

Hizmete geçirilen parametreler ařaęıdaki gibi bildirilir:

```
MQCHAR48 QMgrName;          /* Queue manager name */  
MQZED EntityData;          /* Entity data */  
MQLONG EntityType;         /* Entity type */  
MQCHAR48 ObjectName;       /* Object name */  
MQLONG ObjectType;         /* Object type */  
MQLONG Authority;          /* Authority to be checked */  
MQBYTE ComponentData[n];   /* Component data */  
MQLONG Continuation;       /* Continuation indicator set by  
component */  
MQLONG CompCode;           /* Completion code */  
MQLONG Reason;             /* Reason code qualifying CompCode */
```

## **MQZ\_CHECK\_PRIVATIONAL-Kullanıcının ayrıcalıklı olup olmadığını denetle**

Bu iřlev bir MQZAS\_VERSION\_6 yetkilendirme hizmeti bileřeni tarafından saęlanır ve belirtilen kullanıcının ayrıcalıklı bir kullanıcı olup olmadığını belirlemek için kuyruk yöneticisi tarafından çağrılır.

Bu iřleve iliřkin iřlev tanıtıcısı (MQZEP için) MQZID\_CHECK\_PRIVACITEDIR.

## Sözdizimi

```
MQZ_CHECK_PRIVILEGED( QMgrName , EntityData , EntityType , ComponentData ,  
Continuation , CompCode , Reason )
```

## Parametreler

### QMgrName

Tip: MQCHAR48 -giriş

Kuyruk yöneticisi adı. Bileşeni çağıran kuyruk yöneticisinin adı. Bu ad, değıştirgenin tam uzunluđuna kadar boşluklarla dolduruldu; ad boş değeri bir karakterle sonlandırılmaz.

Kuyruk yöneticisi adı, bileşene bilgi için iletilir; yetki hizmeti arabirimi, bileşenin bu bileşeni tanımlı bir şekilde kullanmasını gerektirmez.

### EntityData

Tip: MQZED-giriş

Varlık verileri. Denetlenecek olan varlıkla ilgili veriler. Daha fazla bilgi için bkz. [“MQZED-Varlık tanımlayıcısı” sayfa 1653.](#)

### EntityType

Tip: MQHOME-giriş

Varlık tipi. EntityDatatarafından belirtilen varlık tipi. Bu değeri aşıđıdaki değeriilerden biri olmalıdır:

#### **MQZAET\_PRINCIPAL**

Müdür.

#### **MQZAET\_GRP**

Grup.

### ComponentData

Tip: MQBYTEXComponentDataLength -giriş/çıkış

Bileşen verileri. Bu veriler, bu bileşen adına kuyruk yöneticisi tarafından tutulur; bu bileşen tarafından sağlanan işlevlerden herhangi biri tarafından yapılan değışiklikler korunur ve bu bileşen işlevlerinden biri bir sonraki defa çağrılır.

Bu veri alanının uzunluđu, MQZ\_INIT\_AUTHORITY çağrısının **ComponentDataLength** parametresindeki kuyruk yöneticisi tarafından geçirilir.

### Devam

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Bileşene göre devam göstergesi kümesi. Aşıđıdaki değeriiler belirlenebilir:

#### **MQZCI\_VARSAYILAN**

Devam eden kuyruk yöneticisine bağımlı.

MQZ\_CHECK\_AUTHORITY için bu, MQZCI\_STOP ile aynı etkiye sahiptir.

#### **MQZCI\_CONTINY**

Sonraki bileşenle devam edin.

#### **MQZCI\_STOP**

Sonraki bileşenle devam ettirmeyin.

Bir bileşene çağrı başarısız olursa (yani, *CompCode* , MQCC\_FAILED değeriini döndürür) ve *Continuation* değıştirgesi MQZCI\_default ya da MQZCI\_CONTINUE ise, kuyruk yöneticisi varsa diđer bileşenleri çağrırmaya devam eder.

Çağrı başarılı olursa (yani, *CompCode* , MQCC\_OK değeriini döndürür), *Devam* ayarı ne olursa olsun, başka bileşen çağrılanmaz.

Çağrı başarısız olursa ve *Continuation* parametresi MQZCI\_STOP ise, başka bileşen çağrılamaz ve hata kuyruk yöneticisine döndürülür. Components have no knowledge of previous calls, so the *Devam* parameter is always set to MQZCI\_DEFAULT before the call.

### CompCode

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Tamamlanma kodu. Bu değeri aşıđıdaki değeriilerden biri olmalıdır:

## **MQCC\_OK**

İşlem başarıyla tamamlandı.

## **MQCC\_FAILED**

Arama başarısız oldu.

### **Neden**

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Neden kodu ön eleme *CompCode*.

*CompCode* , MQCC\_OK ise:

## **MQRC\_NONE**

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_FAILED ise:

## **MQRC\_NOT\_AYRICALIKSIZ**

(2584, X'A18') Bu kullanıcı ayrıcalıklı bir kullanıcı kimliği değil.

## **MQRC\_UNKNOWN\_ENTITY**

(2292, X'8F4') Hizmet için bilinmeyen varlık.

## **MQRC\_SERVICE\_ERROR**

(2289, X'8F1') Hizmet erişimi beklenmeyen bir hata oluştu.

## **MQRC\_SERVICE\_NOT\_VAR**

(2285, X'8ED') Undering hizmeti kullanılmıyor.

Bu neden kodlarına ilişkin ek bilgi için [API tamamlama ve neden kodları](#) başlıklı konuya bakın.

## **C çağırma**

```
MQZ_CHECK_PRIVILEGED (QMgrName, &EntityData, EntityType,  
                      ComponentData, &Continuation,  
                      &CompCode, &Reason);
```

Hizmete geçirilen parametreler aşağıdaki gibi bildirilir:

```
MQCHAR48  QMgrName;           /* Queue manager name */  
MQZED     EntityData;        /* Entity name */  
MQLONG    EntityType;        /* Entity type */  
MQBYTE    ComponentData[n]; /* Component data */  
MQLONG    Continuation;      /* Continuation indicator set by  
                             component */  
MQLONG    CompCode;          /* Completion code */  
MQLONG    Reason;            /* Reason code qualifying CompCode */
```

## **MQZ\_COPY\_ALL\_AUTHORITY-Tüm yetkiyi kopyala**

Bu işlev, bir yetkilendirme hizmeti bileşeni tarafından sağlanır. Bir başvuru nesnesi için yürürlükte olan yetkilerin tümünü başka bir nesneye kopyalamak için kuyruk yöneticisi tarafından başlatılır.

Bu işleve ilişkin işlev tanıtıcısı (MQZEP için) MQZID\_COPY\_ALL\_AUTHORITY olur.

### **Sözdizimi**

`MQZ_COPY_ALL_AUTHORITY( QMgrName , RefObjectName , ObjectName , ObjectType ,  
ComponentData , Continuation , CompCode , Reason )`

### **Parametreler**

#### **QMgrName**

Tip: MQCHAR48 -giriş

Kuyruk yöneticisi adı. Bileşeni çağıran kuyruk yöneticisinin adı. Bu ad, değıştirgenin tam uzunluđuna kadar boşluklarla dolduruldu; ad boş değeri bir karakterle sonlandırılmaz.

Kuyruk yöneticisi adı, bileşene bilgi için iletilir; yetki hizmeti arabirimi, bileşenin bu bileşeni tanımlı bir şekilde kullanmasını gerektirmez.

### **RefObjectAdı**

Tip: MQCHAR48 -giriş

Başvuru nesnesi adı. Başvuru nesnesinin adı, kopyalanacak yetkiler. Dizilimin uzunluk üst sınırı 48 karakterdir; boşlukla sağa yaslanmış ise, bu karakter uzunluđuna göre daha kısadır. Ad boş değeri bir karakter tarafından sonlandırılmadı.

### **ObjectName**

Tip: MQCHAR48 -giriş

Nesne adı. Erişimlerin ayarlanacağı nesnenin adı. Dizilimin uzunluk üst sınırı 48 karakterdir; boşlukla sağa yaslanmış ise, bu karakter uzunluđuna göre daha kısadır. Ad boş değeri bir karakter tarafından sonlandırılmadı.

### **ObjectType**

Tip: MQHOME-giriş

Nesne tipi. *RefObjectName* ve *ObjectName* tarafından belirtilen varlık tipi. Bu değeri aşağıdaki değeriilerden biri olmalıdır:

#### **MQOT\_AUTH\_INFO**

Kimlik doğrulama bilgileri.

#### **MQOT\_KANAL**

Kanal.

#### **MQOT\_CLNTCONN\_CHANNEL**

İstemci bağlantı kanalı.

#### **MQOT\_LISTENER**

Dinleyici.

#### **MQOT\_NAMELIST**

İsim listesi.

#### **MQOT\_PROCESS**

Süreç tanımlaması.

#### **MQOT\_Q**

Kuyruk.

#### **MQOT\_Q\_MGR**

Kuyruk yöneticisi.

#### **MQOT\_SERVICE**

Servis.

#### **MQOT\_KONUSU**

Konu.

### **ComponentData**

Tip: MQBYTEExComponentDataLength -giriş/çıkış

Bileşen verileri. Bu veriler, bu bileşen adına kuyruk yöneticisi tarafından tutulur; bu bileşen tarafından sağlanan işlevlerden herhangi biri tarafından yapılan değışiklikler korunur ve bu bileşen işlevlerinden biri bir sonraki defa çağrılır.

Bu veri alanının uzunluđu, MQZ\_INIT\_AUTHORITY çağrısının ComponentDataLength parametresindeki kuyruk yöneticisi tarafından geçirilir.

### **Devam**

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Bileşene göre devam göstergesi kümesi. Aşağıdaki değeriiler belirlenebilir:

### **MQZCI\_VARSAYILAN**

Devam eden kuyruk yöneticisine bağımlı.

MQZ\_CHECK\_AUTHORITY için bu, MQZCI\_STOP ile aynı etkiye sahiptir.

### **MQZCI\_CONTINY**

Sonraki bileşenle devam edin.

### **MQZCI\_STOP**

Sonraki bileşenle devam ettirmeyin.

### **CompCode**

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Tamamlanma kodu. Bu değer aşağıdaki değerlerden biri olmalıdır:

### **MQCC\_OK**

İşlem başarıyla tamamlandı.

### **MQCC\_FAILED**

Arama başarısız oldu.

### **Neden**

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Neden kodu ön eleme *CompCode*.

*CompCode* , MQCC\_OK ise:

### **MQRC\_NONE**

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_FAILED ise:

### **MQRC\_SERVICE\_ERROR**

(2289, X'8F1') Hizmet erişimi beklenmeyen bir hata oluştu.

### **MQRC\_SERVICE\_NOT\_VAR**

(2285, X'8ED') Undering hizmeti kullanılmıyor.

### **MQRC\_UNKNOWN\_REF\_OBJECT**

(2294, X'8F6') Başvuru nesnesi bilinmiyor.

Bu neden kodlarına ilişkin ek bilgi için [API tamamlama ve neden kodları](#) başlıklı konuya bakın.

## **C çağırma**

```
MQZ_COPY_ALL_AUTHORITY (QMgrName, RefObjectName, ObjectName, ObjectType,  
ComponentData, &Continuation, &CompCode,  
&Reason);
```

Hizmete geçirilen parametreler aşağıdaki gibi bildirilir:

```
MQCHAR48 QMgrName;          /* Queue manager name */  
MQCHAR48 RefObjectName;     /* Reference object name */  
MQCHAR48 ObjectName;       /* Object name */  
MQLONG   ObjectType;       /* Object type */  
MQBYTE   ComponentData[n]; /* Component data */  
MQLONG   Continuation;     /* Continuation indicator set by  
                           component */  
MQLONG   CompCode;        /* Completion code */  
MQLONG   Reason;         /* Reason code qualifying CompCode */
```

## **MQZ\_DELETE\_AUTHORITY-Silme yetkisi**

Bu işlev, bir yetki hizmeti bileşeni tarafından sağlanır ve kuyruk yöneticisi tarafından, belirtilen nesneyle ilişkili tüm yetkileri silmek için başlatılır.

Bu işleve ilişkin işlev tanıtıcısı (MQZEP için) MQZID\_DELETE\_AUTHORITY 'dir.

## Sözdizimi

`MQZ_DELETE_AUTHORITY( QMgrName , ObjectName , ObjectType , ComponentData , Continuation , CompCode , Reason )`

## Parametreler

### QMgrName

Tip: MQCHAR48 -giriş

Kuyruk yöneticisi adı. Bileşeni çağıran kuyruk yöneticisinin adı. Bu ad, değiştirgenin tam uzunluğuna kadar boşluklarla dolduruldu; ad boş değerli bir karakterle sonlandırılmaz.

Kuyruk yöneticisi adı, bileşene bilgi için iletilir; yetki hizmeti arabirimi, bileşenin bu bileşeni tanımlı bir şekilde kullanmasını gerektirmez.

### ObjectName

Tip: MQCHAR48 -giriş

Nesne adı. Erişimlerin silineceği nesnenin adı. Dizilimin uzunluk üst sınırı 48 karakterdir; boşlukla sağa yaslanmış ise, bu karakter uzunluğuna göre daha kısadır. Ad boş değerli bir karakter tarafından sonlandırılmadı.

*ObjectType* MQOT\_Q\_MGR ise, bu ad *QMgrName* ile aynıdır.

### ObjectType

Tip: MQHOME-girişi

Nesne tipi. *ObjectName* tarafından belirtilen varlık tipi. Bu değer aşağıdaki değerlerden biri olmalıdır:

#### **MQOT\_AUTH\_INFO**

Kimlik doğrulama bilgileri.

#### **MQOT\_KANAL**

Kanal.

#### **MQOT\_CLNTCONN\_CHANNEL**

İstemci bağlantı kanalı.

#### **MQOT\_LISTENER**

Dinleyici.

#### **MQOT\_NAMELIST**

İsim listesi.

#### **MQOT\_PROCESS**

Süreç tanımlaması.

#### **MQOT\_Q**

Kuyruk.

#### **MQOT\_Q\_MGR**

Kuyruk yöneticisi.

#### **MQOT\_SERVICE**

Servis.

#### **MQOT\_KONUSU**

Konu.

### ComponentData

Tip: MQBYTE x ComponentDataUzunluk-giriş/çıkış

Bileşen verileri. Bu veriler, bu bileşen adına kuyruk yöneticisi tarafından tutulur; bu bileşen tarafından sağlanan işlevlerden herhangi biri tarafından yapılan değişiklikler korunur ve bu bileşen işlevlerinden biri bir sonraki defa çağrılır.

Bu veri alanının uzunluğu, MQZ\_INIT\_AUTHORITY çağrısının ComponentDataLength parametresindeki kuyruk yöneticisi tarafından geçirilir.

## Devam

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Bileşene göre devam göstergesi kümesi. Aşağıdaki değerler belirlenebilir:

### **MQZCI\_VARSAYILAN**

Devam eden kuyruk yöneticisine bağımlı.

MQZ\_CHECK\_AUTHORITY için bu, MQZCI\_STOP ile aynı etkiye sahiptir.

### **MQZCI\_CONTINY**

Sonraki bileşenle devam edin.

### **MQZCI\_STOP**

Sonraki bileşenle devam ettirmeyin.

## CompCode

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Tamamlanma kodu. Bu değer aşağıdaki değerlerden biri olmalıdır:

### **MQCC\_OK**

İşlem başarıyla tamamlandı.

### **MQCC\_FAILED**

Arama başarısız oldu.

## Neden

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Neden kodu ön eleme *CompCode*.

*CompCode* , MQCC\_OK ise:

### **MQRC\_NONE**

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_FAILED ise:

### **MQRC\_SERVICE\_ERROR**

(2289, X'8F1') Hizmet erişimi beklenmeyen bir hata oluştu.

### **MQRC\_SERVICE\_NOT\_VAR**

(2285, X'8ED') Undering hizmeti kullanılmıyor.

Bu neden kodlarına ilişkin ek bilgi için [API tamamlama ve neden kodları](#) başlıklı konuya bakın.

## C çağırma

```
MQZ_DELETE_AUTHORITY (QMgrName, ObjectName, ObjectType, ComponentData,  
&Continuation, &CompCode, &Reason);
```

Hizmete geçirilen parametreler aşağıdaki gibi bildirilir:

```
MQCHAR48 QMgrName;          /* Queue manager name */  
MQCHAR48 ObjectName;        /* Object name */  
MQLONG   ObjectType;        /* Object type */  
MQBYTE   ComponentData[n]; /* Component data */  
MQLONG   Continuation;      /* Continuation indicator set by  
                             component */  
MQLONG   CompCode;          /* Completion code */  
MQLONG   Reason;            /* Reason code qualifying CompCode */
```

## **MQZ\_ENUMERATE\_AUTHORITY\_DATA-Yetki verilerini numaralandırır**

Bu işlev bir MQZAS\_VERSION\_4 yetki hizmeti bileşeni tarafından sağlanır ve kuyruk yöneticisi tarafından, ilk çağrıda belirtilen seçim ölçütleriyle eşleşen tüm yetki verilerini almak için yinelemeli olarak başlatılır.



Bu işleve ilişkin işlev tanıtıcısı (MQZEP için) MQZID\_ENUMERATE\_AUTHORITY\_DATA olur.

## Sözdizimi

MQZ\_ENUMERATE\_AUTHORITY\_DATA( *QMgrName* , *StartEnumeration* , *Filter* , *AuthorityBufferLength* , *AuthorityBuffer* , *AuthorityDataLength* , *ComponentData* , *Continuation* , *CompCode* , *Reason* )

## Parametreler

### QMgrName

Tip: MQCHAR48 -giriş

Kuyruk yöneticisi adı. Bileşeni çağıran kuyruk yöneticisinin adı. Bu ad, değiştirgenin tam uzunluğuna kadar boşluklarla dolduruldu; ad boş değerli bir karakterle sonlandırılmaz.

Kuyruk yöneticisi adı, bileşene bilgi için iletilir; yetki hizmeti arabirimi, bileşenin bu bileşeni tanımlı bir şekilde kullanmasını gerektirmez.

### StartEnumeration

Tip: MQHOME-girişi

Çağrıların sıralamayı başlatabileceğini belirten işaret. Bu, çağrı işlemi için yetki verilerinin sıralamasını başlatabileceğini ya da önceki bir MQZ\_ENUMERATE\_AUTHORITY\_DATA çağrısıyla başlatılan yetki verilerinin sıralamasına devam edip etmeyeceğini gösterir. Değer, aşağıdaki değerlerden biridir:

#### MQZSE\_START

Sıralamayı başlat. Çağrı, yetki verisinin sıralamasını başlatmak için bu değerle başlatılır. **Filter** parametresi, bu ve art arda gelen çağrılar döndürdüğü yetki verilerini seçmek için kullanılacak seçim ölçütlerini belirtir.

#### MQZSE\_DEVAM

Sıralamaya devam et. Çağrı, yetki verilerinin sıralamasını devam ettirmek için bu değer ile başlatılır. **Filter** parametresi bu durumda yoksayılr ve boş (null) gösterge olarak belirtilebilir (seçim ölçütleri, *StartEnumeration* tarafından MQZSE\_START değerine ayarlanmış, çağrıyla belirtilen **Filter** parametresi tarafından belirlenir).

### Filtre

Tip: MQZAD-input

Süzgeç uygula. *StartEnumeration* MQZSE\_START ise, *Filter* döndürülebilme için yetki verilerini seçmek için kullanılacak seçim ölçütlerini belirtir. *Filter* boş değerli göstergedir, seçim ölçütü kullanılmazsa, tüm yetki verileri döndürülür. Kullanılabilecek seçim ölçütlerine ilişkin ayrıntılar için bkz. "MQZAD-Yetki verileri" sayfa 1650 .

*StartEnumeration* MQZSE\_CONTINUE ise, *Filter* yoksayılr ve boş değerli gösterge olarak belirtilebilir.

### AuthorityBufferUzunluğu

Tip: MQHOME-girişi

*AuthorityBuffer* uzunluğu. Bu, **AuthorityBuffer** parametresindeki bayt cinsinden uzunluğudur. Yetki arabelleği, döndürülebilme için gereken verileri sığdırabilme için yeterince büyük olmalıdır.

### AuthorityBuffer

Tip: MQZAD-output

Yetki verileri. Bu, yetki verilerinin döndürüldüğü arabelleğidir. Arabellek, bir MQZAD yapısını, MQZED yapısını ve tanımlı en uzun varlık adını ve en uzun etki alanı adını barındıracak kadar büyük olmalıdır.

**Not:** Not: Bu değiştirge, arabellek başlangıcındaki MQZAD ' in her zaman oluştuğu için, MQZAD olarak tanımlıdır. Ancak arabellek bir MQZAD olarak bildirilirse, arabellek çok küçük olur-MQZAD, MQZED, artı varlık ve etki alanı adlarını barındırabilmesi için, bir MQZAD ' dan büyük olmalıdır.

### AuthorityDataUzunluğu

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

*AuthorityBuffer*' ta döndürülen veri uzunluğu. Yetki arabelleği çok küçükse, *AuthorityDataLength* gerekli arabelleğin uzunluğuna ayarlanır ve çağrı, MQCC\_FAILED tamamlama kodunu döndürür ve neden kodu MQRC\_BUFFER\_LENGTH\_ERROR değerini döndürür.

### ComponentData

Tip: MQBYTE x ComponentDataUzunluk-giriş/çıkış

Bileşen verileri. Bu veriler, bu bileşen adına kuyruk yöneticisi tarafından tutulur; bu bileşen tarafından sağlanan işlemlerden herhangi biri tarafından yapılan değişiklikler korunur ve bu bileşen işlemlerinden biri bir sonraki defa çağrılır.

Bu veri alanının uzunluğu, MQZ\_INIT\_AUTHORITY çağrısının ComponentDataLength parametresindeki kuyruk yöneticisi tarafından geçirilir.

### Devam

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Bileşene göre devam göstergesi kümesi. Aşağıdaki değerler belirlenebilir:

#### **MQZCI\_VARSAYILAN**

Devam eden kuyruk yöneticisine bağlıdır.

MQZ\_ENUMERATE\_AUTHORITY\_DATA için, bu, MQZCI\_CONTINUE ile aynı etkiye sahiptir.

#### **MQZCI\_CONTINY**

Sonraki bileşenle devam edin.

#### **MQZCI\_STOP**

Sonraki bileşenle devam ettirmeyin.

### CompCode

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Tamamlanma kodu. Bu değer aşağıdaki değerlerden biri olmalıdır:

#### **MQCC\_OK**

İşlem başarıyla tamamlandı.

#### **MQCC\_FAILED**

Arama başarısız oldu.

### Neden

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Neden kodu ön eleme *CompCode*.

*CompCode* , MQCC\_OK ise:

#### **MQRC\_NONE**

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_FAILED ise:

#### **MQRC\_BUFFER\_LENGTH\_ERROR**

(2005, X'7D5') Arabellek uzunluğu parametresi geçerli değil.

#### **MQRC\_NO\_DATA\_AVAILABLE**

(2379, X'94B') Kullanılabilir veri yok.

#### **MQRC\_SERVICE\_ERROR**

(2289, X'8F1') Hizmet erişimi beklenmeyen bir hata oluştu.

Bu neden kodlarına ilişkin ek bilgi için [API tamamlama ve neden kodları](#) başlıklı konuya bakın.

## C çağırma

```
MQZ_ENUMERATE_AUTHORITY_DATA (QMgrName, StartEnumeration, &Filter,  
                                AuthorityBufferLength,  
                                &AuthorityBuffer,  
                                &AuthorityDataLength, ComponentData,
```

```
&Continuation, &CompCode,  
&Reason);
```

Hizmete geçirilen parametreler aşağıdaki gibi bildirilir:

```
MQCHAR48  QMgrName;          /* Queue manager name */  
MQLONG    StartEnumeration; /* Flag indicating whether call should  
start enumeration */  
MQZAD     Filter;          /* Filter */  
MQLONG    AuthorityBufferLength; /* Length of AuthorityBuffer */  
MQZAD     AuthorityBuffer; /* Authority data */  
MQLONG    AuthorityDataLength; /* Length of data returned in  
AuthorityBuffer */  
MQBYTE    ComponentData[n]; /* Component data */  
MQLONG    Continuation;    /* Continuation indicator set by  
component */  
MQLONG    CompCode;        /* Completion code */  
MQLONG    Reason;          /* Reason code qualifying CompCode */
```

## MQZ\_FREE\_USER-Serbest kullanıcı

Bu işlev bir MQZAS\_VERSION\_5 yetki hizmeti bileşeni tarafından sağlanır ve kuyruk yöneticisi tarafından ilişkili ayrılmış kaynak serbest olacak şekilde başlatılır.

Bir uygulama tüm kullanıcı bağlamları altında çalışmayı bitirdiğinde (örneğin, bir MQDISC MQI çağırısı sırasında) başlatılır.

Bu işleve ilişkin işlev tanıtıcısı (MQZEP için) MQZID\_FREE\_USER.

## Sözdizimi

```
MQZ_FREE_USER( QMgrName , FreeParms , ComponentData , Continuation , CompCode ,  
Reason )
```

## Parametreler

### QMgrName

Tip: MQCHAR48 -giriş

Kuyruk yöneticisi adı. Bileşeni çağırın kuyruk yöneticisinin adı. Bu ad, değiştirgenin tam uzunluğuna kadar boşluklarla dolduruldu; ad boş değerli bir karakterle sonlandırılmaz.

Kuyruk yöneticisi adı, bileşene bilgi için iletilir; yetki hizmeti arabirimi, bileşenin bu bileşeni tanımlı bir şekilde kullanmasını gerektirmez.

### FreeParms

Tip: MQZFP-input

Serbest parametreler. Serbest bırakılacak kaynakla ilgili verileri içeren bir yapı. Ayrıntılar için bkz. "MQZFP-Serbest değiştirgeler" sayfa 1655.

### ComponentData

Tip: MQBYTE x ComponentDataUzunluk-giriş/çıkış

Bileşen verileri. Bu veriler, bu bileşen adına kuyruk yöneticisi tarafından tutulur; bu bileşen tarafından sağlanan işlevlerden herhangi biri tarafından yapılan değişiklikler korunur ve bu bileşen işlevlerinden biri bir sonraki defa çağrılır.

Bu veri alanının uzunluğu, MQZ\_INIT\_AUTHORITY çağırısının ComponentDataLength parametresindeki kuyruk yöneticisi tarafından geçirilir.

### Devam

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Devam işareti. Aşağıdaki değerler belirlenebilir:

### MQZCI\_VARSAYILAN

Diğer bileşenlere bağlı olarak devam edin.

## **MQZCI\_STOP**

Sonraki bileşenle devam ettirmeyin.

### **CompCode**

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Tamamlanma kodu. Bu değer aşağıdaki değerlerden biri olmalıdır:

### **MQCC\_OK**

İşlem başarıyla tamamlandı.

### **MQCC\_FAILED**

Arama başarısız oldu.

### **Neden**

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Neden kodu ön eleme *CompCode*.

*CompCode* , MQCC\_OK ise:

### **MQRC\_NONE**

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_FAILED ise:

### **MQRC\_SERVICE\_ERROR**

(2289, X'8F1') Hizmet erişimi beklenmeyen bir hata oluştu.

Bu neden kodlarına ilişkin ek bilgi için [API tamamlama ve neden kodları](#) başlıklı konuya bakın.

## **C çağırma**

```
MQZ_AUTHENTICATE_USER (QMgrName, SecurityParms, ApplicationContext,  
IdentityContext, CorrelationPtr, ComponentData,  
&Continuation, &CompCode, &Reason);
```

Hizmete geçirilen parametreler aşağıdaki gibi bildirilir:

```
MQCHAR48  QMgrName;          /* Queue manager name */  
MQZFP     FreeParms;       /* Resource to be freed */  
MQBYTE    ComponentData[n]; /* Component data */  
MQLONG    Continuation;    /* Continuation indicator set by  
                           component */  
MQLONG    CompCode;        /* Completion code */  
MQLONG    Reason;          /* Reason code qualifying CompCode */
```

## **MQZ\_GET\_AUTHORITY-Yetki al**

Bu işlev bir MQZAS\_VERSION\_1 yetki hizmeti bileşeni tarafından sağlanır ve kuyruk yöneticisi tarafından, bir varlığın, birincil kullanıcının üyesi olduğu grupların sahip olduğu (varlık bir birincil kullanıcı) yetkiler de dahil olmak üzere, belirtilen nesneye erişmek için sahip olduğu yetkiyi almak üzere başlatılmış olur. Genel tanımlardaki yetkiler, döndürülen yetki kümesine eklenir.

Bu işleve ilişkin işlev tanıtıcısı (MQZEP için) MQZID\_GET\_AUTHORITY olur.

### **Sözdizimi**

```
MQZ_GET_AUTHORITY( QMgrName , EntityName , EntityType , ObjectName ,  
ObjectType , Authority , ComponentData , Continuation , CompCode , Reason )
```

### **Parametreler**

#### **QMgrName**

Tip: MQCHAR48 -giriş

Kuyruk yöneticisi adı. Bileşeni çağıran kuyruk yöneticisinin adı. Bu ad, değıştirgenin tam uzunluđuna kadar boşluklarla dolduruldu; ad boş değeri bir karakterle sonlandırılmaz.

Kuyruk yöneticisi adı, bileşene bilgi için iletilir; yetki hizmeti arabirimi, bileşenin bu bileşeni tanımlı bir şekilde kullanmasını gerektirmez.

### **EntityName**

Tip: MQCHAR12 -giriş

Varlık adı. Nesneye erişimi alınmak üzere olan varlığın adı. Dizilimin uzunluk üst sınırı 12 karakterdir; boşlukla doldurursa boşluk karakterinden daha kısadır. Ad boş değeri bir karakter tarafından sonlandırılmadı.

### **EntityType**

Tip: MQHOME-giriş

Varlık tipi. *EntityName* tarafından belirtilen varlık tipi. Bu değeri aşağıdaki değerlerden biri olmalıdır:

#### **MQZAET\_PRINCIPAL**

Müdür.

#### **MQZAET\_GRP**

Grup.

### **ObjectName**

Tip: MQCHAR48 -giriş

Nesne adı. Erişimin alınmak üzere olduđu nesnenin adı. Dizilimin uzunluk üst sınırı 48 karakterdir; boşlukla sağa yaslanmış ise, bu karakter uzunluđuna göre daha kısadır. Ad boş değeri bir karakter tarafından sonlandırılmadı.

*ObjectType* MQOT\_Q\_MGR ise, bu ad *QMgrName* ile aynıdır.

### **ObjectType**

Tip: MQHOME-giriş

Nesne tipi. *ObjectName* tarafından belirtilen varlık tipi. Bu değeri aşağıdaki değerlerden biri olmalıdır:

#### **MQOT\_AUTH\_INFO**

Kimlik doğrulama bilgileri.

#### **MQOT\_KANAL**

Kanal.

#### **MQOT\_CLNTCONN\_CHANNEL**

İstemci bağlantı kanalı.

#### **MQOT\_LISTENER**

Dinleyici.

#### **MQOT\_NAMELIST**

İsim listesi.

#### **MQOT\_PROCESS**

Süreç tanımlaması.

#### **MQOT\_Q**

Kuyruk.

#### **MQOT\_Q\_MGR**

Kuyruk yöneticisi.

#### **MQOT\_SERVICE**

Servis.

#### **MQOT\_KONUSU**

Konu.

### **Yetki**

Tip: MQHOME-giriş

Varlık yetkisi. Varlığın bir yetkisi varsa, bu alan uygun yetkilendirme işlemine eşittir (MQZAO\_ \* sabiti). Birden fazla yetkiniz varsa, bu alan, ilgili MQZAO\_ \* değişmezlerinin bit ya da bit ya da bitmezidir.

### ComponentData

Tip: MQBYTE xComponentDataUzunluk-giriş/çıkış

Bileşen verileri. Bu veriler, bu bileşen adına kuyruk yöneticisi tarafından tutulur; bu bileşen tarafından sağlanan işlemlerden herhangi biri tarafından yapılan değişiklikler korunur ve bu bileşen işlemlerinden biri bir sonraki defa çağrılır.

Bu veri alanının uzunluğu, MQZ\_INIT\_AUTHORITY çağrısının **ComponentDataLength** parametresindeki kuyruk yöneticisi tarafından geçirilir.

### Devam

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Bileşene göre devam göstergesi kümesi. Aşağıdaki değerler belirlenebilir:

#### **MQZCI\_VARSAYILAN**

Devam eden kuyruk yöneticisine bağımlı.

MQZ\_GET\_AUTHORITY için bu, MQZCI\_CONTINUE ile aynı etkiye sahiptir.

#### **MQZCI\_CONTINY**

Sonraki bileşenle devam edin.

#### **MQZCI\_STOP**

Sonraki bileşenle devam ettirmeyin.

### CompCode

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Tamamlanma kodu. Bu değer aşağıdaki değerlerden biri olmalıdır:

#### **MQCC\_OK**

İşlem başarıyla tamamlandı.

#### **MQCC\_FAILED**

Arama başarısız oldu.

### Neden

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Neden kodu ön eleme *CompCode*.

*CompCode* , MQCC\_OK ise:

#### **MQRC\_NONE**

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_FAILED ise:

#### **MQRC\_NOT\_YETKILI**

(2035, X'7F3') Erişim için yetkili değil.

#### **MQRC\_SERVICE\_ERROR**

(2289, X'8F1') Hizmet erişimi beklenmeyen bir hata oluştu.

#### **MQRC\_SERVICE\_NOT\_VAR**

(2285, X'8ED') Undering hizmeti kullanılamıyor.

#### **MQRC\_UNKNOWN\_ENTITY**

(2292, X'8F4') Hizmet için bilinmeyen varlık.

Bu neden kodlarına ilişkin ek bilgi için [API tamamlama ve neden kodları](#) başlıklı konuya bakın.

## C çağırma

```
MQZ_GET_AUTHORITY (QMgzName, EntityName, EntityType, ObjectName,
```

```
ObjectType, &Authority, ComponentData,  
&Continuation, &CompCode, &Reason);
```

Hizmete geçirilen parametreler aşağıdaki gibi bildirilir:

```
MQCHAR48  QMgrName;           /* Queue manager name */  
MQCHAR12  EntityName;        /* Entity name */  
MQLONG    EntityType;        /* Entity type */  
MQCHAR48  ObjectName;        /* Object name */  
MQLONG    ObjectType;        /* Object type */  
MQLONG    Authority;         /* Authority of entity */  
MQBYTE    ComponentData[n]; /* Component data */  
MQLONG    Continuation;      /* Continuation indicator set by  
                             component */  
MQLONG    CompCode;          /* Completion code */  
MQLONG    Reason;           /* Reason code qualifying CompCode */
```

## MQZ\_GET\_AUTHORITY\_2 -Yetki alma (genişletilmiş)

Bu işlev bir MQZAS\_VERSION\_2 yetki hizmeti bileşeni tarafından sağlanır ve kuyruk yöneticisi tarafından, bir varlığın belirtilen nesneye erişmesi için sahip olduğu yetkiyi almak üzere başlatılır.

Bu işleve ilişkin işlev tanıtıcısı (MQZEP için) MQZID\_GET\_AUTHORITY olur.

MQZ\_GET\_AUTHORITY\_2 , MQZ\_GET\_AUTHORITY gibi, ancak **EntityName** parametresi **EntityData** parametresiyle değiştiriliyor.

### Sözdizimi

MQZ\_GET\_AUTHORITY\_2( *QMgrName* , *EntityData* , *EntityType* , *ObjectName* , *ObjectType* , *Authority* , *ComponentData* , *Continuation* , *CompCode* , *Reason* )

### Parametreler

#### QMgrName

Tip: MQCHAR48 -giriş

Kuyruk yöneticisi adı. Bileşeni çağıran kuyruk yöneticisinin adı. Bu ad, değiştirgenin tam uzunluğuna kadar boşluklarla dolduruldu; ad boş değerli bir karakterle sonlandırılmaz.

Kuyruk yöneticisi adı, bileşene bilgi için iletilir; yetki hizmeti arabirimi, bileşenin bu bileşeni tanımlı bir şekilde kullanmasını gerektirmez.

#### EntityData

Tip: MQZED-giriş

Varlık verileri. Nesneye ilişkin yetkilendirmenin alınacağı varlıkla ilgili veriler. Ayrıntılar için bkz. ["MQZED-Varlık tanımlayıcısı" sayfa 1653.](#)

#### EntityType

Tip: MQHOME-girişi

Varlık tipi. *EntityData* tarafından belirtilen varlık tipi. Bu değer aşağıdaki değerlerden biri olmalıdır:

#### MQZAET\_PRINCIPAL

Müdür.

#### MQZAET\_GRP

Grup.

#### ObjectName

Tip: MQCHAR48 -giriş

Nesne adı. Varlık yetkisinin alınacağı nesnenin adı. Dizilimin uzunluk üst sınırı 48 karakterdir; boşlukla sağa yaslanmış ise, bu karakter uzunluğuna göre daha kısadır. Ad boş değerli bir karakter tarafından sonlandırılmadı.

*ObjectType* MQOT\_Q\_MGR ise, bu ad *QMgrName* ile aynıdır.

## ObjectType

Tip: MQHOME-girişi

Nesne tipi. *ObjectName* tarafından belirtilen varlık tipi. Bu değer aşağıdaki değerlerden biri olmalıdır:

### **MQOT\_AUTH\_INFO**

Kimlik doğrulama bilgileri.

### **MQOT\_KANAL**

Kanal.

### **MQOT\_CLNTCONN\_CHANNEL**

İstemci bağlantı kanalı.

### **MQOT\_LISTENER**

Dinleyici.

### **MQOT\_NAMELIST**

İsim listesi.

### **MQOT\_PROCESS**

Süreç tanımlaması.

### **MQOT\_Q**

Kuyruk.

### **MQOT\_Q\_MGR**

Kuyruk yöneticisi.

### **MQOT\_SERVICE**

Servis.

### **MQOT\_KONUSU**

Konu.

## Yetki

Tip: MQHOME-girişi

Varlık yetkisi. Varlığın bir yetkisi varsa, bu alan uygun yetkilendirme işlemine eşittir (MQZAO\_ \* sabiti). Birden fazla yetkiniz varsa, bu alan, ilgili MQZAO\_ \* değişmezlerinin bit ya da bit ya da bitmezidir.

## ComponentData

Tip: MQBYTE xComponentDataUzunluk-giriş/çıkış

Bileşen verileri. Bu veriler, bu bileşen adına kuyruk yöneticisi tarafından tutulur; bu bileşen tarafından sağlanan işlemlerden herhangi biri tarafından yapılan değişiklikler korunur ve bu bileşen işlemlerinden biri bir sonraki defa çağrılır.

Bu veri alanının uzunluğu, MQZ\_INIT\_AUTHORITY çağrısının **ComponentDataLength** parametresindeki kuyruk yöneticisi tarafından geçirilir.

## Devam

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Bileşene göre devam göstergesi kümesi. Aşağıdaki değerler belirlenebilir:

### **MQZCI\_VARSAYILAN**

Devam eden kuyruk yöneticisine bağımlı.

MQZ\_CHECK\_AUTHORITY için bu, MQZCI\_STOP ile aynı etkiye sahiptir.

### **MQZCI\_CONTINY**

Sonraki bileşenle devam edin.

### **MQZCI\_STOP**

Sonraki bileşenle devam ettirmeyin.

## CompCode

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Tamamlanma kodu. Bu değer aşağıdaki değerlerden biri olmalıdır:



## **MQCC\_OK**

İşlem başarıyla tamamlandı.

## **MQCC\_FAILED**

Arama başarısız oldu.

### **Neden**

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Neden kodu ön eleme *CompCode*.

*CompCode* , MQCC\_OK ise:

## **MQRC\_NONE**

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_FAILED ise:

## **MQRC\_NOT\_YETKILI**

(2035, X'7F3') Erişim için yetkili değil.

## **MQRC\_SERVICE\_ERROR**

(2289, X'8F1') Hizmet erişimi beklenmeyen bir hata oluştu.

## **MQRC\_SERVICE\_NOT\_VAR**

(2285, X'8ED') Undering hizmeti kullanılmıyor.

## **MQRC\_UNKNOWN\_ENTITY**

(2292, X'8F4') Hizmet için bilinmeyen varlık.

Bu neden kodlarına ilişkin ek bilgi için [API tamamlama ve neden kodları](#) başlıklı konuya bakın.

## **C çağırma**

```
MQZ_GET_AUTHORITY_2 (QMgrName, &EntityData, EntityType, ObjectName,  
Objectype, &Authority, ComponentData,  
&Continuation, &CompCode, &Reason);
```

Hizmete geçirilen parametreler aşağıdaki gibi bildirilir:

```
MQCHAR48 QMgrName; /* Queue manager name */  
MQZED EntityData; /* Entity data */  
MQLONG EntityType; /* Entity type */  
MQCHAR48 ObjectName; /* Object name */  
MQLONG Objectype; /* Object type */  
MQLONG Authority; /* Authority of entity */  
MQBYTE ComponentData[n]; /* Component data */  
MQLONG Continuation; /* Continuation indicator set by  
component */  
MQLONG CompCode; /* Completion code */  
MQLONG Reason; /* Reason code qualifying CompCode */
```

## **MQZ\_GET\_REZILIT\_AUTHORITY-Belirtik yetkisi al**

Bu işlev bir MQZAS\_VERSION\_1 yetki hizmeti bileşeni tarafından sağlanır ve kuyruk yöneticisi tarafından, bir varlığın, birincil kullanıcının üyesi olduğu grupların sahip olduğu (varlık bir birincil kullanıcı) yetkiler de dahil olmak üzere, belirtilen nesneye erişmek için sahip olduğu yetkiyi almak üzere başlatılmış olur. Genel tanımlardaki yetkiler, döndürülen yetki kümesine eklenir.

On AIX and Linux, for the built-in IBM MQ object authority manager (OAM), the returned authority is that possessed only by the principal's primary group.

Bu işleve ilişkin işlev tanıtıcısı (MQZEP için) MQZID\_GET\_REGICIIT\_AUTHORITY.

### **Sözdizimi**

```
MQZ_GET_EXPLICIT_AUTHORITY( QMgrName , EntityName , EntityType , ObjectName ,  
Objectype , Authority , ComponentData , Continuation , CompCode , Reason )
```

## Parametreler

### QMgrName

Tip: MQCHAR48 -giriş

Kuyruk yöneticisi adı. Bileşeni çağıran kuyruk yöneticisinin adı. Bu ad, değıştirgenin tam uzunluđuna kadar boşluklarla dolduruldu; ad boş değeri bir karakterle sonlandırılmaz.

Kuyruk yöneticisi adı, bileşene bilgi için iletilir; yetki hizmeti arabirimi, bileşenin bu bileşeni tanımlı bir şekilde kullanmasını gerektirmez.

### EntityName

Tip: MQCHAR12 -giriş

Varlık adı. Nesneye erişimin alınacağı varlığın adı. Dizilimin uzunluk üst sınırı 12 karakterdir; boşlukla doldurulsa boşluk karakterinden daha kısadır. Ad boş değeri bir karakter tarafından sonlandırılmadı.

### EntityType

Tip: MQHOME-giriş

Varlık tipi. *EntityName* tarafından belirtilen varlık tipi. Bu değeri aşağıdaki değerilerden biri olmalıdır:

#### **MQZAET\_PRINCIPAL**

Müdür.

#### **MQZAET\_GRP**

Grup.

### ObjectName

Tip: MQCHAR48 -giriş

Nesne adı. Varlık yetkisinin alınacağı nesnenin adı. Dizilimin uzunluk üst sınırı 48 karakterdir; boşlukla sağa yaslanmış ise, bu karakter uzunluđuna göre daha kısadır. Ad boş değeri bir karakter tarafından sonlandırılmadı.

*ObjectType* MQOT\_Q\_MGR ise, bu ad *QMgrName* ile aynıdır.

### ObjectType

Tip: MQHOME-giriş

Nesne tipi. *ObjectName* tarafından belirtilen varlık tipi. Bu değeri aşağıdaki değerilerden biri olmalıdır:

#### **MQOT\_AUTH\_INFO**

Kimlik doğrulama bilgileri.

#### **MQOT\_KANAL**

Kanal.

#### **MQOT\_CLNTCONN\_CHANNEL**

İstemci bağlantı kanalı.

#### **MQOT\_LISTENER**

Dinleyici.

#### **MQOT\_NAMELIST**

İsim listesi.

#### **MQOT\_PROCESS**

Süreç tanımlaması.

#### **MQOT\_Q**

Kuyruk.

#### **MQOT\_Q\_MGR**

Kuyruk yöneticisi.

#### **MQOT\_SERVICE**

Servis.

#### **MQOT\_KONUSU**

Konu.

## Yetki

Tip: MQHOME-giriş

Varlık yetkisi. Varlığın bir yetkisi varsa, bu alan uygun yetkilendirme işlemine eşittir (MQZAO\_ \* sabiti). Birden fazla yetkiniz varsa, bu alan, ilgili MQZAO\_ \* değişmezlerinin bit ya da bit ya da bitmezidir.

## ComponentData

Tip: MQBYTE x ComponentDataUzunluk-giriş/çıkış

Bileşen verileri. Bu veriler, bu bileşen adına kuyruk yöneticisi tarafından tutulur; bu bileşen tarafından sağlanan işlevlerden herhangi biri tarafından yapılan değişiklikler korunur ve bu bileşen işlevlerinden biri bir sonraki defa çağrılır.

Bu veri alanının uzunluğu, MQZ\_INIT\_AUTHORITY çağrısının **ComponentDataLength** parametresindeki kuyruk yöneticisi tarafından geçirilir.

## Devam

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Bileşene göre devam göstergesi kümesi. Aşağıdaki değerler belirlenebilir:

### **MQZCI\_VARSAYILAN**

Devam eden kuyruk yöneticisine bağlıdır.

MQZ\_GET\_AUTHORITY için bu, MQZCI\_CONTINUE ile aynı etkiye sahiptir.

### **MQZCI\_CONTINY**

Sonraki bileşenle devam edin.

### **MQZCI\_STOP**

Sonraki bileşenle devam ettirmeyin.

## CompCode

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Tamamlanma kodu. Bu değer aşağıdaki değerlerden biri olmalıdır:

### **MQCC\_OK**

İşlem başarıyla tamamlandı.

### **MQCC\_FAILED**

Arama başarısız oldu.

## Neden

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Neden kodu ön eleme *CompCode*.

*CompCode* , MQCC\_OK ise:

### **MQRC\_NONE**

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_FAILED ise:

### **MQRC\_NOT\_YETKILI**

(2035, X'7F3') Erişim için yetkili değil.

### **MQRC\_SERVICE\_ERROR**

(2289, X'8F1') Hizmet erişimi beklenmeyen bir hata oluştu.

### **MQRC\_SERVICE\_NOT\_VAR**

(2285, X'8ED') Udering hizmeti kullanılamıyor.

### **MQRC\_UNKNOWN\_ENTITY**

(2292, X'8F4') Hizmet için bilinmeyen varlık.

Bu neden kodlarına ilişkin ek bilgi için [API tamamlama ve neden kodları](#) başlıklı konuya bakın.

## C çağırma

```
MQZ_GET_EXPLICIT_AUTHORITY (QMgrName, EntityName, EntityType,  
ObjectName, ObjectType, &Authority,  
ComponentData, &Continuation,  
&CompCode, &Reason);
```

Hizmete geçirilen parametreler aşağıdaki gibi bildirilir:

```
MQCHAR48 QMgrName;          /* Queue manager name */  
MQCHAR12 EntityName;        /* Entity name */  
MQLONG   EntityType;        /* Entity type */  
MQCHAR48 ObjectName;        /* Object name */  
MQLONG   ObjectType;        /* Object type */  
MQLONG   Authority;         /* Authority of entity */  
MQBYTE   ComponentData[n];  /* Component data */  
MQLONG   Continuation;      /* Continuation indicator set by  
                             component */  
MQLONG   CompCode;          /* Completion code */  
MQLONG   Reason;           /* Reason code qualifying CompCode */
```

## MQZ\_GET\_EXPLICIT\_AUTHORITY\_2 -Belirtik yetkisi al (genişletilmiş)

Bu işlev, bir MQZAS\_VERSION\_2 yetki hizmeti bileşeni tarafından sağlanır ve kuyruk yöneticisi tarafından belirtilen bir grubun belirtilen bir nesneye ( **Kimse** grubunun ek yetkisi olmadan) erişmesi gereken yetkisi ya da belirtilen birincil kullanıcının birincil grubunun belirtilen bir nesneye erişmek zorunda olduğu yetkiyi almak için başlatılmıştır.

Bu işleve ilişkin işlev tanıtıcısı (MQZEP için) MQZID\_GET\_REGICIIT\_AUTHORITY.

MQZ\_GET\_EXPLICIT\_AUTHORITY\_2 is like MQZ\_GET\_EXPLICIT\_AUTHORITY, but with the **EntityName** parameter replaced by the **EntityData** parameter.

### Sözdizimi

```
MQZ_GET_EXPLICIT_AUTHORITY_2( QMgrName , EntityData , EntityType , ObjectName ,  
ObjectType , Authority , ComponentData , Continuation , CompCode , Reason )
```

### Parametreler

#### QMgrName

Tip: MQCHAR48 -giriş

Kuyruk yöneticisi adı. Bileşeni çağıran kuyruk yöneticisinin adı. Bu ad, değiştirgenin tam uzunluğuna kadar boşluklarla dolduruldu; ad boş değerli bir karakterle sonlandırılmaz.

Kuyruk yöneticisi adı, bileşene bilgi için iletilir; yetki hizmeti arabirimi, bileşenin bu bileşeni tanımlı bir şekilde kullanmasını gerektirmez.

#### EntityData

Tip: MQZED-giriş

Varlık verileri. Nesneye ilişkin yetkinin alınabilecek varlıkla ilgili veriler. Ayrıntılar için bkz. "[MQZED-Varlık tanımlayıcısı](#)" sayfa 1653.

#### EntityType

Tip: MQHOME-girişi

Varlık tipi. *EntityData* tarafından belirtilen varlık tipi. Bu değer aşağıdaki değerlerden biri olmalıdır:

#### MQZAET\_PRINCIPAL

Müdür.

#### MQZAET\_GRP

Grup.

**ObjectName**

Tip: MQCHAR48 -giriş

Nesne adı. Varlık yetkisinin alınacağı nesnenin adı. Dizilimin uzunluk üst sınırı 48 karakterdir; boşlukla sağa yaslanmış ise, bu karakter uzunluğuna göre daha kısadır. Ad boş değerli bir karakter tarafından sonlandırılmadı.

*ObjectType* MQOT\_Q\_MGR ise, bu ad *QMgrName* ile aynıdır.

**ObjectType**

Tip: MQHOME-girişi

Nesne tipi. *ObjectName* tarafından belirtilen varlık tipi. Bu değer aşağıdaki değerlerden biri olmalıdır:

**MQOT\_AUTH\_INFO**

Kimlik doğrulama bilgileri.

**MQOT\_KANAL**

Kanal.

**MQOT\_CLNTCONN\_CHANNEL**

İstemci bağlantı kanalı.

**MQOT\_LISTENER**

Dinleyici.

**MQOT\_NAMELIST**

İsim listesi.

**MQOT\_PROCESS**

Süreç tanımlaması.

**MQOT\_Q**

Kuyruk.

**MQOT\_Q\_MGR**

Kuyruk yöneticisi.

**MQOT\_SERVICE**

Servis.

**MQOT\_KONUSU**

Konu.

**Yetki**

Tip: MQHOME-girişi

Varlık yetkisi. Varlığın bir yetkisi varsa, bu alan uygun yetkilendirme işlemine eşittir (MQZAO\_ \* sabiti). Birden fazla yetkiniz varsa, bu alan, ilgili MQZAO\_ \* değişmezlerinin bit ya da bit ya da bitmezidir.

**ComponentData**

Tip: MQBYTE xComponentDataUzunluk-giriş/çıkış

Bileşen verileri. Bu veriler, bu bileşen adına kuyruk yöneticisi tarafından tutulur; bu bileşen tarafından sağlanan işlevlerden herhangi biri tarafından yapılan değişiklikler korunur ve bu bileşen işlevlerinden biri bir sonraki defa çağrılır.

Bu veri alanının uzunluğu, MQZ\_INIT\_AUTHORITY çağrısının **ComponentDataLength** parametresindeki kuyruk yöneticisi tarafından geçirilir.

**Devam**

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Bileşene göre devam göstergesi kümesi. Aşağıdaki değerler belirlenebilir:

**MQZCI\_VARSAYILAN**

Devam eden kuyruk yöneticisine bağımlı.

MQZ\_CHECK\_AUTHORITY için bu, MQZCI\_STOP ile aynı etkiye sahiptir.

## **MQZCI\_CONTINY**

Sonraki bileşenle devam edin.

## **MQZCI\_STOP**

Sonraki bileşenle devam ettirmeyin.

### **CompCode**

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Tamamlanma kodu. Bu değer aşağıdaki değerlerden biri olmalıdır:

### **MQCC\_OK**

İşlem başarıyla tamamlandı.

### **MQCC\_FAILED**

Arama başarısız oldu.

### **Neden**

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Neden kodu ön eleme *CompCode*.

*CompCode* , MQCC\_OK ise:

### **MQRC\_NONE**

(0, X'000') Raporlamak için bir neden yok.

*CompCode* MQCC\_FAILED ise:

### **MQRC\_NOT\_YETKILI**

(2035, X'7F3') Erişim için yetkili değil.

### **MQRC\_SERVICE\_ERROR**

(2289, X'8F1') Hizmet erişimi beklenmeyen bir hata oluştu.

### **MQRC\_SERVICE\_NOT\_VAR**

(2285, X'8ED') Undering hizmeti kullanılamıyor.

### **MQRC\_UNKNOWN\_ENTITY**

(2292, X'8F4') Hizmet için bilinmeyen varlık.

Bu neden kodlarına ilişkin ek bilgi için [API tamamlama ve neden kodları](#) başlıklı konuya bakın.

## **C çağırma**

```
MQZ_GET_EXPLICIT_AUTHORITY_2 (QMgrName, &EntityData, EntityType,  
ObjectName, ObjectType, &Authority,  
ComponentData, &Continuation,  
&CompCode, &Reason);
```

Hizmete geçirilen parametreler aşağıdaki gibi bildirilir:

```
MQCHAR48 QMgrName;          /* Queue manager name */  
MQZED EntityData;          /* Entity data */  
MQLONG EntityType;        /* Entity type */  
MQCHAR48 ObjectName;      /* Object name */  
MQLONG ObjectType;        /* Object type */  
MQLONG Authority;         /* Authority of entity */  
MQBYTE ComponentData[n]; /* Component data */  
MQLONG Continuation;      /* Continuation indicator set by  
component */  
MQLONG CompCode;         /* Completion code */  
MQLONG Reason;           /* Reason code qualifying CompCode */
```

## **MQZ\_INIT\_AUTHORITY-Yetkilendirme hizmetini kullanıma hazırla**

Bu işlem, bir yetkilendirme hizmeti bileşeni tarafından sağlanır ve bileşen yapılandırması sırasında kuyruk yöneticisi tarafından başlatılır. Kuyruk yöneticisine bilgi sağlamak için MQZEP çağırılması beklenir.

Bu işleve ilişkin işlev tanıtıcısı (MQZEP için) MQZID\_INIT\_AUTHORITY 'dir.

## Sözdizimi

MQZ\_INIT\_AUTHORITY( *Hconfig* , *Options* , *QMgrName* , *ComponentDataLength* , *ComponentData* , *Version* , *CompCode* , *Reason* )

## Parametreler

### Hconfig

Tip: MQHCONFIG-input

Yapılandırma tanıtıcısı. Bu tanıtıcı, ilk kullanıma hazırlanmakta olan bileşeni gösterir. Bu bileşen, kuyruk yöneticisini MQZEP işleviyle çağırırken, bu bileşen tarafından kullanılır.

### Seçenekler

Tip: MQHOME-girişi

Başlatma seçenekleri. Bu değer aşağıdaki değerlerden biri olmalıdır:

#### MQZI\_BIRINCIL

Birincil kullanıma hazırlama.

#### MQZIO\_IKINCIL

İkincil kullanıma hazırlama.

### QMgrName

Tip: MQCHAR48 -giriş

Kuyruk yöneticisi adı. Bileşeni çağıran kuyruk yöneticisinin adı. Bu ad, değiştirgenin tam uzunluğuna kadar boşluklarla dolduruldu; ad boş değerli bir karakterle sonlandırılmaz.

Kuyruk yöneticisi adı, bileşene bilgi için iletilir; yetki hizmeti arabirimi, bileşenin bu bileşeni tanımlı bir şekilde kullanmasını gerektirmez.

### ComponentDataUzunluğu

Tip: MQHOME-girişi

Bileşen verilerinin uzunluğu. *ComponentData* alanının bayt cinsinden uzunluğu. Bu uzunluk, bileşen yapısını verilerinde tanımlanır.

### ComponentData

Tip: MQBYTE x ComponentDataUzunluk-giriş/çıkış

Bileşen verileri. Bileşen birincil kullanıma hazırlama işlevini çağırılmadan önce, bu değer tüm sıfırlar için başlatılır. Bu veriler, bu bileşen adına kuyruk yöneticisi tarafından tutulur; bu bileşen tarafından sağlanan işlevler (kullanıma hazırlama işlevi de içinde olmak üzere) tarafından bu bileşen için yapılan değişiklikler korunur ve bu bileşen işlevlerinden bir sonraki kez çağırılır.

Bu veri alanının uzunluğu, MQZ\_INIT\_AUTHORITY çağırısının **ComponentDataLength** parametresindeki kuyruk yöneticisi tarafından geçirilir.

### S\u00fcr\u00fcm

Tip: MQHOME-giriş/çıkış

Sürüm numarası. Kullanıma hazırlama işlevine ilişkin giriş, kuyruk yöneticisinin desteklediği en yüksek sürüm numarasını belirtir. Başlatma işlevinin desteklediği arabirimin sürümüne (gerekirse) bunu değiştirmesi gerekir. Geri dönerse, kuyruk yöneticisi bileşenin döndürdüğü sürümü desteklemiyorsa, MQZ\_TERM\_AUTHORITY işlevini çağırır ve bu bileşeni başka bir şekilde kullanmaz.

Aşağıdaki değerler desteklenir:

#### MQZAS\_VERSION\_1

Sürüm 1.

#### MQZAS\_VERSION\_2

Sürüm 2.

### **MQZAS\_VERSION\_3**

Sürüm 3.

### **MQZAS\_VERSION\_4**

Sürüm 4.

### **MQZAS\_VERSION\_5**

Sürüm 5.

### **MQZAS\_VERSION\_6**

IBM WebSphere MQ 6.

#### **CompCode**

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Tamamlanma kodu. Bu değer aşağıdaki değerlerden biri olmalıdır:

#### **MQCC\_OK**

İşlem başarıyla tamamlandı.

#### **MQCC\_FAILED**

Arama başarısız oldu.

#### **Neden**

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Neden kodu ön eleme *CompCode*.

*CompCode* , MQCC\_OK ise:

#### **MQRC\_NONE**

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_FAILED ise:

#### **MQRC\_INITIALIZATION\_FAILED**

(2286, X'8EE') Kullanıma hazırlama, tanımlanmamış bir neden için başarısız oldu.

#### **MQRC\_SERVICE\_NOT\_VAR**

(2285, X'8ED') Undering hizmeti kullanılamıyor.

Bu neden kodlarına ilişkin ek bilgi için [API tamamlama ve neden kodları](#) başlıklı konuya bakın.

## **C çağırma**

```
MQZ_INIT_AUTHORITY (Hconfig, Options, QMgrName, ComponentDataLength,  
ComponentData, &Version, &CompCode,  
&Reason);
```

Hizmete geçirilen parametreler aşağıdaki gibi bildirilir:

```
MQHCONFIG  Hconfig;           /* Configuration handle */  
MQLONG     Options;           /* Initialization options */  
MQCHAR48   QMgrName;         /* Queue manager name */  
MQLONG     ComponentDataLength; /* Length of component data */  
MQBYTE     ComponentData[n]; /* Component data */  
MQLONG     Version;          /* Version number */  
MQLONG     CompCode;         /* Completion code */  
MQLONG     Reason;           /* Reason code qualifying CompCode */
```

## **MQZ\_SORMA-YETKILENDİRME HİZMETİ**

Bu işlev bir MQZAS\_VERSION\_5 yetkilendirme hizmeti bileşeni tarafından sağlanır ve kuyruk yöneticisi tarafından desteklenen işlevselliği sorgulamak için başlatılır.

Birden çok hizmet bileşeni kullanıldığı durumlarda, hizmet bileşenleri, takılı oldukları siparişe ters sırayla çağrılır.

Bu işleve ilişkin işlev tanıtıcısı (MQZEP için) MQZID\_SORGULADIR.



## Sözdizimi

MQZ\_INQUIRE( *QMgrName* , *SelectorCount* , *Selectors* , *IntAttrCount* , *IntAttrs* , *CharAttrLength* , *CharAttrs* , *SelectorReturned* , *ComponentData* , *Continuation* , *CompCode* , *Reason* )

## Parametreler

### QMgrName

Tip: MQCHAR48 -giriş

Kuyruk yöneticisi adı. Bileşeni çağıran kuyruk yöneticisinin adı. Bu ad, değiştirgenin tam uzunluğuna kadar boşluklarla dolduruldu; ad boş değerli bir karakterle sonlandırılmaz.

Kuyruk yöneticisi adı, bileşene bilgi için iletilir; yetki hizmeti arabirimi, bileşenin bu bileşeni tanımlı bir şekilde kullanmasını gerektirmez.

### SelectorCount

Tip: MQHOME-girişi

Seçicilerin sayısı. **Selectors** parametresine sağlanan seçicilerin sayısı.

Değer 0 ile 256 aralığında olmalıdır.

### Seçiciler

Tip: MQLONGxSelectorCount-input

Seçiciler dizisi. Her seçici, gerekli bir özniteliği tanımlar ve aşağıdakilerden biri olmalıdır:

- MQIACF\_INTERFACE\_VERSION (tamsayı)
- MQIACF\_USER\_ID\_DESPORT (tamsayı)
- MQCACF\_SERVICE\_COMPONENT (karakter)

Seçiciler herhangi bir sırada belirtilebilir. Dizideki seçicilerin sayısı **SelectorCount** parametresiyle gösterilir.

Seçiciler tarafından tanımlanan tamsayı öznitelikleri, **IntAttrs** parametresine *Selectors* içinde görüldükleri sırayla geri döndürülür.

Character attributes identified by selectors are returned in the **CharAttrs** parameter in the same order as they in appear *Selectors*.

### IntAttrSayı

Tip: MQHOME-girişi

IntAttrs değiştirgesinde sağlanan tamsayı özniteliklerinin sayısı.

Değer 0 ile 256 aralığında olmalıdır.

### IntAttrs

Tip: MQUXX IntAttrSayı-çıkış

Tamsayı öznitelikleri. Tamsayı öznitelikleri dizisi. Tamsayı öznitelikleri, *Selectors* dizisinde karşılık gelen tamsayı seçiciyle aynı sırayla döndürülür.

### CharAttrSayı

Tip: MQHOME-girişi

Karakter öznitelikleri arabelleğindeki uzunluk. **CharAttrs** parametresindeki bayt cinsinden uzunluk.

Değer, istenen karakter özniteliklerinin uzunluklarının en az toplamı kadar olmalıdır. Karakter öznitelikleri istenmiyorsa, sıfır geçerli bir değerdir.

### CharAttrs

Tip: MQHOX x CharAttrSayı-çıkış

Karakter öznitelikleri arabelleği. Birlikte bitleştirilmiş karakter özniteliklerini içeren arabellek. Karakter öznitelikleri, *Selectors* dizisinde karşılık gelen karakter seçiciyle aynı sırayla döndürülür.

Arabelleğin uzunluğu CharAttrSayı parametresi tarafından verilir.

### **SelectorReturned**

Tip: MQLong x SelectorCount -giriş

Seçici geri döndü. Seçiciler (Selectors) parametresindeki seçiciler tarafından istenen kümeden hangi özniteliklerin döndürüldüğünü tanımlayan değerler dizisi. Bu dizideki değer sayısı, **SelectorCount** parametresiyle belirtilir. Dizideki her bir değer, Seçiciler dizisinde karşılık gelen konumdaki seçiciye ilişkinir. Her değer aşağıdakilerden biridir:

#### **MQZSL\_RETURNED**

**Selectors** parametresindeki karşılık gelen seçici tarafından istenen öznitelik döndürüldü.

#### **MQZSL\_NOT\_RETURNED**

**Selectors** parametresindeki karşılık gelen seçici tarafından istenen öznitelik iade edilmedi.

Dizi, tüm değerler **MQZSL\_NOT\_RETURNED** olarak başlatılır. Bir yetki hizmeti bileşeni bir öznitelik döndürdüğünde, dizideki uygun değeri **MQZSL\_NOT\_RETURNED** olarak ayarlar. Bu, sorgu çağrısının yapıldığı diğer yetki hizmeti bileşenlerinin hangi özniteliklerin önceden geri gönderildiğini belirlemek için sağlanmasını sağlar.

### **ComponentData**

Tip: MQBYTE x ComponentDataUzunluk-giriş/çıkış

Bileşen verileri. Bu veriler, bu bileşen adına kuyruk yöneticisi tarafından tutulur; bu bileşen tarafından sağlanan işlevlerden herhangi biri tarafından yapılan değişiklikler korunur ve bu bileşen işlevlerinden biri bir sonraki defa çağrılır.

Bu veri alanının uzunluğu, MQZ\_INIT\_AUTHORITY çağrısının **ComponentDataLength** parametresindeki kuyruk yöneticisi tarafından geçirilir.

### **Devam**

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Bileşene göre devam göstergesi kümesi. Aşağıdaki değerler belirlenebilir:

#### **MQZCI\_VARSAYILAN**

Devam eden kuyruk yöneticisine bağımlı.

MQZ\_CHECK\_AUTHORITY için bu, MQZCI\_STOP ile aynı etkiye sahiptir.

#### **MQZCI\_STOP**

Sonraki bileşenle devam ettirmeyin.

### **CompCode**

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Tamamlanma kodu. Bu değer aşağıdaki değerlerden biri olmalıdır:

#### **MQCC\_OK**

İşlem başarıyla tamamlandı.

#### **MQCC\_UYARI**

Kısmi tamamlama.

#### **MQCC\_FAILED**

Arama başarısız oldu.

### **Neden**

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Neden kodu ön eleme *CompCode*.

*CompCode* , MQCC\_OK ise:

#### **MQRC\_NONE**

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* , MQCC\_UYARI ise:

### **MQRC\_CHAR\_ATTRS\_TOO\_SHORT**

Karakter öznitelikleri için yeterli alan yok.

### **MQRC\_INT\_COUNT\_TOO\_SMALL**

Tamsayı öznitelikleri için yeterli alan yok.

*CompCode* MQCC\_FAILED ise:

### **MQRC\_SELECTOR\_COUNT\_ERROR**

Seçicilerin sayısı geçerli değil.

### **MQRC\_SELECTOR\_ERROR**

Öznitelik seçici geçerli değil.

### **MQRC\_SELECTOR\_LIMIT\_EXACID**

Çok sayıda seçici belirtildi.

### **MQRC\_INT\_ATTR\_COUNT\_ERROR**

Tamsayı özniteliklerinin sayısı geçerli değil.

### **MQRC\_INT\_ATTRS\_ARRAY\_ERROR**

Tamsayı öznitelikleri dizisi geçerli değil.

### **MQRC\_CHAR\_ATTR\_LENGTH\_ERROR**

Karakter özniteliklerinin sayısı geçerli değil.

### **MQRC\_CHAR\_ATTRS\_ERROR**

Karakter öznitelikleri dizisi geçerli değil.

### **MQRC\_SERVICE\_ERROR**

(2289, X'8F1') Hizmet erişimi beklenmeyen bir hata oluştu.

Bu neden kodlarına ilişkin ek bilgi için [API tamamlama ve neden kodları](#) başlıklı konuya bakın.

## **C çağırma**

```
MQZ_INQUIRE (QMgrName, SelectorCount, Selectors, IntAttrCount,  
              &IntAttrs, CharAttrLength, &CharAttrs,  
              SelectorReturned, ComponentData, &Continuation,  
              &CompCode, &Reason);
```

Hizmete geçirilen parametreler aşağıdaki gibi bildirilir:

```
MQCHAR48  QMgrName;           /* Queue manager name */  
MQLONG    SelectorCount;     /* Selector count */  
MQLONG    Selectors[n];      /* Selectors */  
MQLONG    IntAttrCount;      /* IntAttrs count */  
MQLONG    IntAttrs[n];       /* Integer attributes */  
MQLONG    CharAttrCount;     /* CharAttrs count */  
MQLONG    CharAttrs[n];     /* Character attributes */  
MQLONG    SelectorReturned[n]; /* Selector returned */  
MQBYTE    ComponentData[n];  /* Component data */  
MQLONG    Continuation;      /* Continuation indicator set by  
                               component */  
MQLONG    CompCode;          /* Completion code */  
MQLONG    Reason;           /* Reason code qualifying CompCode */
```

## **MQZ\_REFRESH\_CACHE-Tüm yetkileri yenile**

Bu işlev bir MQZAS\_VERSION\_3 yetki hizmeti bileşeni tarafından sağlanır ve kuyruk yöneticisi tarafından, bileşen tarafından dahili olarak tutulan yetkiler listesini yenilemek için çağrılır.

Bu işleve ilişkin işlev tanıtıcısı (MQZEP için) MQZID\_REFRESH\_CACHE (8L) olur.

### **Sözdizimi**

```
MQZ_REFRESH_CACHE( QMgrName , ComponentData , Continuation , CompCode ,  
Reason )
```

## Parametreler

### QMgrName

Tip: MQCHAR48 -giriş

Kuyruk yöneticisi adı. Bileşeni çağıran kuyruk yöneticisinin adı. Bu ad, deęiřtirgenin tam uzunluęuna kadar boşluklarla dolduruldu; ad boş deęerli bir karakterle sonlandırılmaz.

Kuyruk yöneticisi adı, bileşene bilgi için iletilir; yetki hizmeti arabirimi, bileşenin herhangi bir şekilde tanımlanmış bir şekilde kullanılmasını gerektirmez.

### ComponentData

Tip: MQBYTE xComponentDataUzunluk-giriş/çıkış

Bileşen verileri. Bu veriler, bu bileşen adına kuyruk yöneticisi tarafından tutulur; bu bileşen tarafından sağlanan işlevlerden herhangi biri tarafından yapılan deęişiklikler korunur ve bu bileşenin sonraki işlevlerinden biri çağrıldığında sunulur.

Bu veri alanının uzunluęu, MQZ\_INIT\_AUTHORITY çağrısının **ComponentDataLength** parametresindeki kuyruk yöneticisi tarafından geçirilir.

### Devam

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Bileşene göre devam göstergesi kümesi. Aşağıdaki deęerler belirlenebilir:

#### MQZCI\_VARSAYILAN

Devam eden kuyruk yöneticisine baęımlı.

MQZ\_CHECK\_AUTHORITY için bu, MQZCI\_STOP ile aynı etkiye sahiptir.

#### MQZCI\_CONTINY

Sonraki bileşenle devam edin.

#### MQZCI\_STOP

Sonraki bileşenle devam ettirmeyin.

### CompCode

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Tamamlanma kodu. Bu deęer aşağıdaki deęerlerden biri olmalıdır:

#### MQCC\_OK

İşlem başarıyla tamamlandı.

#### MQCC\_FAILED

Arama başarısız oldu.

### Neden

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Neden kodu ön eleme *CompCode*.

*CompCode* , MQCC\_OK ise:

#### MQRC\_NONE

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* , MQCC\_UYARI ise:

#### MQRC\_SERVICE\_ERROR

(2289, X'8F1') Hizmet erişimi beklenmeyen bir hata oluştu.

## C çağırma

```
MQZ_REFRESH_CACHE (QMgrName, ComponentData,  
&Continuation, &CompCode, &Reason);
```

Deęiřtirgeleri aşağıdaki gibi bildirin:

```
MQCHAR48  QMgrName;          /* Queue manager name */
MQBYTE    ComponentData[n]; /* Component data */
MQLONG    Continuation;    /* Continuation indicator set by
                             component */
MQLONG    CompCode;        /* Completion code */
MQLONG    Reason;          /* Reason code qualifying CompCode */
```

## MQZ\_SET\_AUTHORITY-Yetki tanımla

Bu işlev bir MQZAS\_VERSION\_1 yetki hizmeti bileşeni tarafından sağlanır ve kuyruk yöneticisi tarafından, bir varlığın belirtilen nesneye erişmesi için sahip olduğu yetkiyi ayarlamak üzere başlatılır.

Bu işleve ilişkin işlev tanıtıcısı (MQZEP için) MQZID\_SET\_AUTHORITY olur.

**Not:** Bu işlev, var olan yetkileri geçersiz kılar. Varolan yetkileri korumak için, bu işlevi bu işlevle yeniden ayarlamanız gerekir.

### Sözdizimi

MQZ\_SET\_AUTHORITY( *QMgrName* , *EntityName* , *EntityType* , *ObjectName* , *ObjectType* , *Authority* , *ComponentData* , *Continuation* , *CompCode* , *Reason* )

### Parametreler

#### QMgrName

Tip: MQCHAR48 -giriş

Kuyruk yöneticisi adı. Bileşeni çağıran kuyruk yöneticisinin adı. Bu ad, değiştirgenin tam uzunluğuna kadar boşluklarla dolduruldu; ad boş değerli bir karakterle sonlandırılmaz.

Kuyruk yöneticisi adı, bileşene bilgi için iletilir; yetki hizmeti arabirimi, bileşenin bu bileşeni tanımlı bir şekilde kullanmasını gerektirmez.

#### EntityName

Tip: MQCHAR12 -giriş

Varlık adı. Nesneye erişimin alınacağı varlığın adı. Dizilimin uzunluk üst sınırı 12 karakterdir; boşlukla doldurulsa boşluk karakterinden daha kısadır. Ad boş değerli bir karakter tarafından sonlandırılmaz.

#### EntityType

Tip: MQHOME-girişi

Varlık tipi. *EntityName* tarafından belirtilen varlık tipi. Bu değer aşağıdaki değerlerden biri olmalıdır:

#### MQZAET\_PRINCIPAL

Müdür.

#### MQZAET\_GRP

Grup.

#### ObjectName

Tip: MQCHAR48 -giriş

Nesne adı. Erişimin gerekli olduğu nesnenin adı. Dizilimin uzunluk üst sınırı 48 karakterdir; boşlukla sağa yaslanmış ise, bu karakter uzunluğuna göre daha kısadır. Ad boş değerli bir karakter tarafından sonlandırılmaz.

*ObjectType* MQOT\_Q\_MGR ise, bu ad *QMgrName* ile aynıdır.

#### ObjectType

Tip: MQHOME-girişi

Nesne tipi. *ObjectName* tarafından belirtilen varlık tipi. Bu değer aşağıdaki değerlerden biri olmalıdır:

#### MQOT\_AUTH\_INFO

Kimlik doğrulama bilgileri.

**MQOT\_KANAL**

Kanal.

**MQOT\_CLNTCONN\_CHANNEL**

İstemci bağlantı kanalı.

**MQOT\_LISTENER**

Dinleyici.

**MQOT\_NAMELIST**

İsim listesi.

**MQOT\_PROCESS**

Süreç tanımlaması.

**MQOT\_Q**

Kuyruk.

**MQOT\_Q\_MGR**

Kuyruk yöneticisi.

**MQOT\_SERVICE**

Servis.

**MQOT\_KONUSU**

Konu.

**Yetki**

Tip: MQHOME-girişi

Varlık yetkisi. Bir yetki belirleniyorsa, bu alan uygun yetki işlemine eşittir (MQZAO\_\* sabiti). Birden fazla yetki belirleniyorsa, bu alan, ilgili MQZAO\_\* değişmezlerinin bit ya da bitişiklerinden biridir.

**ComponentDataname>**

Tip: MQBYTEComponentDataLength -giriş/çıkış

Bileşen verileri. Bu veriler, bu bileşen adına kuyruk yöneticisi tarafından tutulur; bu bileşen tarafından sağlanan işlemlerden herhangi biri tarafından yapılan değişiklikler korunur ve bu bileşen işlemlerinden biri bir sonraki defa çağrılır.

Bu veri alanının uzunluğu, MQZ\_INIT\_AUTHORITY çağrısının **ComponentDataLength** parametresindeki kuyruk yöneticisi tarafından geçirilir.

**Devam**

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Bileşene göre devam göstergesi kümesi. Aşağıdaki değerler belirlenebilir:

**MQZCI\_VARSAYILAN**

Devam eden kuyruk yöneticisine bağımlı.

MQZ\_GET\_AUTHORITY için bu, MQZCI\_CONTINUE ile aynı etkiye sahiptir.

**MQZCI\_CONTINY**

Sonraki bileşenle devam edin.

**MQZCI\_STOP**

Sonraki bileşenle devam ettirmeyin.

**CompCode**

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Tamamlanma kodu. Bu değer aşağıdaki değerlerden biri olmalıdır:

**MQCC\_OK**

İşlem başarıyla tamamlandı.

**MQCC\_FAILED**

Arama başarısız oldu.

## Neden

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Neden kodu ön eleme *CompCode*.

*CompCode* , MQCC\_OK ise:

### MQRC\_NONE

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_FAILED ise:

### MQRC\_NOT\_YETKILI

(2035, X'7F3') Erişim için yetkili değil.

### MQRC\_SERVICE\_ERROR

(2289, X'8F1') Hizmet erişimi beklenmeyen bir hata oluştu.

### MQRC\_SERVICE\_NOT\_VAR

(2285, X'8ED') Udering hizmeti kullanılamıyor.

### MQRC\_UNKNOWN\_ENTITY

(2292, X'8F4') Hizmet için bilinmeyen varlık.

Bu neden kodlarına ilişkin ek bilgi için [API tamamlama ve neden kodları](#) başlıklı konuya bakın.

## C çağırma

```
MQZ_SET_AUTHORITY (QMgrName, EntityName, EntityType, ObjectName,  
Objectype, Authority, ComponentData,  
&Continuation, &CompCode, &Reason);
```

Hizmete geçirilen parametreler aşağıdaki gibi bildirilir:

```
MQCHAR48 QMgrName;          /* Queue manager name */  
MQCHAR12 EntityName;       /* Entity name */  
MQLONG   EntityType;       /* Entity type */  
MQCHAR48 ObjectName;       /* Object name */  
MQLONG   Objectype;        /* Object type */  
MQLONG   Authority;        /* Authority to be checked */  
MQBYTE   ComponentData[n]; /* Component data */  
MQLONG   Continuation;     /* Continuation indicator set by  
component */  
MQLONG   CompCode;         /* Completion code */  
MQLONG   Reason;          /* Reason code qualifying CompCode */
```

## MQZ\_SET\_AUTHORITY\_2 -Yetki tanımla (genişletilmiş)

Bu işlev bir MQZAS\_VERSION\_2 yetki hizmeti bileşeni tarafından sağlanır ve kuyruk yöneticisi tarafından, bir varlığın belirtilen nesneye erişmesi için sahip olduğu yetkiyi ayarlamak üzere başlatılır.

Bu işleve ilişkin işlev tanıtıcısı (MQZEP için) MQZID\_SET\_AUTHORITY olur.

**Not:** Bu işlev, var olan yetkileri geçersiz kılar. Varolan yetkileri korumak için, bu işlevi bu işlevle yeniden ayarlamamız gerekir.

MQZ\_SET\_AUTHORITY\_2 is like MQZ\_SET\_AUTHORITY, but with the **EntityName** parameter replaced by the **EntityData** parameter.

## Sözdizimi

```
MQZ_SET_AUTHORITY_2( QMgrName , EntityData , EntityType , ObjectName ,  
Objectype , Authority , ComponentData , Continuation , CompCode , Reason )
```

## Parametreler

### QMgrName

Tip: MQCHAR48 -giriş

Kuyruk yöneticisi adı. Bileşeni çağıran kuyruk yöneticisinin adı. Bu ad, değıştirgenin tam uzunluđuna kadar boşluklarla dolduruldu; ad boş değeri bir karakterle sonlandırılmaz.

Kuyruk yöneticisi adı, bileşene bilgi için iletilir; yetki hizmeti arabirimi, bileşenin bu bileşeni tanımlı bir şekilde kullanmasını gerektirmez.

### EntityData

Tip: MQZED-giriş

Varlık verileri. Nesneye ilişkin yetkisi belirlenmek üzere olan varlıkla ilgili veriler. Ayrıntılar için bkz. [“MQZED-Varlık tanımlayıcısı” sayfa 1653.](#)

### EntityType

Tip: MQHOME-giriş

Varlık tipi. *EntityData* tarafından belirtilen varlık tipi. Bu değeri aşağıdaki değeriardan biri olmalıdır:

#### **MQZAET\_PRINCIPAL**

Müdür.

#### **MQZAET\_GRP**

Grup.

### ObjectName

Tip: MQCHAR48 -giriş

Nesne adı. Varlık yetkisinin ayarlanacağı nesnenin adı. Dizimin uzunluk üst sınırı 48 karakterdir; boşlukla sağa yaslanmış ise, bu karakter uzunluđuna göre daha kısadır. Ad boş değeri bir karakter tarafından sonlandırılmadı.

*ObjectType* MQOT\_Q\_MGR ise, bu ad *QMgrName* ile aynıdır.

### ObjectType

Tip: MQHOME-giriş

Nesne tipi. *ObjectName* tarafından belirtilen varlık tipi. Bu değeri aşağıdaki değeriardan biri olmalıdır:

#### **MQOT\_AUTH\_INFO**

Kimlik doğrulama bilgileri.

#### **MQOT\_KANAL**

Kanal.

#### **MQOT\_CLNTCONN\_CHANNEL**

İstemci bağlantı kanalı.

#### **MQOT\_LISTENER**

Dinleyici.

#### **MQOT\_NAMELIST**

İsim listesi.

#### **MQOT\_PROCESS**

Süreç tanımlaması.

#### **MQOT\_Q**

Kuyruk.

#### **MQOT\_Q\_MGR**

Kuyruk yöneticisi.

#### **MQOT\_SERVICE**

Servis.

#### **MQOT\_KONUSU**

Konu.



## Yetki

Tip: MQHOME-giriş

Varlık yetkisi. Bir yetki belirleniyorsa, bu alan uygun yetki işlemine eşittir (MQZAO\_\* sabiti). Birden fazla yetki belirleniyorsa, bu alan, ilgili MQZAO\_\* değişmezlerinin bit ya da bitişiklerinden biridir.

## ComponentData

Tip: MQBYTE xComponentDataUzunluk-giriş/çıkış

Bileşen verileri. Bu veriler, bu bileşen adına kuyruk yöneticisi tarafından tutulur; bu bileşen tarafından sağlanan işlevlerden herhangi biri tarafından yapılan değişiklikler korunur ve bu bileşen işlevlerinden biri bir sonraki defa çağrılır.

Bu veri alanının uzunluğu, MQZ\_INIT\_AUTHORITY çağrısının **ComponentDataLength** parametresindeki kuyruk yöneticisi tarafından geçirilir.

## Devam

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Bileşene göre devam göstergesi kümesi. Aşağıdaki değerler belirlenebilir:

### **MQZCI\_VARSAYILAN**

Devam eden kuyruk yöneticisine bağımlı.

MQZ\_CHECK\_AUTHORITY için bu, MQZCI\_STOP ile aynı etkiye sahiptir.

### **MQZCI\_CONTINY**

Sonraki bileşenle devam edin.

### **MQZCI\_STOP**

Sonraki bileşenle devam ettirmeyin.

## CompCode

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Tamamlanma kodu. Bu değer aşağıdaki değerlerden biri olmalıdır:

### **MQCC\_OK**

İşlem başarıyla tamamlandı.

### **MQCC\_FAILED**

Arama başarısız oldu.

## Neden

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Neden kodu ön eleme *CompCode*.

*CompCode* , MQCC\_OK ise:

### **MQRC\_NONE**

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_FAILED ise:

### **MQRC\_NOT\_YETKILI**

(2035, X'7F3') Erişim için yetkili değil.

### **MQRC\_SERVICE\_ERROR**

(2289, X'8F1') Hizmet erişimi beklenmeyen bir hata oluştu.

### **MQRC\_SERVICE\_NOT\_VAR**

(2285, X'8ED') Undering hizmeti kullanılamıyor.

### **MQRC\_UNKNOWN\_ENTITY**

(2292, X'8F4') Hizmet için bilinmeyen varlık.

Bu neden kodlarına ilişkin ek bilgi için [API tamamlama ve neden kodları](#) başlıklı konuya bakın.

## C çağırma

```
MQZ_SET_AUTHORITY_2 (QMgrName, &EntityData, EntityType, ObjectName,  
Objectype, Authority, ComponentData,  
&Continuation, &CompCode, &Reason);
```

Hizmete geçirilen parametreler aşağıdaki gibi bildirilir:

```
MQCHAR48 QMgrName;          /* Queue manager name */  
MQZED EntityData;          /* Entity data */  
MQLONG EntityType;         /* Entity type */  
MQCHAR48 ObjectName;       /* Object name */  
MQLONG Objectype;         /* Object type */  
MQLONG Authority;         /* Authority to be checked */  
MQBYTE ComponentData[n];  /* Component data */  
MQLONG Continuation;      /* Continuation indicator set by  
component */  
MQLONG CompCode;          /* Completion code */  
MQLONG Reason;           /* Reason code qualifying CompCode */
```

## MQZ\_TERM\_AUTHORITY-Yetki hizmetini sonlandır

Bu işlev, bir yetkilendirme hizmeti bileşeni tarafından sağlanır ve kuyruk yöneticisi tarafından bu bileşenin hizmetlerinin artık gerekmediği durumlarda başlatılır. İşlev, bileşenin gerektirdiği herhangi bir temizleme işlemini gerçekleştirmelidir.

Bu işleve ilişkin işlev tanıtıcısı (MQZEP için) MQZID\_TERM\_AUTHORITY olur.

### Sözdizimi

```
MQZ_TERM_AUTHORITY( Hconfig , Options , QMgrName , ComponentData , CompCode ,  
Reason )
```

### Parametreler

#### Hconfig

Tip: MQHCONFIG-input

Yapılandırma tanıtıcısı. Bu tanıtıcı, sonlandırılmakta olan belirli bir bileşeni gösterir. Bu bileşen, kuyruk yöneticisini MQZEP işleviyle çağırırken, bu bileşen tarafından kullanılır.

#### Seçenekler

Tip: MQHOME-giriş

Sonlandırma seçenekleri. Bu değer aşağıdaki değerlerden biri olmalıdır:

#### MQZTO\_BIRINCIL

Birincil sonlandırma.

#### MQZTO\_IKINCIL

İkincil sonlandırma.

#### QMgrName

Tip: MQCHAR48 -giriş

Kuyruk yöneticisi adı. Bileşeni çağırın kuyruk yöneticisinin adı. Bu ad, değiştirgenin tam uzunluğuna kadar boşluklarla dolduruldu; ad boş değerli bir karakterle sonlandırılmaz.

Kuyruk yöneticisi adı, bileşene bilgi için iletilir; yetki hizmeti arabirimi, bileşenin bu bileşeni tanımlı bir şekilde kullanmasını gerektirmez.

#### ComponentData

Tip: MQBYTE x ComponentDataUzunluk-giriş/çıkış

Bileşen verileri. Bu veriler, bu bileşen adına kuyruk yöneticisi tarafından tutulur; bu bileşen tarafından sağlanan işlevlerden herhangi biri tarafından yapılan değişiklikler korunur ve bu bileşen işlevlerinden biri bir sonraki defa çağrılır.

Bu veri alanının uzunluğu, MQZ\_INIT\_AUTHORITY çağrısında ComponentDataLength değiştirilmesindeki kuyruk yöneticisi tarafından geçirilir.

MQZ\_TERM\_AUTHORITY çağrısı tamamlandığında, kuyruk yöneticisi bu verileri atar.

### CompCode

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Tamamlanma kodu. Bu değer aşağıdaki değerlerden biri olmalıdır:

#### MQCC\_OK

İşlem başarıyla tamamlandı.

#### MQCC\_FAILED

Arama başarısız oldu.

### Neden

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Neden kodu ön eleme *CompCode*.

*CompCode* , MQCC\_OK ise:

#### MQRC\_NONE

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_FAILED ise:

#### MQRC\_SERVICE\_NOT\_VAR

(2285, X'8ED') Undering hizmeti kullanılamıyor.

#### MQRC\_TERMINATION\_FAILED

(2287, X'8FF') Sona erdirme tanımlanmamış bir nedenden dolayı başarısız oldu.

Bu neden kodlarına ilişkin ek bilgi için [API tamamlama ve neden kodları](#) başlıklı konuya bakın.

## C çağırma

```
MQZ_TERM_AUTHORITY (Hconfig, Options, QMgrName, ComponentData,  
&CompCode, &Reason);
```

Hizmete geçirilen parametreler aşağıdaki gibi bildirilir:

```
MQHCONFIG  Hconfig;           /* Configuration handle */  
MQLONG     Options;          /* Termination options */  
MQCHAR48   QMgrName;        /* Queue manager name */  
MQBYTE     ComponentData[n]; /* Component data */  
MQLONG     CompCode;        /* Completion code */  
MQLONG     Reason;          /* Reason code qualifying CompCode */
```

## MQZ\_DELETE\_NAME-Silme adı

Bu işlev bir ad hizmeti bileşeni tarafından sağlanır ve kuyruk yöneticisi tarafından, belirtilen kuyruğa ilişkin bir girişi silmek için başlatılır.

Bu işleve ilişkin işlev tanıtıcısı (MQZEP için) MQZID\_DELETE\_NAME olur.

### Sözdizimi

```
MQZ_DELETE_NAME( QMgrName , QName , ComponentData , Continuation , CompCode ,  
Reason )
```

## Parametreler

### QMgrName

Tip: MQCHAR48 -giriş

Kuyruk yöneticisi adı. Bileşeni çağıran kuyruk yöneticisinin adı. Bu ad, değiştirgenin tam uzunluğuna kadar boşluklarla dolduruldu; ad boş değerli bir karakterle sonlandırılmaz.

Kuyruk yöneticisi adı, bileşene bilgi için iletilir; yetki hizmeti arabirimi, bileşenin bu bileşeni tanımlı bir şekilde kullanmasını gerektirmez.

### QName

Tip: MQCHAR48 -giriş

Kuyruk adı. Bir girişin silineceği kuyruğun adı. Bu ad, değiştirgenin tam uzunluğuna kadar boşluklarla dolduruldu; ad boş değerli bir karakterle sonlandırılmaz.

### ComponentData

Tip: MQBYTE x ComponentDataUzunluk-giriş/çıkış

Bileşen verileri. Bu veriler, bu bileşen adına kuyruk yöneticisi tarafından tutulur; bu bileşen tarafından sağlanan işlevlerden herhangi biri tarafından yapılan değişiklikler korunur ve bu bileşen işlevlerinden biri bir sonraki defa çağrılır.

Bu veri alanının uzunluğu, MQZ\_INIT\_NAME çağrısındaki ComponentDataLength parametresindeki kuyruk yöneticisi tarafından geçirilir.

### Devam

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Bileşene göre devam göstergesi kümesi. Bu değer aşağıdaki değerlerden biri olmalıdır:

#### **MQZCI\_VARSAYILAN**

Devam eden kuyruk yöneticisine bağımlı.

#### **MQZCI\_STOP**

Sonraki bileşenle devam ettirmeyin.

**MQZ\_DELETE\_NAME** komutu için, kuyruk yöneticisi **Continuation** değiştirgeninde ne döndürüldüğü önemli değil, başka bir bileşeni başlatmaya çalışmamaktadır.

### CompCode

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Tamamlanma kodu. Bu değer aşağıdaki değerlerden biri olmalıdır:

#### **MQCC\_OK**

İşlem başarıyla tamamlandı.

#### **MQCC\_UYARI**

Uyarı (kısmi tamamlama).

#### **MQCC\_FAILED**

Arama başarısız oldu.

### Neden

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Neden kodu ön eleme *CompCode*.

*CompCode* , MQCC\_OK ise:

#### **MQRC\_NONE**

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* , MQCC\_UYARI ise:

#### **MQRC\_UNKNOWN\_**

(2288, X'8F0') Kuyruk adı bulunamadı.

**Not:** Temeldeki hizmet bu vakaya ilişkin başarılarla yanıt veriyorsa, bu kodu iade etmek mümkün olmayabilir.

*CompCode* MQCC\_FAILED ise:

#### **MQRC\_SERVICE\_ERROR**

(2289, X'8F1') Hizmet erişimi beklenmeyen bir hata oluştu.

#### **MQRC\_SERVICE\_NOT\_VAR**

(2285, X'8ED') Undering hizmeti kullanılmıyor.

Bu neden kodlarına ilişkin ek bilgi için [API tamamlama ve neden kodları](#) başlıklı konuya bakın.

## C çağırma

```
MQZ_DELETE_NAME (QMgrName, QName, ComponentData, &Continuation,  
&CompCode, &Reason);
```

Hizmete geçirilen parametreler aşağıdaki gibi bildirilir:

```
MQCHAR48 QMgrName;          /* Queue manager name */  
MQCHAR48 QName;            /* Queue name */  
MQBYTE ComponentData[n];   /* Component data */  
MQLONG Continuation;       /* Continuation indicator set by  
                             component */  
MQLONG CompCode;           /* Completion code */  
MQLONG Reason;             /* Reason code qualifying CompCode */
```

## MQZ\_INIT\_NAME-Ad hizmetini kullanıma hazırla

Bu işlev bir ad hizmeti bileşeni tarafından sağlanır ve bileşen yapılandırması sırasında kuyruk yöneticisi tarafından başlatılır. Kuyruk yöneticisine bilgi sağlamak için MQZEP çağırılması beklenir.

Bu işleve ilişkin işlev tanıtcısı (MQZEP için) MQZID\_INIT\_NAME olur.

### Sözdizimi

```
MQZ_INIT_NAME( Hconfig , Options , QMgrName , ComponentDataLength ,  
ComponentData , Version , CompCode , Reason )
```

## Parametreler

### Hconfig

Tip: MQHCONFIG-input

Yapılandırma tanıtcısı. Bu tanıtcı, ilk kullanıma hazırlanmakta olan bileşeni gösterir. Bu bileşen, kuyruk yöneticisini MQZEP işleviyle çağırırken, bu bileşen tarafından kullanılır.

### Seçenekler

Tip: MQHOME-girişi

Başlatma seçenekleri. Bu değer aşağıdaki değerlerden biri olmalıdır:

#### **MQZI\_BIRINCIL**

Birincil kullanıma hazırlama.

#### **MQZIO\_IKINCIL**

İkincil kullanıma hazırlama.

### QMgrName

Tip: MQCHAR48 -giriş

Kuyruk yöneticisi adı. Bileşeni çağırın kuyruk yöneticisinin adı. Bu ad, değiştirgenin tam uzunluğuna kadar boşluklarla dolduruldu; ad boş değerli bir karakterle sonlandırılmaz.

Kuyruk yöneticisi adı, bileşene bilgi için iletilir; yetki hizmeti arabirimi, bileşenin bu bileşeni tanımlı bir şekilde kullanmasını gerektirmez.

### **ComponentDataUzunluğu**

Tip: MQHOME-girişi

Bileşen verilerinin uzunluğu. *ComponentData* alanının bayt cinsinden uzunluğu. Bu uzunluk, bileşen yapısını verilerinde tanımlanır.

### **ComponentData**

Tip: MQBYTE x ComponentDataUzunluk-giriş/çıkış

Bileşen verileri. Bileşen birincil kullanıma hazırlama işlevini çağırılmadan önce, bu değer tüm sıfırlar için başlatılır. Bu veriler, bu bileşen adına kuyruk yöneticisi tarafından tutulur; bu bileşen tarafından sağlanan işlevler (kullanıma hazırlama işlevi de içinde olmak üzere) tarafından bu bileşen için yapılan değişiklikler korunur ve bu bileşen işlevlerinden bir sonraki kez çağırılır.

Bu veri alanının uzunluğu, MQZ\_INIT\_AUTHORITY çağırısının **ComponentDataLength** parametresindeki kuyruk yöneticisi tarafından geçirilir.

### **S\00fcr\00fcm**

Tip: MQHOME-giriş/çıkış

Sürüm numarası. Kullanıma hazırlama işlevine ilişkin giriş, kuyruk yöneticisinin desteklediği en yüksek sürüm numarasını belirtir. Başlatma işlevinin desteklediği arabirimin sürümüne (gerekirse) bunu değiştirmesi gerekir. Geri dönerse, kuyruk yöneticisi bileşenin döndürdüğü sürümü desteklemiyorsa, bu bileşen MQZ\_TERM\_NAME işlevini çağırır ve bu bileşenin başka bir kullanımını yapmaz.

Aşağıdaki değerler desteklenir:

#### **MQZAS\_VERSION\_1**

Sürüm 1.

### **CompCode**

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Tamamlanma kodu. Bu değer aşağıdaki değerlerden biri olmalıdır:

#### **MQCC\_OK**

İşlem başarıyla tamamlandı.

#### **MQCC\_FAILED**

Arama başarısız oldu.

### **Neden**

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Neden kodu ön eleme *CompCode*.

*CompCode* , MQCC\_OK ise:

#### **MQRC\_NONE**

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_FAILED ise:

#### **MQRC\_INITIALIZATION\_FAILED**

(2286, X'8EE') Kullanıma hazırlama, tanımlanmamış bir neden için başarısız oldu.

#### **MQRC\_SERVICE\_NOT\_VAR**

(2285, X'8ED') Undering hizmeti kullanılamıyor.

Bu neden kodlarına ilişkin ek bilgi için [API tamamlama ve neden kodları](#) başlıklı konuya bakın.

## **C çağırma**

```
MQZ_INIT_NAME (Hconfig, Options, QMgrName, ComponentDataLength,  
ComponentData, &Version, &CompCode, &Reason);
```

Hizmete geçirilen parametreler aşağıdaki gibi bildirilir:

```
MQHCONFIG  Hconfig;          /* Configuration handle */
MQLONG     Options;        /* Initialization options */
MQCHAR48   QMgrName;      /* Queue manager name */
MQLONG     ComponentDataLength; /* Length of component data */
MQBYTE     ComponentData[n]; /* Component data */
MQLONG     Version;       /* Version number */
MQLONG     CompCode;      /* Completion code */
MQLONG     Reason;        /* Reason code qualifying CompCode */
```

## MQZ\_INSERT\_NAME-Ad ekle

Bu işlev, bir ad hizmeti bileşeni tarafından sağlanır ve kuyruk yöneticisi tarafından, kuyruğa sahip olan kuyruk yöneticisinin adını içeren, belirtilen kuyruğa bir giriş eklemek için başlatılır. Kuyruk hizmette önceden tanımlandıysa, arama başarısız olur.

Bu işleve ilişkin işlev tanıtıcısı (MQZEP için) MQZID\_INSERT\_NAME olur.

### Sözdizimi

MQZ\_INSERT\_NAME( QMgrName , QName , ResolvedQMgrName , ComponentData , Continuation , CompCode , Reason )

### Parametreler

#### QMgrName

Tip: MQCHAR48 -giriş

Kuyruk yöneticisi adı. Bileşeni çağıran kuyruk yöneticisinin adı. Bu ad, değiştirgenin tam uzunluğuna kadar boşluklarla dolduruldu; ad boş değerli bir karakterle sonlandırılmaz.

Kuyruk yöneticisi adı, bileşene bilgi için iletilir; yetki hizmeti arabirimi, bileşenin bu bileşeni tanımlı bir şekilde kullanmasını gerektirmez.

#### QName

Tip: MQCHAR48 -giriş

Kuyruk adı. Girişin ekleneceği kuyruğun adı. Bu ad, değiştirgenin tam uzunluğuna kadar boşluklarla dolduruldu; ad boş değerli bir karakterle sonlandırılmaz.

#### ResolvedQMgrAdı

Tip: MQCHAR48 -giriş

Çözülmüş kuyruk yöneticisi adı. Kuyruk çözümleyicisinin bulunduğu kuyruk yöneticisinin adı. Bu ad, değiştirgenin tam uzunluğuna kadar boşluklarla dolduruldu; ad boş değerli bir karakterle sonlandırılmaz.

#### ComponentData

Tip: MQBYTE xComponentDataUzunluk-giriş/çıkış

Bileşen verileri. Bu veriler, bu bileşen adına kuyruk yöneticisi tarafından tutulur; bu bileşen tarafından sağlanan işlevler (kullanıma hazırlama işlevi de içinde olmak üzere) tarafından bu bileşen için yapılan değişiklikler korunur ve bu bileşen işlevlerinden bir sonraki kez çağrılır.

Bu veri alanının uzunluğu, MQZ\_INIT\_NAME çağrısının **ComponentDataLength** parametresindeki kuyruk yöneticisi tarafından geçirilir.

#### Devam

Tip: MQHOME-giriş/çıkış

Bileşene göre devam göstergesi kümesi. MQZ\_INSERT\_NAME için, kuyruk yöneticisi **Continuation** değiştirgesinde geri döndürülen başka bir bileşeni başlatmaya çalışmamaktadır.

Aşağıdaki değerler desteklenir:

## **MQZCI\_VARSAYILAN**

Devam eden kuyruk yöneticisine bağımlı.

## **MQZCI\_STOP**

Sonraki bileşenle devam ettirmeyin.

### **CompCode**

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Tamamlanma kodu. Bu değer aşağıdaki değerlerden biri olmalıdır:

### **MQCC\_OK**

İşlem başarıyla tamamlandı.

### **MQCC\_FAILED**

Arama başarısız oldu.

### **Neden**

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Neden kodu ön eleme *CompCode*.

*CompCode* , MQCC\_OK ise:

### **MQRC\_NONE**

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_FAILED ise:

### **MQRC\_Q\_ALREADY\_EXISTS**

(2290, X'8F2') Kuyruk nesnesi zaten var.

### **MQRC\_SERVICE\_ERROR**

(2289, X'8F1') Hizmet erişimi beklenmeyen bir hata oluştu.

### **MQRC\_SERVICE\_NOT\_VAR**

(2285, X'8ED') Undering hizmeti kullanılamıyor.

Bu neden kodlarına ilişkin ek bilgi için [API tamamlama ve neden kodları](#) başlıklı konuya bakın.

## **C çağırma**

```
MQZ_INSERT_NAME (QMgrName, QName, ResolvedQMgrName, ComponentData,  
&Continuation, &CompCode, &Reason);
```

Hizmete geçirilen parametreler aşağıdaki gibi bildirilir:

```
MQCHAR48 QMgrName;          /* Queue manager name */  
MQCHAR48 QName;            /* Queue name */  
MQCHAR48 ResolvedQMgrName; /* Resolved queue manager name */  
MQBYTE ComponentData[n];  /* Component data */  
MQLONG Continuation;      /* Continuation indicator set by  
                           component */  
MQLONG CompCode;          /* Completion code */  
MQLONG Reason;           /* Reason code qualifying CompCode */
```

## **MQZ\_LOOKUP\_NAME-Arama adı**

Bu işlev, bir ad hizmeti bileşeni tarafından sağlanır ve kuyruk yöneticisi tarafından, belirtilen bir kuyruk için sahip olan kuyruk yöneticisinin adını almak üzere başlatılır.

Bu işleve ilişkin işlev tanıtıcısı (MQZEP için) MQZID\_LOOKUP\_NAME olur.

### **Sözdizimi**

```
MQZ_LOOKUP_NAME( QMgrName , QName , ResolvedQMgrName , ComponentData ,  
Continuation , CompCode , Reason )
```



## Parametreler

### QMgrName

Tip: MQCHAR48 -giriş

Kuyruk yöneticisi adı. Bileşeni çağıran kuyruk yöneticisinin adı. Bu ad, değiştirgenin tam uzunluğuna kadar boşluklarla dolduruldu; ad boş değerli bir karakterle sonlandırılmaz.

Kuyruk yöneticisi adı, bileşene bilgi için iletilir; yetki hizmeti arabirimi, bileşenin bu bileşeni tanımlı bir şekilde kullanmasını gerektirmez.

### QName

Tip: MQCHAR48 -giriş

Kuyruk adı. Bir girişin çözümlenmesi için kullanılan kuyruğun adı. Bu ad, değiştirgenin tam uzunluğuna kadar boşluklarla dolduruldu; ad boş değerli bir karakterle sonlandırılmaz.

### ResolvedQMgrAdı

Tip: MQCHAR48 -çıkış

Çözülmüş kuyruk yöneticisi adı. İşlev başarıyla tamamlanırsa, kuyruğun sahibi olan kuyruk yöneticisinin adı olur.

Hizmet bileşeni tarafından döndürülen ad, parametrenin tam uzunluğuna kadar boşluklarla doldurulmalıdır; ad boş bir karakterle sonlandırılmamalı ya da baştaki ya da gömülü boşluklar içermemelidir.

### ComponentData

Tip: MQBYTEComponentDataLength -giriş/çıkış

Bileşen verileri. Bu veriler, bu bileşen adına kuyruk yöneticisi tarafından tutulur; bu bileşen tarafından sağlanan işlevler (kullanıma hazırlama işlevi de içinde olmak üzere) tarafından bu bileşen için yapılan değişiklikler korunur ve bu bileşen işlevlerinden bir sonraki kez çağrılır.

Bu veri alanının uzunluğu, MQZ\_INIT\_NAME çağrısının **ComponentDataLength** parametresindeki kuyruk yöneticisi tarafından geçirilir.

### Devam

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Bileşene göre devam göstergesi kümesi. MQZ\_LOOKUP\_NAME için, kuyruk yöneticisi aşağıdaki gibi başka bir ad hizmeti bileşenin başlatılıp başlatılmayacağını belirtir:

- *CompCode* MQCC\_OK ise, başka bileşen başlatılmıyorsa, *Devami* içinde döndürülen değer ne olursa olsun.
- *CompCode* MQCC\_OK değilse, *Continuation* MQZCI\_STOP değilse, başka bir bileşen başlatılır.

Aşağıdaki değerler desteklenir:

#### MQZCI\_VARSAYILAN

Devam eden kuyruk yöneticisine bağımlı.

#### MQZCI\_CONTINY

Sonraki bileşenle devam edin.

#### MQZCI\_STOP

Sonraki bileşenle devam ettirmeyin.

### CompCode

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Tamamlanma kodu. Bu değer aşağıdaki değerlerden biri olmalıdır:

#### MQCC\_OK

İşlem başarıyla tamamlandı.

#### MQCC\_FAILED

Arama başarısız oldu.

## Neden

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Neden kodu ön eleme *CompCode*.

*CompCode* , MQCC\_OK ise:

### **MQRC\_NONE**

(0, X'000') Raporlamak için bir neden yok.

*CompCode* MQCC\_FAILED ise:

### **MQRC\_SERVICE\_ERROR**

(2289, X'8F1') Hizmet erişimi beklenmeyen bir hata oluştu.

### **MQRC\_SERVICE\_NOT\_VAR**

(2285, X'8ED') Undering hizmeti kullanılamıyor.

### **MQRC\_UNKNOWN\_Q\_NAME**

(2288, X'8F0') Kuyruk adı bulunamadı.

Bu neden kodlarına ilişkin ek bilgi için [API tamamlama ve neden kodları](#) başlıklı konuya bakın.

## C çağırma

```
MQZ_LOOKUP_NAME (QMgrName, QName, ResolvedQMgrName, ComponentData,  
&Continuation, &CompCode, &Reason);
```

Hizmete geçirilen parametreler aşağıdaki gibi bildirilir:

```
MQCHAR48 QMgrName;          /* Queue manager name */  
MQCHAR48 QName;            /* Queue name */  
MQCHAR48 ResolvedQMgrName; /* Resolved queue manager name */  
MQBYTE ComponentData[n];  /* Component data */  
MQLONG Continuation;      /* Continuation indicator set by  
                           component */  
MQLONG CompCode;          /* Completion code */  
MQLONG Reason;            /* Reason code qualifying CompCode */
```

## MQZ\_TERM\_NAME-Ad hizmetini sonlandır

Bu işlev bir ad hizmeti bileşeni tarafından sağlanır ve kuyruk yöneticisi tarafından bu bileşenin hizmetlerinin artık gerekmediği durumlarda başlatılır. İşlev, bileşenin gerektirdiği herhangi bir temizleme işlemini gerçekleştirilmelidir.

Bu işleve ilişkin işlev tanıttıcısı (MQZEP için) MQZID\_TERM\_NAME olur.

## Sözdizimi

```
MQZ_TERM_NAME( Hconfig , Options , QMgrName , ComponentData , CompCode ,  
Reason )
```

## Parametreler

### Hconfig

Tip: MQHCONFIG-input

Yapılandırma tanıttıcısı. Bu tanıttıcı, sonlandırılmakta olan belirli bir bileşeni gösterir. Bu bileşen, MQZEP işleviyle kuyruk yöneticisi çağırılırken bileşen tarafından kullanılır.

### Seçenekler

Tip: MQHOME-girişi

Sonlandırma seçenekleri. Bu değer aşağıdaki değerlerden biri olmalıdır:

## **MQZTO\_BIRINCIL**

Birincil sonlandırma.

## **MQZTO\_IKINCIL**

İkincil sonlandırma.

### **QMgrName**

Tip: MQCHAR48 -giriş

Kuyruk yöneticisi adı. Bileşeni çağıran kuyruk yöneticisinin adı. Bu ad, değıştirgenin tam uzunluđuna kadar boşluklarla dolduruldu; ad boş değeri bir karakterle sonlandırılmaz.

Kuyruk yöneticisi adı, bileşene bilgi için iletilir; yetki hizmeti arabirimi, bileşenin bu bileşeni tanımlı bir şekilde kullanmasını gerektirmez.

### **ComponentData**

Tip: MQBYTE x ComponentDataUzunluk-giriş/çıkış

Bileşen verileri. Bu veriler, bu bileşen adına kuyruk yöneticisi tarafından tutulur; bu bileşen tarafından sağlanan işlevler (kullanıma hazırlama işlevi de içinde olmak üzere) tarafından bu bileşen için yapılan değışiklikler korunur ve bu bileşen işlevlerinden bir sonraki kez çağrılır.

Bileşen verileri, paylaşılan belleđde tüm süreçler tarafından erişilebilir.

Bu veri alanının uzunluđu, MQZ\_INIT\_NAME çağrısının **ComponentDataLength** parametresindeki kuyruk yöneticisi tarafından geçirilir.

MQZ\_TERM\_NAME çağrısı tamamlanınca, kuyruk yöneticisi bu verileri atar.

### **CompCode**

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Tamamlanma kodu. Bu değeri aşıđdaki değerlerden biri olmalıdır:

#### **MQCC\_OK**

İşlem başarıyla tamamlandı.

#### **MQCC\_FAILED**

Arama başarısız oldu.

### **Neden**

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Neden kodu ön eleme *CompCode*.

*CompCode* , MQCC\_OK ise:

#### **MQRC\_NONE**

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_FAILED ise:

#### **MQRC\_TERMINATION\_FAILED**

(2287, X'8FF') Sona erdirme tanımlanmamış bir nedenden dolayı başarısız oldu.

#### **MQRC\_SERVICE\_NOT\_VAR**

(2285, X'8ED') Undering hizmeti kullanılmıyor.

Bu neden kodlarına ilişkin ek bilgi için [API tamamlama ve neden kodları](#) başlıklı konuya bakın.

## **C çağırma**

```
MQZ_TERM_NAME (Hconfig, Options, QMgrName, ComponentData, &CompCode,  
&Reason);
```

Hizmete geçirilen parametreler aşıđdaki gibi bildirilir:

```
MQHCONFIG Hconfig; /* Configuration handle */
```

```

MQLONG      Options;          /* Termination options */
MQCHAR48    QMgrName;        /* Queue manager name */
MQBYTE      ComponentData[n]; /* Component data */
MQLONG      CompCode;        /* Completion code */
MQLONG      Reason;          /* Reason code qualifying CompCode */

```

## MQZAC-Uygulama bağlamı

MQZAC yapısı, *ApplicationContext* parametresi için MQZ\_AUTHENTICATE\_USER çağrısında kullanılıyor. Bu parametre, çağırılan uygulamaya ilişkin verileri belirtir.

Tablo 1, yapıdaki alanları özetler.

Çizelge 839. MQZAC 'deki alanlar	
Alan	Tanım
<u>StrucId</u>	Yapı tanıtcısı
<u>S\u00fcr\u00fcm</u>	Yapı sürüm numarası
<u>ProcessId</u>	İşlem tanıtcısı
<u>ThreadId</u>	İş parçacığı tanıtcısı
<u>ApplName</u>	Uygulama adı
<u>UserID</u>	Kullanıcı kimliği
<u>EffectiveUserTanıtcısı</u>	Etkin kullanıcı kimliği
<u>Ortam</u>	Ortam
<u>CallerType</u>	Çağırılan tipi
<u>AuthenticationType</u>	Kimlik denetimi tipi
<u>BindType</u>	Bağ Tipi

## Alanlar

### StrucId

Tip: MQCHAR4 -giriş

Yapı tanıtcısı. Değer aşağıdaki gibidir:

#### **MQZAC\_STRUC\_ID**

Uygulama bağlamı yapısına ilişkin tanıtcı.

C programlama dili için, sabit MQZAC\_STRUC\_ID\_ARRAY değişmezi de tanımlanır; bu, MQZAC\_STRUC\_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisi olur.

### S\u00fcr\u00fcm

Tip: MQHOME-giriş

Yapı sürüm numarası. Değer aşağıdaki gibidir:

#### **MQZAC\_VERSION\_1**

Version-1 uygulama bağlamı yapısı. Değişmez MQZAC\_CURRENT\_VERSION, yürürlükteki sürümün sürüm numarasını belirtir.

### ProcessId

Tip: MQPID-giriş

Uygulamaya ilişkin süreç tanıtcısı.

### ThreadId

Tip: MQTID-input

Uygulamanın iş parçacığı tanıtcısı.

**ApplName**

Tip: MQCHAR28 -giriş

Uygulama adı.

**UserID**

Tip: MQCHAR12 -giriş

Kullanıcı kimliği. AIX and Linux üzerinde bu alan, uygulamanın gerçek kullanıcı kimliğini belirtir. Windows tarihinde bu alan, uygulamanın kullanıcı kimliğini belirtir.

**EffectiveUserkimliği**

Tip: MQCHAR12 -giriş

Etkin kullanıcı kimliği. AIX and Linux tarihinde bu alan, uygulamanın etkin kullanıcı kimliğini belirtir. Windows üzerinde bu alan boştur.

**Ortam**

Tip: MQHOME-girişi

Ortam. Bu alan, çağrılarının yapıldığı ortamı belirtir. Alan, aşağıdaki değerlerden biridir:

**MQXE\_COMMAND\_SERVER**

Komut sunucusu

**MQXE\_MQSC**

runmqsc komut yorumlayıcısı

**MQXE\_MCA**

İleti kanalı aracısı MQXE\_OTHER

**MQXE\_DİĞER**

Tanımsız ortam

**CallerType**

Tip: MQHOME-girişi

Arayan Tipi. Bu alan, çağrıyı yapan programın tipini belirtir. Alan, aşağıdaki değerlerden biridir:

**MQXACT\_EXTERNAL**

Arama, kuyruk yöneticisi dışıdır.

**MQXACT\_INTERNAL**

Arama, kuyruk yöneticisinin iç arasıdır.

**AuthenticationType**

Tip: MQHOME-girişi

Kimlik Doğrulama Tipi. Bu alan, gerçekleştirilmekte olan kimlik doğrulamasının tipini belirtir. Alan, aşağıdaki değerlerden biridir:

**MQZAT\_INITIAL\_CONTEXT**

Kimlik doğrulama çağrısı, kullanıcı bağlamının başlatıldığı için sona ermiş. Bu değer bir MQCONN ya da MQCONNX çağrısı sırasında kullanılır.

**MQZAT\_CHANGE\_CONTEXT**

Kimlik doğrulama çağrısı, değiştirilmekte olan kullanıcı bağlamından kaynaklanır. Bu değer, MCA kullanıcı bağlamını değiştirdiğinde kullanılır. Üst konu: MQZAC-

**BindType**

Tip: MQHOME-girişi

Bağ Tanımlama Tipi. Bu alan, kullanımda olan bağlayıcının tipini belirtir. Alan, aşağıdaki değerlerden biridir:

**MQCNO\_FASTPATH\_BINDING**

Fastpath bağ tanımı.

**MQCNO\_SHARED\_BINDING**

Paylaşılan bağ tanımı.

## MQCNO\_ISOLATED\_BINDING

Yalıtılmış bağ.

### C bildirimini

Yapı alanlarını aşağıdaki gibi bildirin:

```
typedef struct tagMQZAC MQZAC;
struct tagMQZAC {
    MQCHAR4    StrucId;           /* Structure identifier */
    MQLONG     Version;          /* Structure version number */
    MQPID      ProcessId;        /* Process identifier */
    MQTID      ThreadId;        /* Thread identifier */
    MQCHAR28   ApplName;        /* Application name */
    MQCHAR12   UserID;          /* User identifier */
    MQCHAR12   EffectiveUserID;  /* Effective user identifier */
    MQLONG     Environment;      /* Environment */
    MQLONG     CallerType;       /* Caller type */
    MQLONG     AuthenticationType; /* Authentication type */
    MQLONG     BindType;        /* Bind type */
};
```

### MQZAD-Yetki verileri

MQZAD yapısı, iki değiştirge, bir giriş ve bir çıkış için MQZ\_ENUMERATE\_AUTHORITY\_DATA çağrısında kullanılıyor.

**Filter** ve **AuthorityBuffer** değiştirgeleriyle ilgili ek bilgi için “MQZ\_ENUMERATE\_AUTHORITY\_DATA-Yetki verilerini numaralandırır” sayfa 1612 konusuna bakın:

- Çağrıya giriş yapan **Filter** parametresi için MQZAD kullanılır. Bu parametre, çağrı tarafından döndürülen yetki verilerini seçmek için kullanılacak seçim ölçütlerini belirtir.
- MQZAD, çağrıdan çıkış olan **AuthorityBuffer** parametresi için de kullanılır. Bu parametre, bir profil adı, nesne türü ve varlık birleşimine ilişkin yetkileri belirtir.

Tablo 1. Yapıdaki alanları özetler.

Çizelge 840. MQZAD 'daki alanlar	
Alan	Tanım
<u>StrucId</u>	Yapı tanıtıcısı
<u>S\u00f0fcr\u00f0fcm</u>	Yapı sürüm numarası
<u>ProfileName</u>	Tanıtım adı
<u>ObjectType</u>	Nesne tipi
<u>Yetki</u>	Yetki
<u>EntityDataPtr</u>	Varlık verilerine işaretçi
<u>EntityType</u>	Varlık tipi
<u>Seçenekler</u>	Seçenekler

### Alanlar

#### StrucId

Tip: MQCHAR4 -giriş

Yapı tanıtıcısı. Değer aşağıdaki gibidir:

#### MQZAD\_STRUC\_ID

Yetki verileri yapısına ilişkin tanıtıcı.

C programlama dili için, MQZAD\_STRUC\_ID\_ARRAY değişmezi de tanımlanır; bu, MQZAD\_STRUC\_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisidir.

### **S\**

Tip: MQHOME-girişi

Yapı sürüm numarası. Değer aşağıdaki gibidir:

### **MQZAD\_VERSION\_1**

Version-1 uygulama bağlamı yapısı. Değişmez MQZAD\_CURRENT\_VERSION, yürürlükteki sürümün sürüm numarasını belirtir.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

### **MQZAD\_CURRENT\_VERSION**

Yetki verileri yapısının yürürlükteki sürümü.

### **ProfileName**

Tip: MQCHAR48 -giriş

Profil adı.

**Filter** parametresi için bu alan, yetki verilerinin gerekli olduğu profil adıdır. Ad alanın sonuna kadar boş bırakılırsa ya da ilk boş karakter, tüm tanımlı adları için yetki verileri döndürülür.

**AuthorityBuffer** parametresi için bu alan, belirtilen seçim ölçütleriyle eşleşen bir profilin adıdır.

### **ObjectType**

Tip: MQHOME-girişi

Nesne tipi.

**Filter** parametresi için bu alan, yetki verilerinin gerekli olduğu nesne tipidir. Bu değer MQOT\_ALL ise, tüm nesne tiplerine ilişkin yetki verileri döndürülür.

**AuthorityBuffer** parametresi için bu alan, **ProfileName** parametresiyle tanımlanan profilin geçerli olduğu nesne tipidir.

Değer aşağıdakilerden biridir; **Filter** parametresi için MQOT\_ALL değeri de geçerlidir:

### **MQOT\_AUTH\_INFO**

Kimlik doğrulama bilgileri

### **MQOT\_KANAL**

Kanal

### **MQOT\_CLNTCONN\_CHANNEL**

İstemci bağlantı kanalı

### **MQOT\_LISTENER**

Dinleyici

### **MQOT\_NAMELIST**

Ad Listesi

### **MQOT\_PROCESS**

Süreç tanımlaması

### **MQOT\_Q**

Kuyruk

### **MQOT\_Q\_MGR**

Kuyruk yöneticisi

### **MQOT\_SERVICE**

Hizmet

### **Yetki**

Tip: MQHOME-girişi

Otorite.

**Filter** parametresi için bu alan yok sayılır.

For the **AuthorityBuffer** parameter, this field represents the authorizations that the entity has to the objects identified by **ProfileName** and **ObjectType**. Varlığın yalnızca bir yetkisi varsa, bu alan uygun yetki değerine eşittir (MQZAO\_ \* sabiti). Varlığın birden çok yetkisi varsa, alan bit bilgedir ya da karşılık gelen MQZAO\_ \* sabitlerinden biridir.

### EntityDataPtr

Tip: PMQZED-giriş

Bir varlığı tanımlayan MQZED yapısının adresi.

**Filter** parametresi için, bu alan, yetki verilerinin gerekli olduğu varlığı tanımlayan bir MQZED yapısına işaret eder. **EntityDataPtr** boş gösterge ise, tüm varlıklar için yetki verileri döndürülür.

**AuthorityBuffer** parametresi için, bu alan, yetki verilerinin döndürüldüğü varlığı tanımlayan bir MQZED yapısına işaret eder.

### EntityType

Tip: MQHOME-girişi

Varlık tipi.

**Filter** parametresi için bu alan, yetki verilerinin gerekli olduğu varlık tipini belirtir. Bu değer MQZAET\_NONE ise, tüm varlık tiplerine ilişkin yetki verileri döndürülür.

**AuthorityBuffer** parametresi için bu alan, **EntityDataPtr** parametresine işaret edilen MQZED yapısı tarafından tanımlanan varlığın tipini belirtir.

Değer aşağıdakilerden biridir; **Filter** parametresi için, MQZAET\_NONE değeri de geçerlidir:

#### MQZAET\_PRINCIPAL

principal

#### MQZAET\_GRP

Grup

### Seçenekler

Tip: MQAUTHOPT-input

Seçenekler. Bu alan, görüntülenen tanımlar üzerinde denetim veren seçenekleri belirtir. Aşağıdaki değerlerden biri belirtilmeli:

#### MQAUTHOPT\_NAME\_ALL\_MATCHING

Tüm profilleri görüntüler

#### MQAUTHOPT\_NAME\_EXPLICIT

**ProfileName** alanında belirtildiği gibi, tam olarak aynı ada sahip profilleri görüntüler.

Ayrıca, aşağıdakilerden birinin de belirtilmesi gerekir:

#### MQAUTHOPT\_ENTITY\_SET

Varlığın, **ProfileName** parametresi tarafından belirtilen nesneye sahip olduğu birikmeli yetkiyi hesaplamak için kullanılan tüm profilleri görüntüler. **ProfileName** parametresi genel arama karakteri içermemelidir.

- If the specified entity is a principal, for each member of the set {entity, groups} the most applicable profile that applies to the object is displayed.
- Belirtilen varlık bir grupsa, nesne için geçerli olan gruptan en geçerli profil görüntülenir.
- If this value is specified, then the values of **ProfileName**, **ObjectType**, **EntityType**, and the entity name specified in the **EntityDataPtr** MQZED structure, must all be non-blank.

MQAUTHOPT\_NAME\_ALL\_MATCHING belirlendiyseniz, aşağıdaki değeri de belirtebilirsiniz:

#### MQAUTHOPT\_ENTITY\_EXPLICIT

**EntityDataPtr** MQZED yapısında belirtilen varlık adıyla tam olarak aynı varlık adına sahip profilleri görüntüler.



## C bildirimi

```
typedef struct tagMQZAD MQZAD;
struct tagMQZAD {
    MQCHAR4   StrucId;           /* Structure identifier */
    MQLONG    Version;          /* Structure version number */
    MQCHAR48  ProfileName;      /* Profile name */
    MQLONG    ObjectType;       /* Object type */
    MQLONG    Authority;        /* Authority */
    PMQZED    EntityDataPtr;    /* Address of MQZED structure identifying an
    entity */
    MQLONG    EntityType;       /* Entity type */
    MQAUTHOPT Options;         /* Options */
};
```

## MQZED-Varlık tanımlayıcısı

MQZED yapısı, yetkisinin denetleneceği varlığı belirtmek için bir dizi yetki hizmeti çağrısında kullanılır.

Tablo 1. Yapıdaki alanları özetler.

Çizelge 841. MQZED 'deki alanlar	
Alan	Tanım
<u>StrucId</u>	Yapı tanıtcısı
<u>S\u00fcr\u00fcm</u>	S\u00fcr\u00fcm
<u>EntityName Ptr</u>	Varlık Adı
<u>EntityDomainPtr</u>	Varlık etki alanı göstergesi
<u>SecurityId</u>	Güvenlik tanıtcısı
<u>CorrelationPtr</u>	İlinti göstergesi

## Alanlar

### StrucId

Tip: MQCHAR4 -giriş

Yapı tanıtcısı. Değer aşağıdaki gibidir:

#### MQZED\_STRUC\_ID

Varlık tanımlayıcısı yapısına ilişkin tanıtcı.

C programlama dili için, MQZED\_STRUC\_ID\_ARRAY değişmezi de tanımlanır; bu, MQZED\_STRUC\_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisidir.

### S\u00fcr\u00fcm

Tip: MQHOME-girişi

Yapı sürüm numarası. Değer aşağıdaki gibidir:

#### MQZED\_VERSION\_1

Version-1 varlık tanımlayıcı yapısı.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

#### MQZED\_CURRENT\_VERSION

Varlık tanımlayıcı yapısının yürürlükteki sürümü.

### EntityNamePtr

Tip: PMQCHAR-giriş

Profil adı.

Varlık adı adresi. Bu, yetkisini denetlenecek olan varlığın adına gösteren bir işarettir.

## EntityDomainPtr

Tip: PMQCHAR-giriş

Varlık etki alanı adının adresi. Bu, yetkisini denetlenecek olan varlığın tanımını içeren etki alanının adına işaretler.

## SecurityId

Tip: MQBYTE40 -giriş

Otorite.

Güvenlik tanıtcısı. Bu, yetkisini denetlenecek güvenlik tanımlayıcısıdır.

## CorrelationPtr

Tip: MQPTR-input

İlinti göstergesi. Bu, kimlik doğrulama kullanıcı işlevi ile diğer uygun OAM işlevleri arasında ilişki verilerin aktarılmasını kolaylaştırır.

## C bildirimi

```
typedef struct tagMQZED MQZED;
struct tagMQZED {
    MQCHAR4    StrucId;           /* Structure identifier */
    MQLONG     Version;          /* Structure version number */
    PMQCHAR    EntityNamePtr;    /* Address of entity name */
    PMQCHAR    EntityDomainPtr;  /* Address of entity domain name */
    MQBYTE40   SecurityId;       /* Security identifier */
    MQPTR      CorrelationPtr;   /* Address of correlation data */
}
```

## MQZEP-Bileşen giriş noktası ekle

Bir hizmet bileşeni, kullanıma hazırlama sırasında, o hizmet bileşenine ilişkin giriş noktası vektörüne bir giriş noktası eklemek için bu işlevi başlatır.

## Sözdizimi

MQZEP ( *Hconfig* , *Function* , *EntryPoint* , *CompCode* , *Reason* )

## Parametreler

### Hconfig

Tip: MQHCONFIG-input

Yapılandırma tanıtcısı. Bu tanıtcı, bu kurulabilir hizmet için yapılandırılmakta olan bileşeni gösterir. Bileşen başlatma çağrısındaki kuyruk yöneticisi tarafından bileşen yapılandırma işlevine geçirilen bileşenle aynı olmalıdır.

### İşlev

Tip: MQHOME-giriş

İşlev tanıtcısı. Bu, kurulabilir her hizmet için geçerli değerler tanımlanır.

MQZEP, aynı işlev için bir kereden fazla çağrılırsa, yapılan son çağrı, kullanılan giriş noktasını sağlar.

### EntryPoint

Tip: PMQFUNC-giriş

İşlev giriş noktası. Bu, işlevi gerçekleştirmek için bileşenin sağladığı giriş noktasının adresidir.

NULL değeri geçerli ve işlevin bu bileşen tarafından sağlanmamış olduğunu gösterir. MQZEP kullanılarak tanımlanmamış giriş noktaları için boş değer (NULL) varsayılır.

### CompCode

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Tamamlanma kodu. Bu değer aşağıdaki değerlerden biri olmalıdır:

## **MQCC\_OK**

İşlem başarıyla tamamlandı.

## **MQCC\_FAILED**

Arama başarısız oldu.

### **Neden**

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Neden kodu ön eleme *CompCode*.

*CompCode* , MQCC\_OK ise:

## **MQRC\_NONE**

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_FAILED ise:

## **MQRC\_FUNCTION\_HATASI**

(2281, X'8E9') İşlev tanıtıcısı geçerli değil.

## **MQRC\_HCONFIL\_ERROR**

(2280, X'8E8') Yapılandırma tutamacı geçerli değil.

Bu neden kodlarına ilişkin ek bilgi için [API tamamlama ve neden kodları](#) başlıklı konuya bakın.

## **C çağırma**

```
MQZEP (Hconfig, Function, EntryPoint, &CompCode, &Reason);
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
MQHCONFIG  Hconfig;      /* Configuration handle */
MQLONG     Function;    /* Function identifier */
PMQFUNC    EntryPoint;  /* Function entry point */
MQLONG     CompCode;    /* Completion code */
MQLONG     Reason;      /* Reason code qualifying CompCode */
```

## **MQZFP-Serbest değiştirgeler**

MQZFP yapısı, *FreeParms* parametresine ilişkin MQZ\_FREE\_USER çağrısında kullanılır. Bu parametre, serbest bırakılacak kaynakla ilgili verileri belirtir.

*Tablo 1.* Yapıdaki alanları özetler.

Çizelge 842. MQZFP içindeki alanlar	
Alan	Tanım
<u>StrucId</u>	Yapı tanıtıcısı
<u>S\u00f0fcr\u00f0fcm</u>	S\u00f0fcr\u00f0fcm
<u>Ayrıldı</u>	Ayrılmış alan
<u>CorrelationPtr</u>	İlinti göstergesi

## **Alanlar**

### **StrucId**

Tip: MQCHAR4 -giriş

Yapı tanıtıcısı. Değer aşağıdaki gibidir:

## MQZIC\_STRUC\_ID

Kimlik bağlamı yapısına ilişkin tanıtıcı. C programlama dili için, MQZIC\_STRUC\_ID\_ARRAY değişmezi de tanımlanır; bu, MQZIC\_STRUC\_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisidir.

## S\u00fcr\u00fcm

Tip: MQHOME-giriş

Yapı sürüm numarası. Değer aşağıdaki gibidir:

## MQZFP\_VERSION\_1

Version-1 boş parametreler yapısı.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

## MQZFP\_CURRENT\_VERSION

Serbest parametre yapısının geçerli sürümü.

## Ayrıldı

Tip: MQBYTE8 -giriş

Ayrılmış alan. Başlangıç değeri boş değerli.

## CorrelationPtr

Tip: MQPTR-input

İlinti göstergesi. Serbest bırakılacak kaynakla ilgili ilinti verilerinin adresi.

## C bildirimi

```
typedef struct tagMQZFP MQZFP;
struct tagMQZFP {
    MQCHAR4    StrucId;           /* Structure identifier */
    MQLONG     Version;          /* Structure version number */
    MQBYTE8    Reserved;        /* Reserved field */
    MQPTR      CorrelationPtr;   /* Address of correlation data */
};
```

## MQZIC-Kimlik bağlamı

MQZIC yapısı, *IdentityContext* parametresi için MQZ\_AUTHENTICATE\_USER çağrısında kullanılıyor.

MQZIC yapısı, iletiyi ilk olarak bir kuyruğa koyan uygulamanın kullanıcıyı tanıyan kimlik bağlamı bilgilerini içerir:

- Kuyruk yöneticisi, *UserIdentifier* alanını kullanıcıyı tanıyan bir adla doldurur; kuyruk yöneticisinin bunu yapabileceği biçimi, uygulamanın çalışmakta olduğu ortama bağlıdır.
- Kuyruk yöneticisi, *AccountingToken* alanını, iletiyi koyan uygulamadan saptadığı bir simgeyle ya da sayıyla doldurur.
- Uygulamalar, kullanıcı hakkında içermek istedikleri ek bilgiler (örneğin, şifrelenmiş bir parola) için *ApplIdentityVerileri* alanını kullanabilir.

Uygun şekilde yetkili uygulamalar, kimlik bağlamını MQZ\_AUTHENTICATE\_USER işlevini kullanarak ayarlayabilir.

IBM MQ for Windows altında bir ileti oluşturulduğunda, bir Windows sistem güvenlik tanıtıcısı (SID) *AccountingToken* alanında saklanır. SID, *UserIdentifier* (Kullanıcı Kimliği) alanını tamamlamak ve bir kullanıcının kimlik bilgilerini oluşturmak için kullanılabilir.

*Tablo 1.* Yapıdaki alanları özetler.

Çizelge 843. MQZIC 'deki alanlar	
Alan	Tanım
StrucId	Yapı tanıtıcısı

Çizelge 843. MQZIC ' deki alanlar (devamı var)

Alan	Tanım
<u>S\u00fcr\u00fcm</u>	S\u00fcr\u00fcm
<u>UserIdentifier</u>	Kullanıcı kimliği
<u>AccountingToken</u>	Hesap simgesi
<u>ApplIdentityVerileri</u>	Uygulama kimliği verileri

## Alanlar

### StrucId

Tip: MQCHAR4 -giriş

Yapı tanıtıcısı. Değer aşağıdaki gibidir:

#### MQZIC\_STRUC\_ID

Kimlik bağlamı yapısına ilişkin tanıtıcı. C programlama dili için, MQZIC\_STRUC\_ID\_ARRAY değişmezi de tanımlanır; bu, MQZIC\_STRUC\_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisidir.

### S\u00fcr\u00fcm

Tip: MQHOME-giriş

Yapı sürüm numarası. Değer aşağıdaki gibidir:

#### MQZIC\_VERSION\_1

Version-1 kimlik bağlamı yapısı.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

#### MQZIC\_CURRENT\_VERSION

Kimlik bağlamı yapısının yürürlükteki sürümü.

### UserIdentifier

Tip: MQCHAR12 -giriş

Kullanıcı kimliği. Bu, iletinin tanıtıcı bağlamının bir parçasıdır. *UserIdentifier* , iletiyi oluşturan uygulamanın kullanıcı tanımlayıcısını belirtir. Kuyruk yöneticisi bu bilgileri karakter verisi olarak değerlendirir, ancak biçiminin biçimini tanımlamaz. *UserIdentifier* alanıyla ilgili daha fazla bilgi için bkz. [“UserIdentifier \(MQCHAR12\)” sayfa 451.](#)

### AccountingToken

Tip: MQBYTE32 -giriş

Muhasebe simgesi. Bu, iletinin tanıtıcı bağlamının bir parçasıdır. *AccountingToken* , iletinin uygun şekilde ücretlendirilmesinin sonucu olarak bir uygulamanın çalışmasına olanak sağlar. Kuyruk yöneticisi bu bilgileri bir bit dizgisi olarak değerlendirir ve içeriğini denetmez. *AccountingToken* alanıyla ilgili daha fazla bilgi için bkz. [“AccountingToken \(MQBYTE32\)” sayfa 452.](#)

### ApplIdentityVerileri

Tip: MQCHAR32 -giriş

Kimlikle ilgili uygulama verileri. Bu, iletinin tanıtıcı bağlamının bir parçasıdır. *ApplIdentityVerileri* , iletinin kökeniyle ilgili ek bilgi sağlamak için kullanılacak uygulama takımı tarafından tanımlanan bilgilerdir. Örneğin, kimlik verilerinin güvenilir olup olmadığını göstermek için uygun kullanıcı yetkisiyle çalışan uygulamalar tarafından ayarlanabilir. *ApplIdentityVeri* alanıyla ilgili daha fazla bilgi için bkz. [“ApplIdentityVerileri \(MQCHAR32\)” sayfa 454.](#)

## C bildirim

```
typedef struct tagMQZED MQZED;
```

```
struct tagMQZED {
    MQCHAR4   StrucId;           /* Structure identifier */
    MQLONG    Version;          /* Structure version number */
    MQCHAR12  UserIdentifier;    /* User identifier */
    MQBYTE32  AccountingToken;  /* Accounting token */
    MQCHAR32  ApplIdentityData; /* Application data relating to identity */
};
```

## IBM i Installable services interface reference information on IBM i

Use this information to understand the reference information for the installable services for IBM i.

Her işlev için, işlev tanıtıcısı (MQZEP için) de içinde olmak üzere, bir tanımlama vardır.

*Parametreler* , ortaya çıkmaları gereken sırayla gösterilir. Hepsi hazır olmalı.

Her parametre adının ardından, veri tipi parantez içinde olur. Bunlar, [“Temel veri tipleri” sayfa 982](#) içinde açıklanan temel veri tipleridir.

Parametrelerin açıklamasından sonra, C dili çağrısına da verilir.

### İlgili başvurular

**IBM i** [IBM üzerindeki kurulabilir hizmetler ve bileşenler](#)

**ALW** [AIX, Linux, and Windows için kurulabilir hizmetler ve bileşenler](#)

[“Kurulabilir hizmetler arabirimi başvuru bilgileri” sayfa 1595](#)

Bu konu derlemi, kurulabilir hizmetlere ilişkin başvuru bilgileri sağlar.

## IBM i IBM üzerinde MQZEP (Bileşen giriş noktası ekle)

Bu işlev, başlatma sırasında bir hizmet bileşeni tarafından çağrılır ve bu hizmet bileşenine ilişkin giriş noktası vektörüne bir giriş noktası eklemek için bu işlev başlatılır.

### Sözdizimi

```
MQZEP (Hconfig, Function, EntryPoint, CompCode, Reason)
```

### Parametreler

MQZEP çağrısı aşağıdaki parametrelere sahiptir.

#### Hconfig (MQHCONFIG)-giriş

Yapılandırma tanıtıcısı.

Bu tanıtıcı, bu kurulabilir hizmet için yapılandırılmakta olan bileşeni temsil eder. Bileşen başlatma çağrısındaki kuyruk yöneticisi tarafından bileşen yapısını işlevine geçirilen bir işleyle aynı olmalıdır.

#### İşlev (MQUZE)-giriş

İşlev tanıtıcısı.

Bu, kurulabilir her hizmet için geçerli değerler tanımlanır. MQZEP aynı işlev için bir kereden fazla çağrılırsa, yapılan son çağrı, kullanılan giriş noktasını sağlar.

#### EntryPoint (PMQFUNC)-giriş

İşlev giriş noktası.

Bu, işlevi gerçekleştirmek için bileşenin sağladığı giriş noktasının adresidir. NULL değeri geçerli ve işlevin bu bileşen tarafından sağlanmamış olduğunu gösterir. BOŞ is assumed for entry points which are not defined using MQZEP.

#### CompCode (MQHOT)-çıkış

Tamamlanma kodu.

Bu, aşağıdakilerden biridir:

## **MQCC\_OK**

İşlem başarıyla tamamlandı.

## **MQCC\_FAILED**

Arama başarısız oldu.

## **Neden (MQUZE)-çıkış**

Neden kodu ön eleme *CompCode*.

*CompCode* MQCC\_OK ise:

## **MQRC\_NONE**

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_FAILED ise:

## **MQRC\_FUNCTION\_HATASI**

(2281, X'8E9') İşlev tanıtıcısı geçerli değil.

## **MQRC\_HCONFIL\_ERROR**

(2280, X'8E8') Yapılandırma tutamacı geçerli değil.

Bu neden kodlarına ilişkin ek bilgi için [İletiler ve neden kodları](#) başlıklı konuya bakın.

## **C çağırma**

```
MQZEP (Hconfig, Function, EntryPoint, &CompCode, &Reason);
```

Değiştirgeleri aşağıdaki gibi bildirin:

```
MQHCONFIG Hconfig; /* Configuration handle */
MQLONG Function; /* Function identifier */
PMQFUNC EntryPoint; /* Function entry point */
MQLONG CompCode; /* Completion code */
MQLONG Reason; /* Reason code qualifying CompCode */
```

## **IBM i IBM üzerinde MQHCONFIG (Yapılandırma tanıtıcısı)**

MQHCONFIG veri tipi, belirli bir kurulabilir hizmet için yapılandırılmakta olan bir yapılanış tanıtıcısını (yani, bileşen) gösterir. Bir yapılandırma tanıtıcısı, doğal sınırı üzerinde hizalanmalıdır.

Uygulamalar, yalnızca eşitlik için bu tipteki değişkenleri test etmelidir.

## **C bildirimi**

```
typedef void MQPOINTER MQHCONFIG;
```

## **IBM i PMQFUNC (Pointer to function) on IBM i**

Bir işleve işaretçi.

## **C bildirimi**

```
typedef void MQPOINTER PMQFUNC;
```

## IBM üzerinde MQZ\_AUTHENTICATE\_USER (Kullanıcı kimliği kullanıcı kimliği)

Bu işlev, bir MQZAS\_VERSION\_5 yetki hizmeti bileşeni tarafından sağlanır. Bir kullanıcının kimliğini doğrulamak ya da kimlik bağlamı alanlarını ayarlamak için kuyruk yöneticisi tarafından çağrılır.

Bu, bir IBM MQ kullanıcı uygulaması bağlamı oluşturulduğunda çağrılır. Bu durum, bağlantı çağruları sırasında, uygulamanın kullanıcı bağlamının ilk kullanıma hazırlandığı noktada ve her bir noktada uygulamanın kullanıcı bağlamının değiştirildiği durumlarda olur. Bir bağlantı çağrısı yapıldığında, uygulamanın kullanıcı bağlamı bilgileri *IdentityContext* alanında yeniden edinilir.

Bu işleve ilişkin işlev tanıtıcısı (MQZEP için) MQZID\_AUTHENTICATE\_USER ' tır.

### Sözdizimi

**MQZ\_AUTHENTICATE\_USER (QMgrName, SecurityParms, ApplicationContext, IdentityContext, CorrelationPtr, ComponentData, Continuation, CompCode, Reason)**

### Parametreler

MQZ\_AUTHENTICATE\_USER çağrısı aşağıdaki parametrelere sahiptir.

#### QMgrName (MQCHAR48)-giriş

Kuyruk yöneticisi adı.

Bileşeni çağırın kuyruk yöneticisinin adı. Bu ad, değiştirgenin tam uzunluğuna kadar boşluklarla dolduruldu; ad boş değerli bir karakterle sonlandırılmaz. Kuyruk yöneticisi adı, bileşene bilgi için iletilir; yetki hizmeti arabirimi, bileşenin bu bileşeni tanımlı bir şekilde kullanmasını gerektirmez.

#### SecurityParms (MQCSP)-giriş

Güvenlik parametreleri.

Kullanıcı kimliği, parola ve kimlik doğrulama türleriyle ilgili veriler.

Bir MQCONN MQI çağrısı sırasında bu değiştirge boş değerli (null) ya da varsayılan değerler içeriyor.

#### ApplicationContext (MQZAC)-giriş

Uygulama bağlamı.

Çağrılan uygulamayla ilgili veriler. Ayrıntılar için bkz. [“IBM üzerinde MQZAC \(Uygulama bağlamı\)” sayfa 1689](#). Tüm MQCONN ya da MQCONNX MQI çağrısı sırasında, MQZAC yapısındaki kullanıcı bağlamı bilgileri yeniden alınır.

#### IdentityContext (MQZIC)-giriş/çıkış

Kimlik bağlamı.

Kullanıcı kimliği doğrulamasına giriş sırasında, bu değer yürürlükteki kimlik bağlamını tanımlar. Kullanıcı kimliği doğrulama işlevi, kuyruk yöneticisinin yeni kimlik bağlamını benimsediği noktada bu değişikliği değiştirebilir. MQZIC yapısıyla ilgili ayrıntılar için [“IBM üzerinde MQZIC \(Kimlik bağlamı\)” sayfa 1696](#) konusuna bakın.

#### CorrelationPtr (MQPTR)-çıkış

İlinti göstergesi.

Herhangi bir ilinti verisinin adresini belirtir. Bu işaretçi daha sonra diğer OAM çağrılarında iletilir.

#### ComponentData (MQBYTE x ComponentDataLength)-giriş/çıkış

Bileşen verileri.

Bu veriler, bu bileşen adına kuyruk yöneticisi tarafından tutulur; bu bileşen tarafından sağlanan işlevlerden herhangi biri tarafından yapılan değişiklikler korunur ve bu bileşen işlevlerinden biri bir sonraki defa çağrılır. Bu veri alanının uzunluğu, MQZ\_INIT\_AUTHORITY çağrısının **ComponentDataLength** parametresindeki kuyruk yöneticisi tarafından geçirilir.



### Devam (MQLONG)-çıkış

Devam işareti.

Aşağıdaki değerler belirlenebilir:

#### MQZCI\_VARSAYILAN

Diğer bileşenlere bağlı olarak devam edin.

#### MQZCI\_STOP

Sonraki bileşenle devam ettirmeyin.

### CompCode (MQHOT)-çıkış

Tamamlanma kodu.

Bu, aşağıdakilerden biridir:

#### MQCC\_OK

İşlem başarıyla tamamlandı.

#### MQCC\_FAILED

Arama başarısız oldu.

### Neden (MQUZE)-çıkış

Neden kodu ön eleme *CompCode*.

*CompCode* MQCC\_OK ise:

#### MQRC\_NONE

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_FAILED ise:

#### MQRC\_SERVICE\_ERROR

(2289, X'8F1') Hizmet erişimi beklenmeyen bir hata oluştu.

Bu neden kodlarına ilişkin ek bilgi için [İletiler ve neden kodları](#) başlıklı konuya bakın.

## C çağırma

```
MQZ_AUTHENTICATE_USER (QMgrName, SecurityParms, ApplicationContext,  
IdentityContext, &CorrelationPtr, ComponentData,  
&Continuation, &CompCode, &Reason);
```

Hizmete geçirilen parametreler aşağıdaki gibi bildirilir:

```
MQCHAR48  QMgrName;          /* Queue manager name */  
MQCSP     SecurityParms;  /* Security parameters */  
MQZAC     ApplicationContext; /* Application context */  
MQZIC     IdentityContext; /* Identity context */  
MQPTR     CorrelationPtr;  /* Correlation pointer */  
MQBYTE    ComponentData[n]; /* Component data */  
MQLONG    Continuation;    /* Continuation indicator set by  
                           component */  
MQLONG    CompCode;        /* Completion code */  
MQLONG    Reason;          /* Reason code qualifying CompCode */
```

IBM i

## IBM üzerinde MQZ\_CHECK\_AUTHORITY (Denetim yetkisi)

Bu işlev bir MQZAS\_VERSION\_1 yetki hizmeti bileşeni tarafından sağlanır ve kuyruk yöneticisi tarafından, bir varlığın belirli bir nesne üzerinde belirli bir eylemi ya da eylemleri gerçekleştirme yetkisinin olup olmadığını kontrol etmek için çağrılır.

Bu işleve ilişkin işlev tanıtıcısı (MQZEP için) MQZID\_CHECK\_AUTHORITY olur.

## Sözdizimi

**MQZ\_CHECK\_AUTHORITY** (*QMgrName, EntityName, EntityType, ObjectName, ObjectType, Authority, ComponentData, Continuation, CompCode, Reason*)

## Parametreler

MQZ\_CHECK\_AUTHORITY çağrısının değıştirgeleri şunlardır:

### **QMgrName (MQCHAR48)-giriş**

Kuyruk yöneticisi adı.

Bileşeni çağırın kuyruk yöneticisinin adı. Bu ad, değıştirgenin tam uzunluğuna kadar boşluklarla dolduruldu; ad boş değeri bir karakterle sonlandırılmaz. Kuyruk yöneticisi adı, bileşene bilgi için iletilir; yetki hizmeti ararımı, bileşenin herhangi bir şekilde tanımlanmış bir şekilde kullanılmasını gerektirmez.

### **EntityName (MQCHAR12)-giriş**

Varlık adı.

Nesneye ilişkin yetkisi denetlenmek üzere olan varlığın adı. Dizilimin uzunluk üst sınırı 12 karakterdir; boşlukla doldurursa boşluk karakterinden daha kısadır. Ad boş değeri bir karakter tarafından sonlandırılmadı.

Bu varlığın temel güvenlik hizmetiyle tanınması şart değil. Bu bilgi bilinmiyorsa, denetim için özel **kimse** grubunun (tüm varlıkların ait olduğu varsayıldığı) yetkilendirmeleri kullanılır. Tüm boş ad geçerli ve bu şekilde kullanılabilir.

### **EntityType (MQUBE)-giriş**

Varlık tipi.

*EntityName* tarafından belirtilen varlık tipi. Bu, aşağıdakilerden biridir:

#### **MQZAET\_PRINCIPAL**

Müdür.

#### **MQZAET\_GRUP**

Grup.

### **ObjectName (MQCHAR48)-giriş**

Nesne adı.

Erişimin gerekli olduğu nesnenin adı. Dizilimin uzunluk üst sınırı 48 karakterdir; boşlukla sağa yaslanmış ise, bu karakter uzunluğuna göre daha kısadır. Ad boş değeri bir karakter tarafından sonlandırılmadı.

*ObjectType* MQOT\_Q\_MGR ise, bu ad *QMgrName* ile aynıdır.

### **ObjectType (MQUZE)-giriş**

Nesne tipi.

*ObjectName* tarafından belirtilen varlık tipi. Bu, aşağıdakilerden biridir:

#### **MQOT\_AUTH\_INFO**

Kimlik doğrulama bilgileri.

#### **MQOT\_KANAL**

Kanal.

#### **MQOT\_CLNTCONN\_CHANNEL**

İstemci bağlantı kanalı.

#### **MQOT\_LISTENER**

Dinleyici.

#### **MQOT\_NAMELIST**

İsim listesi.

#### **MQOT\_PROCESS**

Süreç tanımlaması.

**MQOT\_Q**

Kuyruk.

**MQOT\_Q\_MGR**

Kuyruk yöneticisi.

**MQOT\_SERVICE**

Servis.

**Yetki (MQUBE)-giriş**

Denetlenecek yetki.

Bir yetki imleniyorsa, bu alan uygun yetki işlemine eşittir (MQZAO\_ \* sabiti). Birden fazla yetki denetleniyorsa, bu değer, ilgili MQZAO\_ \* değişmezlerinin bit ya da ya da alt öğelerinden biri olur.

MQI çağrılarının kullanılması için aşağıdaki yetkiler geçerlidir:

**MQZAO\_CONNECT**

MQCONN çağrısını kullanma yeteneği.

**MQZAO\_GÖZAT**

Bir göz atma seçeneği ile MQGET çağrısını kullanma yeteneği.

Bu, MQGET çağrısında MQGMO\_BROWSE\_FIRST, MQGMO\_BROWSE\_MSG\_UNDER\_CURSOR ya da MQGMO\_BROWSE\_NEXT seçeneğinin belirtilmesine olanak sağlar.

**MQZAO\_INPUT**

MQGET çağrısını bir giriş seçeneği ile kullanma yeteneği.

Bu, MQOO\_INPUT\_SHARED, MQOO\_INPUT\_EXCLUSIVE YA DA MQOPEN çağrısında MQOO\_INPUT\_AS\_Q\_DEF seçeneğinin belirtilmesini sağlar.

**MQZAO\_OUTPUT**

MQPUT çağrısını kullanma yeteneği.

Bu, MQOPEN çağrısında MQOO\_OUTPUT seçeneğinin belirtilmesini sağlar.

**MQZAO\_SORGULAMA**

MQINQ çağrısını kullanma yeteneği.

Bu, MQOPEN çağrısında MQOO\_ENSORE seçeneğinin belirtilmesini sağlar.

**MQZAO\_SET**

MQSET çağrısını kullanma yeteneği.

Bu, MQOO\_SET seçeneğinin MQOPEN çağrısında belirtilmesini sağlar.

**MQZAO\_PASS\_IDENTITY\_CONTEXT**

Kimlik bağlamını geçirebilme yeteneği.

Bu, MQPUT çağrısında MQOO\_PASS\_IDENTITY\_CONTEXT seçeneğinin belirtilmesini ve MQPUT ve MQPUT1 çağrılarında MQPMO\_PASS\_IDENTITY\_CONTEXT seçeneğinin belirtilmesini sağlar.

**MQZAO\_PASS\_ALL\_CONTEXT**

Tüm bağlamı geçirebilme yeteneği.

Bu, MQPUT çağrısında MQOO\_PASS\_ALL\_CONTEXT seçeneğinin belirtilmesine ve MQPUT ve MQPUT1 çağrılarında belirtilecek MQPMO\_PASS\_ALL\_CONTEXT seçeneğinin geçerli olduğunu sağlar.

**MQZAO\_SET\_IDENTITY\_CONTEXT**

Kimlik bağlamını ayarlama yeteneği.

Bu, MQPUT çağrısında MQOO\_SET\_IDENTITY\_CONTEXT seçeneğinin belirtilmesini ve MQPUT ve MQPUT1 çağrılarında MQPMO\_SET\_IDENTITY\_CONTEXT seçeneğinin belirtilmesini sağlar.

**MQZAO\_SET\_ALL\_CONTEXT**

Tüm bağlamı ayarlayabilme yeteneği.

Bu, MQPUT çağrısında MQOO\_SET\_ALL\_CONTEXT seçeneğinin belirtilmesine ve MQPUT ve MQPUT1 çağrılarında belirtilecek MQPMO\_SET\_ALL\_CONTEXT seçeneğinin belirlenmesine olanak sağlar.

#### **MQZAO\_ALTERNATE\_USER\_AUTHORITY**

Diğer kullanıcı yetkisini kullanma yeteneği.

Bu, MQOPEN çağrısında MQOO\_ALTERNATE\_USER\_AUTHORITY seçeneğinin belirtilmesine ve MQPUT1 çağrısında MQPMO\_ALTERNATE\_USER\_AUTHORITY seçeneğinin belirtilmesine olanak tanır.

#### **MQZAO\_ALL\_MQI**

MQI yetkilerinin tümü.

Bu, daha önce açıklanan yetkilerin tümünü etkinleştirir.

Kuyruk yöneticisinin denetimi için aşağıdaki yetkiler geçerlidir:

#### **MQZAO\_CREATE**

Belirli bir tipte nesnelere oluşturma yeteneği.

#### **MQZAO\_DELETE**

Belirli bir nesneyi silme yeteneği.

#### **MQZAO\_GÖRÜNTÜLE**

Belirli bir nesnenin özniteliklerini görüntüleme yeteneği.

#### **MQZAO\_CHANGE**

Belirlenen bir nesnenin özniteliklerini değiştirme yeteneği.

#### **MQZAO\_CLEAR**

Belirtilen bir kuyruktan tüm iletileri silme yeteneği.

#### **MQZAO\_YETKİLİVER**

Belirli bir nesne için diğer kullanıcılara yetki verme yeteneği.

#### **MQZAO\_CONTROL**

İstemci olmayan bir kanal nesnesini başlatma, durdurma ya da ping etme yeteneği.

#### **MQZAO\_CONTROL\_EXTENDED**

Bir sıra numarasını sıfırlamak ya da istemci dışındaki bir kanal nesnesindeki belirsiz bir iletiyi çözebilme yeteneği.

#### **MQZAO\_ALL\_ADMIN**

MQZAO\_CREATE dışında tüm denetim yetkileri.

Aşağıdaki yetkiler, MQI ' nin her iki kullanımı ve bir kuyruk yöneticisinin denetlenmesine uygulanır:

#### **MQZAO\_ALL**

Tüm yetkiler, MQZAO\_CREATE dışındaki tüm yetkiler.

#### **MQZAO\_NONE**

Yetki yok.

#### **ComponentData (MQBYTE x ComponentDataLength)-giriş/çıkış**

Bileşen verileri.

Bu veriler, bu bileşen adına kuyruk yöneticisi tarafından tutulur; bu bileşen tarafından sağlanan işlemlerden herhangi biri tarafından yapılan değişiklikler korunur ve bu bileşenin sonraki işlemlerinden biri çağrıldığında sunulur.

Bu veri alanının uzunluğu, MQZ\_INIT\_AUTHORITY çağrısının **ComponentDataLength** parametresindeki kuyruk yöneticisi tarafından geçirilir.

#### **Devam (MQLONG)-çıkış**

Bileşene göre devam göstergesi kümesi.

Aşağıdaki değerler belirlenebilir:

#### **MQZCI\_VARSAYILAN**

Devam eden kuyruk yöneticisine bağımlı.

MQZ\_CHECK\_AUTHORITY için bu, MQZCI\_STOP ile aynı etkiye sahiptir.

#### **MQZCI\_CONTINY**

Sonraki bileşenle devam edin.

#### **MQZCI\_STOP**

Sonraki bileşenle devam ettirmeyin.

#### **CompCode (MQHOT)-çıkış**

Tamamlanma kodu.

Bu, aşağıdakilerden biridir:

#### **MQCC\_OK**

İşlem başarıyla tamamlandı.

#### **MQCC\_FAILED**

Arama başarısız oldu.

#### **Neden (MQUZE)-çıkış**

Neden kodu ön eleme *CompCode*.

*CompCode* MQCC\_OK ise:

#### **MQRC\_NONE**

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_FAILED ise:

#### **MQRC\_NOT\_YETKILI**

(2035, X'7F3') Erişim için yetkili değil.

#### **MQRC\_SERVICE\_ERROR**

(2289, X'8F1') Hizmet erişimi beklenmeyen bir hata oluştu.

#### **MQRC\_SERVICE\_NOT\_VAR**

(2285, X'8ED') Undering hizmeti kullanılamıyor.

Bu neden kodlarına ilişkin ek bilgi için [İletiler ve neden kodları](#) başlıklı konuya bakın.

## **C çağırma**

```
MQZ_CHECK_AUTHORITY (QMgrName, EntityName, EntityType, ObjectName,  
Objectype, Authority, ComponentData,  
&Continuation, &CompCode, &Reason);
```

Hizmete geçirilen parametreler aşağıdaki gibi bildirilir:

```
MQCHAR48 QMgrName;          /* Queue manager name */  
MQCHAR12 EntityName;       /* Entity name */  
MQLONG EntityType;         /* Entity type */  
MQCHAR48 ObjectName;       /* Object name */  
MQLONG Objectype;         /* Object type */  
MQLONG Authority;         /* Authority to be checked */  
MQBYTE ComponentData[n];  /* Component data */  
MQLONG Continuation;      /* Continuation indicator set by  
component */  
MQLONG CompCode;          /* Completion code */  
MQLONG Reason;           /* Reason code qualifying CompCode */
```

## **MQZ\_CHECK\_PRIVATIONAL-Kullanıcının ayrıcalıklı olup olmadığını denetle**

Bu işlev bir MQZAS\_VERSION\_6 yetkilendirme hizmeti bileşeni tarafından sağlanır ve belirtilen kullanıcının ayrıcalıklı bir kullanıcı olup olmadığını belirlemek için kuyruk yöneticisi tarafından çağrılır.

Bu işleve ilişkin işlev tanıtıcısı (MQZEP için) MQZID\_CHECK\_PRIVACITEDIR.

## Sözdizimi

MQZ\_CHECK\_PRIVILEGED( *QMgrName* , *EntityData* , *EntityType* , *ComponentData* , *Continuation* , *CompCode* , *Reason* )

## Parametreler

### QMgrName

Tip: MQCHAR48 -giriş

Kuyruk yöneticisi adı. Bileşeni çağıran kuyruk yöneticisinin adı. Bu ad, değiştirgenin tam uzunluğuna kadar boşluklarla dolduruldu; ad boş değerli bir karakterle sonlandırılmaz.

Kuyruk yöneticisi adı, bileşene bilgi için iletilir; yetki hizmeti arabirimi, bileşenin bu bileşeni tanımlı bir şekilde kullanmasını gerektirmez.

### EntityData

Tip: MQZED-giriş

Varlık verileri. Denetlenecek olan varlıkla ilgili veriler. Daha fazla bilgi için bkz. “MQZED-Varlık tanımlayıcısı” sayfa 1653.

### EntityType

Tip: MQHOME-girişi

Varlık tipi. EntityDatatarafından belirtilen varlık tipi. Bu değer aşağıdaki değerlerden biri olmalıdır:

#### **MQZAET\_PRINCIPAL**

Müdür.

#### **MQZAET\_GRP**

Grup.

### ComponentData

Tip: MQBYTEExComponentDataLength -giriş/çıkış

Bileşen verileri. Bu veriler, bu bileşen adına kuyruk yöneticisi tarafından tutulur; bu bileşen tarafından sağlanan işlemlerden herhangi biri tarafından yapılan değişiklikler korunur ve bu bileşen işlemlerinden biri bir sonraki defa çağrılır.

Bu veri alanının uzunluğu, MQZ\_INIT\_AUTHORITY çağrısının **ComponentDataLength** parametresindeki kuyruk yöneticisi tarafından geçirilir.

### Devam

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Bileşene göre devam göstergesi kümesi. Aşağıdaki değerler belirlenebilir:

#### **MQZCI\_VARSAYILAN**

Devam eden kuyruk yöneticisine bağımlı.

MQZ\_CHECK\_AUTHORITY için bu, MQZCI\_STOP ile aynı etkiye sahiptir.

#### **MQZCI\_CONTINY**

Sonraki bileşenle devam edin.

#### **MQZCI\_STOP**

Sonraki bileşenle devam ettirmeyin.

Bir bileşene çağrı başarısız olursa (yani, *CompCode* , MQCC\_FAILED değerini döndürür) ve *Continuation* değiştirgesi MQZCI\_default ya da MQZCI\_CONTINUE ise, kuyruk yöneticisi varsa diğer bileşenleri çağdırmaya devam eder.

Çağrı başarılı olursa (yani, *CompCode* , MQCC\_OK değerini döndürür), *Devam* ayarı ne olursa olsun, başka bileşen çağdırılmaz.

Çağrı başarısız olursa ve *Continuation* parametresi MQZCI\_STOP ise, başka bileşen çağdırılmaz ve hata kuyruk yöneticisine döndürülür. Components have no knowledge of previous calls, so the *Devam* parameter is always set to MQZCI\_DEFAULT before the call.

## CompCode

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Tamamlanma kodu. Bu deęer ařaęıdaki deęerlerden biri olmalıdır:

### **MQCC\_OK**

İřlem bařarıyla tamamlandı.

### **MQCC\_FAILED**

Arama bařarisız oldu.

## Neden

Tip: MQXX\_ENCODE\_CASE\_ONE long-output

Neden kodu ön eleme *CompCode*.

*CompCode* , MQCC\_OK ise:

### **MQRC\_NONE**

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_FAILED ise:

### **MQRC\_NOT\_AYRICALIKSIZ**

(2584, X'A18') Bu kullanıcı ayrıcalıklı bir kullanıcı kimlięi deęil.

### **MQRC\_UNKNOWN\_ENTITY**

(2292, X'8F4') Hizmet için bilinmeyen varlık.

### **MQRC\_SERVICE\_ERROR**

(2289, X'8F1') Hizmet eriřimi beklenmeyen bir hata oluřtu.

### **MQRC\_SERVICE\_NOT\_VAR**

(2285, X'8ED') Undering hizmeti kullanılamıyor.

Bu neden kodlarına iliřkin ek bilgi için [API tamamlama ve neden kodlarıbařlıklı konuya](#) bakın.

## C çağırma

```
MQZ_CHECK_PRIVILEGED (QMgrName, &EntityData, EntityType,  
ComponentData, &Continuation,  
&CompCode, &Reason);
```

Hizmete geçirilen parametreler ařaęıdaki gibi bildirilir:

```
MQCHAR48  QMgrName;          /* Queue manager name */  
MQZED     EntityData;       /* Entity name */  
MQLONG    EntityType;      /* Entity type */  
MQBYTE    ComponentData[n]; /* Component data */  
MQLONG    Continuation;    /* Continuation indicator set by  
                           component */  
MQLONG    CompCode;        /* Completion code */  
MQLONG    Reason;         /* Reason code qualifying CompCode */
```

**IBM i**

## **IBM üzerindeki MQZ\_COPY\_ALL\_AUTHORITY (Tüm yetkiyi kopyala)**

Bu iřlev, bir yetkilendirme hizmeti bileřeni tarafından saęlanır. Bir bařvuru nesnesi için yürürlükte olan yetkilerin tümünü bařka bir nesneye kopyalamak için kuyruk yöneticisi tarafından çağrılır.

Bu iřleve iliřkin iřlev tanıtıcısı (MQZEP için) MQZID\_COPY\_ALL\_AUTHORITY olur.

## Sözdizimi

**MQZ\_COPY\_ALL\_AUTHORITY (QMgrName, RefObjectName, ObjectName, ObjectName, ComponentData, Continuation, CompCode, Reason)**

## Parametreler

MQZ\_COPY\_ALL\_AUTHORITY çağrısına aşağıdaki parametreler yer aldı.

### QMgrName (MQCHAR48)-giriş

Kuyruk yöneticisi adı.

Bileşeni çağıran kuyruk yöneticisinin adı. Bu ad, değiştirgenin tam uzunluğuna kadar boşluklarla dolduruldu; ad boş değerli bir karakterle sonlandırılmaz.

Kuyruk yöneticisi adı, bileşene bilgi için iletilir; yetki hizmeti arabirimi, bileşenin herhangi bir şekilde tanımlanmış bir şekilde kullanılmasını gerektirmez.

### RefObjectAd (MQCHAR48)-giriş

Başvuru nesnesi adı.

Başvuru nesnesinin adı, kopyalanacak yetkiler. Dizilimin uzunluk üst sınırı 48 karakterdir; boşlukla sağa yaslanmış ise, bu karakter uzunluğuna göre daha kısadır. Ad boş değerli bir karakter tarafından sonlandırılmadı.

### ObjectName (MQCHAR48)-giriş

Nesne adı.

Erişimlerin ayarlanacağı nesnenin adı. Dizilimin uzunluk üst sınırı 48 karakterdir; boşlukla sağa yaslanmış ise, bu karakter uzunluğuna göre daha kısadır. Ad boş değerli bir karakter tarafından sonlandırılmadı.

### ObjectType (MQUZE)-giriş

Nesne tipi.

*RefObjectName* ve *ObjectName* tarafından belirtilen nesne tipi. Bu, aşağıdakilerden biridir:

#### MQOT\_AUTH\_INFO

Kimlik doğrulama bilgileri.

#### MQOT\_KANAL

Kanal.

#### MQOT\_CLNTCONN\_CHANNEL

İstemci bağlantı kanalı.

#### MQOT\_LISTENER

Dinleyici.

#### MQOT\_NAMELIST

İsim listesi.

#### MQOT\_PROCESS

Süreç tanımlaması.

#### MQOT\_Q

Kuyruk.

#### MQOT\_Q\_MGR

Kuyruk yöneticisi.

#### MQOT\_SERVICE

Servis.

### ComponentData (MQBYTE x ComponentDataLength)-giriş/çıkış

Bileşen verileri.

Bu veriler, bu bileşen adına kuyruk yöneticisi tarafından tutulur; bu bileşen tarafından sağlanan işlevlerden herhangi biri tarafından yapılan değişiklikler korunur ve bu bileşenin sonraki işlevlerinden biri çağrıldığında sunulur.

Bu veri alanının uzunluğu, MQZ\_INIT\_AUTHORITY çağrısının **ComponentDataLength** parametresindeki kuyruk yöneticisi tarafından geçirilir.



### Devam (MQLONG)-çıkış

Bileşene göre devam göstergesi kümesi.

Aşağıdaki değerler belirlenebilir:

#### MQZCI\_VARSAYILAN

Devam eden kuyruk yöneticisine bağımlı.

MQZ\_COPY\_ALL\_AUTHORITY için bu, MQZCI\_STOP ile aynı etkiye sahiptir.

#### MQZCI\_CONTINY

Sonraki bileşenle devam edin.

#### MQZCI\_STOP

Sonraki bileşenle devam ettirmeyin.

### CompCode (MQHOT)-çıkış

Tamamlanma kodu.

Bu, aşağıdakilerden biridir:

#### MQCC\_OK

İşlem başarıyla tamamlandı.

#### MQCC\_FAILED

Arama başarısız oldu.

### Neden (MQUZE)-çıkış

Neden kodu ön eleme *CompCode*.

*CompCode* MQCC\_OK ise:

#### MQRC\_NONE

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_FAILED ise:

#### MQRC\_SERVICE\_ERROR

(2289, X'8F1') Hizmet erişimi beklenmeyen bir hata oluştu.

#### MQRC\_SERVICE\_NOT\_VAR

(2285, X'8ED') Undering hizmeti kullanılamıyor.

#### MQRC\_UNKNOWN\_REF\_OBJECT

(2294, X'8F6') Başvuru nesnesi bilinmiyor.

Bu neden kodlarına ilişkin ek bilgi için [İletiler ve neden kodları](#) başlıklı konuya bakın.

## C çağırma

```
MQZ_COPY_ALL_AUTHORITY (QMgrName, RefObjectName, ObjectName, ObjectType,  
ComponentData, &Continuation, &CompCode,  
&Reason);
```

Hizmete geçirilen parametreler aşağıdaki gibi bildirilir:

```
MQCHAR48 QMgrName;          /* Queue manager name */  
MQCHAR48 RefObjectName;     /* Reference object name */  
MQCHAR48 ObjectName;       /* Object name */  
MQLONG   ObjectType;       /* Object type */  
MQBYTE   ComponentData[n]; /* Component data */  
MQLONG   Continuation;     /* Continuation indicator set by  
                           component */  
MQLONG   CompCode;        /* Completion code */  
MQLONG   Reason;         /* Reason code qualifying CompCode */
```

## IBM üzerinde MQZ\_DELETE\_AUTHORITY (Silme yetkisi)

Bu işlev, bir yetki hizmeti bileşeni tarafından sağlanır ve belirtilen nesneyle ilişkili tüm yetkileri silmek için kuyruk yöneticisi tarafından çağrılır.

Bu işleve ilişkin işlev tanıtıcısı (MQZEP için) MQZID\_DELETE\_AUTHORITY 'dir.

### Sözdizimi

**MQZ\_DELETE\_AUTHORITY** (*QMgrName, ObjectName, ObjectType, ComponentData, Continuation, CompCode, Reason*)

### Parametreler

MQZ\_DELETE\_AUTHORITY çağrısının değiştirgeleri şunlardır:

#### **QMgrName (MQCHAR48)-giriş**

Kuyruk yöneticisi adı.

Bileşeni çağıran kuyruk yöneticisinin adı. Bu ad, değiştirgenin tam uzunluğuna kadar boşluklarla dolduruldu; ad boş değerli bir karakterle sonlandırılmaz.

Kuyruk yöneticisi adı, bileşene bilgi için iletilir; yetki hizmeti arabirimi, bileşenin herhangi bir şekilde tanımlanmış bir şekilde kullanılmasını gerektirmez.

#### **ObjectName (MQCHAR48)-giriş**

Nesne adı.

Erişimlerin silineceği nesnenin adı. Dizilimin uzunluk üst sınırı 48 karakterdir; boşlukla sağa yaslanmış ise, bu karakter uzunluğuna göre daha kısadır. Ad boş değerli bir karakter tarafından sonlandırılmaz.

*ObjectType* MQOT\_Q\_MGR ise, bu ad *QMgrName* ile aynıdır.

#### **ObjectType (MQUZE)-giriş**

Nesne tipi.

*ObjectName* tarafından belirtilen varlık tipi. Bu, aşağıdakilerden biridir:

#### **MQOT\_AUTH\_INFO**

Kimlik doğrulama bilgileri.

#### **MQOT\_KANAL**

Kanal.

#### **MQOT\_CLNTCONN\_CHANNEL**

İstemci bağlantı kanalı.

#### **MQOT\_LISTENER**

Dinleyici.

#### **MQOT\_NAMELIST**

İsim listesi.

#### **MQOT\_PROCESS**

Süreç tanımlaması.

#### **MQOT\_Q**

Kuyruk.

#### **MQOT\_Q\_MGR**

Kuyruk yöneticisi.

#### **MQOT\_SERVICE**

Servis.

#### **ComponentData (MQBYTE x ComponentDataLength)-giriş/çıkış**

Bileşen verileri.

Bu veriler, bu bileşen adına kuyruk yöneticisi tarafından tutulur; bu bileşen tarafından sağlanan işlevlerden herhangi biri tarafından yapılan değişiklikler korunur ve bu bileşenin sonraki işlevlerinden biri çağrıldığında sunulur.

Bu veri alanının uzunluğu, MQZ\_INIT\_AUTHORITY çağrısının **ComponentDataLength** parametresindeki kuyruk yöneticisi tarafından geçirilir.

### Devam (MQLONG)-çıkış

Bileşene göre devam göstergesi kümesi.

Aşağıdaki değerler belirlenebilir:

#### MQZCI\_VARSAYILAN

Devam eden kuyruk yöneticisine bağımlı.

MQZ\_DELETE\_AUTHORITY için bu, MQZCI\_STOP ile aynı etkiye sahiptir.

#### MQZCI\_CONTINY

Sonraki bileşenle devam edin.

#### MQZCI\_STOP

Sonraki bileşenle devam ettirmeyin.

### CompCode (MQHOT)-çıkış

Tamamlanma kodu.

Bu, aşağıdakilerden biridir:

#### MQCC\_OK

İşlem başarıyla tamamlandı.

#### MQCC\_FAILED

Arama başarısız oldu.

### Neden (MQUZE)-çıkış

Neden kodu ön eleme *CompCode*.

*CompCode* MQCC\_OK ise:

#### MQRC\_NONE

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_FAILED ise:

#### MQRC\_SERVICE\_ERROR

(2289, X'8F1') Hizmet erişimi beklenmeyen bir hata oluştu.

#### MQRC\_SERVICE\_NOT\_VAR

(2285, X'8ED') Undering hizmeti kullanılamıyor.

Bu neden kodlarına ilişkin ek bilgi için [İletiler ve neden kodları](#) başlıklı konuya bakın.

## C çağırma

```
MQZ_DELETE_AUTHORITY (QMgrName, ObjectName, ObjectType, ComponentData,  
&Continuation, &CompCode, &Reason);
```

Hizmete geçirilen parametreler aşağıdaki gibi bildirilir:

```
MQCHAR48 QMgrName;          /* Queue manager name */  
MQCHAR48 ObjectName;       /* Object name */  
MQLONG   ObjectType;       /* Object type */  
MQBYTE   ComponentData[n]; /* Component data */  
MQLONG   Continuation;     /* Continuation indicator set by  
                           component */  
MQLONG   CompCode;         /* Completion code */  
MQLONG   Reason;          /* Reason code qualifying CompCode */
```

## IBM üzerinde MQZ\_ENUMERATE\_AUTHORITY\_DATA (Yetki verileri sıralı değer listesi)

Bu işlev bir MQZAS\_VERSION\_4 yetki hizmeti bileşeni tarafından sağlanır ve kuyruk yöneticisi tarafından, ilk çağrıda belirtilen seçim ölçütleriyle eşleşen tüm yetki verilerini almak için sürekli olarak çağrılır.

Bu işleve ilişkin işlev tanıtıcısı (MQZEP için) MQZID\_ENUMERATE\_AUTHORITY\_DATA olur.

### Sözdizimi

**MQZ\_ENUMERATE\_AUTHORITY\_DATA** (*QMgrName, StartEnumeration, Filter, AuthorityBufferLength, AuthorityBuffer, AuthorityDataLength, ComponentData, Continuation, CompCode, Reason*)

### Parametreler

MQZ\_ENUMERATE\_AUTHORITY\_DATA çağrısı aşağıdaki parametrelere sahiptir.

#### QMgrName (MQCHAR48)-giriş

Kuyruk yöneticisi adı.

Bileşeni çağıran kuyruk yöneticisinin adı. Bu ad, değiştirgenin tam uzunluğuna kadar boşluklarla dolduruldu; ad boş değerli bir karakterle sonlandırılmaz.

Kuyruk yöneticisi adı, bileşene bilgi için iletilir; yetki hizmeti arabirimi, bileşenin herhangi bir şekilde tanımlanmış bir şekilde kullanılmasını gerektirmez.

#### StartEnumeration (MQUZE)-giriş

Çağrıların sıralı değer listesi başlatılıp başlatılmayacağını belirten işaret.

Bu durum, çağrıların yetki verilerinin sıralamasını başlatması gerekip gerekmediğini belirtir ya da önceki bir MQZ\_ENUMERATE\_AUTHORITY\_DATA çağrısıyla başlatılan yetki verilerinin sıralamasını devam ettirir. Değer aşağıdakilerden biridir:

##### MQZSE\_START

Sıralamayı başlat.

Çağrı, yetki verisinin sıralamasını başlatmak için bu değerle çağrılır. **Filter** parametresi, bu ve art arda gelen çağrıların döndürdüğü yetki verilerini seçmek için kullanılacak seçim ölçütlerini belirtir.

##### MQZSE\_DEVAM

Sıralamaya devam et.

Çağrı, yetki verisinin sıralamasını sürdürmek için bu değerle çağrılır. **Filter** parametresi bu durumda yoksayılr ve boş (null) gösterge olarak belirtilebilir (seçim ölçütleri, *StartEnumeration* tarafından MQZSE\_START değerine ayarlanmış, çağrıyla belirtilen **Filter** parametresi tarafından belirlenir).

#### Süzgeç (MQZAD)-giriş

Süzgeç uygula.

*StartEnumeration* MQZSE\_START ise, *Filter* döndürülebilmek için yetki verilerini seçmek için kullanılacak seçim ölçütlerini belirtir. *Filter* boş değerli göstergedir, seçim ölçütü kullanılmazsa, tüm yetki verileri döndürülür. Kullanılabilecek seçim ölçütlerine ilişkin ayrıntılar için bkz. [“IBM üzerinde MQZAD \(Yetki verileri\)” sayfa 1691](#).

*StartEnumeration* MQZSE\_CONTINUDE ise, *Filter* yoksayılr ve boş değerli gösterge olarak belirtilebilir.

#### AuthorityBufferUzunluğu (MQUZE)-giriş

*AuthorityBuffer* uzunluğu.

Bu, **AuthorityBuffer** parametresindeki bayt cinsinden uzunluğudur. Yetki arabelleği, döndürülebilmek için gereken verileri sığdırmak için yeterince büyük olmalıdır.

### **AuthorityBuffer (MQZAD)-çıkış**

Yetki verileri.

Bu, yetki verilerinin döndürülediği arabelleğidir. Arabellek, bir MQZAD yapısını, MQZED yapısını ve tanımlı en uzun varlık adını ve en uzun etki alanı adını barındıracak kadar büyük olmalıdır.

**Not:** Bu değiştirge, arabellek başlangıcındaki MQZAD ' in her zaman olduğu için, MQZAD olarak tanımlıdır. Ancak, arabellek bir MQZAD olarak bildirildiyse, arabellek çok küçük olur; MQZAD, MQZED, artı varlık ve etki alanı adlarını barındırabilmesi için, bir MQZAD ' dan büyük olması gerekir.

### **AuthorityDataUzunluk (MQUZE)-çıkış**

*AuthorityBuffer*' ta döndürülen veri uzunluğu.

Bu, *AuthorityBuffer*' ta döndürülen verilerin uzunluğudur. Yetki arabelleği çok küçükse, *AuthorityDataLength* gerekli arabelleğin uzunluğuna ayarlanır ve çağrı, MQCC\_FAILED tamamlama kodunu döndürür ve neden kodu MQRC\_BUFFER\_LENGTH\_ERROR değerini döndürür.

### **ComponentData (MQBYTE x ComponentDataLength)-giriş/çıkış**

Bileşen verileri.

Bu veriler, bu bileşen adına kuyruk yöneticisi tarafından tutulur; bu bileşen tarafından sağlanan işlevlerden herhangi biri tarafından yapılan değişiklikler korunur ve bu bileşenin sonraki işlevlerinden biri çağrıldığında sunulur.

Bu veri alanının uzunluğu, MQZ\_INIT\_AUTHORITY çağrısının **ComponentDataLength** parametresindeki kuyruk yöneticisi tarafından geçirilir.

### **Devam (MQLONG)-çıkış**

Bileşene göre devam göstergesi kümesi.

Aşağıdaki değerler belirlenebilir:

#### **MQZCI\_VARSAYILAN**

Devam eden kuyruk yöneticisine bağımlı.

MQZ\_ENUMERATE\_AUTHORITY\_DATA için bu, MQZCI\_CONTINUE ile aynı etkiye sahiptir.

#### **MQZCI\_CONTINY**

Sonraki bileşenle devam edin.

#### **MQZCI\_STOP**

Sonraki bileşenle devam ettirmeyin.

### **CompCode (MQHOT)-çıkış**

Tamamlanma kodu.

Bu, aşağıdakilerden biridir:

#### **MQCC\_OK**

İşlem başarıyla tamamlandı.

#### **MQCC\_FAILED**

Arama başarısız oldu.

### **Neden (MQUZE)-çıkış**

Neden kodu ön eleme *CompCode*.

*CompCode* MQCC\_OK ise:

#### **MQRC\_NONE**

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_FAILED ise:

#### **MQRC\_BUFFER\_LENGTH\_ERROR**

(2005, X'7D5') Arabellek uzunluğu parametresi geçerli değil.

#### **MQRC\_NO\_DATA\_AVAILABLE**

(2379, X'94B') Kullanılabilir veri yok.

## **MQRC\_SERVICE\_ERROR**

(2289, X'8F1') Hizmet erişimi beklenmeyen bir hata oluştu.

Bu neden kodlarına ilişkin ek bilgi için [İletiler ve neden kodları](#) başlıklı konuya bakın.

## **C çağırma**

```
MQZ_ENUMERATE_AUTHORITY_DATA (QMgrName, StartEnumeration, &Filter,  
    AuthorityBufferLength,  
    &AuthorityBuffer,  
    &AuthorityDataLength, ComponentData,  
    &Continuation, &CompCode,  
    &Reason);
```

Hizmete geçirilen parametreler aşağıdaki gibi bildirilir:

```
MQCHAR48  QMgrName;           /* Queue manager name */  
MQLONG    StartEnumeration; /* Flag indicating whether call should  
                                start enumeration */  
MQZAD     Filter;           /* Filter */  
MQLONG    AuthorityBufferLength; /* Length of AuthorityBuffer */  
MQZAD     AuthorityBuffer;  /* Authority data */  
MQLONG    AuthorityDataLength; /* Length of data returned in  
                                AuthorityBuffer */  
MQBYTE    ComponentData[n]; /* Component data */  
MQLONG    Continuation;     /* Continuation indicator set by  
                                component */  
MQLONG    CompCode;         /* Completion code */  
MQLONG    Reason;          /* Reason code qualifying CompCode */
```

## **MQZ\_FREE\_USER-Serbest kullanıcı**

Bu işlev bir MQZAS\_VERSION\_5 yetki hizmeti bileşeni tarafından sağlanır ve kuyruk yöneticisi tarafından, ilişkili ayrılmış kaynak için serbest olan serbest bir şekilde çağrılır. Bir uygulama tüm kullanıcı bağlamları altında çalışmayı bitirdiğinde (örneğin, bir MQDISC MQI çağrısı sırasında) çağrılır.

Bu işleve ilişkin işlev tanıtıcısı (MQZEP için) MQZID\_FREE\_USER.

## **IBM i IBM üzerinde MQZ\_GET\_AUTHORITY (Yetki al)**

Bu işlev bir MQZAS\_VERSION\_1 yetki hizmeti bileşeni tarafından sağlanır ve kuyruk yöneticisi tarafından bir varlığın belirtilen nesneye erişmesi için sahip olduğu yetkiyi almak için çağrılır.

Bu işleve ilişkin işlev tanıtıcısı (MQZEP için) MQZID\_GET\_AUTHORITY olur.

## **Sözdizimi**

**MQZ\_GET\_AUTHORITY (QMgrName, EntityName, EntityType, ObjectName, ObjectType, Authority, ComponentData, Continuation, CompCode, Reason)**

## **Parametreler**

MQZ\_GET\_AUTHORITY çağrısının değiştirgeleri şunlardır:

### **QMgrName (MQCHAR48)-giriş**

Kuyruk yöneticisi adı.

Bileşeni çağırın kuyruk yöneticisinin adı. Bu ad, değiştirgenin tam uzunluğuna kadar boşluklarla dolduruldu; ad boş değerli bir karakterle sonlandırılmaz.

Kuyruk yöneticisi adı, bileşene bilgi için iletilir; yetki hizmeti arabirimi, bileşenin herhangi bir şekilde tanımlanmış bir şekilde kullanılmasını gerektirmez.

### **EntityName (MQCHAR12)-giriş**

Varlık adı.

Nesneye erişimi alınmak üzere olan varlığın adı. Dizilimin uzunluk üst sınırı 12 karakterdir; boşlukla doldurulsa boşluk karakterinden daha kısadır. Ad boş değerli bir karakter tarafından sonlandırılmadı.

#### **EntityType (MQUBE)-giriş**

Varlık tipi.

*EntityType* tarafından belirtilen varlık tipi. Aşağıdaki değer belirlenebilir:

#### **MQZAET\_PRINCIPAL**

Müdür.

#### **MQZAET\_GRUP**

Grup.

#### **ObjectName (MQCHAR48)-giriş**

Nesne adı.

Varlığın yetkisinin alınacağı nesnenin adı. Dizilimin uzunluk üst sınırı 48 karakterdir; boşlukla sağa yaslanmış ise, bu karakter uzunluğuna göre daha kısadır. Ad boş değerli bir karakter tarafından sonlandırılmadı.

*ObjectType* MQOT\_Q\_MGR ise, bu ad *QMgrName* ile aynıdır.

#### **ObjectType (MQUZE)-giriş**

Nesne tipi.

*ObjectName* tarafından belirtilen varlık tipi. Bu, aşağıdakilerden biridir:

#### **MQOT\_AUTH\_INFO**

Kimlik doğrulama bilgileri.

#### **MQOT\_KANAL**

Kanal.

#### **MQOT\_CLNTCONN\_CHANNEL**

İstemci bağlantı kanalı.

#### **MQOT\_LISTENER**

Dinleyici.

#### **MQOT\_NAMELIST**

İsim listesi.

#### **MQOT\_PROCESS**

Süreç tanımlaması.

#### **MQOT\_Q**

Kuyruk.

#### **MQOT\_Q\_MGR**

Kuyruk yöneticisi.

#### **MQOT\_SERVICE**

Servis.

#### **Yetki (MQHOT)-çıkış**

Varlık yetkisi.

Varlığın bir yetkisi varsa, bu alan uygun yetkilendirme işlemine eşittir (MQZAO\_ \* sabiti). Birden fazla yetkiniz varsa, bu alan, ilgili MQZAO\_ \* değişmezlerinin bit ya da bit ya da bitmezidir.

#### **ComponentData (MQBYTE x ComponentDataLength)-giriş/çıkış**

Bileşen verileri.

Bu veriler, bu bileşen adına kuyruk yöneticisi tarafından tutulur; bu bileşen tarafından sağlanan işlevlerden herhangi biri tarafından yapılan değişiklikler korunur ve bu bileşenin sonraki işlevlerinden biri çağrıldığında sunulur.

Bu veri alanının uzunluğu, MQZ\_INIT\_AUTHORITY çağrısının **ComponentDataLength** parametresindeki kuyruk yöneticisi tarafından geçirilir.

### Devam (MQLONG)-çıkış

Bileşene göre devam göstergesi kümesi.

Aşağıdaki değerler belirlenebilir:

#### MQZCI\_VARSAYILAN

Devam eden kuyruk yöneticisine bağımlı.

MQZ\_GET\_AUTHORITY için bu, MQZCI\_CONTINUE ile aynı etkiye sahiptir.

#### MQZCI\_CONTINY

Sonraki bileşenle devam edin.

#### MQZCI\_STOP

Sonraki bileşenle devam ettirmeyin.

### CompCode (MQHOT)-çıkış

Tamamlanma kodu.

Bu, aşağıdakilerden biridir:

#### MQCC\_OK

İşlem başarıyla tamamlandı.

#### MQCC\_FAILED

Arama başarısız oldu.

### Neden (MQUZE)-çıkış

Neden kodu ön eleme *CompCode*.

*CompCode* MQCC\_OK ise:

#### MQRC\_NONE

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_FAILED ise:

#### MQRC\_NOT\_YETKILI

(2035, X'7F3') Erişim için yetkili değil.

#### MQRC\_SERVICE\_ERROR

(2289, X'8F1') Hizmet erişimi beklenmeyen bir hata oluştu.

#### MQRC\_SERVICE\_NOT\_VAR

(2285, X'8ED') Udering hizmeti kullanılmıyor.

#### MQRC\_UNKNOWN\_ENTITY

(2292, X'8F4') Hizmet için bilinmeyen varlık.

Bu neden kodlarına ilişkin ek bilgi için [İletiler ve neden kodları](#) başlıklı konuya bakın.

## C çağırma

```
MQZ_GET_AUTHORITY (QMgrName, EntityName, EntityType, ObjectName,  
ObjectTyp e, &Authority, ComponentData,  
&Continuation, &CompCode, &Reason);
```

Hizmete geçirilen parametreler aşağıdaki gibi bildirilir:

```
MQCHAR48 QMgrName;          /* Queue manager name */  
MQCHAR12 EntityName;        /* Entity name */  
MQLONG   EntityType;        /* Entity type */  
MQCHAR48 ObjectName;        /* Object name */  
MQLONG   ObjectTyp e;       /* Object type */  
MQLONG   Authority;         /* Authority of entity */  
MQBYTE   ComponentData[n]; /* Component data */  
MQLONG   Continuation;      /* Continuation indicator set by  
component */  
MQLONG   CompCode;          /* Completion code */  
MQLONG   Reason;            /* Reason code qualifying CompCode */
```



## IBM üzerinde MQZ\_GET\_EXPLICIT\_AUTHORITY (Belirtik yetkisi al)

Bu işlev bir MQZAS\_VERSION\_1 yetki hizmeti bileşeni tarafından sağlanır ve kuyruk yöneticisi tarafından belirtilen bir grubun belirtilen bir nesneye erişmesi (ancak **kimse** grubunun ek yetkisi olmadan) ya da belirtilen birincil kullanıcının birincil grubunun belirtilen bir nesneye erişmek zorunda olduğu yetkiyi almak için kuyruk yöneticisi tarafından çağrılır.

Bu işleve ilişkin işlev tanıtıcısı (MQZEP için) MQZID\_GET\_REGICIIT\_AUTHORITY.

### Sözdizimi

**MQZ\_GET\_REMATIONIT\_AUTHORITY** (*QMgrName, EntityName, EntityType, ObjectName, ObjectType, Authority, ComponentData, Continuation, CompCode, Reason*)

### Parametreler

MQZ\_GET\_REGICIIT\_AUTHORITY çağrısında şu deęiřtirgeler var.

#### **QMgrName (MQCHAR48)-giriř**

Kuyruk yöneticisi adı.

Bileşeni çağırın kuyruk yöneticisinin adı. Bu ad, deęiřtirgenin tam uzunluęuna kadar boşluklarla dolduruldu; ad boş deęerli bir karakterle sonlandırılmaz.

Kuyruk yöneticisi adı, bileşene bilgi için iletilir; yetki hizmeti arabirimi, bileşenin herhangi bir şekilde tanımlanmış bir şekilde kullanılmasını gerektirmez.

#### **EntityName (MQCHAR12)-giriř**

Varlık adı.

Nesneye erişimin alınacağı varlığın adı. Dizilimin uzunluk üst sınırı 12 karakterdir; boşlukla doldurulsa boşluk karakterinden daha kısadır. Ad boş deęerli bir karakter tarafından sonlandırılmadı.

#### **EntityType (MQUBE)-giriř**

Varlık tipi.

*EntityName* tarafından belirtilen varlık tipi. Ařaęıdaki deęer belirlenebilir:

##### **MQZAET\_PRINCIPAL**

Müdür.

##### **MQZAET\_GRP**

Grup.

#### **ObjectName (MQCHAR48)-giriř**

Nesne adı.

Varlığın yetkisinin alınacağı nesnenin adı. Dizilimin uzunluk üst sınırı 48 karakterdir; boşlukla saęa yaslanmış ise, bu karakter uzunluęuna göre daha kısadır. Ad boş deęerli bir karakter tarafından sonlandırılmadı.

*ObjectType* MQOT\_Q\_MGR ise, bu ad *QMgrName* ile aynıdır.

#### **ObjectType (MQUZE)-giriř**

Nesne tipi.

*ObjectName* tarafından belirtilen varlık tipi. Bu, ařaęıdakilerden biridir:

##### **MQOT\_AUTH\_INFO**

Kimlik doęrulama bilgileri.

##### **MQOT\_KANAL**

Kanal.

##### **MQOT\_CLNTCONN\_CHANNEL**

İstemci baęlantı kanalı.

**MQOT\_LISTENER**

Dinleyici.

**MQOT\_NAMELIST**

İsim listesi.

**MQOT\_PROCESS**

Süreç tanımlaması.

**MQOT\_Q**

Kuyruk.

**MQOT\_Q\_MGR**

Kuyruk yöneticisi.

**MQOT\_SERVICE**

Servis.

**Yetki (MQHOT)-çıkış**

Varlık yetkisi.

Varlığın bir yetkisi varsa, bu alan uygun yetkilendirme işlemine eşittir (MQZAO\_ \* sabiti). Birden fazla yetkiniz varsa, bu alan, ilgili MQZAO\_ \* değişmezlerinin bit ya da bit ya da bitmezidir.

**ComponentData (MQBYTE x ComponentDataLength)-giriş/çıkış**

Bileşen verileri.

Bu veriler, bu bileşen adına kuyruk yöneticisi tarafından tutulur; bu bileşen tarafından sağlanan işlevlerden herhangi biri tarafından yapılan değişiklikler korunur ve bu bileşenin sonraki işlevlerinden biri çağrıldığında sunulur.

Bu veri alanının uzunluğu, MQZ\_INIT\_AUTHORITY çağrısının **ComponentDataLength** parametresindeki kuyruk yöneticisi tarafından geçirilir.

**Devam (MQLONG)-çıkış**

Bileşene göre devam göstergesi kümesi.

Aşağıdaki değerler belirlenebilir:

**MQZCI\_VARSAYILAN**

Devam eden kuyruk yöneticisine bağlı.

MQZ\_GET\_RECELENIT\_AUTHORITY için bu, MQZCI\_CONTINUE ile aynı etkiye sahiptir.

**MQZCI\_CONTINY**

Sonraki bileşenle devam edin.

**MQZCI\_STOP**

Sonraki bileşenle devam ettirmeyin.

**CompCode (MQHOT)-çıkış**

Tamamlanma kodu.

Bu, aşağıdakilerden biridir:

**MQCC\_OK**

İşlem başarıyla tamamlandı.

**MQCC\_FAILED**

Arama başarısız oldu.

**Neden (MQUZE)-çıkış**

Neden kodu ön eleme *CompCode*.

*CompCode* MQCC\_OK ise:

**MQRC\_NONE**

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_FAILED ise:

### **MQRC\_NOT\_YETKILI**

(2035, X'7F3') Erişim için yetkili değil.

### **MQRC\_SERVICE\_ERROR**

(2289, X'8F1') Hizmet erişimi beklenmeyen bir hata oluştu.

### **MQRC\_SERVICE\_NOT\_VAR**

(2285, X'8ED') Underlying hizmeti kullanılamıyor.

### **MQRC\_UNKNOWN\_ENTITY**

(2292, X'8F4') Hizmet için bilinmeyen varlık.

Bu neden kodlarına ilişkin ek bilgi için [İletiler ve neden kodları](#) başlıklı konuya bakın.

## **C çağırma**

```
MQZ_GET_EXPLICIT_AUTHORITY (QMgrName, EntityName, EntityType,  
                             ObjectName, ObjectType, &Authority,  
                             ComponentData, &Continuation,  
                             &CompCode, &Reason);
```

Hizmete geçirilen parametreler aşağıdaki gibi bildirilir:

```
MQCHAR48  QMgrName;           /* Queue manager name */  
MQCHAR12  EntityName;        /* Entity name */  
MQLONG    EntityType;        /* Entity type */  
MQCHAR48  ObjectName;       /* Object name */  
MQLONG    ObjectType;       /* Object type */  
MQLONG    Authority;        /* Authority of entity */  
MQBYTE    ComponentData[n]; /* Component data */  
MQLONG    Continuation;     /* Continuation indicator set by  
                             component */  
MQLONG    CompCode;         /* Completion code */  
MQLONG    Reason;          /* Reason code qualifying CompCode */
```

## **IBM i IBM üzerinde MQZ\_INIT\_AUTHORITY (Yetkilendirme hizmetini kullanıma hazırla)**

Bu işlev bir yetkilendirme hizmeti bileşeni tarafından sağlanır ve bileşen yapılandırması sırasında kuyruk yöneticisi tarafından çağrılır. Kuyruk yöneticisine bilgi sağlamak için MQZEP çağırılması beklenir.

Bu işleve ilişkin işlev tanıtıcısı (MQZEP için) MQZID\_INIT\_AUTHORITY 'dir.

### **Sözdizimi**

**MQZ\_INIT\_AUTHORITY (Hconfig, Options, QMgrName, ComponentDataLength, ComponentData, Version, CompCode, Reason)**

### **Parametreler**

MQZ\_INIT\_AUTHORITY çağırısının deęiřtirgeleri řunlardır:

#### **Hconfig (MQHCONFIG)-giriř**

Yapılandırma tanıtıcısı.

Bu tanıtıcı, ilk kullanıma hazırlanmakta olan bileşeni gösterir. Bu bileşen, kuyruk yöneticisini MQZEP işleviyle çağırırken, bu bileşen tarafından kullanılır.

#### **Seçenekler (MQUZE)-giriř**

Başlatma seçenekleri.

Bu, aşağıdakilerden biridir:

#### **MQZI\_BIRINCIL**

Birincil kullanıma hazırlama.

## **MQZIO\_IKINCIL**

İkincil kullanıma hazırlama.

### **QMgrName (MQCHAR48)-giriş**

Kuyruk yöneticisi adı.

Bileşeni çağıran kuyruk yöneticisinin adı. Bu ad, değiştirgenin tam uzunluğuna kadar boşluklarla dolduruldu; ad boş değerli bir karakterle sonlandırılmaz.

Kuyruk yöneticisi adı, bileşene bilgi için iletilir; yetki hizmeti arabirimi, bileşenin herhangi bir şekilde tanımlanmış bir şekilde kullanılmasını gerektirmez.

### **ComponentDataUzunluğu (MQUZE)-giriş**

Bileşen verilerinin uzunluğu.

*ComponentData* alanının bayt cinsinden uzunluğu. Bu uzunluk, bileşen yapısını verilerinde tanımlanır.

### **ComponentData (MQBYTE x ComponentDataLength)-giriş/çıkış**

Bileşen verileri.

Bu işlem, bileşenin birincil kullanıma hazırlama işlevini çağırılmadan önce tüm sıfırlar için kullanıma hazırlanır. Bu veriler, bu bileşen adına kuyruk yöneticisi tarafından tutulur; bu bileşen tarafından sağlanan işlevler (kullanıma hazırlama işlevi de içinde olmak üzere) tarafından bu bileşen için yapılan değişiklikler korunur ve bu bileşenin sonraki işlevlerinden biri çağırıldığında sunulur.

### **Sürüm (MQUZE)-giriş/çıkış**

Sürüm numarası.

Kullanıma hazırlama işlevine ilişkin giriş sırasında, kuyruk yöneticisinin desteklediği *en yüksek* sürüm numarasını tanıtır. The initialization function must change this, if necessary, to the version of the interface which - supports. Geri dönerse, kuyruk yöneticisi bileşenin döndürdüğü sürümü desteklemiyorsa, bileşenin MQZ\_TERM\_AUTHORITY işlevini çağırır ve bu bileşeni başka bir şekilde kullanmaz.

Aşağıdaki değerler desteklenir:

#### **MQZAS\_VERSION\_1**

Sürüm 1.

#### **MQZAS\_VERSION\_2**

Sürüm 2.

#### **MQZAS\_VERSION\_3**

Sürüm 3.

#### **MQZAS\_VERSION\_4**

Sürüm 4.

#### **MQZAS\_VERSION\_5**

Sürüm 5.

#### **MQZAS\_VERSION\_6**

IBM WebSphere MQ 6.

### **CompCode (MQHOT)-çıkış**

Tamamlanma kodu.

Bu, aşağıdakilerden biridir:

#### **MQCC\_OK**

İşlem başarıyla tamamlandı.

#### **MQCC\_FAILED**

Arama başarısız oldu.

### **Neden (MQUZE)-çıkış**

Neden kodu ön eleme *CompCode*.

*CompCode* MQCC\_OK ise:

## **MQRC\_NONE**

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_FAILED ise:

## **MQRC\_INITIALIZATION\_FAILED**

(2286, X'8EE') Kullanıma hazırlama, tanımlanmamış bir neden için başarısız oldu.

## **MQRC\_SERVICE\_NOT\_VAR**

(2285, X'8ED') Undering hizmeti kullanılmıyor.

Bu neden kodlarına ilişkin ek bilgi için [İletiler ve neden kodları](#) başlıklı konuya bakın.

## **C çağırma**

```
MQZ_INIT_AUTHORITY (Hconfig, Options, QMgrName, ComponentDataLength,  
ComponentData, &Version, &CompCode,  
&Reason);
```

Hizmete geçirilen parametreler aşağıdaki gibi bildirilir:

```
MQHCONFIG  Hconfig;          /* Configuration handle */  
MQLONG     Options;         /* Initialization options */  
MQCHAR48   QMgrName;       /* Queue manager name */  
MQLONG     ComponentDataLength; /* Length of component data */  
MQBYTE     ComponentData[n]; /* Component data */  
MQLONG     Version;        /* Version number */  
MQLONG     CompCode;       /* Completion code */  
MQLONG     Reason;        /* Reason code qualifying CompCode */
```

**IBM i**

## **IBM üzerinde MQZ\_SORGULAMAK (Sorgulama Yetkilendirme Hizmeti)**

Bu işlev bir MQZAS\_VERSION\_5 yetkilendirme hizmeti bileşeni tarafından sağlanır ve kuyruk yöneticisi tarafından desteklenen işlevselliği sorgulamak için çağrılır. Birden çok hizmet bileşeni kullanıldığı durumlarda, hizmet bileşenleri, takılı oldukları siparişe ters sırayla çağrılır.

Bu işleve ilişkin işlev tanıtıcısı (MQZEP için) MQZID\_SORGULADIR.

## **Sözdizimi**

### **MQZ\_SORGULAMA**

(*QMgrName, SelectorCount, Selectors, IntAttrCount, IntAttrs, CharAttrLength, CharAttrs, SelectorReturned, ComponentData, Continuation, CompCode, Reason*)

## **Parametreler**

MQZ\_SORSORE çağrısında aşağıdaki parametreler yer aldı.

### **QMgrName (MQCHAR48)-giriş**

Kuyruk yöneticisi adı.

Bileşeni çağıran kuyruk yöneticisinin adı. Bu ad, değiştirgenin tam uzunluğuna kadar boşluklarla dolduruldu; ad boş değerli bir karakterle sonlandırılmaz.

Kuyruk yöneticisi adı, bileşene bilgi için iletilir; yetki hizmeti arabirimi, bileşenin herhangi bir şekilde tanımlanmış bir şekilde kullanılmasını gerektirmez.

### **SelectorCount (MQUZE)-giriş**

Seçicilerin sayısı.

Seçiciler parametresinde sağlanan seçicilerin sayısı.

Değer sıfır ile 256 arasında bir değer olmalıdır.

### **Seçiciler (MQUBIT x SelectorCount)-giriş**

Seçiciler.

Seçiciler dizisi. Her seçici, gerekli bir özniteliği tanımlar ve aşağıdaki tiplerden biri olmalıdır:

- MQIACF\_\* (tamsayı)
- MQCACF\_\* (karakter)

Seçiciler herhangi bir sırada belirtilebilir. Dizideki seçicilerin sayısı SelectorCount parametresiyle gösterilir.

Integer attributes identified by selectors are returned in the IntAttrs parameter in the same order as they appear in Selectors.

Character attributes identified by selectors are returned in the CharAttrs parameter in the same order as they in appear Selectors.

### **IntAttrSayı (MQUZE)-giriş**

Tamsayı özniteliklerinin sayısı.

IntAttrs değiştirilmesinde sağlanan tamsayı özniteliklerinin sayısı.

Değer 0 ile 256 aralığında olmalıdır.

### **IntAttrs (MQUXX IntAttrSayı)-çıkış**

Tamsayı öznitelikleri.

Tamsayı öznitelikleri dizisi. Tamsayı öznitelikleri, Seçiciler dizisinde karşılık gelen tamsayı seçiciyle aynı sırayla döndürülür.

### **CharAttrSayı (MQUZE)-giriş**

Karakter öznitelikleri arabelleğindeki uzunluk.

CharAttrs parametresinin bayt cinsinden uzunluğu.

Değer, istenen karakter özniteliklerinin uzunluklarının en az toplamını elde eder. Karakter öznitelikleri istenmiyorsa, sıfır geçerli bir değerdir.

### **CharAttrs (MQUXX CharAttrSayı)-çıkış**

Karakter öznitelikleri arabelleği.

Birlikte bitleştirilmiş karakter özniteliklerini içeren arabellek. Karakter öznitelikleri, Seçiciler dizisinde karşılık gelen karakter seçiciyle aynı sırayla döndürülür.

Arabelleğin uzunluğu CharAttrSayı parametresi tarafından verilir.

### **SelectorReturned (MQLONGxSelectorCount)-giriş**

Seçici geri döndü.

Seçiciler (Selectors) parametresindeki seçiciler tarafından istenen kümeden hangi özniteliklerin döndürüldüğünü tanımlayan değerler dizisi. Bu dizideki değer sayısı, SelectorCount parametresiyle belirtilir. Dizideki her bir değer, Seçiciler dizisinde karşılık gelen konumdaki seçiciye ilişkinir. Her değer aşağıdakilerden biridir:

#### **MQZSL\_RETURNED**

Seçiciler parametresindeki ilgili seçici tarafından istenen öznitelik döndürüldü.

#### **MQZSL\_NOT\_RETURNED**

Seçiciler parametresindeki ilgili seçici tarafından istenen öznitelik döndürülmedi.

Dizi tüm değerlerle ilk kullanıma hazırlandı; *MQZL\_NOT\_RETURNED*. Bir yetki hizmeti bileşeni bir öznitelik döndürdüğünde, dizideki uygun değeri *MQZL\_RETURNED* olarak ayarlar. Bu, sorgu çağrısının yapıldığı diğer yetki hizmeti bileşenlerinin hangi özniteliklerin önceden geri gönderildiğini belirlemek için sağlanmasını sağlar.

### **ComponentData (MQBYTE x ComponentDataLength)-giriş/çıkış**

Bileşen verileri.

Bu veriler, bu bileşen adına kuyruk yöneticisi tarafından tutulur; bu bileşen tarafından sağlanan işlevlerden herhangi biri tarafından yapılan değişiklikler korunur ve bu bileşenin sonraki işlevlerinden biri çağrıldığında sunulur.

Bu veri alanının uzunluğu, MQZ\_INIT\_AUTHORITY çağrısının **ComponentDataLength** parametresindeki kuyruk yöneticisi tarafından geçirilir.

#### **Devam (MQLONG)-çıkış**

Devam işareti.

Aşağıdaki değerler belirlenebilir:

#### **MQZCI\_VARSAYILAN**

Diğer bileşenlere bağlı olarak devam edin.

#### **MQZCI\_STOP**

Sonraki bileşenle devam ettirmeyin.

#### **CompCode (MQHOT)-çıkış**

Tamamlanma kodu.

Bu, aşağıdakilerden biridir:

#### **MQCC\_OK**

İşlem başarıyla tamamlandı.

#### **MQCC\_UYARI**

Kısmi tamamlama.

#### **MQCC\_FAILED**

Arama başarısız oldu.

#### **Neden (MQUZE)-çıkış**

Neden kodu ön eleme *CompCode*.

*CompCode* MQCC\_OK ise:

#### **MQRC\_NONE**

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_UYARI ise:

#### **MQRC\_CHAR\_ATTRS\_TOO\_SHORT**

Karakter öznitelikleri için yeterli alan yok.

#### **MQRC\_INT\_COUNT\_TOO\_SMALL**

Tamsayı öznitelikleri için yeterli alan yok.

*CompCode* MQCC\_FAILED ise:

#### **MQRC\_SELECTOR\_COUNT\_ERROR**

Seçicilerin sayısı geçerli değil.

#### **MQRC\_SELECTOR\_ERROR**

Öznitelik seçici geçerli değil.

#### **MQRC\_SELECTOR\_LIMIT\_EXACID**

Çok sayıda seçici belirtildi.

#### **MQRC\_INT\_ATTR\_COUNT\_ERROR**

Tamsayı özniteliklerinin sayısı geçerli değil.

#### **MQRC\_INT\_ATTRS\_ARRAY\_ERROR**

Tamsayı öznitelikleri dizisi geçerli değil.

#### **MQRC\_CHAR\_ATTR\_LENGTH\_ERROR**

Karakter özniteliklerinin sayısı geçerli değil.

#### **MQRC\_CHAR\_ATTRS\_ERROR**

Karakter öznitelikleri dizisi geçerli değil.

## **MQR\_SERVICE\_ERROR**

(2289, X'8F1') Hizmet erişimi beklenmeyen bir hata oluştu.

## **C çağırma**

```
MQZ_INQUIRE (QMgrName, SelectorCount, Selectors, IntAttrCount,
              &IntAttrs, CharAttrLength, &CharAttrs,
              SelectorReturned, ComponentData, &Continuation,
              &CompCode, &Reason);
```

Hizmete geçirilen parametreler aşağıdaki gibi bildirilir:

```
MQCHAR48  QMgrName;           /* Queue manager name */
MQLONG    SelectorCount;     /* Selector count */
MQLONG    Selectors[n];      /* Selectors */
MQLONG    IntAttrCount;     /* IntAttrs count */
MQLONG    IntAttrs[n];      /* Integer attributes */
MQLONG    CharAttrCount;    /* CharAttrs count */
MQLONG    CharAttrs[n];     /* Character attributes */
MQLONG    SelectorReturned[n]; /* Selector returned */
MQBYTE    ComponentData[n]; /* Component data */
MQLONG    Continuation;     /* Continuation indicator set by
                             component */
MQLONG    CompCode;         /* Completion code */
MQLONG    Reason;          /* Reason code qualifying CompCode */
```

## **IBM i**

## **MQZ\_REFRESH\_CACHE (Refresh all authorizations) on IBM i**

Bu işlev bir MQZAS\_VERSION\_3 yetkilendirme hizmeti bileşeni tarafından sağlanır. Bu bileşen, bileşen tarafından dahili olarak tutulan yetkilerin listesini yenilemek için kuyruk yöneticisi tarafından çağrılır.

Bu işleve ilişkin işlev tanıtıcısı (MQZEP için) MQZID\_REFRESH\_CACHE (8L) olur.

## **Sözdizimi**

### **MQZ\_REFRESH\_CACHE**

(QMgrName, ComponentData, Continuation, CompCode, Reason)

## **Parametreler**

### **QMgrName (MQCHAR48)-giriş**

Kuyruk yöneticisi adı.

Bileşeni çağıran kuyruk yöneticisinin adı. Bu ad, değiştirgenin tam uzunluğuna kadar boşluklarla dolduruldu; ad boş değerli bir karakterle sonlandırılmaz.

Kuyruk yöneticisi adı, bileşene bilgi için iletilir; yetki hizmeti arabirimi, bileşenin bu bileşeni tanımlı bir şekilde kullanmasını gerektirmez.

### **ComponentData (MQBYTE x ComponentDataLength) -giriş/çıkış**

Bileşen verileri.

Bu veriler, bu belirli bileşen adına kuyruk yöneticisi tarafından tutulur. Bu bileşen tarafından sağlanan işlevlerden herhangi biri tarafından yapılan değişiklikler korunur ve bileşenin bir sonraki işlevi çağrıldığında sunulur.

Bu veri alanının uzunluğu, MQZ\_INIT\_AUTHORITY çağrısının *ComponentDataUzunluğu* değiştirilmesinde kuyruk yöneticisi tarafından geçirilir.

### **Devam (MQHOT)-çıkış**

Bileşene göre devam göstergesi kümesi.

Aşağıdaki değerler belirlenebilir:



### **MQZCI\_VARSAYILAN**

Devam eden kuyruk yöneticisine bağımlı.

MQZ\_REFRESH\_CACHE için bu, MQZCI\_CONTINUE ile aynı etkiye sahiptir.

### **MQZCI\_CONTINY**

Sonraki bileşenle devam edin.

### **MQZCI\_STOP**

Sonraki bileşenle devam ettirmeyin.

### **CompCode (MQHOT)-çıkış**

Tamamlanma kodu.

Bu, aşağıdakilerden biridir:

### **MQCC\_OK**

İşlem başarıyla tamamlandı.

### **MQCC\_FAILED**

Arama başarısız oldu.

### **Neden (MQUZE)-çıkış**

Neden kodu ön eleme *CompCode*.

*CompCode* , MQCC\_OK ise:

### **MQRC\_NONE**

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_FAILED ise:

### **MQRC\_SERVICE\_ERROR**

(2289, X'8F1') Hizmet erişimi beklenmeyen bir hata oluştu.

## **C çağırma**

```
MQZ_REFRESH_CACHE (QMgrName, ComponentData,  
&Continuation, &CompCode, &Reason);
```

Değişirgeleri aşağıdaki gibi bildirin:

```
MQCHAR48 QMgrName;          /* Queue manager name */  
MQBYTE   ComponentData[n]; /* Component data */  
MQLONG   Continuation;     /* Continuation indicator set by  
                           component */  
MQLONG   CompCode;        /* Completion code */  
MQLONG   Reason;         /* Reason code qualifying CompCode */
```

**IBM i**

## **IBM üzerinde MQZ\_SET\_AUTHORITY (Yetki ayarla)**

Bu işlev bir MQZAS\_VERSION\_1 yetki hizmeti bileşeni tarafından sağlanır ve kuyruk yöneticisi tarafından, bir varlığın belirtilen nesneye erişmesi için sahip olduğu yetkiyi ayarlamak üzere çağrılır.

Bu işleve ilişkin işlev tanıtıcısı (MQZEP için) MQZID\_SET\_AUTHORITY olur.

**Not:** Bu işlev, var olan yetkileri geçersiz kılar. Varolan yetkileri korumak için, bu işlevi bu işlevle yeniden ayarlamamız gerekir.

## **Sözdizimi**

**MQZ\_SET\_AUTHORITY (QMgrName, EntityName, EntityType, ObjectName, ObjectType, Authority, ComponentData, Continuation, CompCode, Reason)**

## Parametreler

MQZ\_SET\_AUTHORITY çağrısının deęiřtirgeleri řunlardır:

### **QMgrName (MQCHAR48)-giriř**

Kuyruk yöneticisi adı.

Bileřeni çağırın kuyruk yöneticisinin adı. Bu ad, deęiřtirgenin tam uzunluęuna kadar boşluklarla dolduruldu; ad boş deęerli bir karakterle sonlandırılmaz.

Kuyruk yöneticisi adı, bileřene bilgi için iletilir; yetki hizmeti arabirimi, bileřenin herhangi bir řekilde tanımlanmış bir řekilde kullanılmasını gerektirmez.

### **EntityName (MQCHAR12)-giriř**

Varlık adı.

Nesneye eriřimin ayarlanacaęı varlıęın adı. Dizilimin uzunluk üst sınırı 12 karakterdir; boşlukla doldurulsa boşluk karakterinden daha kısadır. Ad boş deęerli bir karakter tarafından sonlandırılmadı.

### **EntityType (MQUBE)-giriř**

Varlık tipi.

*EntityName* tarafından belirtilen varlık tipi. Ařaęıdaki deęer belirlenebilir:

#### **MQZAET\_PRINCIPAL**

Müdür.

#### **MQZAET\_GRUP**

Grup.

### **ObjectName (MQCHAR48)-giriř**

Nesne adı.

Eriřimin gerekli olduęu nesnenin adı. Dizilimin uzunluk üst sınırı 48 karakterdir; boşlukla saęa yaslanmış ise, bu karakter uzunluęuna göre daha kısadır. Ad boş deęerli bir karakter tarafından sonlandırılmadı.

*ObjectType* MQOT\_Q\_MGR ise, bu ad *QMgrName* ile aynıdır.

### **ObjectType (MQUZE)-giriř**

Nesne tipi.

*ObjectName* tarafından belirtilen varlık tipi. Bu, ařaęıdakilerden biridir:

#### **MQOT\_AUTH\_INFO**

Kimlik doęrulama bilgileri.

#### **MQOT\_KANAL**

Kanal.

#### **MQOT\_CLNTCONN\_CHANNEL**

İstemci baęlantı kanalı.

#### **MQOT\_LISTENER**

Dinleyici.

#### **MQOT\_NAMELIST**

İsim listesi.

#### **MQOT\_PROCESS**

Süreç tanımlaması.

#### **MQOT\_Q**

Kuyruk.

#### **MQOT\_Q\_MGR**

Kuyruk yöneticisi.

#### **MQOT\_SERVICE**

Servis.

## Yetki (MQUBE)-giriş

Denetlenecek yetki.

Bir yetki belirleniyorsa, bu alan uygun yetkilendirme işlemine eşittir (MQZAO\_ \* sabiti). Birden fazla yetki belirleniyorsa, bu değer, ilgili MQZAO\_ \* değişmezlerinin bit ya da bit ya da dışında bir ya da daha çok yetki tanımından farklı olur.

## ComponentData (MQBYTE x ComponentDataLength)-giriş/çıkış

Bileşen verileri.

Bu veriler, bu bileşen adına kuyruk yöneticisi tarafından tutulur; bu bileşen tarafından sağlanan işlevlerden herhangi biri tarafından yapılan değişiklikler korunur ve bu bileşenin sonraki işlevlerinden biri çağrıldığında sunulur.

Bu veri alanının uzunluğu, MQZ\_INIT\_AUTHORITY çağrısının **ComponentDataLength** parametresindeki kuyruk yöneticisi tarafından geçirilir.

## Devam (MQLONG)-çıkış

Bileşene göre devam göstergesi kümesi.

Aşağıdaki değerler belirlenebilir:

### MQZCI\_VARSAYILAN

Devam eden kuyruk yöneticisine bağlıdır.

MQZ\_SET\_AUTHORITY için bu, MQZCI\_STOP ile aynı etkiye sahiptir.

### MQZCI\_CONTINY

Sonraki bileşenle devam edin.

### MQZCI\_STOP

Sonraki bileşenle devam ettirmeyin.

## CompCode (MQHOT)-çıkış

Tamamlanma kodu.

Bu, aşağıdakilerden biridir:

### MQCC\_OK

İşlem başarıyla tamamlandı.

### MQCC\_FAILED

Arama başarısız oldu.

## Neden (MQUZE)-çıkış

Neden kodu ön eleme *CompCode*.

*CompCode* MQCC\_OK ise:

### MQRC\_NONE

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_FAILED ise:

### MQRC\_NOT\_YETKILI

(2035, X'7F3') Erişim için yetkili değil.

### MQRC\_SERVICE\_ERROR

(2289, X'8F1') Hizmet erişimi beklenmeyen bir hata oluştu.

### MQRC\_SERVICE\_NOT\_VAR

(2285, X'8ED') Undering hizmeti kullanılamıyor.

### MQRC\_UNKNOWN\_ENTITY

(2292, X'8F4') Hizmet için bilinmeyen varlık.

## C çağırma

MQZ\_SET\_AUTHORITY (QMgrName, EntityName, EntityType, ObjectName,

```
ObjectType, Authority, ComponentData,  
&Continuation, &CompCode, &Reason);
```

Hizmete geçirilen parametreler aşağıdaki gibi bildirilir:

```
MQCHAR48  QMgrName;           /* Queue manager name */  
MQCHAR12  EntityName;       /* Entity name */  
MQLONG    EntityType;       /* Entity type */  
MQCHAR48  ObjectName;       /* Object name */  
MQLONG    ObjectType;       /* Object type */  
MQLONG    Authority;        /* Authority to be checked */  
MQBYTE    ComponentData[n]; /* Component data */  
MQLONG    Continuation;     /* Continuation indicator set by  
                             component */  
MQLONG    CompCode;         /* Completion code */  
MQLONG    Reason;          /* Reason code qualifying CompCode */
```

## MQZ\_TERM\_AUTHORITY-Yetki hizmetini sonlandır

Bu işlev, bir yetkilendirme hizmeti bileşeni tarafından sağlanır ve kuyruk yöneticisi tarafından bu bileşenin hizmetlerinin artık gerekmediği durumlarda çağrılır. İşlev, bileşenin gerektirdiği herhangi bir temizleme işlemini gerçekleştirmelidir.

Bu işleve ilişkin işlev tanıtıcısı (MQZEP için) MQZID\_TERM\_AUTHORITY olur.

### Sözdizimi

**MQZ\_TERM\_AUTHORITY** (*Hconfig, Options, QMgrName, ComponentData, CompCode, Reason*)

### Parametreler

MQZ\_TERM\_AUTHORITY çağrısının değiştiricileri şunlardır:

#### Hconfig (MQHCONFIG)-giriş

Yapılandırma tanıtıcısı.

Bu tanıtıcı, sonlandırılmakta olan belirli bir bileşeni gösterir.

#### Seçenekler (MQUZE)-giriş

Sonlandırma seçenekleri.

Bu, aşağıdakilerden biridir:

##### MQZTO\_BIRINCIL

Birincil sonlandırma.

##### MQZTO\_IKINCIL

İkincil sonlandırma.

#### QMgrName (MQCHAR48)-giriş

Kuyruk yöneticisi adı.

Bileşeni çağırın kuyruk yöneticisinin adı. Bu ad, değiştirgenin tam uzunluğuna kadar boşluklarla dolduruldu; ad boş değerli bir karakterle sonlandırılmaz.

Kuyruk yöneticisi adı, bileşene bilgi için iletilir; yetki hizmeti arabirimi, bileşenin herhangi bir şekilde tanımlanmış bir şekilde kullanılmasını gerektirmez.

#### ComponentData (MQBYTE x ComponentDataLength)-giriş/çıkış

Bileşen verileri.

Bu veriler, bu bileşen adına kuyruk yöneticisi tarafından tutulur; bu bileşen tarafından sağlanan işlevlerden herhangi biri tarafından yapılan değişiklikler korunur ve bu bileşenin sonraki işlevlerinden biri çağrıldığında sunulur.

Bu veri alanının uzunluğu, MQZ\_INIT\_AUTHORITY çağrısındaki **ComponentDataLength** parametresindeki kuyruk yöneticisi tarafından geçirilir.

MQZ\_TERM\_AUTHORITY çağrısı tamamlandığında, kuyruk yöneticisi bu verileri atar.

### CompCode (MQHOT)-çıkış

Tamamlanma kodu.

Bu, aşağıdakilerden biridir:

#### MQCC\_OK

İşlem başarıyla tamamlandı.

#### MQCC\_FAILED

Arama başarısız oldu.

### Neden (MQUZE)-çıkış

Neden kodu ön eleme *CompCode*.

*CompCode* MQCC\_OK ise:

#### MQRC\_NONE

(0, X'000 ') Raporlamak için bir neden yok.

*CompCode* MQCC\_FAILED ise:

#### MQRC\_SERVICE\_NOT\_VAR

(2285, X'8ED') Undering hizmeti kullanılamıyor.

#### MQRC\_TERMINATION\_FAILED

(2287, X'8FF') Sona erdirme tanımlanmamış bir nedenden dolayı başarısız oldu.

Bu neden kodlarına ilişkin ek bilgi için [İletiler ve neden kodları](#) başlıklı konuya bakın.

## C çağırma

```
MQZ_TERM_AUTHORITY (Hconfig, Options, QMgrName, ComponentData,  
&CompCode, &Reason);
```

Hizmete geçirilen parametreler aşağıdaki gibi bildirilir:

```
MQHCONFIG  Hconfig;          /* Configuration handle */  
MQLONG     Options;         /* Termination options */  
MQCHAR48   QMgrName;       /* Queue manager name */  
MQBYTE     ComponentData[n]; /* Component data */  
MQLONG     CompCode;        /* Completion code */  
MQLONG     Reason;         /* Reason code qualifying CompCode */
```

## IBM i IBM üzerinde MQZAC (Uygulama bağlamı)

Bu parametre, çağırılan uygulamaya ilişkin verileri belirtir.

MQZAC yapısı, **ApplicationContext** parametresine ilişkin MQZ\_AUTHENTICATE\_USER çağrısında kullanılır.

### Alanlar

#### StrucId (MQCHAR4)

Yapı tanıtıcısı.

Değer:

#### MQZAC\_STRUC\_ID

Uygulama bağlamı yapısına ilişkin tanıtıcı.

C programlama dili için, sabit MQZAC\_STRUC\_ID\_ARRAY değişmezi de tanımlanır; bu, MQZAC\_STRUC\_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisi olur.

Bu, hizmete giriş alanıdır.

#### **Sürüm (MQZSN)**

Yapı sürüm numarası.

Değer:

#### **MQZAC\_VERSION\_1**

Version-1 uygulama bağlamı yapısı.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

#### **MQZAC\_CURRENT\_VERSION**

Uygulama bağlamı yapısının yürürlükteki sürümü.

Bu, hizmete giriş alanıdır.

#### **ProcessId (MQPID)**

İşlem tanıtıcısı.

Uygulamaya ilişkin süreç tanıtıcısı.

#### **ThreadId (MQTID)**

İş parçacığı tanıtıcısı.

Uygulamanın iş parçacığı tanıtıcısı.

#### **ApplName (MQCHAR28)**

Uygulama adı.

Uygulama adı.

#### **UserID (MQCHAR12)**

Kullanıcı kimliği.

IBM i sistemleri için, uygulama işinin yaratıldığı kullanıcı tanıtımı. (IBM üzerinde, uygulama işinde QWTSETP API ile bir tanıtım değiştirme işlemi gerçekleştirildiğinde, yürürlükteki kullanıcı tanıtımı döndürülür).

#### **EffectiveUserkimliği (MQCHAR12)**

Etkin kullanıcı kimliği.

IBM i sistemleri için, uygulama işinin yürürlükteki kullanıcı tanıtımı.

#### **Ortam (MQLONG)**

Ortam.

Bu alan, çağrılarının yapıldığı ortamı belirtir.

Bu, aşağıdaki değerlerden birine sahip olabilir:

#### **MQXE\_COMMAND\_SERVER**

Komut sunucusu.

#### **MQXE\_MQSC**

runmqsc komut yorumlayıcısı.

#### **MQXE\_MCA**

İleti kanalı aracısı

#### **MQXE\_DİĞER**

Tanımsız ortam

#### **CallerType (MQUZE)**

Arayan Tipi.

Bu alan, çağrıyı yapan programın tipini belirtir.

Bu, aşağıdaki değerlerden birine sahip olabilir:

## **MQXACT\_EXTERNAL**

Arama, kuyruk yöneticisi dışıdır.

## **MQXACT\_INTERNAL**

Arama, kuyruk yöneticisinin iç arasındır.

## **AuthenticationType (MQUZE)**

Kimlik Doğrulama Tipi.

Bu alan, gerçekleştirilmekte olan kimlik doğrulamasının tipini belirtir.

Bu, aşağıdaki değerlerden birine sahip olabilir:

## **MQZAT\_INITIAL\_CONTEXT**

Kimlik doğrulama çağrısı, kullanıcı bağlamının başlatıldığı için sona ermiş. Bu değer, bir MQCONN ya da MQCONNX çağrısı sırasında kullanılır.

## **MQZAT\_CHANGE\_CONTEXT**

Kimlik doğrulama çağrısı, değiştirilmekte olan kullanıcı bağlamından kaynaklanır. Bu değer, MCA kullanıcı bağlamını değiştirdiğinde kullanılır.

v

## **BindType (MQUZN)**

Bağ Tanımlama Tipi.

Bu alan, kullanımda olan bağlayıcının tipini belirtir.

Bu, aşağıdaki değerlerden birine sahip olabilir:

## **MQCNO\_FASTPATH\_BINDING**

Fastpath bağ tanımı.

## **MQCNO\_SHARED\_BINDING**

Paylaşılan bağ tanımı.

## **MQCNO\_ISOLATED\_BINDING**

Yalıtılmış bağ.

## **C bildirim**

```
typedef struct tagMQZAC MQZAC;
struct tagMQZAC {
    MQCHAR4    StrucId;           /* Structure identifier */
    MQLONG     Version;          /* Structure version number */
    MQPID      ProcessId;        /* Process identifier */
    MQTID      ThreadId;         /* Thread identifier */
    MQCHAR28   ApplName;         /* Application name */
    MQCHAR12   UserID;           /* User identifier */
    MQCHAR12   EffectiveUserID;  /* Effective user identifier */
    MQLONG     Environment;      /* Environment */
    MQLONG     CallerType;       /* Caller type */
    MQLONG     AuthenticationType; /* Authentication type */
    MQLONG     BindType;         /* Bind type */
};
```

## **IBM i IBM üzerinde MQZAD (Yetki verileri)**

MQZAD yapısı, iki değiştirge için MQZ\_ENUMERATE\_AUTHORITY\_DATA çağrısında kullanılıyor.

**Filter** ve **AuthorityBuffer** değiştirgeleriyle ilgili ek bilgi için [“IBM üzerinde MQZ\\_ENUMERATE\\_AUTHORITY\\_DATA \(Yetki verileri sıralı değer listesi\)” sayfa 1672](#) konusuna bakın:

- Çağrıya giriş yapan **Filter** parametresi için MQZAD kullanılır. Bu parametre, çağrı tarafından döndürülen yetki verilerini seçmek için kullanılacak seçim ölçütlerini belirtir.
- MQZAD, çağrıdan çıkış olan **AuthorityBuffer** parametresi için de kullanılır. Bu parametre, bir profil adı, nesne türü ve varlık birleşimine ilişkin yetkileri belirtir.

## Alanlar

### StrucId (MQCHAR4)

Yapı tanıtıcısı.

Değer:

### MQZAD\_STRUC\_ID

Yetki verileri yapısına ilişkin tanıtıcı.

C programlama dili için, MQZAD\_STRUC\_ID\_ARRAY değişmezi de tanımlanır; bu, MQZAD\_STRUC\_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisidir.

Bu, hizmete giriş alanıdır.

### Sürüm (MQUZE)

Yapı sürüm numarası.

Değer:

### MQZAD\_VERSION\_1

Version-1 yetki verileri yapısı.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

### MQZAD\_CURRENT\_VERSION

Yetki verileri yapısının yürürlükteki sürümü.

Bu, hizmete giriş alanıdır.

### ProfileName (MQCHAR48)

Profil adı.

**Filter** parametresi için bu alan, yetki verilerinin gerekli olduğu profil adıdır. Ad alanının sonuna kadar boş bırakılırsa ya da ilk boş karakter, tüm tanıtım adları için yetki verileri döndürülür.

**AuthorityBuffer** parametresi için bu alan, belirtilen seçim ölçütleriyle eşleşen bir profilin adıdır.

### ObjectType (MQUZE)

Nesne tipi.

**Filter** parametresi için bu alan, yetki verilerinin gerekli olduğu nesne tipidir. Bu değer MQOT\_ALL ise, tüm nesne tiplerine ilişkin yetki verileri döndürülür.

**AuthorityBuffer** parametresi için bu alan, **ProfileName** tarafından tanımlanan profilin geçerli olduğu nesne tipidir.

Değer aşağıdakilerden biridir; **Filter** parametresi için MQOT\_ALL değeri de geçerlidir:

### MQOT\_AUTH\_INFO

Kimlik doğrulama bilgileri.

### MQOT\_KANAL

Kanal.

### MQOT\_CLNTCONN\_CHANNEL

İstemci bağlantı kanalı.

### MQOT\_LISTENER

Dinleyici.

### MQOT\_NAMELIST

İsim listesi.

### MQOT\_PROCESS

Süreç tanımlaması.

### MQOT\_Q

Kuyruk.

### MQOT\_Q\_MGR

Kuyruk yöneticisi.



## **MQOT\_SERVICE**

Servis.

## **Yetki (MQUBE)**

Otorite.

**Filter** parametresi için bu alan yok sayılır.

For the **AuthorityBuffer** parameter, this field represents the authorizations that the entity has to the objects identified by **ProfileName** and **ObjectType**. Varlığın yalnızca bir yetkisi varsa, bu alan uygun yetki değerine eşittir (MQZAO\_ \* sabiti). Varlığın birden çok yetkisi varsa, alan bit bilgedir ya da karşılık gelen MQZAO\_ \* sabitlerinden biridir.

## **EntityDataPtr (PMQZED)**

Bir varlığı tanımlayan MQZED yapısının adresi.

**Filter** parametresi için, bu alan, yetki verilerinin gerekli olduğu varlığı tanımlayan bir MQZED yapısına işaret eder. **EntityDataPtr** boş gösterge ise, tüm varlıklar için yetki verileri döndürülür.

**AuthorityBuffer** parametresi için bu alan, döndürülen yetki verilerinin geldiği varlığı tanımlayan bir MQZED yapısına işaret eder.

## **EntityType (MQUBE)**

Varlık tipi.

**Filter** parametresi için bu alan, yetki verilerinin gerekli olduğu varlık tipini belirtir. Bu değer MQZAET\_NONE ise, tüm varlık tiplerine ilişkin yetki verileri döndürülür.

**AuthorityBuffer** parametresi için bu alan, **EntityDataPtr** tarafından işaret edilen MQZED yapısı tarafından tanımlanan varlığın tipini belirtir.

Değer aşağıdakilerden biridir; **Filter** parametresi için, MQZAET\_NONE değeri de geçerlidir:

### **MQZAET\_PRINCIPAL**

Müdür.

### **MQZAET\_GRP**

Grup.

## **Seçenekler (MQAUTHOPT)**

Seçenekler.

Bu alan, görüntülenen tanımlar üzerinde denetim veren seçenekleri belirtir.

Aşağıdakilerden biri belirtilmeli:

### **MQAUTHOPT\_NAME\_ALL\_MATCHING**

Tüm profilleri görüntüler

### **MQAUTHOPT\_NAME\_EXPLICIT**

**ProfileName** alanında belirtildiği gibi, tam olarak aynı ada sahip profilleri görüntüler.

Ayrıca, aşağıdakilerden birinin de belirtilmesi gerekir:

### **MQAUTHOPT\_ENTITY\_SET**

Display all profiles used to calculate the cumulative authority that the entity has to the object specified by **ProfileName**. **ProfileName** alanı, herhangi bir genel arama karakteri içermemelidir.

- If the specified entity is a principal, for each member of the set {entity, groups} the most applicable profile that applies to the object is displayed.
- Belirtilen varlık bir grupsa, nesne için geçerli olan gruptan en geçerli profil görüntülenir.
- If this value is specified, then the values of **ProfileName**, **ObjectType**, **EntityType**, and the entity name specified in the **EntityDataPtr** MQZED structure, must all be non-blank.

**MQAUTHOPT\_NAME\_ALL\_MATCHING** belirttiyse, aşağıdakileri de belirtebilirsiniz:

## **MQAUTHOPT\_ENTITY\_EXPLICIT**

**EntityDataPtr** MQZED yapısında belirtilen varlık adıyla tam olarak aynı varlık adına sahip profilleri görüntüler.

## **C bildirimi**

```
typedef struct tagMQZAD MQZAD;
struct tagMQZAD {
    MQCHAR4    StrucId;           /* Structure identifier */
    MQLONG     Version;          /* Structure version number */
    MQCHAR48   ProfileName;      /* Profile name */
    MQLONG     ObjectType;        /* Object type */
    MQLONG     Authority;         /* Authority */
    PMQZED     EntityDataPtr;     /* Address of MQZED structure identifying an
                                   entity */
    MQLONG     EntityType;        /* Entity type */
    MQAUTHOPT  Options;          /* Options */
};
```

## **IBM i IBM üzerinde MQZED (Varlık tanımlayıcı)**

MQZED yapısı, yetkisinin denetleneceği varlığı belirtmek için bir dizi yetki hizmeti çağrısında kullanılır.

### **Alanlar**

#### **StrucId (MQCHAR4)**

Yapı tanıtıcısı.

Değer:

#### **MQZED\_STRUC\_ID**

Varlık tanımlayıcısı yapısına ilişkin tanıtıcı.

C programlama dili için, MQZED\_STRUC\_ID\_ARRAY değişmezi de tanımlanır; bu, MQZED\_STRUC\_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisidir.

Bu, hizmete giriş alanıdır.

#### **Sürüm (MQUZE)**

Yapı sürüm numarası.

Değer:

#### **MQZED\_VERSION\_1**

Version-1 varlık tanımlayıcı yapısı.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

#### **MQZED\_CURRENT\_VERSION**

Varlık tanımlayıcı yapısının yürürlükteki sürümü.

Bu, hizmete giriş alanıdır.

#### **EntityNamePtr (PMQCHAR)**

Varlık adı adresi.

Bu, yetkisini denetlenecek olan varlığın adına gösteren bir işaretidir.

#### **EntityDomainPtr (PMQCHAR)**

Varlık etki alanı adresi.

Bu, yetkisini denetlenecek olan varlığın tanımını içeren etki alanının adına işaretidir.

#### **SecurityId (MQBYTE40)**

Güvenlik tanıtıcısı.

Bu, yetkisini denetlenecek güvenlik tanımlayıcısıdır.

## CorrelationPtr (MQPTR)

İlinti göstergesi.

Bu, kimlik doğrulama kullanıcı işlevi ile diğer uygun OAM işlevleri arasında ilişkisel verilerin aktarılmasını kolaylaştırır.

## C bildirim

```
typedef struct tagMQZED MQZED;  
struct tagMQZED {  
    MQCHAR4    StrucId;           /* Structure identifier */  
    MQLONG     Version;          /* Structure version number */  
    PMQCHAR    EntityNamePtr;    /* Address of entity name */  
    PMQCHAR    EntityDomainPtr;  /* Address of entity domain name */  
    MQBYTE40   SecurityId;       /* Security identifier */  
    MQPTR      CorrelationPtr;   /* Address of correlation data */  
};
```

## IBM i MQZFP (Free parameters) on IBM i

Bu parametre, serbest bırakılacak kaynakla ilgili verileri belirtir.

MQZFP yapısı, **FreeParms** parametresine ilişkin MQZ\_FREE\_USER çağrısında kullanılır.

## Alanlar

### StrucId (MQCHAR4)

Yapı tanıtıcısı.

Değer:

#### MQZFP\_STRUC\_ID

Serbest parametreler yapısına ilişkin tanıtıcı.

C programlama dili için, sabit MQZFP\_STRUC\_ID\_ARRAY değişmezi de tanımlanır; bu, MQZFP\_STRUC\_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisi olur.

Bu, hizmete giriş alanıdır.

### Sürüm (MQUZE)

Yapı sürüm numarası.

Değer:

#### MQZFP\_VERSION\_1

Version-1 boş parametreler yapısı.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

#### MQZFP\_CURRENT\_VERSION

Serbest parametre yapısının geçerli sürümü.

Bu, hizmete giriş alanıdır.

### Reserved (Ayrılmış) (MQBYTE8)

Ayrılmış alan.

Başlangıç değeri boş değerli.

### CorrelationPtr (MQPTR)

İlinti göstergesi.

Serbest bırakılacak kaynakla ilgili ilinti verilerinin adresi.

## C bildirim

```
typedef struct tagMQZFP MQZFP;
```

```

struct tagMQZFP {
    MQCHAR4  StrucId;          /* Structure identifier */
    MQLONG   Version;         /* Structure version number */
    MQBYTE8  Reserved;       /* Reserved field */
    MQPTR    CorrelationPtr;  /* Address of correlation data */
};

```

## IBM i IBM üzerinde MQZIC (Kimlik bağlamı)

MQZIC yapısı, **IdentityContext** parametresine ilişkin MQZ\_AUTHENTICATE\_USER çağrısında kullanılır.

MQZIC yapısı, iletiyi ilk olarak bir kuyruğa koyan uygulamanın kullanıcıını tanımlayan kimlik bağlamı bilgilerini içerir:

- Kuyruk yöneticisi, UserIdentifier alanını kullanıcıyı tanıtan bir adla doldurur; kuyruk yöneticisinin bunu yapabilme biçimi, uygulamanın çalışmakta olduğu ortama bağlıdır.
- Kuyruk yöneticisi, AccountingToken alanını, iletiyi koyan uygulamadan saptadığı bir simgeyle ya da numarasıyla doldurur.
- Uygulamalar, kullanıcı hakkında içermek istedikleri ek bilgiler (örneğin, şifrelenmiş bir parola) için ApplIdentityVeri alanını kullanabilir.

Uygun olarak yetkili uygulamalar, MQZ\_AUTHENTICATE\_USER işlevini kullanarak kimlik bağlamını ayarlayabilir.

IBM MQ for Windows altında bir ileti oluşturulduğunda, bir Windows sistem güvenlik tanıtıcısı (SID) AccountingToken alanında saklanır. SID, UserIdentifier (Kullanıcı Kimliği) alanını tamamlamak ve bir kullanıcının kimlik bilgilerini oluşturmak için kullanılabilir.

### Alanlar

#### StrucId (MQCHAR4)

Yapı tanıtıcısı.

Değer:

#### MQZIC\_STRUC\_ID

Kimlik bağlamı yapısına ilişkin tanıtıcı.

C programlama dili için, MQZIC\_STRUC\_ID\_ARRAY değişmezi de tanımlanır; bu, MQZIC\_STRUC\_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisidir.

Bu, hizmete giriş alanıdır.

#### Sürüm (MQUZE)

Yapı sürüm numarası.

Değer:

#### MQZIC\_VERSION\_1

Version-1 kimlik bağlamı yapısı.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

#### MQZIC\_CURRENT\_VERSION

Kimlik bağlamı yapısının yürürlükteki sürümü.

Bu, hizmete giriş alanıdır.

#### UserIdentifier (MQCHAR12)

Kullanıcı kimliği.

Bu, iletinin **kimlik bağlamının** bir parçasıdır.

*UserIdentifier*, iletiyi oluşturan uygulamanın kullanıcı tanımlayıcısını belirtir. Kuyruk yöneticisi bu bilgileri karakter verisi olarak değerlendirir, ancak biçiminin biçimini tanımlamaz. *UserIdentifier* alanıyla ilgili daha fazla bilgi için bkz. [“UserIdentifier \(MQCHAR12\)” sayfa 451.](#)

## AccountingToken (MQBYTE32)

Muhasebe simgesi.

Bu, iletinin **kimlik bağlamının** bir parçasıdır.

*AccountingToken*, uygulamanın uygun şekilde ücretlendirilmesine neden olarak bir uygulamanın çalışmasına neden olur. Kuyruk yöneticisi bu bilgileri bir bit dizgisi olarak değerlendirir ve içeriğini denetmez. *AccountingToken* alanıyla ilgili daha fazla bilgi için bkz. [“AccountingToken \(MQBYTE32\)” sayfa 452.](#)

## ApplIdentityVerileri (MQCHAR32)

Kimlikle ilgili uygulama verileri.

Bu, iletinin **kimlik bağlamının** bir parçasıdır.

*ApplIdentityData*, iletinin kökeniyle ilgili ek bilgi sağlamak için kullanılacak uygulama takımı tarafından tanımlanan bilgilerdir. Örneğin, kimlik verilerinin güvenilir olup olmadığını göstermek için uygun kullanıcı yetkisiyle çalışan uygulamalar tarafından ayarlanabilir. *ApplIdentityData* alanıyla ilgili daha fazla bilgi için bkz. [“ApplIdentityVerileri \(MQCHAR32\)” sayfa 454.](#)

## C bildirimini

```
typedef struct tagMQZED MQZED;
struct tagMQZED {
    MQCHAR4    StrucId;           /* Structure identifier */
    MQLONG     Version;          /* Structure version number */
    MQCHAR12   UserIdentifier;    /* User identifier */
    MQBYTE32   AccountingToken;  /* Accounting token */
    MQCHAR32   ApplIdentityData; /* Application data relating to identity */
};
```

## IBM MQ .NET sınıfları ve arabirimleri

IBM MQ .NET sınıfları ve arabirimleri alfabetik olarak sıralanır. Özellikler, yöntemler ve oluşturucular açıklanır.

### MQAsyncStatus.NET sınıfı

Önceki MQI etkinliğinin durumunu sorgulamak için *MQAsyncStatus* 'u (örneğin, önceki zamanuyumsuz koyma işlemlerinin başarısına ilişkin) sorgulamak için kullanın. *MQAsyncStatus*, *MQSTS* veri yapısının özelliklerini sarsalar.

### Sınıf

```
System.Object
├── IBM.WMQ.MQBase
│   └── IBM.WMQ.MQBaseObject
│       └── IBM.WMQ.MQAsyncStatus
```

```
public class IBM.WMQ.MQAsyncStatus extends IBM.WMQ.MQBaseObject;
```

- [“Özellikler” sayfa 1697](#)
- [“Oluşturucular” sayfa 1698](#)

### Özellikler

Özellikler alınırken *MQException* için test ediliyor.

```
public static int CompCode {get;}
```

İlk hata ya da uyarıdan tamamlanma kodu.

```
public static int Reason {get;}
```

İlk hata ya da uyarıdan neden kodu.

```
public static int PutSuccessCount {get;}
```

Çağrılan zamanuyumsuz MQI ' nin başarılı sayısı.

```
public static int PutWarningCount {get;}
```

Bir uyarıyla başarılı olan zamanuyumsuz MQI çağrılarının sayısı.

```
public static int PutFailureCount {get;}
```

Çağrılan zamanuyumsuz MQI çağrılarının sayısı.

```
public static int ObjectType {get;}
```

İlk hata için nesne tipi. Aşağıdaki değerler olanaklı:

- MQC.MQOT\_ALIAS\_Q
- MQC.MQOT\_LOCAL\_Q
- MQC.MQOT\_MODEL\_Q
- MQC.MQOT\_Q
- MQC.MQOT\_REMOTE\_Q
- MQC.MQOT\_TOPIC
- 0, herhangi bir nesnenin döndürülmediği anlamına gelir

```
public static string ObjectName {get;}
```

Nesne adı.

```
public static string ObjectQMgrName {get;}
```

Nesne kuyruğu yöneticisi adı.

```
public static string ResolvedObjectName {get;}
```

Çözümlenen nesne adı.

```
public static string ResolvedObjectQMgrName {get;}
```

Çözümlenen nesne kuyruğu yöneticisi adı.

## Oluşturucular

```
public MQAsyncStatus() throws MQException;
```

Oluşturucu yöntemi, alanları sıfır ya da boş olarak başlatılan bir nesneyi uygun şekilde yapılandırır.

## MQAuthenticationInformationRecord.NET sınıfı

Bir IBM MQ TLS istemci bağlantısında kullanılacak kimlik doğrulayıcıya ilişkin bilgileri belirtmek için MQAuthenticationInformationRecord değerini kullanın. MQAuthenticationInformationRecord bir kimlik doğrulama bilgisi kaydını kapsüller, MQAIR.

## Sınıf

```
System.Object  
└─ IBM.WMQ.MQAuthenticationInformationRecord
```

```
public class IBM.WMQ.MQAuthenticationInformationRecord extends System.Object;
```

- [“Özellikler” sayfa 1699](#)
- [“Oluşturucular” sayfa 1699](#)

## Özellikler

Özellikler alınırken MQException için test ediliyor.

```
public long Version {get; set;}
```

Yapı sürüm numarası.

```
public long AuthInfoType {get; set;}
```

Kimlik doğrulama bilgilerinin tipi. Bu öznitelik aşağıdaki değerlerden birine ayarlanmalıdır:

- OCSP -Sertifika iptal durumu denetimi OCSP kullanılarak yapılır.
- CRLLDAP -Sertifika iptal durumu denetimi, LDAP sunucularında Sertifika İptal Listeleri kullanılarak yapılır.

```
public string AuthInfoConnName {get; set;}
```

LDAP sunucusunun çalışmakta olduğu anasistemin DNS adı ya da IP adresi, isteğe bağlı bir kapı numarasıyla birlikte. Bu anahtar sözcük gereklidir.

```
public string LDAPPassword {get; set;}
```

LDAP sunucusuna erişen kullanıcının ayırt edici adı ile ilişkilendirilmiş parola. Bu özellik, **AuthInfoType** değeri CRLLDAPolarak ayarlandığında geçerlidir.

```
public string LDAPUserName {get; set;}
```

LDAP sunucusuna erişmekte olan kullanıcının ayırt edici adı. Bu özelliği ayarladığınızda, LDAPUserNameLength ve LDAPUserNamePtr otomatik olarak doğru olarak ayarlanır. Bu özellik yalnızca AuthInfoType değeri CRLLDAPolarak ayarlandığında geçerlidir.

```
public string OCSPResponderURL {get; set;}
```

OCSP yanıtlayıcının iletişim kurabileceği URL. Bu özellik yalnızca AuthInfoType değeri OCSPolarak ayarlandığında geçerlidir.

Bu alan büyük ve küçük harfe duyarlıdır. It must start with the string http:// in lowercase. URL ' nin geri kalanı, OCSP sunucusu uygulamasına bağlı olarak büyük ve küçük harfe duyarlı olabilir.

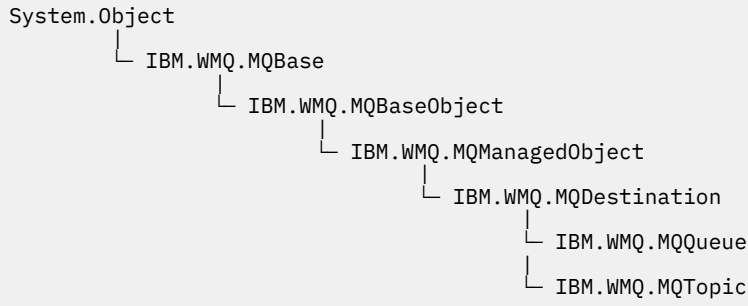
## Oluşturucular

```
MQAuthenticationInformationRecord();
```

## MQDestination.NET sınıfı

Use MQDestination to access methods that are common to MQQueue and MQTopic. MQDestination , soyut bir temel sınıftır ve somutlaştırılmaz.

## Sınıf



```
public class IBM.WMQ.MQDestination extends IBM.WMQ.MQManagedObject;
```

- “Özellikler” sayfa 1700
- “Yöntemler” sayfa 1700
- “Oluşturucular” sayfa 1702

## Özellikler

Özellikler alınırken MQException için test ediliyor.

```
public DateTime CreationDateTime {get;}
```

Kuyruğun ya da konunun yaratıldığı tarih ve saat. Başlangıçta MQQueue içinde yer alan bu özellik, temel MQDestination sınıfına taşınmıştır.

Varsayılan değer yoktur.

```
public int DestinationType {get;}
```

Kullanılmakta olan hedef tipini tanımlayan tamsayı değeri. Bu değer, alt sınıflar oluşturucusundan (MQQueue ya da MQTopic) kullanıma hazırlandı; bu değer şu değerlerden birini alabilir:

- MQOT\_Q
- MQOT\_TOPIC

Varsayılan değer yoktur.

## Yöntemler

```
public void Get(MQMessage message);
public void Get(MQMessage message, MQGetMessageOptions getMessageOptions);
public void Get(MQMessage message, MQGetMessageOptions getMessageOptions, int MaxMsgSize);
```

MQException yayınlıyor.

Hedef bir MQQueue nesnesiyse, kuyruktan ileti alır; hedef MQTopic nesnesiyse, alma işlemi için varsayılan bir MQGetMessageOptions yönetim ortamı kullanılarak bir ileti alır.

Alma işlemi başarısız olursa, MQMessage nesnesi değişmeden kalır. Başarılı olursa, MQMessage 'un ileti tanımlayıcısı ve ileti verileri kısımları, gelen iletiden ileti tanımlayıcısı ve ileti verileriyle değiştirilir.

Belirli bir MQQueueManager 'den IBM MQ 'e yapılan tüm çağrılar zamanuyumlu olur. Bu nedenle, bir bekleme işlemi gerçekleştirdiğinizde, aynı MQQueueManager 'u kullanan diğer tüm iş parçacıklarının, Alma işlemi gerçekleştirilinceye kadar IBM MQ çağrılarının yapılması engellenir. IBM MQ 'a aynı anda erişmek için birden çok iş parçacığının olması gerekiyorsa, her bir iş parçacığının kendi MQQueueManager nesnesi yaratması gerekir.

### ileti

İleti tanımlayıcısını ve döndürülen ileti verilerini içerir. İleti tanımlayıcısındaki bazı alanlar giriş değiştirgeleri olur. MessageId ve CorrelationId giriş parametrelerinin gerektiği şekilde ayarlandığından emin olmak önemlidir.



Yeniden bağlantılanabilir istemci, MQGM\_SYNCPOINT altında alınan iletiler için, başarılı yeniden bağlantı sonrasında MQRC\_BACKED\_OUT neden kodunu döndürür.

### **getMessageSeçenekleri**

Alma işleminin denetlenmesini denetleyen seçenekler.

MQC.MQGMO\_CONVERT seçeneğinin kullanılması, tek byte 'lık karakter kodlarından çift byte kodlarına dönüştürülürken MQC.MQRC\_CONVERTED\_STRING\_TOO\_BIG neden koduyla bir kural dışı duruma neden olabilir. Bu durumda, ileti dönüştürmeden arabelleğe kopyalanır.

*getMessageOptions* belirtilmemişse, kullanılan ileti seçeneği MQGMO\_NOWAIT olur.

Yeniden bağlanabilir bir istemcide MQGMO\_LOGICAL\_ORDER seçeneğini kullanırsanız, MQRC\_RECONNECT\_INCOMPATIBLE neden kodu döndürülür.

### **MaxMsgBoyutu**

Bu ileti nesnesinin alınabileceği en büyük ileti. Kuyruktaki ileti bu boyuttan büyükse, iki şeyden biri oluşur:

- If the MQGMO\_ACCEPT\_TRUNCATED\_MSG flag is set in the MQGetMessageOptions object, the message is filled with as much of the message data as possible. MQCC\_WARNING tamamlanma kodu ve MQRC\_TRUNCATED\_MSG\_ACCEPTED neden koduyla bir kural dışı durum yayınlandı.
- MQGMO\_ACCEPT\_TRUNCATED\_MSG işareti ayarlanmamış ise, ileti kuyruktaki kalır. MQCC\_WARNING tamamlanma kodu ve MQRC\_TRUNCATED\_MSG\_FAILED neden koduyla bir kural dışı durum yayınlandı.

*MaxMsgSize* belirtilmemişse, tüm ileti alınır.

```
public void Put(MQMessage message);  
public void Put(MQMessage message, MQPutMessageOptions putMessageOptions);
```

MQException yayınlıyor.

Hedef bir MQQueue nesnesiyse, bir kuyruğa ileti koyar ya da hedef bir MQTopic nesnesiyse, bir konuya ileti yayınlar.

Put çağrısından sonra MQMessage nesnesinde yapılan değişiklikler, IBM MQ kuyruğunda ya da yayınındaki gerçek iletiyi etkilemez.

Put, MQMessage nesnesinin MessageId ve CorrelationId özelliklerini güncelleştirir ve ileti verilerini temizlememektedir. Daha fazla Put ya da Get çağrısı, MQMessage nesnesindeki güncellenmiş bilgilere gönderme yapıyor. Örneğin, aşağıdaki kod parçasısında ilk ileti a ve ikinci abiletisini içerir.

```
msg.WriteString("a");  
q.Put(msg, pmo);  
msg.WriteString("b");  
q.Put(msg, pmo);
```

### **ileti**

İleti tanımlayıcı verilerini içeren bir MQMessage nesnesi ve gönderilecek ileti. İleti tanımlayıcısı bu yöntemin bir sonucu olarak değiştirilebilir. Bu yöntemin tamamlanmasından hemen sonra ileti tanımlayıcısındaki değerler, kuyruğa konulan ya da konu üzerinde yayınlanan değerlerdir.

Yeniden bağlanabilir bir istemciye döndürülen neden kodları şunlardır:

- MQRC\_CALL\_INTERRUPTED if the connection is broken while running a Put call on a persistent message and the reconnection is successful.
- Bağlantı, kalıcı olmayan bir iletiyle (bkz. [Uygulama Kurtarma](#)) bir put çağrısı çalıştırırken bağlantı başarılı olursa MQRC\_NONE.

### **putMessageSeçenekleri**

Put işleminin denetlenmesini denetleyen seçenekler.

*putMessageOptions* belirtilmemişse, varsayılan MQPutMessageOptions eşgörünümü kullanılır.

Yeniden bağlanabilir bir istemcide MQPMO\_LOGICAL\_ORDER seçeneğini kullanırsanız, MQRC\_RECONNECT\_INCOMPATIBLE neden kodu döndürülür.

**Not:** Basitlik ve performans için, kuyruğa tek bir ileti koymak isterseniz, MQQueueManager.Put nesnesini kullanın. Bunun için bir MQQueue nesnesine sahip olmanız.

## Oluşturucular

MQDestination, soyut bir temel sınıftır ve somutlaştırılmaz. Access destinations using MQQueue and MQTopic constructors, or using MQQueueManager.AccessQueue and MQQueueManager.AccessTopic methods.

## MQEnvironment.NET sınıfı

MQQueueManager oluşturucusunun nasıl çağrıldığı ve bir IBM MQ MQI client bağlantısı seçmesi için MQEnvironment'u kullanın. MQEnvironment sınıfı, IBM MQ' in davranışını denetleyen özellikler içerir.

### Sınıf

```
System.Object
├── IBM.WMQ.MQEnvironment
```

```
public class IBM.WMQ.MQEnvironment extends System.Object;
```

- [“Özellikler-yalnızca istemci” sayfa 1702](#)
- [“Özellikler” sayfa 1703](#)
- [“Oluşturucular” sayfa 1704](#)

### Özellikler-yalnızca istemci

Özellikler alınırken MQException için test ediliyor.

```
public static int CertificateValPolicy {get; set;}
```

Uzak iş ortağı sistemlerinden alınan sayısal sertifikaların geçerliliğini denetlemek için hangi TLS sertifikası geçerlilik denetimi ilkesinin kullanılacağını belirleyin. Geçerli değerler şunlardır:

- MQC.CERTIFICATE\_VALIDATION\_POLICY\_ANY
- MQC.CERTIFICATE\_VALIDATION\_POLICY\_RFC5280

```
public static ArrayList EncryptionPolicySuiteB {get; set;}
```

Takım B uyumlu şifreleme düzeyini ayarlayın. Geçerli değerler şunlardır:

- MQC.MQ\_SUITE\_B\_NONE -Varsayılan değer budur.
- MQC.MQ\_SUITE\_B\_128\_BIT
- MQC.MQ\_SUITE\_B\_192\_BIT

```
public static string Channel {get; set;}
```

Hedef kuyruk yöneticisine bağlanmak için kullanılan kanalın adı. Kanal özelliğini, istemci kipinde bir MQQueueManager yönetim ortamını somutlaştıramadan önce ayarlamanız gerekir.

```
public static int FipsRequired {get; set;}
```

Şifreleme IBM MQ'inde gerçekleştiriliyorsa, yalnızca FIPS onaylı algoritmaları kullanmak için MQC.MQSSL\_FIPS\_YES değerini belirtin. Varsayılan değer MQC.MQSSL\_FIPS\_NO' dir.

Şifreleme donanımı yapılandırıldıysa, kullanılan şifreleme modülleri, donanım ürünü tarafından sağlanır. Kullanıştaki donanıma bağlı olarak, bunlar belirli bir düzey için FIPS onaylı olmayabilir.

**public static string Hostname {get; set;}**

IBM MQ sunucusunun bulunduğu bilgisayarın TCP/IP anasistem adı. Anasistem adı belirlenmezse ve geçersiz kılan özellikler belirlenmezse, sunucu bağ tanımları kipi yerel kuyruk yöneticisine bağlanmak için kullanılır.

**public static int Port {get; set;}**

Bağlanılacak kapı. Bu, IBM MQ sunucusunun gelen bağlantı isteklerini dinlediği kapıdır. Varsayılan değer 1414 'tür.

**public static string SSLCipherSpec {get; set;}**

Bağlantı için TLS ' yi etkinleştirmek üzere SVRCONN kanalındaki CipherSpec kümesinin değerine SSLCipherSpec değerini ayarlayın. Varsayılan değer Boş Değerdir ve TLS bağlantı için etkinleştirilmez.

**public static string sslPeerName {get; set;}**

Ayırt edici bir ad örünü. sslCipherSpec ayarlandıysa, bu değişken, doğru kuyruk yöneticisinin kullanıldığından emin olmak için kullanılabilir. Boş değere ayarlanırsa (varsayılan), kuyruk yöneticisinin ayırt edici adı (DN) gerçekleştirilmez. sslCipherSpec boş değerliyse,sslPeerName yoksayılır.

## Özellikler

Özellikler alınırken MQException için test ediliyor.

**public static ArrayList HdrCompList {get; set;}**

Üstbilgi Verileri Sıkıştırma Listesi

**public static int KeyResetCount {get; set;}**

Gizli anahtar yeniden anlaşılmadan önce bir TLS etkileşimi içinde gönderilen ve alınan şifrelenmemiş baytların sayısını belirtir.

**public static ArrayList MQAIRArray {get; set;}**

MQAuthenticationInformationRecord nesnelere dizisi.

**public static ArrayList MsgCompList {get; set;}**

İleti Verileri Sıkıştırma Listesi

**public static string Password {get; set;}**

Kimliği doğrulanacak parola. Bu Parola özelliği ayarlanarak, MQCSP yapısından gönderme yapılan parolaya veri yerleştirilir.

**public static string ReceiveExit {get; set;}**

Alma çıkışı, bir kuyruk yöneticisinden alınan verileri incelemenizi ve değiştirmenize olanak sağlar. Genellikle kuyruk yöneticisinde karşılık gelen bir gönderme çıkışıyla birlikte kullanılır. ReceiveExit boş değere ayarlıysa, alma çıkışı çağrılır.

**public static string ReceiveUserData {get; set;}**

Alma çıkışıyla ilişkili kullanıcı verileri. 32 karakterle sınırlıdır.

**public static string SecurityExit {get; set;}**

Bir güvenlik çıkışı, bir kuyruk yöneticisine bağlanma girişiminde bulunulduğunda oluşan güvenlik akışlarını uyarlamana olanak sağlar. SecurityExit boş değer olarak ayarlanırsa, güvenlik çıkışı çağrılmaz.

**public static string SecurityUserData {get; set;}**

Bir güvenlik çıkışıyla ilişkili kullanıcı verileri. 32 karakterle sınırlıdır.

**public static string SendExit {get; set;}**

Çıkış gönderme işlemi, kuyruk yöneticisine gönderilen verileri incelemenizi ya da değiştirmenizi sağlar. Genellikle kuyruk yöneticisinde karşılık gelen bir alma çıkışıyla kullanılır. SendExit boş değer olarak ayarlanırsa, gönderme çıkışı çağrılır.

**public static string SendUserData {get; set;}**

Bir gönderme çıkışıyla ilişkili kullanıcı verileri. 32 karakterle sınırlıdır.

## **public static string SharingConversations {get; set;}**

The SharingConversations field is used on connections from .NET applications, when these applications are not using a client channel definition table (CCDT).

SharingConversations , bu bağlantıyla ilişkili bir yuvada paylaşılacak etkileşim sayısı üst sınırını belirler.

0 değeri, kanalın etkileşim paylaşımı, okuma ve sağlıklı işletim bildirim bakımından IBM WebSphere MQ 7.0' dan önce olduğu gibi çalıştığı anlamına gelir.

The field is passed in the hash table of properties as a SHARING\_CONVERSATIONS\_PROPERTY, when instantiating an IBM MQ queue manager.

SharingConversationsseçeneğini belirlemezseniz, varsayılan değer olan 10 kullanılır.

## **public static string SSLCryptoHardware {get; set;}**

Sistemde bulunan şifreleme donanımını yapılandırmak için gereken değiştirge dizgisinin adını belirler. sslCipherSpec boş değerliyse,SSLCryptoHardware yoksayılır.

## **public static string SSLKeyRepository {get; set;}**

Anahtar havuzunun tam olarak nitelenmiş dosya adını belirleyin.

SSLKeyRepository boş değer (varsayılan) olarak ayarlanırsa, anahtar havuzunu bulmak için sertifika MQSSLKEYR ortam değişkeni kullanılır. sslCipherSpec boş değerliyse,SSLCryptoHardware yoksayılır.

**Not:** .kdb uzantısı, dosya adının zorunlu bir parçasıdır, ancak parametrenin değerinin bir parçası olarak içerilmez. Belirttiğiniz izin var olmalıdır. IBM MQ dosyası, yeni anahtar havuzuna ilk kez eriştiği sürece, dosya önceden vardiysa, bu dosyayı yaratır.

## **public static string UserId {get; set;}**

Kimliği doğrulanacak kullanıcı kimliği. MQCSP yapısından gönderme yapılan kullanıcı kimliği, UserIdayarlanarak doldurulur. Bir API ya da Güvenlik çıkışı kullanarak UserId kimliğini doğrulayın.

## **Oluşturucular**

**public MQEnvironment()**

## **MQException.NET sınıfı**

Başarısız olan bir IBM MQ işlevinin tamamlanma ve neden kodunu öğrenmek için MQException kullanın. Bir IBM MQ hatası ortaya çıktığında MQException yayınlanır.

## **Sınıf**

```
System.Object
├── System.Exception
│   └── System.ApplicationException
│       └── IBM.WMQ.MQException
```

**public class IBM.WMQ.MQException extends System.ApplicationException;**

- [“Özellikler” sayfa 1705](#)
- [“Oluşturucular” sayfa 1705](#)

## Özellikler

**public int CompletionCode {get; set;}**

Hatayla ilişkili IBM MQ tamamlanma kodu. Olası değerler şunlardır:

- MQException.MQCC\_OK
- MQException.MQCC\_WARNING
- MQException.MQCC\_FAILED

**public int ReasonCode {get; set;}**

IBM MQ hata kodunu açıklayan neden kodu.

## Oluşturucular

**public MQException(int completionCode, int reasonCode)**

**completionCode**

IBM MQ tamamlanma kodu.

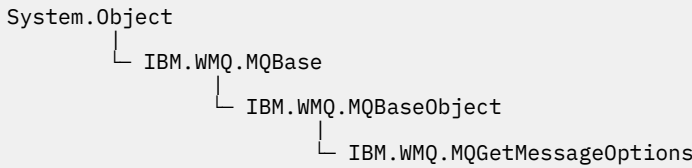
**reasonCode**

IBM MQ tamamlanma kodu.

## MQGetMessageOptions.NET sınıfı

İletilerin nasıl alınacağını belirtmek için MQGetMessageOptions seçeneğini kullanın. MQDestination.Getdavranışını değiştirir.

## Sınıf



```
public class IBM.WMQ.MQGetMessageOptions extends IBM.WMQ.MQBaseObject;
```

- [“Özellikler” sayfa 1705](#)
- [“Oluşturucular” sayfa 1708](#)

## Özellikler

**Not:** Bu sınıftaki bazı seçeneklerin davranışı, kullanıcıların bulunduğu ortama bağlıdır. Bu öğeler yıldız işaretiyle (\*) işaretlenir.

Özellikler alınırken MQException için test ediliyor.

**public int GroupStatus {get;}\***

GroupStatus , alınan iletinin bir grupta olup olmadığını ve gruptaki son iletinin sonuncu olup olmadığını belirtir. Olası değerler şunlardır:

**MQC.MQGS\_LAST\_MSG\_IN\_GROUP**

İleti, gruptaki son ya da tek iletidir.

**MQC.MQGS\_MSG\_IN\_GROUP**

İleti bir grup içinde, ancak gruptaki son kişi değil.

**MQC.MQGS\_NOT\_IN\_GROUP**

İleti bir grupta yer almıyor.

**public int MatchOptions {get; set;}\***

MatchOptions , bir iletinin nasıl seçileceğini belirler. Aşağıdaki eşleşme seçenekleri ayarlanabiliyor:

**MQC.MQMO\_MATCH\_CORREL\_ID**

Eşleştirilecek ilinti tanıtıcısı.

**MQC.MQMO\_MATCH\_GROUP\_ID**

Eşleştirilecek grup tanıtıcısı.

**MQC.MQMO\_MATCH\_MSG\_ID**

Eşleştirilecek ileti tanıtıcısı.

**MQC.MQMO\_MATCH\_MSG\_SEQ\_NUMBER**

İleti sıra numarasını eşleştir.

**MQC.MQMO\_NONE**

Eşleşen herhangi bir eşleşme yok.

**public int Options {get; set;}**

Seçenekler , MQQueue .getişlemini denetler. Aşağıdaki değerlerden herhangi biri belirlenebilir. Birden fazla seçenek gerekliyse, değerler eklenebilir ya da bit ya da işleç kullanılarak birleştirilebilir.

**MQC.MQMO\_ACCEPT\_TRUNCATED\_MSG**

İleti verilerinin kesilmesine izin verir.

**MQC.MQMO\_ALL\_MSGS\_AVAILABLE\***

Yalnızca gruptaki tüm iletiler kullanılabilir olduğunda, bir gruptan ileti alın.

**MQC.MQMO\_ALL\_SEGMENTS\_AVAILABLE\***

Mantıksal iletinin bölümlerini, yalnızca gruptaki tüm kesimler kullanılabilir olduğunda alın.

**MQC.MQMO\_BROWSE\_FIRST**

Kuyruğun başlangıcından göz atın.

**MQC.MQMO\_BROWSE\_MSG\_UNDER\_CURSOR\***

Göz atma imleci altındaki iletiye göz atın.

**MQC.MQMO\_BROWSE\_NEXT**

Kuyrukta yürürlükteki konumdan sisteme göz atın.

**MQC.MQMO\_COMPLETE\_MSG\***

Yalnızca tam mantıksal iletileri alın.

**MQC.MQMO\_CONVERT**

Request the application data to be converted, to conform to the CharSet and Kodlama attributes of the MQMessage, before the data is copied into the message buffer. Veri dönüştürme de, veriler ileti arabelleğinden alındığında de uygulanır; uygulamalar bu seçeneği ayarlamaz.

Bu seçeneğin kullanılması, tek baytlık karakter kümelerinden çift baytlı karakter takımlarına dönüştürülürken sorunlara neden olabilir. Bunun yerine, ileti teslim edildikten sonra readString, readLineve writeString yöntemlerini kullanarak dönüştürmeyi gerçekleştirin.

**MQC.MQMO\_FAIL\_IF QUIESCING**

Kuyruk yöneticisi susturulmuş durumda olduğunda hata oluştu.

**MQC.MQMO\_LOCK\***

Göz atıldığı iletiyi kilitleyin.

**MQC.MQMO\_LOGICAL\_ORDER\***

İletileri gruplar halinde ve mantıksal sırayla, mantıksal sırayla döndürür.

Yeniden bağlanabilir bir istemcide MQMO\_LOGICAL\_ORDER seçeneğini kullanırsanız, uygulamaya MQRC\_RECONNECT\_INCOMPATIBLE neden kodu döndürülür.

**MQC.MQMO\_MARK\_SKIP\_BACKOUT\***

Kuyruktaki iletiyi yeniden yürürlüğe koymadan, bir iş biriminin geriletilmesine izin verir.

**MQC.MQMO\_MSG\_UNDER\_CURSOR**

Göz atma imlecinin altına ileti al.

**MQC.MQMO\_NONE**

Başka bir seçenek belirtilmedi; tüm seçenekler varsayılan değerlerini kabul eder.

**MQC.MQMO\_NO\_PROPERTIES**

İleti tanımlayıcısında (ya da uzantıda) bulunan özellikler dışında, iletinin hiçbir özelliği alınmadı.

**MQC.MQGMO\_NO\_SYNCPOINT**

Eşitleme noktası denetimi olmadan ileti al.

**MQC.MQGMO\_NO\_WAIT**

Uygun bir ileti yoksa hemen geri dönün.

**MQC.MQGMO\_PROPERTIES\_AS\_Q\_DEF**

Retrieve message properties as defined by the PropertyControl attribute of MQQueue. İleti tanımlayıcısındaki ya da uzantıdaki ileti özelliklerine erişim, PropertyControl özneliğinden etkilenmez.

**MQC.MQGMO\_PROPERTIES\_COMPATIBILITY**

Retrieve message properties with a prefix of mcd, jms, usr, or mqext, in MQRFH2 headers. İletinin diğer özellikleri (ileti tanımlayıcısı ya da uzantıda bulunan özellikler dışında) atılır.

**MQC.MQGMO\_PROPERTIES\_FORCE\_MQRFH2**

MQRFH2 üstbilgilerinde, ileti tanımlayıcısında ya da uzantıda bulunan özellikler dışında ileti özelliklerini alır. Özellikleri almayı bekleyen, ancak ileti tanıtıcılarını kullanmak üzere değiştiremeyecek uygulamalarda MQC.MQGMO\_PROPERTIES\_FORCE\_MQRFH2 kullanın.

**MQC.MQGMO\_PROPERTIES\_IN\_HANDLE**

Retrieve message properties using a MsgHandle.

**MQC.MQGMO\_SYNCPOINT**

İletiyi eşitleme noktası denetimi altında alın. İleti, diğer uygulamalar tarafından kullanılamaz olarak işaretlenir, ancak iş birimi kesinleştirildiğinde kuyruktan silinir. İş birimi geriletilirse, ileti yeniden kullanılabilir kılındır.

**MQC.MQGMO\_SYNCPOINT\_IF\_PERSISTENT\***

İleti kalıcıysa, eşitleme noktası denetimi ile ileti alın.

**MQC.MQGMO\_UNLOCK\***

Önceden kilitlemiş bir iletinin kilidini açın.

**MQC.MQGMO\_WAIT**

İletinin gelmesini bekleyin.

**public string ResolvedQueueName {get;}**

Kuyruk yöneticisi, ResolvedQueueAd ' un iletinin alındığı kuyruğun yerel adına göre ayarlandığını ayarlar. ResolvedQueueAd1 , bir diğer ad kuyruğu ya da model kuyruğu açıldıysa, kuyruğu açmak için kullanılan addan farklıdır.

**public char Segmentation {get;}\***

Kesimlere ayırma , alınan ileti için bölümlemeye izin verilip verilmeyeceğini belirtir. Olası değerler şunlardır:

**MQC.MQSEG\_INHIBITED**

Bölümlemeye izin verme.

**MQC.MQSEG\_ALLOWED**

Kesimlere ayırma

**public byte SegmentStatus {get;}\***

SegmentStatus , alınan iletinin bir mantıksal iletinin parçası olup olmadığını gösteren bir çıkış alanıdır. İleti bir bölümse, bu işaret son bölüm olup olmadığını belirtir. Olası değerler şunlardır:

**MQC.MQSS\_LAST\_SEGMENT**

İleti, mantıksal iletinin son ya da tek bölüğüdür.

**MQC.MQSS\_NOT\_A\_SEGMENT**

İleti bir bölüm değil.

**MQC.MQSS\_SEGMENT**

İleti bir bölümdür, ancak mantıksal iletinin son bölümü değildir.

**public int WaitInterval {get; set;}**

WaitInterval , bir MQQueue . get çağrısının uygun bir ileti gelmesi için bekleyeceği sürenin üst sınısıdır. MQC.MQGMO\_WAIT ile WaitInterval seçeneğini kullanın. Bir ileti için sınırsız süre beklemek üzere MQC.MQWI\_UNLIMITED değerini ayarlayın.

## Olusturucular

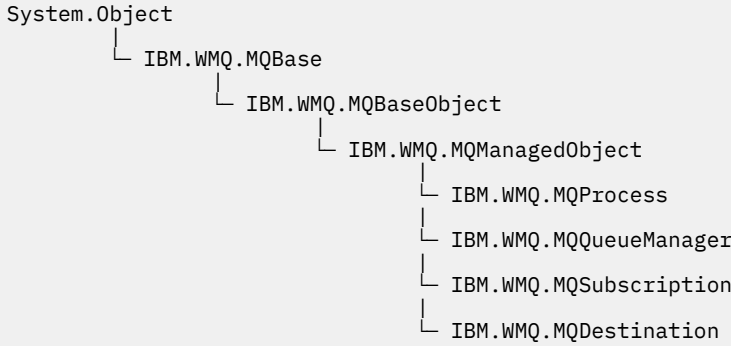
### public MQGetMessageOptions()

Seçenekler ile MQC.MQGMO\_NO\_WAIT olarak ayarlanmış yeni bir MQGetMessageOptions nesnesi oluşturun, WaitInterval sıfır olarak ayarlanır ve ResolvedQueueAd boş olarak ayarlanır.

## MQManagedObject.NET sınıfı

MQDestination, MQProcess, MQQueueManager ve MQSubscription özelliklerini sorgulamak ve ayarlamak için MQManagedObject kullanın. MQManagedObject, bu sınıfların üst sınıfıdır.

## Sınıflar



```
public class IBM.WMQ.MQManagedObject extends IBM.WMQ.MQBaseObject;
```

- [“Özellikler” sayfa 1708](#)
- [“Yöntemler” sayfa 1709](#)
- [“Olusturucular” sayfa 1710](#)

## Özellikler

Özellikler alınırken MQException için test ediliyor.

### public string AlternateUserId {get; set;}

Varsa, diğer kullanıcı kimliği (varsa) kaynak açıldıysa ayarlanır. AlternateUserID.set is ignored when issued for an object that is opened. AlternateUserTnt abonelikler için geçerli değil.

### public int CloseOptions {get; set;}

Kaynağın kapanma şeklini denetlemek için bu özneliği ayarlayın. Varsayılan değer MQC.MQCO\_NONE değeridir. Kalıcı dinamik kuyruklar, geçici dinamik kuyruklar, abonelikler ve bunları yaratan nesnelere tarafından erişilmekte olan konular dışındaki tüm kaynaklar için izin verilebilir tek değer MQC.MQCO\_NONE ' dir.

Kuyruklar ve konular için aşağıdaki ek değerlere izin verilebilir:

#### MQC.MQCO\_DELETE

İleti yoksa, kuyruğu silin.

#### MQC.MQCO\_DELETE\_PURGE

Kuyruğun silinmesi, üzerindeki iletilerin temizlenmesi.

#### MQC.MQCO\_QUIESCE

Herhangi bir ileti kaldıysa (son kapanmadan önce alınmalarına izin veren) bir uyarı almak için kuyruk kapatılmasını isteyin.

Abonelikler için aşağıdaki ek değerlere izin verilebilir:



**MQC.MQCO\_KEEP\_SUB**

Abonelik silinmez. Bu seçenek yalnızca özgün abonelik dayanıklı olduğunda geçerlidir. MQC.MQCO\_KEEP\_SUB , kalıcı bir konu için varsayılan değerdir.

**MQC.MQCO\_REMOVE\_SUB**

Abonelik silinir. MQC.MQCO\_REMOVE\_SUB , kalıcı olmayan, yönetilmeyen bir konu için varsayılan değerdir.

**MQC.MQCO\_PURGE\_SUB**

Abonelik silinir. MQC.MQCO\_PURGE\_SUB , dayanıklı olmayan, yönetilen bir konu için varsayılan değerdir.

**public MQQueueManager ConnectionReference {get;}**

Bu kaynağın ait olduğu kuyruk yöneticisi.

**public string MQDescription {get;}**

Kaynak, kuyruk yöneticisi tarafından tutulan kaynağın tanımı. MQDescription , abonelikler ve konular için boş bir dizgi döndürür.

**public boolean IsOpen {get;}**

Kaynağın şu anda açık olup olmadığını belirtir.

**public string Name {get;}**

Kaynağın adı. Ad, erişim yönteminde sağlanan değer ya da dinamik bir kuyruk için kuyruk yöneticisi tarafından ayrılan ad.

**public int OpenOptions {get; set;}**

OpenOptions , bir IBM MQ nesnesi açıldığında ayarlanır. OpenOptions .set yöntemi yoksayılr ve bir hataya neden olmaz. Abonelikler için OpenOptions yoktur.

**Yöntemler****public virtual void Close();**

MQException yayınlıyor.

Nesneyi kapatır. Close çağırıldıktan sonra, bu kaynağa ilişkin başka bir işlem yapılmasına izin verilmez. Close yönteminin davranışını değiştirmek için closeOptions özniteliğini ayarlayın.

**public string GetAttributeString(int selector, int length);**

MQException yayınlıyor.

Bir öznitelik dizgisi alır.

**seçici**

Hangi özniteliğin sorgulanmakta olduğunu gösteren tamsayı.

**uzunluk**

Gerekten dizilimin uzunluğunu gösteren tamsayı.

**public void Inquire(int[] selectors, int[] intAttrs, byte[] charAttrs);**

MQException yayınlıyor.

Bir tamsayılar dizisini ve bir kuyruğun, işlemin ya da kuyruk yöneticisinin özniteliklerini içeren bir karakter dizgileri kümesini döndürür. Sorgulanacak öznitelikler, seçiciler dizisinde belirtilir.

**Not:** Many of the more common attributes can be queried using the Get methods defined in MQManagedObject, MQQueue and MQQueueManager.

**seçiciler**

Sorgulanacak değerlere sahip öznitelikleri tanımlayan tamsayı dizisi.

**intAttrs**

Tamsayı öznitelik değerlerinin döndürüldüğü dizi. Tamsayı özniteliği değerleri, seçiciler dizisinde tamsayı özniteliği seçiciyle aynı sırada döndürülür.

**charAttrs**

Karakter özniteliklerinin döndürüldüğü arabellek, bitleştirilir. Karakter öznitelikleri, seçiciler dizisinde karakter özniteliği seçiciyle aynı sırada döndürülür. Her bir öznitelik için her öznitelik dizgisinin uzunluğu değişmez.

**public void Set(int[] selectors, int[] intAttrs, byte[] charAttrs);**

MQException yayınlıyor.

Seçiciler vektöründe tanımlanan öznitelikleri ayarlar. Ayarlanacak öznitelikler, seçiciler dizisinde belirtilir.

**seçiciler**

Ayarlanacak değerleri içeren öznitelikleri tanımlayan tamsayı dizisi.

**intAttrs**

Ayarlanacak tamsayı özniteliği değerleri dizisi. Bu değerler, seçiciler dizisinde bulunan tamsayı özniteliği seçicileriyle aynı sırada olmalıdır.

**charAttrs**

Ayarlanacak karakter özniteliklerinin bitişirileceği arabellek. Bu değerler, seçiciler dizisinde karakter özniteliği seçiciyle aynı sırada olmalıdır. Her karakter özniteliğinin uzunluğu sabittir.

**public void SetAttributeString(int selector, string value, int length);**

MQException yayınlıyor.

Bir öznitelik dizisi belirler.

**seçici**

Hangi özniteliğin ayarlanmakta olduğunu gösteren tamsayı.

**değer**

Öznitelik değeri olarak ayarlanacak dizgi.

**uzunluk**

Gereken dizilimin uzunluğunu gösteren tamsayı.

## Oluşturucular

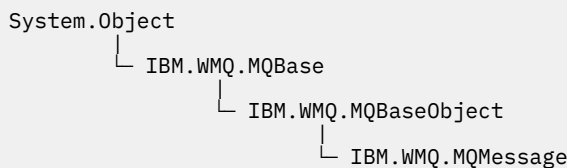
**protected MQManagedObject()**

Oluşturucu yöntemi. Bu nesne, kendisi tarafından somutlaştırılmayan bir soyut temel sınıftır.

## MQMessage.NET sınıfı

Bir IBM MQ iletisine ilişkin ileti tanımlayıcısına ve verilere erişmek için MQMessage ' u kullanın. MQMessage , bir IBM MQ iletisini sarsalıyor.

### Sınıf



```
public class IBM.WMQ.MQMessage extends IBM.WMQ.MQBaseObject;
```

Bir MQMessage nesnesi oluşturun ve daha sonra, iletiyle uygulamadaki diğer nesnelere veri aktarmak için Read ve Write yöntemlerini kullanın. MQDestination, MQQueue ve MQTopic sınıflarının Put ve Get yöntemlerini kullanarak MQMessage nesnelere gönderin ve alın.

MQMessage özelliklerini kullanarak ileti tanımlayıcısının özelliklerini alır ve ayarlar. SetProperty ve GetProperty yöntemlerini kullanarak genişletilmiş ileti özelliklerini ayarlayın ve alın.

- [“Özellikler” sayfa 1711](#)
- [“Read ve Write ileti yöntemleri” sayfa 1716](#)
- [“Arabellek yöntemleri” sayfa 1719](#)
- [“Özellik yöntemleri” sayfa 1720](#)

- [“Oluşturucular” sayfa 1722](#)

## Özellikler

Özellikler alınırken MQException için test ediliyor.

### **public string AccountingToken {get; set;}**

İletinin kimlik bağlamının bir parçası; bir uygulamanın iletinin sonucu olarak yapılan çalışmalardan sorumlu olması için yardımcı olur. Varsayılan değer MQC.MQACT\_NONE değeridir.

### **public string ApplicationIdData {get; set;}**

İletinin kimlik bağlamının bir parçası. ApplicationIdData , uygulama takımı tarafından tanımlanan bilgilerdir ve iletiyle ya da başlatıcısı hakkında ek bilgi sağlamak için kullanılabilir. Varsayılan değer "" değeridir.

### **public string ApplicationOriginData {get; set;}**

İletinin kaynağı hakkında ek bilgi sağlamak için kullanılacak uygulama tarafından tanımlanan bilgiler. Varsayılan değer "" değeridir.

### **public int BackoutCount {get;}**

Bir iş biriminin parçası olarak, iletinin daha önce bir MQQueue .Get çağrısı tarafından geri döndürülmesinin ve yedeklenmesinin kaç kez olduğunu belirten sayı. Varsayılan değer sıfır 'tır.

### **public int CharacterSet {get; set;}**

İletideki karakter verilerinin kodlanmış karakter takımı tanıtıcısı.

İletide karakter verilerinin karakter kümesini tanımlamak için CharSet seçeneğini belirleyin. İletideki karakter verilerini kodlamak için hangi karakter kümesinin kullanıldığını öğrenmek için CharSet ögesini seçin.

.NET uygulamaları her zaman Unicode 'da çalışır; diğer ortam uygulamalarında, kuyruk yöneticisi altında çalışmakta olan aynı karakter kümesinde çalışır.

ReadString ve ReadLine yöntemleri, iletteki karakter verilerini sizin için Unicode 'a dönüştürür.

WriteString yöntemi, Unicode 'dan CharSet' ta kodlanan karakter kümesine dönüştürür. CharSet varsayılan değerine ayarlandıysa, MQC.MQCCSI\_Q\_MGRdeğeri 0 olur, dönüştürme gerçekleşmez ve CharSet değeri 1200 olarak ayarlanır. CharSet değerini başka bir değere ayarladıysanız, WriteString , Unicode 'dan diğer değere dönüştürür.

**Not:** Diğer okuma ve yazma yöntemleri CharSet ögesini kullanmaz.

- ReadChar ve WriteChar , dönüştürmeden ve ileti arabelleğinden bir Unicode karakterini okur ve bu arabelleğden okur.
- ReadUTF ve WriteUTF , uygulamadaki bir Unicode dizesi ile ileti arabelleğindeki 2 baytlık bir alan başına önek olarak önek olarak UTF-8 dizesini dönüştürür.
- Byte yöntemleri, uygulama ve ileti arabelleğiyle arabelleğiyle değiştirilmeden bayt d"

### **public byte[] CorrelationId {get; set;}**

• Bir MQQueue .Get çağrısı için, alınacak iletinin ilinti tanıtıcısı. Kuyruk yöneticisi, ileti tanımlayıcı alanlarıyla eşleşen bir ilinti tanıtıcısı ve ileti tanıtıcısı ile ilk iletiyi döndürür. Varsayılan değer olan MQC.MQCI\_NONE, herhangi bir ilinti tanıtıcısının eşleşmesine yardımcı olur.

• Bir MQQueue .Put çağrısı için, ilişkilendirilecek ilinti tanıtıcısı.

### **public int DataLength {get;}**

Okunabilmek için kalan ileti verilerinin bayt sayısı.

### **public int DataOffset {get; set;}**

İleti verileri içinde imlecin bulunduğu konum. Okuma ve yazma işlemleri yürürlükteki konumda yürürlüğe girmektedir.

### **public int Encoding {get; set;}**

Uygulama iletisi verilerinde sayısal değerler için kullanılan gösterim. Kodlama , ikili, paketlenmiş onlu ve kayan nokta verileri için geçerlidir. Bu sayısal biçimler için okuma ve yazma yöntemlerinin davranışı

uygun şekilde değiştirilir. Bu üç kısımdan her birinden bir değer ekleyerek, kodlama alanı için bir değer oluşturun. Diğer bir seçenek olarak, bit ya da işleci kullanarak üç kısımdan her birinden değerleri birleştiren değeri oluşturun.

1. İkili tamsayı

**MQC.MQENC\_INTEGER\_NORMAL**

Big-endian tamsayılar.

**MQC.MQENC\_INTEGER\_REVERSED**

Intel mimarisinde kullanıldığı şekliyle küçük endian tamsayıları.

2. Paketli-ondalık

**MQC.MQENC\_DECIMAL\_NORMAL**

Büyük-endian paketli-ondalık, z/OS tarafından kullanılıyor.

**MQC.MQENC\_DECIMAL\_REVERSED**

Little-endian paketli-ondalık.

3. Kayan noktalı

**MQC.MQENC\_FLOAT\_IEEE\_NORMAL**

Big-endian IEEE float.

**MQC.MQENC\_FLOAT\_IEEE\_REVERSED**

Küçük-endian IEEE floats (kullanılan Intel mimarisi).

**MQC.MQENC\_FLOAT\_S390**

z/OS biçimi kayan noktaları biçimlendirir.

Varsayılan değer şöyledir:

```
MQC.MQENC_INTEGER_REVERSED |
MQC.MQENC_DECIMAL_REVERSED |
MQC.MQENC_FLOAT_IEEE_REVERSED
```

Varsayılan ayar, WriteInt 'un küçük endian bir tamsayı yazmasına ve ReadInt 'un küçük-endian tamsayısını okumasına neden olur. Onun yerine işaret MQC.MQENC\_INTEGER\_NORMAL işaretini ayarlıyorsanız, WriteInt büyük bir tamsayı (big-endian integer) bir tamsayı yazar ve ReadInt büyük endian bir tamsayı okur.

**Not:** IEEE biçim kayan noktalarından zSeries biçimi kayan noktalara dönüştürülürken duyarlık kaybı ortaya çıkabilir.

**public int Expiry {get; set;}**

Bir saniyenin onda biri olarak ifade edilen bir süre bitimi (uygulamanın, iletiyi koyan uygulama tarafından belirlenir). Bir iletinin süre bitimi süresi dolduktan sonra, kuyruk yöneticisi tarafından atılabilir. İleti, MQC.MQRO\_EXPIRATION işaretlerinden birini belirtiyorsa, ileti atıldığında bir rapor oluşturulur. Varsayılan değer, iletinin hiçbir zaman süre bitimine geçmediği anlamına gelen MQC.MQEI\_UNLIMITED'dedir.

**public int Feedback {get; set;}**

Raporun doğasını belirtmek için MQC.MQMT\_REPORT tipinde bir iletiyle Geribildirim seçeneğini kullanın. Aşağıdaki bildirim kodları sistem tarafından tanımlanır:

- MQC.MQFB\_EXPIRATION
- MQC.MQFB\_COA
- MQC.MQFB\_COD
- MQC.MQFB\_QUIT
- MQC.MQFB\_PAN
- MQC.MQFB\_NAN
- MQC.MQFB\_DATA\_LENGTH\_ZERO
- MQC.MQFB\_DATA\_LENGTH\_NEGATIVE

- MQC.MQFB\_DATA\_LENGTH\_TOO\_BIG
- MQC.MQFB\_BUFFER\_OVERFLOW
- MQC.MQFB\_LENGTH\_OFF\_BY\_ONE
- MQC.MQFB\_IH\_ERROR

MQC.MQFB\_APPL\_FIRST - MQC.MQFB\_APPL\_LAST aralığındaki uygulama tanımlı geribildirim değerleri de kullanılabilir. Bu alanın varsayılan değeri, herhangi bir geri bildirim verilmemesinin sağlanmasıdır. MQC.MQFB\_NONE.

### **public string Format {get; set;}**

İletideki verilerin alıcıya iletilmesinde doğasını belirtmek için iletinin göndereni tarafından kullanılan biçim adı. Kendi biçim adlarınızı kullanabilirsiniz, ancak MQ harfleriyle başlayan adların, kuyruk yöneticisi tarafından tanımlanan anlamları vardır. Kuyruk yöneticisi yerleşik biçimleri şunlardır:

#### **MQC.MQFMT\_ADMIN**

Komut sunucusu isteği/yanıt iletisi.

#### **MQC.MQFMT\_COMMAND\_1**

1 komut yanıtı iletisi yazın.

#### **MQC.MQFMT\_COMMAND\_2**

Tip 2 komut yanıt iletisi.

#### **MQC.MQFMT\_DEAD\_LETTER\_HEADER**

-Ölü harf üstbilgisi.

#### **MQC.MQFMT\_EVENT**

Olay iletisi.

#### **MQC.MQFMT\_NONE**

Biçim adı yok.

#### **MQC.MQFMT\_PCF**

Programlanabilir komut biçiminde kullanıcı tanımlı ileti.

#### **MQC.MQFMT\_STRING**

Tamamen karakterlerden oluşan ileti.

#### **MQC.MQFMT\_TRIGGER**

Tetikleme iletisi

#### **MQC.MQFMT\_XMIT\_Q\_HEADER**

İletim kuyruğu üstbilgisi.

Varsayılan değer MQC.MQFMT\_NONE'dir.

### **public byte[] GroupId {get; set;}**

Fiziksel iletinin ait olduğu ileti grubunu tanımlayan bayt dizesi. Varsayılan değer MQC.MQGI\_NONE'dir.

### **public int MessageFlags {get; set;}**

Bir iletinin bölümlenmesini ve durumunun denetlenmesini işaretler.

### **public byte[] MessageId {get; set;}**

Bir MQQueue.Get çağrısı için bu alan, alınacak iletinin ileti tanıtıcısını belirtir. Olağan durumda kuyruk yöneticisi, ileti tanıtıcısı alanlarıyla eşleşen bir ileti tanıtıcısı ve ilinti tanıtıcısı ile ilk iletiyi döndürür. Herhangi bir ileti tanıtıcısının, MQC.MQMI\_NONE özel değerini kullanarak eşleşmesine izin verin.

MQQueue.Put çağrısı için, bu alan kullanılacak ileti tanıtıcısını belirtir. MQC.MQMI\_NONE belirtilirse, kuyruk yöneticisi, ileti konduğunda benzersiz bir ileti tanıtıcısı oluşturur. Bu üye değişkeninin değeri, kullanılmakta olan ileti tanıtıcısını göstermek için, put işleminden sonra güncellenir. Varsayılan değer MQC.MQMI\_NONE'dir.

### **public int MessageLength {get;}**

MQMessage nesnesindeki ileti verilerinin bayt sayısı.

### **public int MessageSequenceNumber {get; set;}**

Bir grup içindeki mantıksal iletinin sıra numarası.

**public int MessageType {get; set;}**

İletinin tipini gösterir. Şu anda sistem tarafından tanımlanan değerler şunlardır:

- MQC.MQMT\_DATAGRAM
- MQC.MQMT\_REPLY
- MQC.MQMT\_REPORT
- MQC.MQMT\_REQUEST

Application-defined values can also be used, in the range MQC.MQMT\_APPL\_FIRST to MQC.MQMT\_APPL\_LAST. Bu alanın varsayılan değeri MQC.MQMT\_DATAGRAM' dir.

**public int Offset {get; set;}**

Kesimlere ayrılmış bir iletide, mantıksal iletinin başlangıcındaki fiziksel bir iletteki verilerin görelî konumu.

**public int OriginalLength {get; set;}**

Kesimlere ayrılmış bir iletinin özgün uzunluğu.

**public int Persistence {get; set;}**

Mesaj devamlılığı. Aşağıdaki değerler tanımlanır:

- MQC.MQPER\_NOT\_PERSISTENT

Bu seçeneği yeniden bağlanabilir bir istemcide ayarladıysanız, bağlantı başarılı olduğunda MQRC\_NONE neden kodu uygulamaya geri döndürülür.

- MQC.MQPER\_PERSISTENT

Bu seçeneği yeniden bağlanabilir bir istemcide ayarladıysanız, bağlantı başarılı olduktan sonra MQRC\_CALL\_INTERRUPTED neden kodu uygulamaya geri döndürülür.

- MQC.MQPER\_PERSISTENCE\_AS\_Q\_DEF

Varsayılan değer, hedef kuyruğun varsayılan kalıcı saklama özniteliğinden iletiye ilişkin kalıcılığı alan MQC.MQPER\_PERSISTENCE\_AS\_Q\_DEFdeğeridir.

**public int Priority {get; set;}**

İleti önceliği. Özel değer MQC.MQPRI\_PRIORITY\_AS\_Q\_DEF giden iletide de ayarlanabilir. İletiyeye ilişkin öncelik, hedef kuyruğun varsayılan öncelik özniteliğinden alınır. Varsayılan değer MQC.MQPRI\_PRIORITY\_AS\_Q\_DEFdeğeridir.

**public int PropertyValidation {get; set;}**

İletinin bir özelliği ayarlandığında özellikler doğrulamasının gerçekleşip gerçekleştirilmeyeceğini belirtir. Olası değerler şunlardır:

- MQCMHO\_DEFAULT\_VALIDATION
- MQCMHO\_VALIDATE
- MQCMHO\_NO\_VALIDATION

Varsayılan değer MQCMHO\_DEFAULT\_VALIDATIONdeğeridir.

**public string PutApplicationName {get; set;}**

İletiyey koyan uygulamanın adı. Varsayılan değer ""değeridir.

**public int PutApplicationType {get; set;}**

İletiyey koyan uygulamanın tipi. PutApplicationTipi , sistem tarafından tanımlanan ya da kullanıcı tanımlı bir değer olabilir. Aşağıdaki değerler sistem tarafından tanımlanır:

- MQC.MQAT\_AIX
- MQC.MQAT\_CICS
- MQC.MQAT\_DOS
- MQC.MQAT\_IMS
- MQC.MQAT\_MVS
- MQC.MQAT\_OS2

- MQC.MQAT\_OS400
- MQC.MQAT\_QMGR
- MQC.MQAT\_UNIX
- MQC.MQAT\_WINDOWS
- MQC.MQAT\_JAVA

Varsayılan değer, iletide herhangi bir bağlam bilgisinin olmadığını gösteren MQC.MQAT\_NO\_CONTEXT değeridir.

**public DateTime PutDateTime {get; set;}**

İletinin yerleştiği saat ve tarih.

**public string ReplyToQueueManagerName {get; set;}**

Yanıt ya da rapor iletileri göndermek için kuyruk yöneticisinin adı. Varsayılan değer ""'dir ve kuyruk yöneticisi ReplyToQueueManagerName' i sağlar.

**public string ReplyToQueueName {get; set;}**

İleti için alma isteğini yayınlayan uygulamanın MQC.MQMT\_REPLY ve MQC.MQMT\_REPORT iletilerini göndereceği ileti kuyruğunun adı. Varsayılan ReplyToQueueName "" olur.

**public int Report {get; set;}**

Rapor ve yanıt iletilerine ilişkin seçenekleri belirlemek için Rapor ' u kullanın:

- Raporların gerekli olup olmadığını.
- Uygulama iletisi verilerinin raporlara dahil edilip edilmeyeceğini belirleyin.
- Rapordaki ya da yanıtta iletisi ve ilinti tanıtıcılarının nasıl ayarlanacağı.

Dört rapor tipinin herhangi bir birleşimi istenebilir:

- Dört rapor tipinin herhangi bir birleşimini belirtin. Her bir rapor tipi için, uygulama iletisi verilerinin rapor iletisine eklenip eklenmeyeceğine bağlı olarak, her bir rapor tipi için herhangi bir üç seçenektan herhangi birini seçebilirsiniz.

1. Varışta onayla

- MQC.MQRO\_COA
- MQC.MQRO\_COA\_WITH\_DATA
- MQC.MQRO\_COA\_WITH\_FULL\_DATA \*\*

2. Teslim edilmeyi onayla

- MQC.MQRO\_COD
- MQC.MQRO\_COD\_WITH\_DATA
- MQC.MQRO\_COD\_WITH\_FULL\_DATA \*\*

3. Özel durum

- MQC.MQRO\_EXCEPTION
- MQC.MQRO\_EXCEPTION\_WITH\_DATA
- MQC.MQRO\_EXCEPTION\_WITH\_FULL\_DATA \*\*

4. Süre Bitim Tarihi

- MQC.MQRO\_EXPIRATION
- MQC.MQRO\_EXPIRATION\_WITH\_DATA
- MQC.MQRO\_EXPIRATION\_WITH\_FULL\_DATA \*\*

**Not:** Listede \*\* ile işaretlenen değerler, z/OS kuyruk yöneticileri tarafından desteklenmez. Uygulamanızın, uygulamanın çalışmakta olduğu altyapıdan bağımsız olarak bir z/OS kuyruk yöneticisine erişmesi olasıysa, bunları kullanmayın.

- Rapor ya da yanıt iletisi için ileti tanıtıcısının nasıl oluşturulacağını denetlemek için aşağıdakilerden birini belirtin:

- MQC.MQRO\_NEW\_MSG\_ID
- MQC.MQRO\_PASS\_MSG\_ID
- Rapor ya da yanıt iletisine ilişkin ilinti tanıtıcısının nasıl ayarlanacağını denetlemek için aşağıdakilerden birini belirtin:
  - MQC.MQRO\_COPY\_MSG\_ID\_TO\_CORREL\_ID
  - MQC.MQRO\_PASS\_CORREL\_ID
- Hedef kuyruğa teslim edilemediğinde özgün iletinin yok edilmesini denetlemek için aşağıdakilerden birini belirtin:
  - MQC.MQRO\_DEAD\_LETTER\_Q
  - MQC.MQRO\_DISCARD\_MSG \*\*
- Herhangi bir rapor seçeneği belirtilmemişse, varsayılan değer:

```
MQC.MQRO_NEW_MSG_ID |
MQC.MQRO_COPY_MSG_ID_TO_CORREL_ID |
MQC.MQRO_DEAD_LETTER_Q
```

- Alma uygulamasının olumlu bir eylem ya da olumsuz işlem raporu iletisi göndermesini istemek için aşağıdakilerden birini ya da her ikisini belirleyebilirsiniz.
  - MQC.MQRO\_PAN
  - MQC.MQRO\_NAN

#### **public int TotalMessageLength {get;}**

İletideki toplam bayt sayısı, bu iletinin alındığı ileti kuyruğunda saklanır.

#### **public string UserId {get; set;}**

UserId , iletinin tanıtıcı bağlamının bir parçasıdır. Kuyruk yöneticisi genellikle değeri sağlar. Kimlik bağlamını belirleme yetkiniz varsa, değeri geçersiz kılabilirsiniz.

#### **public int Version {get; set;}**

Kullanılan MQMD yapısının sürümü.

## **Read ve Write ileti yöntemleri**

Read ve Write yöntemleri, .NET System.IO ad alanı içindeki BinaryReader ve BinaryWriter sınıflarının üyeleriyle aynı işlevleri gerçekleştirir. Tam dil sözdizimine ve kullanım örneklerine ilişkin MSDN 'ye bakın. İleti arabelleğindeki yürürlükteki konumdan okunan ya da yazan yöntemlerin sayısı. Bunlar, yürürlükteki konumu okunan ya da yazılan bayt sayısına göre ileriye taşılırlar.

**Not:** İleti verileri bir MQRFH ya da MQRFH2 üstbilgisi içeriyorsa, verileri okumak için ReadBytes yöntemini kullanmanız gerekir.

- Tüm yöntemler IOException' e atılıyor.
- ReadFully yöntemleri, iletiyi tam olarak ayarlamak için hedef byte ya da sbyte dizisini otomatik olarak yeniden boyutlandırır. Boş değerli bir dizi de yeniden boyutlandırılır.
- Read yöntem atma EndOfStreamException.
- WriteDecimal yöntem atma MQException.
- ReadString, ReadLine ve WriteString yöntemleri, Unicode ile iletinin karakter kümesi arasında dönüştürme yapar; bkz. [CharacterSet](#) .
- Decimal , big-endian, MQC.MQENC\_DECIMAL\_NORMAL ya da little-endian MQC.MQENC\_DECIMAL\_REVERSE biçiminde kodlanmış, paketlenmiş onlu sayıları okuma ve yazma yöntemlerini Kodlamadeğerine göre okur ve bu yöntemlerin paketlenmesini sağlar. Ondalık aralıklar ve karşılık gelen .NET tipleri şunlardır:

**Decimal2/short**  
-999-999



**Decimal4/int**

-9999999-9999999

**Decimal8/long**

-9999999999999-9999999999999999

- The `Double` and `Float` methods read and write floating values encoded in IEEE big-endian and little-endian formats, `MQC.MQENC_FLOAT_IEEE_NORMAL` and `MQC.MQENC_FLOAT_IEEE_REVERSED`, or in S/390 format, `MQC.MQENC_FLOAT_S390`, according to the value of `Kodlama`.
- The `Int` methods read and write integer values encoded in big-endian, `MQC.MQENC_INTEGER_NORMAL`, or little-endian, `MQC.MQENC_INTEGER_REVERSED`, format, according to the value of `Kodlama`. İmzalanmamış 2 baytlık bir tamsayı tipinin yanı sıra, tamsayılar imzalanır. Tamsayı büyüklükleri ve .NET ve IBM MQ tipleri şunlardır:

**2 bayt**

short, Int2, ushort, UInt2

**4 bayt**

int, Int4

**8 bayt**

long, Int8

- `WriteObject`, bir nesnenin sınıfını, geçici olmayan ve durağan olmayan alanlarının değerlerini ve üst tiplerinde bulunan alanları ileti arabelleğiyle aktarır.
- `ReadObject`, nesnenin sınıfından, sınıfın imzasından ve geçici olmayan ve durağan olmayan alanların değerlerini ve üst tiplerinden oluşan bir nesne yaratır.

Çizelge 844. Okuma ve Yazma ileti yöntemleri	
Hedef türü	Yöntem imzaları
<b>Boolean</b>	<pre>public bool ReadBoolean();  public void WriteBoolean(bool value);</pre>
<b>Byte</b>	<pre>public byte ReadByte() public byte ReadUnsignedByte()  public void Write(int value) public void WriteByte(int value) public void WriteByte(byte value) public void WriteByte(sbyte value)</pre>
<b>Bytes</b>	<pre>public byte[] ReadBytes(int count) public void ReadFully(ref byte[] value) public void ReadFully(ref sbyte[] value) public void ReadFully(ref byte[] value, int offset, int length) public void ReadFully(ref sbyte[] value, int offset, int length)  public void Write(byte[] value) public void Write(sbyte[] value) public void Write(byte[] value, int offset, int length) public void Write(sbyte[] value, int offset, int length) public void WriteBytes(string value)</pre>
<b>Decimal2</b>	<pre>public void WriteDecimal2(short value)</pre>

Çizelge 844. Okuma ve Yazma ileti yöntemleri (devamı var)

Hedef türü	Yöntem imzaları
<b>Decimal4</b>	<code>public void WriteDecimal4(short value)</code>
<b>Decimal8</b>	<code>public void WriteDecimal8(short value)</code>
<b>Double</b>	<code>public double ReadDouble()</code> <code>public void WriteDouble(double value)</code>
<b>Float</b>	<code>public float ReadFloat()</code> <code>public void WriteFloat(float value)</code>
<b>Int2</b>	<code>public void WriteInt2(int value)</code>
<b>Int4</b>	<code>public int readDecimal4()</code> <code>public int ReadInt()</code> <code>public int ReadInt4()</code> <code>public void WriteInt(int value)</code> <code>public void WriteInt4(int value)</code>
<b>Int8</b>	<code>public void WriteInt8(long value)</code>
<b>Long</b>	<code>public long ReadDecimal8()</code> <code>public long ReadLong()</code> <code>public long ReadInt8()</code> <code>public void WriteLong(long value)</code>
<b>Object</b>	<code>public Object ReadObject()</code> <code>public void WriteObject(Object object)</code>
<b>Short</b>	<code>public short ReadShort()</code> <code>public short ReadDecimal2()</code> <code>public short ReadInt2()</code> <code>public void WriteShort(int value)</code>

Çizelge 844. Okuma ve Yazma ileti yöntemleri (devamı var)

Hedef türü	Yöntem imzaları
<b>string</b>	<pre>public string ReadString(int length)  public void WriteString(string string)</pre>
<b>Unsigned Short</b>	<pre>public ushort ReadUnsignedShort() public ushort ReadUInt2()</pre>
<b>Unicode</b>	<pre>public string ReadLine() public char ReadChar()  public void WriteChar(int value) public void WriteChars(string string)</pre>
<b>UTF</b>	<pre>public string ReadUTF()  public void WriteUTF(string string)</pre>

## Arabellek yöntemleri

### **public void ClearMessage();**

IOException yayınlıyor.

İleti arabelleğindeki verileri atar ve veri görelî konumunu sıfır olarak ayarlar.

### **public void ResizeBuffer(int size)**

IOException yayınlıyor.

Sonraki alma işlemleri için gerekebilecek arabellek büyüklüğünün MQMessage nesnesine ilişkin bir ipucu. İleti şu anda ileti verilerini içeriyorsa ve yeni büyüklük yürürlükteki büyükten küçükse, ileti verileri kesilir.

### **public void Seek(int pos)**

Throws IOException, ArgumentOutOfRangeException, ArgumentException.

İmleci, *pos* ile verilen ileti arabelleğindeki mutlak konuma taşır. Sonraki okuma ve yazma işlemleri, arabelleğindeki bu konumda işlem yazar.

### **public int SkipBytes(int i)**

Throws IOException, EndOfStreamException.

İleti arabelleğindeki *n* byte 'ı leri taşır ve *n* değerini döndürür, atlanan bayt sayısını döndürür.

Aşağıdaki olaylardan biri gerçekleşinceye kadar SkipBytes yöntem öbekleri:

- Tüm byte 'lar atlandı
- İleti arabelleğinin sonu saptandı
- Bir kural dışı durum yayımlandı

## Özellik yöntemleri

### **public void DeleteProperty(string name);**

MQException yayılıyor.

İletiden belirtilen adı taşıyan bir özelliği siler.

#### **ad**

Silinecek özelliğin adı.

### **public System.Collections.IEnumerator GetPropertyNames(string name)**

MQException yayılıyor.

Belirtilen adla eşleşen tüm özellik adlarından oluşan bir IEnumerator döndürür. Yüzde işareti '%', adın özelliklerini, sıfır ya da daha fazla karakterle eşleşen, iletinin özelliklerini süzmek için genel arama karakteri olarak kullanılabilir.

#### **ad**

Eşleştirmek için özelliğin adı.

## **SetProperty ve GetProperty yöntemleri**

Tüm SetProperty ve GetProperty yöntemleri MQException yayındır.

The SetProperty method of MQMessage .NET class adds a new property if a property does not exist already. Ancak, özellik önceden varsa, sağlanan özellik değeri listenin sonuna eklenir. When multiple values are set to a property name using SetProperty, calling GetProperty for that name returns those values sequentially in the order that those values were set.

The behavior is same for all the Set\*Property and Get\*Property typed methods such as GetLongProperty, SetLongProperty, GetBooleanProperty, SetBooleanProperty, GetStringProperty, and SetStringProperty.

Tip	Yöntem imzaları
<b>Boolea n</b>	<pre>public boolean GetBooleanProperty(string name); public boolean GetBooleanProperty(string name, MQPropertyDescriptor pd);  public void SetBooleanProperty(string name, boolean value); public void SetBooleanProperty(string name, MQPropertyDescriptor pd, boolean value);</pre>
<b>Byte</b>	<pre>public sbyte GetByteProperty(string name); public sbyte GetByteProperty(string name, MQPropertyDescriptor pd);  public void SetByteProperty(string name, sbyte value); public void SetByteProperty(string name, MQPropertyDescriptor pd, sbyte value);</pre>
<b>Bytes</b>	<pre>public sbyte[] GetBytesProperty(string name); public sbyte[] GetBytesProperty(string name, MQPropertyDescriptor pd);  public void SetBytesProperty(string name, sbyte[] value); public void SetBytesProperty(string name, MQPropertyDescriptor pd, sbyte[] value);</pre>

Çizelge 845. SetProperty ve GetProperty yöntemleri (devamı var)

Tip	Yöntem imzaları
<b>Double</b>	<pre>public double GetDoubleProperty(string name); public double GetDoubleProperty(string name, MQPropertyDescriptor pd);  public void SetDoubleProperty(string name, double value); public void SetDoubleProperty(string name, MQPropertyDescriptor pd, double value);</pre>
<b>Float</b>	<pre>public float GetFloatProperty(string name); public float GetFloatProperty(string name, MQPropertyDescriptor pd);  public void SetFloatProperty(string name, float value); public void SetFloatProperty(string name, MQPropertyDescriptor pd, float value);</pre>
<b>Int2</b>	<pre>public short GetInt2Property(string name); public short GetInt2Property(string name, MQPropertyDescriptor pd);  public void SetInt2Property(string name, short value); public void SetInt2Property(string name, MQPropertyDescriptor pd, short value);</pre>
<b>Int4</b>	<pre>public int GetInt4Property(string name); public int GetInt4Property(string name, MQPropertyDescriptor pd);  public void SetInt4Property(string name, int value); public void SetInt4Property(string name, MQPropertyDescriptor pd, int value);</pre>
<b>Int8</b>	<pre>public long GetInt8Property(string name); public long GetInt8Property(string name, MQPropertyDescriptor pd);  public void SetInt8Property(string name, long value); public void SetInt8Property(string name, MQPropertyDescriptor pd, long value);</pre>
<b>Long</b>	<pre>public long GetLongProperty(string name); public long GetLongProperty(string name, MQPropertyDescriptor pd);  public void SetLongProperty(string name, long value); public void SetLongProperty(string name, MQPropertyDescriptor pd, long value);</pre>
<b>Object</b>	<pre>public Object GetObjectProperty(string name); public Object GetObjectProperty(string name, MQPropertyDescriptor pd);  public void SetObjectProperty(string name, Object value); public void SetObjectProperty(string name, MQPropertyDescriptor pd, Object value);</pre>

Çizelge 845. SetProperty ve GetProperty yöntemleri (devamı var)

Tip	Yöntem imzaları
<b>Short</b>	<pre>public short GetShortProperty(string name); public short GetShortProperty(string name, MQPropertyDescriptor pd);  public void SetShortProperty(string name, short value); public void SetShortProperty(string name, MQPropertyDescriptor pd, short value);</pre>
<b>string</b>	<pre>public string GetStringProperty(string name); public string GetStringProperty(string name, MQPropertyDescriptor pd);  public void SetStringProperty(string name, string value); public void SetStringProperty(string name, MQPropertyDescriptor pd, string value);</pre>

## Oluşturucular

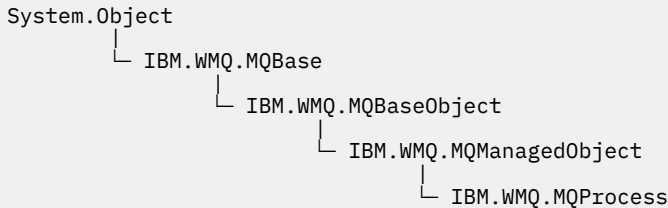
**public MQMessage();**

Varsayılan ileti tanımlayıcı bilgileri ve boş ileti arabelleğiyle bir MQMessage nesnesi yaratır.

## MQProcess.NET sınıfı

Bir IBM MQ işleminin özniteliklerini sorgulamak için MQProcess kullanın. Bir oluşturucu ya da bir MQQueueManager AccessProcess yöntemi kullanarak bir MQProcess nesnesi yaratın.

## Sınıf



```
public class IBM.WMQ.MQProcess extends IBM.WMQ.MQManagedObject;
```

- [“Özellikler” sayfa 1722](#)
- [“Oluşturucular” sayfa 1723](#)

## Özellikler

Özellikler alınırken MQException için test ediliyor.

**public string ApplicationId {get;}**

Başlatılacak uygulamayı tanıtan karakter dizilimini alır. ApplicationId , bir tetikleyici Monitor uygulaması tarafından kullanılır. ApplicationId , başlatma kuyruğuna, tetikleyici iletisinin bir parçası olarak gönderilir.

Varsayılan değer boş değerdir.

## **public int ApplicationType {get;}**

Bir tetikleyici izleyicisi uygulaması tarafından başlatılacak işlemin tipini tanımlar. Standart tipler tanımlanır, ancak diğerleri kullanılabilir:

- MQAT\_AIX
- MQAT\_CICS
- MQAT\_IMS
- MQAT\_MVS
- MQAT\_NATIVE
- MQAT\_OS400
- MQAT\_UNIX
- MQAT\_WINDOWS
- MQAT\_JAVA
- MQAT\_USER\_FIRST
- MQAT\_USER\_LAST

Varsayılan değer MQAT\_NATIVE'dir.

## **public string EnvironmentData {get;}**

Başlatılacak uygulamanın ortasıyla ilgili bilgi alır.

Varsayılan değer boş değerdir.

## **public string UserData {get;}**

Kullanıcının, başlatılacak uygulama hakkında bilgi verdiği bilgileri alır.

Varsayılan değer boş değerdir.

## **Oluşturucular**

```
public MQProcess(MQQueueManager queueManager, string processName, int openOptions);
```

```
public MQProcess(MQQueueManager qMgr, string processName, int openOptions, string queueManagerName, string alternateUserId);
```

MQException yayınlıyor.

Süreç özniteliklerini sorgulamak için *qMgr* kuyruk yöneticisine ilişkin bir IBM MQ işleme erişin.

### **qMgr**

Erişilecek kuyruk yöneticisi.

### **processName**

Açılacak sürecin adı.

### **openOptions**

İşlemin açılmasını denetleyen seçenekler. Bit kullanan OR kullanılarak eklenebilecek ya da birleştirilebilen geçerli seçenekler şunlardır:

- MQC.MQ00\_FAIL\_IF QUIESCING
- MQC.MQ00\_INQUIRE
- MQC.MQ00\_SET
- MQC.MQ00\_ALTERNATE\_USER\_AUTHORITY

### **queueManagerAdı**

İşlemin tanımlandığı kuyruk yöneticisinin adı. Kuyruk yöneticisi, sürecin erişmekte olduğu kuyruk yöneticisi aynıysa, boş değerli ya da boş bir kuyruk yöneticisi adı bırakabilirsiniz.

### alternateUserTanıtıcısı

**openOptions** parametresinde MQC.MQ00\_ALTERNATE\_USER\_AUTHORITY belirtilirse, *alternateUserId*, işleme ilişkin yetkiyi denetlemek için kullanılan alternatif kullanıcı kimliğini belirtir. MQ00\_ALTERNATE\_USER\_AUTHORITY belirtilmemişse, *alternateUserId* boş ya da boş değerli olabilir.

Default user authority is used for connection to the queue manager if MQC.MQ00\_ALTERNATE\_USER\_AUTHORITY is not specified.

```
public MQProcess MQQueueManager.AccessProcess(string processName, int
openOptions);
public MQProcess MQQueueManager.AccessProcess(string processName, int
openOptions, string queueManagerName, string alternateUserId);
```

MQException yayınıyor.

Süreç özniteliklerine ilişkin bilgi edinmek için bu kuyruk yöneticisine ilişkin bir IBM MQ işlemine erişin.

### processName

Açılacak sürecin adı.

### openOptions

İşlemin açılmasını denetleyen seçenekler. Bit kullanan OR kullanılarak eklenebilecek ya da birleştirilebilen geçerli seçenekler şunlardır:

- MQC.MQ00\_FAIL\_IF QUIESCING
- MQC.MQ00\_INQUIRE
- MQC.MQ00\_SET
- MQC.MQ00\_ALTERNATE\_USER\_AUTHORITY

### queueManagerAdı

İşlemin tanımlandığı kuyruk yöneticisinin adı. Kuyruk yöneticisi, sürecin erişmekte olduğu kuyruk yöneticisi aynıysa, boş değerli ya da boş bir kuyruk yöneticisi adı bırakabilirsiniz.

### alternateUserTanıtıcısı

**openOptions** parametresinde MQC.MQ00\_ALTERNATE\_USER\_AUTHORITY belirtilirse, *alternateUserId*, işleme ilişkin yetkiyi denetlemek için kullanılan alternatif kullanıcı kimliğini belirtir. MQ00\_ALTERNATE\_USER\_AUTHORITY belirtilmemişse, *alternateUserId* boş ya da boş değerli olabilir.

Default user authority is used for connection to the queue manager if MQC.MQ00\_ALTERNATE\_USER\_AUTHORITY is not specified.

## MQPropertyDescriptor.NET sınıfı

Use `MQPropertyDescriptor` as a parameter to `MQMessage.GetProperty` and `MQMessage.SetProperty` methods. `MQPropertyDescriptor`, bir `MQMessage` özelliğini açıklar.

### Sınıf

```
System.Object
└─ IBM.WMQ.MQPropertyDescriptor
```

```
public class IBM.WMQ.MQPropertyDescriptor extends System.Object;
```

- [“Özellikler” sayfa 1725](#)
- [“Oluşturucular” sayfa 1726](#)



## Özellikler

Özellikler alınırken MQException için test ediliyor.

**public int Context {get; set;}**

Özelliğin ait olduğu ileti bağlamı. Olası değerler şunlardır:

**MQC.MQPD\_NO\_CONTEXT**

Özellik bir ileti bağlamıyla ilişkilendirilmemiş.

**MQC.MQPD\_USER\_CONTEXT**

Özellik, kullanıcı bağlamıyla ilişkilendirilir.

Kullanıcı yetkilendirildiyse, bir ileti alındığında, kullanıcı bağlamıyla ilişkilendirilen bir özellik kaydedilir. Saklanan bağlamına gönderme yapan sonraki bir Put yöntemi, özelliği yeni iletiye geçirebilir.

**public int CopyOptions {get; set;}**

CopyOptions , özelliğin hangi tip iletiye kopyalanabileceğini açıklar.

Kuyruk yöneticisi, kuyruk yöneticisinin yanlış olarak tanıdığı IBM MQ tanımlı bir özelliği içeren bir ileti aldığı anda, kuyruk yöneticisi CopyOptions alanının değerini düzeltir.

Aşağıdaki seçeneklerin herhangi bir birleşimi belirtilebilir. Değerleri ekleyerek ya da bit akıllı ORseçeneğini kullanarak seçenekleri birleştirin.

**MQC.MQCOPY\_ALL**

Özellik sonraki iletilerin tüm türlerine kopyalanır.

**MQC.MQCOPY\_FORWARD**

Özellik iletilmekte olan bir iletiye kopyalanır.

**MQC.MQCOPY\_PUBLISH**

Özellik, bir ileti yayınlanırken bir abone tarafından alınan iletiye kopyalanır.

**MQC.MQCOPY\_REPLY**

Özellik bir yanıt iletisine kopyalanır.

**MQC.MQCOPY\_REPORT**

Özellik bir rapor iletisine kopyalanır.

**MQC.MQCOPY\_DEFAULT**

Belirtilen değer, başka bir kopya seçeneği belirtilmemiş. Özellik ile sonraki iletiler arasında hiçbir ilişki yok. MQC.MQCOPY\_DEFAULT is always returned for message descriptor properties.

**MQC.MQCOPY\_NONE**

MQC.MQCOPY\_DEFAULT ile aynı

**public int Options { set; }**

Seçenekler , varsayılan olarak CMQC.MQPD\_NONE değerine ayarlanır. Başka bir değer ayarlayamazsınız.

**public int Support { get; set; }**

IBM MQ tanımlı ileti özellikleri için gereken destek düzeyini belirtmek için Support (Destek) seçeneğini belirleyin. Diğer tüm özellikler için destek isteğe bağlıdır. Aşağıdaki değerlerin hiçbiri ya da hiçbiri belirlenebilir.

**MQC.MQPD\_SUPPORT\_OPTIONAL**

Özellik, bir kuyruk yöneticisi tarafından desteklenmese de kabul edilir. İletinin ileti özelliklerini desteklemeyen bir kuyruk yöneticisine akabilmesi için özellik atılabilir. Bu değer, IBM MQ tanımlı olmayan özelliklere de atanır.

**MQC.MQPD\_SUPPORT\_REQUIRED**

Özelliğe ilişkin destek gereklidir. İletiyi IBM MQ-tanımlı özelliği desteklemeyen bir kuyruk yöneticisine koyarsanız, yöntem başarısız olur. Bu, tamamlanma kodu MQC.MQCC\_FAILED ve neden kodu MQC.MQRC\_UNSUPPORTED\_PROPERTY değerini döndürür.

## MQC.MQPD\_SUPPORT\_REQUIRED\_IF\_LOCAL

İleti yerel bir kuyruk için yazılmışsa, özelliğe ilişkin destek gereklidir. If you put the message to a local queue on a queue manager that does not support the IBM MQ-defined property, the method fails. Bu, tamamlanma kodu MQC.MQCC\_FAILED ve neden kodu MQC.MQRC\_UNSUPPORTED\_PROPERTYdeğerini döndürür.

İleti uzak bir kuyruk yöneticisine konursa, denetim yapılmaz.

## Oluşturucular

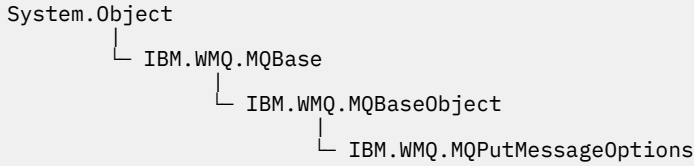
### PropertyDescriptor();

Bir özellik tanımlayıcısı yaratın.

## MQPutMessageOptions.NET sınıfı

İletilerin nasıl gönderileceğini belirlemek için MQPutMessageOptions seçeneğini kullanın. MQDestination.Putdavranışını değiştirir.

## Sınıf



```
public class IBM.WMQ.MQPutMessageOptions extends IBM.WMQ.MQBaseObject;
```

- [“Özellikler” sayfa 1726](#) [“Oluşturucular” sayfa 1728](#)

## Özellikler

Özellikler alınırken MQException için test ediliyor.

**Not:** Bu sınıftaki bazı seçeneklerin davranışı, kullanıcıların bulunduğu ortama bağlıdır. Bu öğeler yıldız işaretiyle (\*) işaretlenir.

### public MQQueue ContextReference {get; set;}

options alanı MQC.MQPMO\_PASS\_IDENTITY\_CONTEXT ya da MQC.MQPMO\_PASS\_ALL\_CONTEXTiçeriyorsa, bu alanı bağlam bilgilerinin içereceği MQQueue 'e gönderme yapmak için bu alanı ayarlayın.

Bu alanın ilk değeri boş değerli.

### public int InvalidDestCount {get;} \*

Genel olarak, dağıtım listeleri için, InvalidDestCount , dağıtım listesindeki kuyruklara gönderilmeyecek iletilerin sayısını belirtir. Bu sayı, açılmamış olan kuyrukları ve başarıyla açılan kuyrukları, ancak koyma işleminin başarısız olduğu kuyrukları içerir.

.NET dağıtım listelerini desteklemez, ancak tek bir kuyruk açılırken InvalidDestCount (InvalidDest Sayısı) ayarı ayarlanır.

### public int KnownDestCount {get;} \*

Genellikle dağıtım listeleri için kullanılan KnownDestCount , yürürlükteki çağrıyı yerel kuyruklara çözümlen kuyruklara başarılı bir şekilde gönderdiğini gösterir.

.NET dağıtım listelerini desteklemez, ancak tek bir kuyruk açılırken InvalidDestCount (InvalidDest Sayısı) ayarı ayarlanır.

## **public int Options {get; set;}**

`MQDestination.put` ve `MQQueueManager.put` yöntemini denetleyen seçenekler. Aşağıdaki değerlerin hiçbiri ya da hiçbiri belirlenebilir. Birden fazla seçenek gerekiyorsa, bit ya da işleç kullanılarak değerler eklenebilir ya da birleştirilebilir.

### **MQC.MQPMO\_ASYNC\_RESPONSE**

Bu seçenek, bazı yanıt verileri ile `MQDestination.put` çağrısının zamanuyumsuz olarak çağrılmasına neden olur.

### **MQC.MQPMO\_DEFAULT\_CONTEXT**

Varsayılan bağlamı iletiyle ilişkilendirin.

### **MQC.MQPMO\_FAIL\_IF\_QUIESCING**

Kuyruk yöneticisi susturulmuş durumda olduğunda hata oluştu.

### **MQC.MQPMO\_LOGICAL\_ORDER \***

İleti gruplarındaki mantıksal iletileri ve bölümleri mantıksal sıralarına koyun.

Yeniden bağlanabilir bir istemcide `MQPMO_LOGICAL_ORDER` seçeneğini kullanırsanız, uygulamaya `MQRC_RECONNECT_INCOMPATIBLE` neden kodu döndürülür.

### **MQC.MQPMO\_NEW\_CORREL\_ID \***

Gönderilen her ileti için yeni bir ilinti tanıtıcısı oluşturun.

### **MQC.MQPMO\_NEW\_MSG\_ID \***

Gönderilen her ileti için yeni bir ileti tanıtıcısı oluşturun.

### **MQC.MQPMO\_NONE**

Seçenek belirtilmedi. Diğer seçeneklerle kullanmayın.

### **MQC.MQPMO\_NO\_CONTEXT**

İletiyle ilişkilendirilecek bağlam yok.

### **MQC.MQPMO\_NO\_SYNCPOINT**

Eşitleme noktası denetimi olmadan bir ileti girin. Eşitleme noktası denetim seçeneği belirtilmediyse, varsayılan olarak bir eşitleme noktası varsayılmaz.

### **MQC.MQPMO\_PASS\_ALL\_CONTEXT**

Bir giriş kuyruğu tutamacından tüm bağlamı geçirin.

### **MQC.MQPMO\_PASS\_IDENTITY\_CONTEXT**

Kimlik bağlamını bir giriş kuyruğu tanıtıcısından geçirin.

### **MQC.MQPMO\_RESPONSE\_AS\_Q\_DEF**

Bir `MQDestination.put` çağrısı için, bu seçenek koyma yanıt tipini kuyruğun `DEFPRESP` özneliğinden alır.

Bir `MQQueueManager.put` çağrısı için bu seçenek, çağrıların zamanuyumlu hale getirilmesine neden olur.

### **MQC.MQPMO\_RESPONSE\_AS\_TOPIC\_DEF**

`MQC.MQPMO_RESPONSE_AS_TOPIC_DEF`, konu nesneleriyle birlikte kullanılmak üzere `MQC.MQPMO_RESPONSE_AS_Q_DEF` ile eşanımlıdır.

### **MQC.MQPMO\_RETAIN**

Gönderilmekte olan yayın kuyruk yöneticisi tarafından alıkonabilecektir. Bu seçenek kullanılırsa ve yayın saklanamazsa, ileti yayınlanmaz ve çağrı `MQC.MQRC_PUT_NOT_RETAINED` ile başarısız olur.

`MQSubscription.RequestPublicationUpdate` yöntemini çağırarak, bu yayının bir kopyasını yayımlandıktan sonra isteyin. Kaydedilen yayın, `MQC.MQSO_NEW_PUBLICATIONS_ONLY` seçeneğini ayarlamadan abonelik yaratan uygulamalara gönderilir. Saklanacak yayın olup olmadığını öğrenmek için, alındığı zaman yayının `MQIsRetained` ileti özelliğini denetleyin.

Alıkonan yayınlar bir abone tarafından istendiğinde, kullanılan abonelik, konu dizisinde bir genel arama karakteri içerebilir. Konu ağacında, abonelikle eşleşen birden çok yayın varsa, bu yayınlar gönderilmektedir.

#### **MQC.MQPMO\_SET\_ALL\_CONTEXT**

Uygulamadaki tüm bağlamı ayarlayın.

#### **MQC.MQPMO\_SET\_IDENTITY\_CONTEXT**

Kimlik bağlamını uygulamadan ayarlayın.

#### **MQC.MQPMO\_SYNC\_RESPONSE**

Bu seçenek, MQDestination.put ya da MQQueueManager.put çağrısının tam yanıt verisiyle zaman uyumlu hale getirilmesine neden olur.

#### **MQC.MQPMO\_SUPPRESS\_REPLYTO**

Yayının ReplyToQueueName ve ReplyToQueueManagerAd alanlarına veri girildiğinde, bu bilgiler abonelere iletilmez. Bu seçenek, ReplyToQueueName gerektiren bir rapor seçeneği ile birlikte kullanılırsa, çağrı MQC.MQRC\_MISSING\_REPLY\_TO\_Q ile başarısız olur.

#### **MQC.MQPMO\_SYNCPOINT**

Eşitleme noktası denetimi içeren bir ileti girin. İleti, iş birimi kesinleştirilinceye kadar iş biriminin dışında görünmez. İş birimi yedeklendiyse, ileti silinir.

#### **public int RecordFields {get; set;} \***

Dağıtım listeleriyle ilgili bilgiler. Dağıtım listeleri .NET' ta desteklenmiyor.

#### **public string ResolvedQueueManagerName {get;}**

Kuyruk yöneticisi tarafından belirlenen bir çıkış alanı, uzak kuyruk adı tarafından belirlenen kuyruğa sahip olan kuyruk yöneticisi adına göre belirlenir. Kuyruk uzak bir kuyruksa, ResolvedQueueManagerName kuyruğun eriştiği kuyruk yöneticisi adından farklı olabilir.

Boş olmayan bir değer yalnızca, nesne tek bir kuyruksa döndürülür. Nesne bir dağıtım listesi ya da bir konu ise, döndürülen değer tanımsız olur.

#### **public string ResolvedQueueName {get;}**

Kuyruk yöneticisi tarafından, iletinin yerleştirileceği kuyruğun adına ayarlanacak çıkış alanı. Açılan kuyruk bir diğer ad ya da model kuyruğuydu ise, ResolvedQueueAd kuyruğu açmak için kullanılan addan farklı olabilir.

Boş olmayan bir değer yalnızca, nesne tek bir kuyruksa döndürülür. Nesne bir dağıtım listesi ya da bir konu ise, döndürülen değer tanımsız olur.

#### **public int UnknownDestCount {get;} \***

Genellikle dağıtım listeleri için kullanılan UnknownDestCount , kuyruk yöneticisi tarafından ayarlanan bir çıkış alanıdır. Yürürlükteki çağrının, uzak kuyruklara çözümlen kuyruklara başarılı bir şekilde gönderdiği ileti sayısını bildirir.

.NET dağıtım listelerini desteklemez, ancak tek bir kuyruk açılırken InvalidDestCount (InvalidDest Sayısı) ayarı ayarlanır.

### **Oluşturucular**

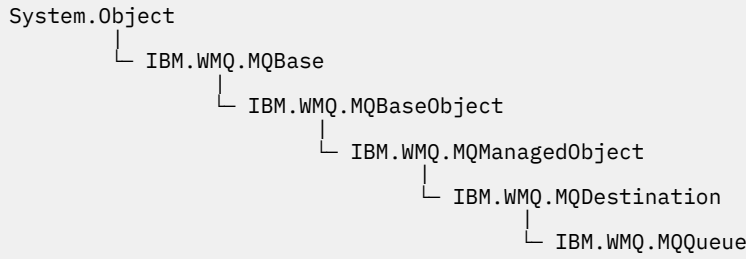
#### **public MQPutMessageOptions();**

Seçenek kümesi olmadan yeni bir MQPutMessageOptions nesnesi oluşturun ve boş bir ResolvedQueueAd ve ResolvedQueueManagerName girin.

### **MQQueue.NET sınıfı**

İleti göndermek ve almak için MQQueue , bir IBM MQ kuyruğuna ilişkin sorgu özneliklerini de kullanabilirsiniz. Bir oluşturucu ya da MQQueueManager .AccessProcess yöntemi kullanarak bir MQQueue nesnesi yaratın.

### **Sınıf**



```
public class IBM.WMQ.MQQueue extends IBM.WMQ.MQDestination;
```

- “Özellikler” sayfa 1729
- “Yöntemler” sayfa 1731
- “Oluşturucular” sayfa 1733

## Özellikler

Özellikler alınırken MQException için test ediliyor.

**public int ClusterWorkLoadPriority {get;}**

Kuyruğun önceliğini belirler. Bu parametre yalnızca yerel, uzak ve diğer ad kuyrukları için geçerlidir.

**public int ClusterWorkLoadRank {get;}**

Kuyruğun sırasını belirtir. Bu parametre yalnızca yerel, uzak ve diğer ad kuyrukları için geçerlidir.

**public int ClusterWorkLoadUseQ {get;}**

Hedef kuyruk yerel bir yönetim ortamına ve en az bir uzak küme örneğine sahip olduğunda, MQPUT işleminin davranışını belirtir. MQPUT bir küme kanalından kaynaklanırsa bu değiştirge geçerli değildir. Bu değiştirge yalnızca yerel kuyruklar için geçerlidir.

**public DateTime CreationDateTime {get;}**

Bu kuyruğun yaratıldığı tarih ve saat.

**public int CurrentDepth {get;}**

Şu anda kuyruklardaki ileti sayısını alır. Bu değer, bir put çağrısı sırasında ve bir alma çağrısının geri çekmesi sırasında artırılır. Bu, göz atmayan bir alma sırasında ve geri alma sırasında oluşan bir arama sırasında azaltılır.

**public int DefinitionType {get;}**

Kuyruğun nasıl tanımlandığını. Olası değerler şunlardır:

- MQC.MQQDT\_PREDEFINED
- MQC.MQQDT\_PERMANENT\_DYNAMIC
- MQC.MQQDT\_TEMPORARY\_DYNAMIC

**public int InhibitGet {get; set;}**

Bu kuyruktaki ya da bu konu için ileti alıp alamayacağınızı denetler. Olası değerler şunlardır:

- MQC.MQQA\_GET\_INHIBITED
- MQC.MQQA\_GET\_ALLOWED

**public int InhibitPut {get; set;}**

Bu kuyruğa ya da bu konuya ilişkin iletiler yerleştirip koyamayacağınızı denetler. Olası değerler şunlardır:

- MQQA\_PUT\_INHIBITED
- MQQA\_PUT\_ALLOWED

**public int MaximumDepth {get;}**

Herhangi bir zamanda kuyrukta bulunabilecek ileti sayısı üst sınırı. An attempt to put a message to a queue that already contains this many messages fails with reason code MQC.MQRC\_Q\_FULL.

**public int MaximumMessageLength {get;}**

Bu kuyruktaki her iletide var olabilen uygulama verilerinin uzunluk üst sınırı. Bu değerden daha büyük bir ileti koyma girişimi başarısız olur. Neden kodu: MQC.MQRC\_MSG\_TOO\_BIG\_FOR\_Q.

**public int NonPersistentMessageClass {get;}**

Bu kuyruğa koyulan kalıcı olmayan iletilerin güvenilirliği düzeyi.

**public int OpenInputCount {get;}**

Şu anda kuyruktan ileti kaldırmak için geçerli olan tutamaçların sayısı. OpenInputCount , yerel kuyruk yöneticisi tarafından bilinen geçerli giriş tanıtıcılarının toplam sayısıdır; yalnızca uygulama tarafından oluşturulan tutamaçlar değildir.

**public int OpenOutputCount {get;}**

Kuyruğa ileti eklemek için geçerli olan tutamaçların sayısı. OpenOutputCount , yerel kuyruk yöneticisi tarafından bilinen geçerli çıkış tanıtıcılarının toplam sayısıdır; yalnızca uygulama tarafından yaratılan tanıtıcıların tanıtıcısıdır.

**public int QueueAccounting {get;}**

Kuyruğa ilişkin hesap bilgileri toplamanın etkinleştirilip etkinleştirilmeyeceğini belirler.

**public int QueueMonitoring {get;}**

Kuyruğun izlenmesini etkinleştirip etkinleştirmeyip etkinleştirmeyeceğini belirler.

**public int QueueStatistics {get;}**

Kuyruğa ilişkin istatistik toplamanın etkinleştirilip etkinleştirilmeyeceğini belirler.

**public int QueueType {get;}**

Bu kuyruğun tipi aşağıdaki değerlerden birine sahip:

- MQC.MQQT\_ALIAS
- MQC.MQQT\_LOCAL
- MQC.MQQT\_REMOTE
- MQC.MQQT\_CLUSTER

**public int Shareability {get;}**

Kuyruğun birden çok kez giriş için açılıp açılmayacağı. Olası değerler şunlardır:

- MQC.MQQA\_SHAREABLE
- MQC.MQQA\_NOT\_SHAREABLE

**public string TPIPE {get;}**

The TPIPE name used for communication with OTMA using the IBM MQ IMS bridge.

**public int TriggerControl {get; set;}**

Tetikleme iletilerinin bir başlatma kuyruğuna yazılıp yazılmayacağı, kuyruğa hizmet vermek üzere bir uygulama başlatmak için. Olası değerler şunlardır:

- MQC.MQTC\_OFF
- MQC.MQTC\_ON

**public string TriggerData {get; set;}**

Kuyruk yöneticisinin tetikleme iletilerine eklediği serbest biçimli veriler. Bu kuyruğa gelen bir ileti, bir tetikleme iletilerinin başlangıç kuyruğuna yazılmasına neden olduğunda TriggerData ekler. Dizilimin izin verilen uzunluk üst sınırı MQC.MQ\_TRIGGER\_DATA\_LENGTH tarafından verilir.

**public int TriggerDepth {get; set;}**

Tetikleme tipi MQC.MQTT\_DEPTH olarak ayarlandığında bir tetikleme iletilerinin yazılmadan önce kuyruğun üzerinde olması gereken ileti sayısı.

**public int TriggerMessagePriority {get; set;}**

İletilerin hangi iletilerinin oluşturulacağı, tetikleme iletilerinin oluşturulmasına katkıda bulunmamasını sağlar. Yani, kuyruk yöneticisi bir tetikleme oluşturulup oluşturulmayacağına karar verirken bu iletileri yoksayar. Sıfır değeri, tüm iletilerinin, tetikleme iletilerinin oluşturulmasına katkıda bulunmasına neden olur.

## **public int TriggerType {get; set;}**

Bu kuyruğa gelen iletilerin bir sonucu olarak tetikleme iletilerinin yazıldığı koşullar. Olası değerler şunlardır:

- MQC.MQTT\_NONE
- MQC.MQTT\_FIRST
- MQC.MQTT\_EVERY
- MQC.MQTT\_DEPTH

## **Yöntemler**

### **public void Get(MQMessage message);**

### **public void Get(MQMessage message, MQGetMessageOptions getMessageOptions);**

### **public void Get(MQMessage message, MQGetMessageOptions getMessageOptions, int MaxMsgSize);**

MQException yayınlıyor.

Kuyruktan ileti alır.

Alma işlemi başarısız olursa, MQMessage nesnesi değişmeden kalır. Başarılı olursa, MQMessage ' un ileti tanımlayıcısı ve ileti verileri kısımları, gelen iletiden ileti tanımlayıcısı ve ileti verileriyle değiştirilir.

Belirli bir MQQueueManager 'den IBM MQ ' e yapılan tüm çağrılar zamanuyumlu olur. Bu nedenle, bir bekleme işlemi gerçekleştirdiğinizde, aynı MQQueueManager ' u kullanan diğer tüm iş parçacıklarının, Alma işlemi gerçekleştirilinceye kadar IBM MQ çağrılarının yapılması engellenir. IBM MQ ' a aynı anda erişmek için birden çok iş parçacığının olması gerekiyorsa, her bir iş parçacığının kendi MQQueueManager nesnesi yaratması gerekir.

### **ileti**

İleti tanımlayıcısını ve döndürülen ileti verilerini içerir. İleti tanımlayıcısındaki bazı alanlar giriş değiştirgeleri olur. MessageId ve CorrelationId giriş parametrelerinin gerektiği şekilde ayarlandığından emin olmak önemlidir.

Yeniden bağlantılanabilir istemci, MQGM\_SYNCPOINT altında alınan iletiler için, başarılı yeniden bağlantı sonrasında MQRC\_BACKED\_OUT neden kodunu döndürür.

### **getMessageSeçenekleri**

Alma işleminin denetlenmesini denetleyen seçenekler.

MQC.MQGMO\_CONVERT seçeneğinin kullanılması, tek byte 'lık karakter kodlarından çift byte kodlarına dönüştürülürken MQC.MQRC\_CONVERTED\_STRING\_TOO\_BIG neden koduyla bir kural dışı duruma neden olabilir. Bu durumda, ileti dönüştürmeden arabelleğe kopyalanır.

*getMessageOptions* belirtilmemişse, kullanılan ileti seçeneği MQGMO\_NOWAIT olur.

Yeniden bağlanabilir bir istemcide MQGMO\_LOGICAL\_ORDER seçeneğini kullanırsanız, MQRC\_RECONNECT\_INCOMPATIBLE neden kodu döndürülür.

### **MaxMsgBoyutu**

Bu ileti nesnesinin alınabileceği en büyük ileti. Kuyruktaki ileti bu boyuttan büyükse, iki şeyden biri oluşur:

- If the MQGMO\_ACCEPT\_TRUNCATED\_MSG flag is set in the MQGetMessageOptions object, the message is filled with as much of the message data as possible. MQCC\_WARNING tamamlanma kodu ve MQRC\_TRUNCATED\_MSG\_ACCEPTED neden koduyla bir kural dışı durum yayınlandı.
- MQGMO\_ACCEPT\_TRUNCATED\_MSG işareti ayarlanmamış ise, ileti kuyruktaki kalır. MQCC\_WARNING tamamlanma kodu ve MQRC\_TRUNCATED\_MSG\_FAILED neden koduyla bir kural dışı durum yayınlandı.

*MaxMsgSize* belirtilmemişse, tüm ileti alınır.

```
public void Put(MQMessage message);  
public void Put(MQMessage message, MQPutMessageOptions putMessageOptions);
```

MQException yayınlıyor.

Bir iletiyi kuyruğa yerleştirir.

Put çağrısından sonra MQMessage nesnesinde yapılan değişiklikler, IBM MQ kuyruğunda ya da yayınındaki gerçek iletiyi etkilemez.

Put , MQMessage nesnesinin MessageId ve CorrelationId özelliklerini güncelleştirir ve ileti verilerini temizlememektedir. Daha fazla Put ya da Get çağrısı, MQMessage nesnesindeki güncellenmiş bilgilere gönderme yapıyor. Örneğin, aşağıdaki kod parçasısında ilk ileti a ve ikinci abiletisini içerir.

```
msg.WriteString("a");  
q.Put(msg, pmo);  
msg.WriteString("b");  
q.Put(msg, pmo);
```

### ileti

İleti tanımlayıcı verilerini içeren bir MQMessage nesnesi ve gönderilecek ileti. İleti tanımlayıcısı bu yöntemin bir sonucu olarak değiştirilebilir. Bu yöntemin tamamlanmasından hemen sonra ileti tanımlayıcısındaki değerler, kuyruğa konulan ya da konu üzerinde yayınlanan değerlerdir.

Yeniden bağlanabilir bir istemciye döndürülen neden kodları şunlardır:

- MQRC\_CALL\_INTERRUPTED if the connection is broken while running a Put call on a persistent message and the reconnection is successful.
- Bağlantı, kalıcı olmayan bir iletiyle (bkz. [Uygulama Kurtarma](#) ) bir put çağrısı çalıştırırken bağlantı başarılı olursa MQRC\_NONE .

### putMessageSeçenekleri

Put işleminin denetlenmesini denetleyen seçenekler.

*putMessageOptions* belirtilmemişse, varsayılan MQPutMessageOptions eşgörünümü kullanılır.

Yeniden bağlanabilir bir istemcide MQPMO\_LOGICAL\_ORDER seçeneğini kullanırsanız, MQRC\_RECONNECT\_INCOMPATIBLE neden kodu döndürülür.

**Not:** Basitlik ve performans için, kuyruğa tek bir ileti koymak isterseniz, MQQueueManager.Put nesnesini kullanın. Bunun için bir MQQueue nesnesine sahip olmanız.

```
public void PutForwardMessage(MQMessage message);  
public void PutForwardMessage(MQMessage message, MQPutMessageOptions  
putMessageOptions);
```

Throws MQException

Kuyruğa iletilmekte olan bir ileti yazın; burada *message* , özgün iletidir.

### ileti

İleti tanımlayıcı verilerini içeren bir MQMessage nesnesi ve gönderilecek ileti. İleti tanımlayıcısı bu yöntemin bir sonucu olarak değiştirilebilir. Bu yöntemin tamamlanmasından hemen sonra ileti tanımlayıcısındaki değerler, kuyruğa konulan ya da konu üzerinde yayınlanan değerlerdir.

Yeniden bağlanabilir bir istemciye döndürülen neden kodları şunlardır:

- MQRC\_CALL\_INTERRUPTED if the connection is broken while running a Put call on a persistent message and the reconnection is successful.
- Bağlantı, kalıcı olmayan bir iletiyle (bkz. [Uygulama Kurtarma](#) ) bir put çağrısı çalıştırırken bağlantı başarılı olursa MQRC\_NONE .

### putMessageSeçenekleri

Put işleminin denetlenmesini denetleyen seçenekler.

*putMessageOptions* belirtilmemişse, varsayılan MQPutMessageOptions eşgörünümü kullanılır.



Yeniden bağlanabilir bir istemcide MQPMO\_LOGICAL\_ORDER seçeneğini kullanırsanız, MQRC\_RECONNECT\_INCOMPATIBLE neden kodu döndürülür.

```
public void PutReplyMessage(MQMessage message)  
public void PutReplyMessage(MQMessage message, MQPutMessageOptions  
putMessageOptions)
```

MQException yayınlıyor.

Kuyruğa bir yanıt iletisi koyun; burada *message* özgün iletidir.

#### **ileti**

İleti tanımlayıcısını ve döndürülen ileti verilerini içerir. İleti tanımlayıcısındaki bazı alanlar giriş değişirgeleri olur. MessageId ve CorrelationId giriş parametrelerinin gerektiği şekilde ayarlandığından emin olmak önemlidir.

Yeniden bağlantılanabilir istemci, MQGM\_SYNCPOINT altında alınan iletiler için, başarılı yeniden bağlantı sonrasında MQRC\_BACKED\_OUT neden kodunu döndürür.

#### **putMessageSeçenekleri**

Put işleminin denetlenmesini denetleyen seçenekler.

*putMessageOptions* belirtilmemişse, varsayılan MQPutMessageOptions eşgörünümü kullanılır.

Yeniden bağlanabilir bir istemcide MQPMO\_LOGICAL\_ORDER seçeneğini kullanırsanız, MQRC\_RECONNECT\_INCOMPATIBLE neden kodu döndürülür.

```
public void PutReportMessage(MQMessage message)  
public void PutReportMessage(MQMessage message, MQPutMessageOptions  
putMessageOptions)
```

MQException yayınlıyor.

Kuyruğa bir rapor iletisi koyun; burada *message* özgün iletidir.

#### **ileti**

İleti tanımlayıcısını ve döndürülen ileti verilerini içerir. İleti tanımlayıcısındaki bazı alanlar giriş değişirgeleri olur. MessageId ve CorrelationId giriş parametrelerinin gerektiği şekilde ayarlandığından emin olmak önemlidir.

Yeniden bağlantılanabilir istemci, MQGM\_SYNCPOINT altında alınan iletiler için, başarılı yeniden bağlantı sonrasında MQRC\_BACKED\_OUT neden kodunu döndürür.

#### **putMessageSeçenekleri**

Put işleminin denetlenmesini denetleyen seçenekler.

*putMessageOptions* belirtilmemişse, varsayılan MQPutMessageOptions eşgörünümü kullanılır.

Yeniden bağlanabilir bir istemcide MQPMO\_LOGICAL\_ORDER seçeneğini kullanırsanız, MQRC\_RECONNECT\_INCOMPATIBLE neden kodu döndürülür.

## **Oluşturucular**

```
public MQQueue MQQueueManager.AccessQueue(string queueName, int openOptions);  
public MQQueue MQQueueManager.AccessQueue(string queueName, int openOptions,  
string queueManagerName, string dynamicQueueName, string alternateUserId);
```

MQException yayınlıyor.

Bu kuyruk yöneticilerindeki bir kuyruğa erişir.

İletiler alabilir ya da bunlara göz atabilir, iletileri yerleştirebilir, kuyruğun özneliklerini sorgulayabilir ya da kuyruğun özneliklerini ayarlayabilirsiniz. Kuyruk adı bir model kuyruğa, dinamik bir yerel kuyruk yaratılır. Dinamik kuyruğun adını bulmak için, sonuçtaki MQQueue nesnesinin name özneliğini sorgulayın.

#### **queueName**

Açılacak kuyruğun adı.

## openOptions

Kuyruğun açıldığını denetleyen seçenekler.

### **MQC.MQOO\_ALTERNATE\_USER\_AUTHORITY**

Belirtilen kullanıcı kimliğiyle geçerliliğini denetleyin.

### **MQC.MQOO\_BIND\_AS\_QDEF**

Kuyruk için varsayılan bağ tanımını kullan.

### **MQC.MQOO\_BIND\_NOT\_FIXED**

Belirli bir hedef için bağ tanımlamayın.

### **MQC.MQOO\_BIND\_ON\_OPEN**

Kuyruk açıldığında bağ tanımlama tanıtıcısı hedef değerine bağlanır.

### **MQC.MQOO\_BROWSE**

İletiyeye göz atmak için açın.

### **MQC.MQOO\_FAIL\_IF QUIESCING**

Kuyruk yöneticisi susturulmuş durumda olduğunda hata oluştu.

### **MQC.MQOO\_INPUT\_AS\_Q\_DEF**

Kuyruk tanımlı varsayılan değeri kullanarak ileti almak için açın.

### **MQC.MQOO\_INPUT\_SHARED**

Paylaşılan erişimi olan iletileri almak için açın.

### **MQC.MQOO\_INPUT\_EXCLUSIVE**

Özel erişimi olan iletileri almak için açın.

### **MQC.MQOO\_INQUIRE**

Özellikleri sorgulamak istiyorsanız, sorgu için açma işlemi için açın.

### **MQC.MQOO\_OUTPUT**

İleti koymak için açın.

### **MQC.MQOO\_PASS\_ALL\_CONTEXT**

Tüm bağlamın iletilmesine izin verir.

### **MQC.MQOO\_PASS\_IDENTITY\_CONTEXT**

Kimlik bağlamının iletilmesine izin verir.

### **MQC.MQOO\_SAVE\_ALL\_CONTEXT**

İleti alındığında bağlam kaydedin.

### **MQC.MQOO\_SET**

Özellikleri ayarlamak istiyorsanız, özellikleri ayarlamak için açın-özellikleri ayarlamak için.

### **MQC.MQOO\_SET\_ALL\_CONTEXT**

Tüm bağlamın ayarlanmasını sağlar.

### **MQC.MQOO\_SET\_IDENTITY\_CONTEXT**

Kimlik bağlamının ayarlanmasını sağlar.

## queueManagerAdı

Kuyruğun tanımlı olduğu kuyruk yöneticisinin adı. Tamamen boş ya da boş değer olan bir ad, MQQueueManager nesnesinin bağlı olduğu kuyruk yöneticisini belirtir.

## dynamicQueueAdı

queueName bir model kuyruğunun adını belirtmedikçe, *dynamicQueueName* yoksayıdır. Ters durumda, *dynamicQueueName*, yaratılacak dinamik kuyruğun adını belirtir. queueName, bir model kuyruğunun adını belirtiyorsa, boş değerli ya da boş değerli ad kullanılamaz. Addaki son boş olmayan karakter bir yıldız işareti ( \*) ise, kuyruk yöneticisi yıldız işaretini bir karakter dizisiyle değiştirir. Kuyruk için oluşturulan adın, bu kuyruk yöneticisiyle benzersiz olduğunu garanti eder.

## alternateUserTanıtıcısı

openOptions değiştirilmesinde MQC.MQOO\_ALTERNATE\_USER\_AUTHORITY belirtilirse, *alternateUserId*, açık ara yetkilendirmeyi denetlemek için kullanılan diğer kullanıcı kimliğini belirtir. MQC.MQOO\_ALTERNATE\_USER\_AUTHORITY belirtilmemişse, *alternateUserId* boş bırakılabilir ya da boş bırakılabilir.

```
public MQQueue(MQQueueManager queueManager, string queueName, int openOptions,  
string queueManagerName, string dynamicQueueName, string alternateUserId);
```

MQException yayınlıyor.

queueManager üzerindeki bir kuyruğa erişir.

İletiler alabilir ya da bunlara göz atabilir, iletileri yerleştirebilir, kuyruğun özniteliklerini sorgulayabilir ya da kuyruğun özniteliklerini ayarlayabilirsiniz. Kuyruk adı bir model kuyruğsa, dinamik bir yerel kuyruk yaratılır. Dinamik kuyruğun adını bulmak için, sonuçtaki MQQueue nesnesinin name özniteliğini sorgulayın.

#### **queueManager**

Kuyruğa erişmek için kuyruk yöneticisi.

#### **queueName**

Açılacak kuyruğun adı.

#### **openOptions**

Kuyruğun açıldığını denetleyen seçenekler.

##### **MQC.MQOO\_ALTERNATE\_USER\_AUTHORITY**

Belirtilen kullanıcı kimliğiyle geçerliliğini denetleyin.

##### **MQC.MQOO\_BIND\_AS\_QDEF**

Kuyruk için varsayılan bağ tanımını kullan.

##### **MQC.MQOO\_BIND\_NOT\_FIXED**

Belirli bir hedef için bağ tanımlamayın.

##### **MQC.MQOO\_BIND\_ON\_OPEN**

Kuyruk açıldığında bağ tanımlama tanıtıcısı hedef değerine bağlanır.

##### **MQC.MQOO\_BROWSE**

İletiyeye göz atmak için açın.

##### **MQC.MQOO\_FAIL\_IF QUIESCING**

Kuyruk yöneticisi susturulmuş durumda olduğunda hata oluştu.

##### **MQC.MQOO\_INPUT\_AS\_Q\_DEF**

Kuyruk tanımlı varsayılan değeri kullanarak ileti almak için açın.

##### **MQC.MQOO\_INPUT\_SHARED**

Paylaşılan erişimi olan iletileri almak için açın.

##### **MQC.MQOO\_INPUT\_EXCLUSIVE**

Özel erişimi olan iletileri almak için açın.

##### **MQC.MQOO\_INQUIRE**

Özellikleri sorgulamak istiyorsanız, sorgu için açma işlemi için açın.

##### **MQC.MQOO\_OUTPUT**

İleti koymak için açın.

##### **MQC.MQOO\_PASS\_ALL\_CONTEXT**

Tüm bağlamın iletilmesine izin verir.

##### **MQC.MQOO\_PASS\_IDENTITY\_CONTEXT**

Kimlik bağlamının iletilmesine izin verir.

##### **MQC.MQOO\_SAVE\_ALL\_CONTEXT**

İleti alındığında bağlam kaydedin.

##### **MQC.MQOO\_SET**

Özellikleri ayarlamak istiyorsanız, özellikleri ayarlamak için açın-özellikleri ayarlamak için.

##### **MQC.MQOO\_SET\_ALL\_CONTEXT**

Tüm bağlamın ayarlanmasını sağlar.

##### **MQC.MQOO\_SET\_IDENTITY\_CONTEXT**

Kimlik bağlamının ayarlanmasını sağlar.

### queueManagerAdı

Kuyruğun tanımlı olduğu kuyruk yöneticisinin adı. Tamamen boş ya da boş değer olan bir ad, MQQueueManager nesnesinin bağlı olduğu kuyruk yöneticisini belirtir.

### dynamicQueueAdı

queueName bir model kuyruğunun adını belirtmedikçe, *dynamicQueueName* yoksayılr. Ters durumda, *dynamicQueueName*, yaratılacak dinamik kuyruğun adını belirtir. queueName, bir model kuyruğunun adını belirtiyorsa, boş değerli ya da boş değerli ad kullanılamaz. Addaki son boş olmayan karakter bir yıldız işareti (\*) ise, kuyruk yöneticisi yıldız işaretini bir karakter dizisiyle değiştirir. Kuyruk için oluşturulan adın, bu kuyruk yöneticisiyle benzersiz olduğunu garanti eder.

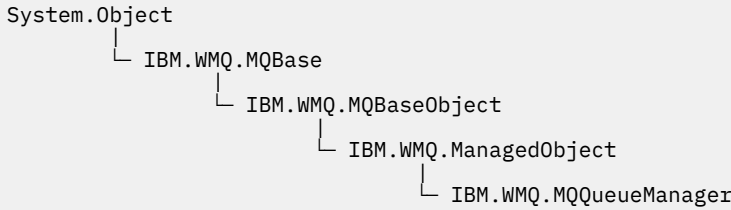
### alternateUserTanıtıcısı

openOptions değiştirgesinde MQC.MQ00.ALTERNATE.USER.AUTHORITY belirtilirse, *alternateUserId*, açık ara yetkilendirmeyi denetlemek için kullanılan diğer kullanıcı kimliğini belirtir. MQC.MQ00.ALTERNATE.USER.AUTHORITY belirtilmemişse, *alternateUserId* boş bırakılabilir ya da boş bırakılabilir.

## MQQueueManager.NET sınıfı

Bir kuyruk yöneticisine ve erişim kuyruğu yöneticisi nesnelere bağlanmak için MQQueueManager kullanın. Ayrıca hareketleri de denetler. MQQueueManager oluşturucusu, bir istemci ya da sunucu bağlantısı yaratır.

### Sınıf



```
public class IBM.WMQ.MQQueueManager extends IBM.WMQ.MQManagedObject;
```

- [“Özellikler” sayfa 1736](#)
- [“Yöntemler” sayfa 1739](#)
- [“Oluşturucular” sayfa 1745](#)

### Özellikler

Özellikler alınırken MQException için test ediliyor.

```
public int AccountingConnOverride {get;}
```

Uygulamaların MQI muhasebe ve kuyruk hesap değerlerinin ayarını geçersiz kılıp geçersiz kılamayacağı.

```
public int AccountingInterval {get;}
```

Ara muhasebe kayıtlarının yazıldığı süre (saniye cinsinden).

```
public int ActivityRecording {get;}
```

Etkinlik raporlarının oluşturulmasını denetler.

```
public int AdoptNewMCACheck {get;}
```

MCA 'nın yeni bir gelen kanalı algılandığında kabul edilip edilmediğini belirlemek için hangi öğelerin denetleneceğini belirtir. Kabul edilmesi için, MCA adının etkin bir MCA 'nın adıyla eşleşmesi gerekir.

```
public int AdoptNewMCAInterval {get;}
```

Yeni kanalın artık kalan kanalı sona erdirmek için bekleyeceği süre (saniye olarak).

**public int AdoptNewMCAType {get;}**

AdoptNewMCACheck değeriyle eşleşen yeni bir gelen kanal isteği saptandığında, artık bir MCA yönetim ortamının (yeniden başlatılıp başlatılmayacağı) benimsenip benimsenmeyeceğini (yeniden başlatılacak).

**public int BridgeEvent {get;}**

IMS köprüsü olaylarının oluşturulup oluşturulmayacağı.

**public int ChannelEvent {get;}**

Kanal olaylarının oluşturulup oluşturulmayacağı.

**public int ChannelInitiatorControl {get;}**

Kuyruk yöneticisi başlatıldığında, kanal başlatıcının otomatik olarak başlatılıp başlatılmayacağını.

**public int ChannelInitiatorAdapters {get;}**

IBM MQ çağrılarını işlemek için kullanılan bağdaştırıcı alt görevlerinin sayısı.

**public int ChannelInitiatorDispatchers {get;}**

Kanal başlatıcısı için kullanılacak dağıtıcıların sayısı.

**public int ChannelInitiatorTraceAutoStart {get;}**

Kanal başlatıcı izinin otomatik olarak başlatılıp başlatılmayacağını belirler.

**public int ChannelInitiatorTraceTableSize {get;}**

Kanal başlatıcısının izleme verileri alanının megabayt cinsinden büyüklüğü.

**public int ChannelMonitoring {get;}**

Kanal izlemesinin kullanılıp kullanılmayacağını.

**public int ChannelStatistics {get;}**

Kanallar için istatistik verilerinin toplanmasını denetler.

**public int CharacterSet {get;}**

Kuyruk yöneticisinin CCSID ' yi (kodlanmış karakter takımı tanıtıcısı) döndürür. CharacterSet , kuyruk yöneticisi tarafından uygulama programlama arabirimindeki tüm karakter dizgisi alanları için kullanılır.

**public int ClusterSenderMonitoring {get;}**

Otomatik olarak tanımlanmış küme gönderen kanallarına ilişkin çevrimiçi izleme verilerinin toplanmasını denetler.

**public int ClusterSenderStatistics {get;}**

Otomatik olarak tanımlanmış küme gönderen kanallarına ilişkin istatistik verileri toplamanın denetlenmesini sağlar.

**public int ClusterWorkLoadMRU {get;}**

Giden küme kanalı sayısı üst sınırı.

**public int ClusterWorkLoadUseQ {get;}**

The default value of the MQQueue property, ClusterWorkLoadUseQ, if it specifies a value of QMGR.

**public int CommandEvent {get;}**

Komut olaylarının oluşturulup oluşturulmayacağını belirtir.

**public string CommandInputQueueName {get;}**

Kuyruk yöneticilikinde tanımlı olan komut giriş kuyruğunun adını döndürür. Bu kuyruğa erişim yetkisi varsa, uygulamalar bu kuyruğa komut gönderebilirler.

**public int CommandLevel {get;}**

Kuyruk yöneticisinin işlev düzeyini gösterir. Belirli bir işlev düzeyine karşılık gelen işlev kümesi altyapıya bağlıdır. Belirli bir altyapıda, tüm kuyruk yöneticilerine ortak olan en düşük işlevsel düzeydeki işlevleri destekleyen her kuyruk yöneticisine güvenebilirsiniz.

**public int CommandLevel {get;}**

Kuyruk yöneticisi başlatıldığında komut sunucusunun da otomatik olarak başlatılıp başlatılmayacağını.

**public string DNSGroup {get;}**

Artık kullanılmadı.

**public int DNSWLM {get;}**

Artık kullanılmadı.

**public int IPAddressVersion {get;}**

Kanal bağlantısı için kullanılacak IP protokolü (IPv4 ya da IPv6).

**public boolean IsConnected {get;}**

isConnected(isConnected) değerinin değerini döndürür.

Değeri true ise, kuyruk yöneticisine yönelik bir bağlantı yapılmıştır ve bozuk olduğu bilinmez. IsConnected çağrıları, kuyruk yöneticisine etkin bir şekilde ulaşmayı denemez; dolayısıyla, fiziksel bağlantı kopabilir, ancak IsConnected yine de true değerini döndürebilir. IsConnected durumu yalnızca, etkinlik olduğunda (örneğin, ileti almak, ileti almak), kuyruk yöneticisinde gerçekleştirildiğinde güncellenir.

Değeri false (yanlış) ise, kuyruk yöneticisine yönelik bir bağlantı yapılmamıştır ya da kopmuştur ya da bağlantısı kesilmiştir.

**public int KeepAlive {get;}**

Bağlantının diğer ucunun kullanılabilir olup olmadığını denetlemek için TCP KEPENSAILE olanağının kullanılıp kullanılmayacağını belirler. Kullanılmayan bir kanal varsa, kanal kapatılır.

**public int ListenerTimer {get;}**

Bir APPC ya da TCP/IP hatasından sonra dinleyiciyi yeniden başlatmak için IBM MQ ile denemeler arasındaki zaman aralığı (saniye).

**public int LoggerEvent {get;}**

Kaydedici olaylarının oluşturulup oluşturulmayacağı.

**public string LU62ARMSuffix {get;}**

The suffix of the APPCPM member of SYS1.PARMLIB. Bu sonek, bu kanal başlatıcı için LUADD adını alır. Otomatik yeniden başlatma yöneticisi (ARM) kanal başlatıcısını yeniden başlattığında, z/OS komut SET APPC=xx yayınlanır.

**public string LUGroupName {get; z/os}**

Kuyruk paylaşım grubu için gelen iletimleri işleyen LU 6.2 dinleyicisi tarafından kullanılacak soysal LU adı.

**public string LUName {get;}**

Giden LU 6.2 iletimleri için kullanılacak LU ' nun adı.

**public int MaximumActiveChannels {get;}**

Herhangi bir zamanda etkin olabilecek kanal sayısı üst sınırı.

**public int MaximumCurrentChannels {get;}**

Herhangi bir zamanda güncel olabilecek kanal sayısı üst sınırı (bağlı istemcilere sahip sunucu bağlantı kanalları da içinde olmak üzere).

**public int MaximumLU62Channels {get;}**

LU 6.2 iletim protokolünü kullanan, geçerli olabilecek kanal sayısı üst sınırı ya da bağlanabilen istemciler sayısı.

**public int MaximumMessageLength {get;}**

Kuyruk yöneticisi tarafından işlenebilecek bir iletinin (bayt cinsinden) uzunluk üst sınırını döndürür. İleti uzunluğu üst sınırı MaximumMessageLengthdeğerinden büyük olan bir kuyruk tanımlanmaz.

**public int MaximumPriority {get;}**

Kuyruk yöneticisi tarafından desteklenen ileti önceliği üst sınırını döndürür. Öncelikler, bu değere sıfırdan (en düşük) göre değer elde eder. Bu yöntemi, kuyruk yöneticisinden bağlantıyı kestikten sonra çağırırsanız, MQException ' ı atar.

**public int MaximumTCPChannels {get;}**

TCP/IP iletim protokolünü kullanan, geçerli olabilecek kanal sayısı üst sınırı ya da bağlanabilen istemci sayısı.

**public int MQIAccounting {get;}**

MQI verilerine ilişkin muhasebe bilgilerinin toplanmasını denetler.

**public int MQIStatistics {get;}**

Kuyruk yöneticisine ilişkin istatistik izleme bilgilerinin derlemesini denetler.

**public int OutboundPortMax {get;}**

Giden kanallar için bağ tanımlanırken kullanılacak kapı numaraları aralığındaki değer üst sınırı.

**public int OutboundPortMin {get;}**

Giden kanallar için bağ tanımlanırken kullanılacak kapı numaraları aralığındaki değer alt sınırı.

**public int QueueAccounting {get;}**

Sınıf 3 muhasebesi (iş parçacığı düzeyi ve kuyruk düzeyinde muhasebe) verilerinin tüm kuyruklar için kullanılıp kullanılmayacağını belirleyin.

**public int QueueMonitoring {get;}**

Kuyruklar için çevrimiçi izleme verilerinin toplanmasını denetler.

**public int QueueStatistics {get;}**

Kuyruklar için istatistik verilerinin toplanmasını denetler.

**public int ReceiveTimeout {get;}**

Bir TCP/IP kanalının etkin olmayan duruma dönmeden önce iş ortasından kalp atışları da içinde olmak üzere veri almak için bekleyeceği süre.

**public int ReceiveTimeoutMin {get;}**

Bir TCP/IP kanalının etkin olmayan bir duruma dönmeden önce iş ortasından kalp atışları da içinde olmak üzere veri almak için bekleyeceği en az süre.

**public int ReceiveTimeoutType {get;}**

ReceiveTimeout' ta değer için uygulanacak niteleyici.

**public int SharedQueueQueueManagerName {get;}**

İletilerin paylaşılan bir kuyruğa nasıl teslim gönderileceğini belirtir. Put, hedef kuyruk yöneticisiyle aynı kuyruk paylaşım grubundan farklı bir kuyruk yöneticisi belirtiyorsa, ileti iki şekilde teslim edilir:

**MQC.MQSQQM\_USE**

İletiler, paylaşılan kuyruğa konmadan önce nesne kuyruk yöneticisine teslim edilir.

**MQCMQSQQM\_IGNORE**

İletiler doğrudan paylaşılan kuyruğa konadır.

**public int SSLEvent {get;}**

TLS olaylarının oluşturulup oluşturulmayacağı.

**public int SSLFips {get;}**

Şifreleme donanımı yerine IBM MQ' de şifreleme gerçekleştiriliyorsa, yalnızca FIPS onaylı algoritmaların kullanılıp kullanılmayacağını.

**public int SSLKeyResetCount {get;}**

Gizli anahtar yeniden anlaşılmadan önce bir TLS iletişimde gönderilen ve alınan şifrelenmemiş baytların sayısını belirtir.

**public int ClusterSenderStatistics {get;}**

İstatistiklerin ardışık toplanmaları arasında dakika cinsinden aralığı belirtir.

**public int SyncpointAvailability {get;}**

Kuyruk yöneticisinin MQQueue . get ve MQQueue . put yöntemleriyle iş birimlerini ve eşitleme noktalarını destekleyip desteklemediğini belirtir.

**public string TCPName {get;}**

TCPStackType değerine bağlı olarak, kullanılacak TCP/IP sisteminin yalnızca ya da varsayılan adı.

**public int TCPStackType {get;}**

Kanal başlatıcısının yalnızca TCPName içinde belirlenen TCP/IP adres alanını kullanıp kullanmadığını belirler. Diğer bir seçenek olarak, kanal başlatıcısı herhangi bir TCP/IP adresine bağ tanımlayabilir.

**public int TraceRouteRecording {get;}**

Rota izleme bilgilerinin kaydedilmesini denetler.

## Yöntemler

**public MQProcess AccessProcess(string processName, int openOptions);**

**public MQProcess AccessProcess(string processName, int openOptions, string queueManagerName, string alternateUserId);**

MQException yayınlıyor.

Süreç özniteliklerine ilişkin bilgi edinmek için bu kuyruk yöneticisine ilişkin bir IBM MQ işlemine erişin.

### **processName**

Açılacak sürecin adı.

### **openOptions**

İşlemin açılmasını denetleyen seçenekler. Bit kullanan OR kullanılarak eklenebilecek ya da birleştirilebilen geçerli seçenekler şunlardır:

- MQC.MQOO\_FAIL\_IF QUIESCING
- MQC.MQOO\_INQUIRE
- MQC.MQOO\_SET
- MQC.MQOO\_ALTERNATE\_USER\_AUTHORITY

### **queueManagerAdı**

İşlemin tanımlandığı kuyruk yöneticisinin adı. Kuyruk yöneticisi, sürecin erişmekte olduğu kuyruk yöneticisi aynıysa, boş değerli ya da boş bir kuyruk yöneticisi adı bırakabilirsiniz.

### **alternateUserTanıtıcısı**

**openOptions** parametresinde MQC.MQOO\_ALTERNATE\_USER\_AUTHORITY belirtilirse, *alternateUserId*, işleme ilişkin yetkiyi denetlemek için kullanılan alternatif kullanıcı kimliğini belirtir. MQOO\_ALTERNATE\_USER\_AUTHORITY belirtilmemişse, *alternateUserId* boş ya da boş değerli olabilir.

Default user authority is used for connection to the queue manager if MQC.MQOO\_ALTERNATE\_USER\_AUTHORITY is not specified.

```
public MQQueue AccessQueue(string queueName, int openOptions);  
public MQQueue AccessQueue(string queueName, int openOptions, string  
queueManagerName, string dynamicQueueName, string alternateUserId);
```

MQException yayınlıyor.

Bu kuyruk yöneticilerindeki bir kuyruğa erişir.

İletiler alabilir ya da bunlara göz atabilir, iletileri yerleştirebilir, kuyruğun özniteliklerini sorgulayabilir ya da kuyruğun özniteliklerini ayarlayabilirsiniz. Kuyruk adı bir model kuyruğusa, dinamik bir yerel kuyruk yaratılır. Dinamik kuyruğun adını bulmak için, sonuçtaki MQQueue nesnesinin name öznitelikliğini sorgulayın.

### **queueName**

Açılacak kuyruğun adı.

### **openOptions**

Kuyruğun açıldığını denetleyen seçenekler.

#### **MQC.MQOO\_ALTERNATE\_USER\_AUTHORITY**

Belirtilen kullanıcı kimliğiyle geçerliliğini denetleyin.

#### **MQC.MQOO\_BIND\_AS\_QDEF**

Kuyruk için varsayılan bağ tanımını kullan.

#### **MQC.MQOO\_BIND\_NOT\_FIXED**

Belirli bir hedef için bağ tanımlamayın.

#### **MQC.MQOO\_BIND\_ON\_OPEN**

Kuyruk açıldığında bağ tanımlama tanıtıcısı hedef değerine bağlanır.

#### **MQC.MQOO\_BROWSE**

İletiyeye göz atmak için açın.

#### **MQC.MQOO\_FAIL\_IF QUIESCING**

Kuyruk yöneticisi susturulmuş durumda olduğunda hata oluştu.

#### **MQC.MQOO\_INPUT\_AS\_Q\_DEF**

Kuyruk tanımlı varsayılan değeri kullanarak ileti almak için açın.



**MQC.MQOO\_INPUT\_SHARED**

Paylaşılan erişimi olan iletileri almak için açın.

**MQC.MQOO\_INPUT\_EXCLUSIVE**

Özel erişimi olan iletileri almak için açın.

**MQC.MQOO\_INQUIRE**

Özellikleri sorgulamak istiyorsanız, sorgu için açma işlemi için açın.

**MQC.MQOO\_OUTPUT**

İleti koymak için açın.

**MQC.MQOO\_PASS\_ALL\_CONTEXT**

Tüm bağlamın iletilmesine izin verir.

**MQC.MQOO\_PASS\_IDENTITY\_CONTEXT**

Kimlik bağlamının iletilmesine izin verir.

**MQC.MQOO\_SAVE\_ALL\_CONTEXT**

İleti alındığında bağlam kaydedin.

**MQC.MQOO\_SET**

Özellikleri ayarlamak istiyorsanız, özellikleri ayarlamak için açın-özellikleri ayarlamak için.

**MQC.MQOO\_SET\_ALL\_CONTEXT**

Tüm bağlamın ayarlanmasını sağlar.

**MQC.MQOO\_SET\_IDENTITY\_CONTEXT**

Kimlik bağlamının ayarlanmasını sağlar.

**queueManagerAdı**

Kuyruğun tanımlı olduğu kuyruk yöneticisinin adı. Tamamen boş ya da boş değer olan bir ad, MQQueueManager nesnesinin bağlı olduğu kuyruk yöneticisini belirtir.

**dynamicQueueAdı**

queueName bir model kuyruğunun adını belirtmedikçe, *dynamicQueueName* yoksayılr. Ters durumda, *dynamicQueueName*, yaratılacak dinamik kuyruğun adını belirtir. queueName, bir model kuyruğunun adını belirtiyorsa, boş değerli ya da boş değerli ad kullanılamaz. Addaki son boş olmayan karakter bir yıldız işareti (\*) ise, kuyruk yöneticisi yıldız işaretini bir karakter dizgisiyle değiştirir. Kuyruk için oluşturulan adın, bu kuyruk yöneticisiyle benzersiz olduğunu garanti eder.

**alternateUserTanıtıcısı**

openOptions değiştirgesinde MQC.MQOO\_ALTERNATE\_USER\_AUTHORITY belirtilirse, *alternateUserId*, açık ara yetkilendirmeyi denetlemek için kullanılan diğer kullanıcı kimliğini belirtir. MQC.MQOO\_ALTERNATE\_USER\_AUTHORITY belirtilmemişse, *alternateUserId* boş bırakılabilir ya da boş bırakılabilir.

```
public MQTopic AccessTopic( MQDestination destination, string topicName, string
topicObject, int options);
public MQTopic AccessTopic( MQDestination destination, string topicName, string
topicObject, int options, string alternateUserId);
public MQTopic AccessTopic( MQDestination destination, string topicName, string
topicObject, int options, string alternateUserId, string subscriptionName);
public MQTopic AccessTopic( MQDestination destination, string topicName, string
topicObject, int options, string alternateUserId, string subscriptionName,
System.Collections.Hashtable properties);
public MQTopic AccessTopic(string topicName, string topicObject, int openAs,
int options);
public MQTopic AccessTopic(string topicName, string topicObject, int openAs,
int options, string alternateUserId);
public MQTopic AccessTopic(string topicName, string topicObject, int options,
string alternateUserId, string subscriptionName);
public MQTopic AccessTopic(string topicName, string topicObject, int options,
string alternateUserId, string subscriptionName, System.Collections.Hashtable
properties);
```

Bu kuyruk yöneticisinde bir konuya erişin.

MQTopic nesnelere, bazen konu nesnelere olarak adlandırılan denetim konusu nesnelere yakından ilişkilidir. Girişte, *topicObject* bir yönetici konu nesnesini işaret eder. MQTopic oluşturucusu, konu nesnesinden bir konu dizgisini alır ve bir konu adı yaratmak için bunu *topicName* ile birleştirir. *topicObject* ya da *topicName* boş değerli olabilir. Konu adı, konu ağacına eşleştirilir ve *topicObject* içinde en yakın eşleşen denetim konusu nesnesinin adı döndürülür.

MQTopic nesnesiyle ilişkili konular iki konu dizisini birleştirmenin sonudur. Birinci konu dizisi, *topicObject* ile tanımlanan yönetimle ilgili konu nesnesi tarafından tanımlanır. İkinci konu dizisi: *topicString*. The resulting topic string associated with the MQTopic object can identify multiple topics by including wildcards.

Konunun yayınlama ya da abone olma için açılıp açılmadığına bağlı olarak, konu başlıklarını yayınlamak için MQTopic .Put yöntemlerini ya da konulara ilişkin yayınları almak için MQTopic .Get yöntemlerini kullanabilirsiniz. Aynı konuyu yayınlamak ve bu konuya abone olmak istiyorsanız, yayınlamak için bir kez ve bir kez abone olmak üzere iki kez konuya erişmeniz gerekir.

Abonelik için bir MQTopic nesnesi oluşturursanız, bir MQDestination nesnesi sağlamadan yönetilen bir abonelik varsayılan olarak kabul edilir. Bir kuyruğu MQDestination nesnesi olarak geçerseniz, yönetilmeyen bir abonelik varsayılır. Ayarladığınız abonelik seçeneklerinin yönetilen ya da yönetilmeyen abonelik tutarlı olduğundan emin olmanız gerekir.

### Hedef

*destination* , bir MQQueue yönetim ortağıdır. *destinations* sağlanarak, MQTopic yönetilmeyen bir abonelik olarak açılır. Konuyla ilgili yayınlar, *destination* . olarak erişilen kuyruğa teslim edilir.

### topicName

Konu adının ikinci bölümü olan bir konu dizisi. *topicName* , *topicObject* yönetim konusu nesnesinde tanımlanan konu dizisiyle bitleştirilir. *topicName* ' ı boş değer olarak ayarlayabilirsiniz; bu durumda konu adı *topicObject* içindeki konu dizisiyle tanımlanır.

### topicObject

Giriş sırasında, *topicObject* , konu adının ilk bölümünü oluşturan konu dizisini içeren konu nesnesinin adıdır. The topic string in *topicObject* is concatenated with *topicName*. Konu dizileri oluşturulmasına ilişkin kurallar, [Konu dizilerinin birleştirilmesi](#) altında tanımlanır.

On output, *topicObject* contains the name of the administrative topic object that is the closest match in the topic tree to the topic identified by the topic string.

### openAs

Yayınlamak ya da abone olmak için konuya erişin. Parametre, şu seçeneklerden yalnızca birini içerebilir:

- MQC.MQTOPIC\_OPEN\_AS\_SUBSCRIPTION
- MQC.MQTOPIC\_OPEN\_AS\_PUBLICATION

### seçenekler

Yayın ya da abonelik için konunun açıklarını denetleyen seçenekleri birleştirin. Abonelik ve MQC.MQOO\_\* değişmezleri için bir konuya erişmek üzere bir konuya erişmek üzere MQC.MQSO\_\* sabitlerini kullanın.

If more than one option is required, add the values together, or combine the option values using the bitwise OR operator.

### alternateUserTanıtıcısı

İşlemi bitirmek için gereken yetkiyi denetlemek için kullanılan diğer kullanıcı kimliğini belirtin. You must specify *alternateUserId*, if either MQC.MQOO\_ALTERNATE\_USER\_AUTHORITY or MQC.MQSO\_ALTERNATE\_USER\_AUTHORITY is set in the options parameter.

### subscriptionName

MQC.MQSO\_DURABLE ya da MQC.MQSO\_ALTER seçenekleri sağlandıysa, *subscriptionName* gereklidir. Her iki durumda da MQTopic , abonelik için örtük olarak açılmıştır. MQC.MQSO\_DURABLE ayarlıysa ve abonelik varsa ya da MQC.MQSO\_ALTER ayarlandıysa ve abonelik yoksa, kural dışı durum oluşur.

## özellikler

HASH çizelgesi kullanılarak listelenen özel abonelik özelliklerinden herhangi birini ayarlayın. HASH çizelgesindeki belirlenen girişler çıkış değerleriyle güncellenir. Çıkış değerlerini bildirmek için, HASH çizelgesine girişler eklenmez.

- MQC.MQSUB\_PROP\_ALTERNATE\_SECURITY\_ID
- MQC.MQSUB\_PROP\_SUBSCRIPTION\_EXPIRY
- MQC.MQSUB\_PROP\_SUBSCRIPTION\_USER\_DATA
- MQC.MQSUB\_PROP\_SUBSCRIPTION\_CORRELATION\_ID
- MQC.MQSUB\_PROP\_PUBLICATION\_PRIORITY
- MQC.MQSUB\_PROP\_PUBLICATION\_ACCOUNTING\_TOKEN
- MQC.MQSUB\_PROP\_PUBLICATION\_APPLICATIONID\_DATA

## **public MQAsyncStatus GetAsyncStatus();**

Throws MQException

Kuyruk yöneticisi bağlantısına ilişkin zamanuyumsuz etkinliği temsil eden bir MQAsyncStatus nesnesi döndürür.

## **public void Backout();**

MQException yayınlıyor.

Son eşitleme noktasından bu yana, eşitleme noktası içinde okunan ya da yazılan tüm iletileri geri al.

MQC.MQPMO\_SYNCPOINT işaret kümesiyle yazılan iletiler kuyruklardan kaldırılır.

MQC.MQGMO\_SYNCPOINT işaretiyle okunan iletiler, geldikleri kuyruklara geri dönerler. İletiler kalıcıysa, değişiklikler günlüğe kaydedilir.

Yeniden bağlantılanabilir istemciler için, yeniden bağlantı başarılı olduktan sonra MQRC\_NONE neden kodu istemciye döndürülür.

## **public void Begin();**

MQException yayınlıyor.

Begin yalnızca sunucu bağ tanımları kipinde desteklenir. Küresel bir çalışma birimi başlatır.

## **public void Commit();**

MQException yayınlıyor.

Son eşitleme noktasından bu yana, eşitleme noktası içinde okunan ya da yazılan tüm iletileri kesinleştirin.

MQC.MQPMO\_SYNCPOINT işaret kümesiyle yazılan iletiler, diğer uygulamalar tarafından kullanılabilir kılınmaya başlanır. MQC.MQGMO\_SYNCPOINT işaret kümesiyle alınan iletiler silinir. İletiler kalıcıysa, değişiklikler günlüğe kaydedilir.

Yeniden bağlanabilir bir istemciye döndürülen neden kodları şunlardır:

- MQRC\_CALL\_INTERRUPTED if connection is lost while carrying out the commit call.
- Yeniden bağlantıdan sonra kesinleştirme çağrısı yayımlandıysa, MQRC\_BACKED\_OUT .

## **Disconnect();**

MQException yayınlıyor.

Kuyruk yöneticisiyle bağlantıyı kapatın. Bu kuyruk yöneticisinde erişilen tüm nesnelere bu uygulama tarafından erişilemez. Nesnelere yeniden erişmek için bir MQQueueManager nesnesi oluşturun.

Genel olarak, bir iş biriminin bir parçası olarak gerçekleştirilen herhangi bir çalışma kesinleştirilmektedir. Ancak, iş birimi .NET tarafından yönetiliyorsa, iş birimi geri döndürülebilirdi.

```

public void Put(int type, string destinationName, MQMessage message);
public void Put(int type, string destinationName, MQMessage message
MQPutMessageOptions putMessageOptions);
public void Put(int type, string destinationName, string queueManagerName,
string topicString, MQMessage message);
public void Put(string queueName, MQMessage message);
public void Put(string queueName, MQMessage message, MQPutMessageOptions
putMessageOptions);
public void Put(string queueName, string queueManagerName, MQMessage message);
public void Put(string queueName, string queueManagerName, MQMessage message,
MQPutMessageOptions putMessageOptions);
public void Put(string queueName, string queueManagerName, MQMessage message,
MQPutMessageOptions putMessageOptions, string alternateUserId);

```

MQException yayınlıyor.

Places a single message onto a queue or topic without creating an MQQueue or MQTopic object first.

#### **queueName**

İletinin yerleştirilecek kuyruğun adı.

#### **destinationName**

Hedef nesnenin adı. Bu, *type* değerine bağlı olarak bir kuyruk ya da bir konudur.

#### **tip**

Hedef nesnenin tipi. Seçenekleri birleştirmemelisiniz.

#### **MQC.MQOT\_Q**

Kuyruk

#### **MQC.MQOT\_TOPIC**

Konu

#### **queueManagerAdı**

Kuyruğun tanımlandığı kuyruk yöneticisi ya da kuyruk yöneticisi diğer adının adı.

MQC.MQOT\_TOPIC tipi belirtilirse, bu parametre yoksayılr.

Kuyruk bir model kuyruksa ve çözülen kuyruk yöneticisi adı bu kuyruk yöneticisi değilse, MQException atılır.

#### **topicString**

*topicString*, *destinationName* konu nesnesindeki konu adı ile birleştirilir.

*destinationName* bir kuyruksa, *topicString* yoksayılr.

#### **ileti**

Gönderilecek ileti. İleti bir giriş/çıkış nesnesidir.

Yeniden bağlanabilir bir istemciye döndürülen neden kodları şunlardır:

- Kalıcı bir ileti üzerinde bir put çağrısı gerçekleştirilirken bağlantı kesilirse MQRC\_CALL\_INTERRUPTED .
- MQRC\_NONE if the connection is successful while performing a Put call on a non-persistent message (see [Uygulama Kurtarma](#) ).

#### **putMessageSeçenekleri**

Put işlemlerinin hareketlerini denetleyen seçenekler.

*putMessageOptions* ögesini çıkarırsanız, varsayılan bir *putMessageOptions* yönetim ortamı yaratılır. *putMessageOptions* bir giriş/çıkış nesnesidir.

Yeniden bağlanabilir bir istemcide MQPMO\_LOGICAL\_ORDER seçeneğini kullanırsanız, MQRC\_RECONNECT\_INCOMPATIBLE neden kodu döndürülür.

## alternateUserTanıtıcısı

İletiyi bir kuyruğa yerleştirirken yetkilendirmeyi denetlemek için kullanılan diğer bir kullanıcı kimliğini belirtir.

*putMessageOptions* içinde MQC.MQOO\_ALTERNATE\_USER\_AUTHORITY ayarlamadıysanız, *alternateUserId* ögesini çıkarabilirsiniz. MQC.MQOO\_ALTERNATE\_USER\_AUTHORITY'ı ayarladıysanız, *alternateUserId*'ı da ayarlamanız gerekir. *alternateUserId* has not effect unless you also set MQC.MQOO\_ALTERNATE\_USER\_AUTHORITY.

## Oluşturucular

```
public MQQueueManager();
public MQQueueManager(string queueManagerName);
public MQQueueManager(string queueManagerName, Int options);
public MQQueueManager(string queueManagerName, Int options, string channel,
string connName);
public MQQueueManager(string queueManagerName, string channel, string
connName);
public MQQueueManager(string queueManagerName, System.Collections.Hashtable
properties);
```

MQException yayınlıyor.

Kuyruk yöneticisiyle bağlantı yaratır. İstemci bağlantısı ya da sunucu bağlantısı oluşturmak arasında seçim yapın.

Kuyruk yöneticisine bağlanma girişimi sırasında kuyruk yöneticisine ilişkin sorgulama (inq) yetkinizin olması gerekir. Sorgu yetkisi olmadan, bağlantı kurma girişimi başarısız olur.

Aşağıdaki koşullardan biri geçerliyse, istemci bağlantısı oluşturulur:

1. *channel* or *connName* are specified in the constructor.
2. *HostName*, *Portya* da *Channel*, *properties* içinde belirtilir.
3. *MQEnvironment.HostName*, *MQEnvironment.Portya* da *MQEnvironment.Channel* belirtilmektedir.

Bağlantı özelliklerinin değerleri, gösterilen sırada varsayılan olarak belirlenir. Oluşturucudaki *channel* ve *connName*, oluşturucudaki özellik değerlerinden öncelikli olarak uygulanır. Oluşturucu özellik değerleri, *MQEnvironment* özelliklerinden önceliklidir.

Anasistem adı, kanal adı ve kapı, *MQEnvironment* sınıfında tanımlanır.

### queueManagerAdı

Bağlanılacak kuyruk yöneticisinin ya da kuyruk yöneticisi grubunun adı.

Varsayılan kuyruk yöneticisi seçimi yapmak için, parametreyi kaldırın ya da boş değerli ya da boş bırakın. Bir sunucu üzerindeki varsayılan kuyruk yöneticisi bağlantısı, sunucudaki varsayılan kuyruk yöneticisidir. İstemci bağlantısıyla ilgili varsayılan kuyruk yöneticisi bağlantısı, dinleyiciye bağlı olan kuyruk yöneticisinedir.

### seçenekler

MQCNO bağlantı seçeneklerini belirtin. Değerlerin, yapılmakta olan bağlantı tipine uygun olması gerekir. Örneğin, bir clientistemci bağlantısı için aşağıdaki sunucu bağlantısı özelliklerini belirtirseniz, bir *MQException* dosyası atılır.

- MQC.MQCNO\_FASTPATH\_BINDING
- MQC.MQCNO\_STANDARD\_BINDING

### özellikler

Özellikler parametresi, *MQEnvironment* tarafından ayarlanan özellikleri geçersiz kılan bir dizi anahtar/değer çiftini alır; Örneğin, “*MQEnvironment* özelliklerini geçersiz kıl” sayfa 1748 örneğine bakın. Aşağıdaki özellikler geçersiz kılınabilir:

- MQC.CONNECT\_OPTIONS\_PROPERTY
- MQC.CONNECTION\_NAME\_PROPERTY
- MQC.ENCRYPTION\_POLICY\_SUITE\_B
- MQC.HOST\_NAME\_PROPERTY
- MQC.PORT\_PROPERTY
- MQC.CHANNEL\_PROPERTY
- MQC.SSL\_CIPHER\_SPEC\_PROPERTY
- MQC.SSL\_PEER\_NAME\_PROPERTY
- MQC.SSL\_CERT\_STORE\_PROPERTY
- MQC.SSL\_CRYPTOHARDWARE\_PROPERTY
- MQC.SECURITY\_EXIT\_PROPERTY
- MQC.SECURITY\_USERDATA\_PROPERTY
- MQC.SEND\_EXIT\_PROPERTY
- MQC.SEND\_USERDATA\_PROPERTY
- MQC.RECEIVE\_EXIT\_PROPERTY
- MQC.RECEIVE\_USERDATA\_PROPERTY
- MQC.USER\_ID\_PROPERTY
- MQC.PASSWORD\_PROPERTY
- MQC.MQAIR\_ARRAY
- MQC.KEY\_RESET\_COUNT
- MQC.FIPS\_REQUIRED
- MQC.HDR\_CMP\_LIST
- MQC.MSG\_CMP\_LIST
- MQC.TRANSPORT\_PROPERTY

### **channel**

Sunucu bağlantı kanalının adı

### **connName**

Connection name in the format *HostName (Kapı)*.

You can supply a list of *anasistem adları* and *kapılar* as an argument to the constructor `MQQueueManager (String queueManagerName, Hashtable properties)` using `CONNECTION_NAME_PROPERTY`.

Örneğin:

```
ConnectionName = "fred.mq.com(2344),nick.mq.com(3746),tom.mq.com(4288)";
Hashtable Properties=new Hashtable();
properties.Add(MQC.CONNECTION_NAME_PROPERTY,ConnectionName);
MQQueueManager qmgr=new MQQueue Manager("qmgrname",properties);
```

Bir bağlantı girişiminde bulunulduğunda, bağlantı adı listesi sırayla işlenir. İlk anasistem adı ve bağlantı noktasına bağlanma girişimi başarısız olursa, ikinci öznitelik çiftiyle bağlantı kurma girişiminde bulunmaya çalışılır. İstemci bu işlemi, başarılı bir bağlantı yapılıncaya ya da liste tükeninceye kadar yineler. Liste tüketildiyse, istemci uygulamasına uygun bir neden kodu ve tamamlanma kodu döndürülür.

Bağlantı adı için bir kapı numarası sağlanmadığında, varsayılan kapı (`mqclient.ini` içinde yapılandırılır) kullanılır.

## Bağlantı Listesini Ayarla

Otomatik istemci yeniden bağlanma seçenekleri ayarlandığında, aşağıdaki yöntemleri kullanarak bağlantı listesini ayarlayabilirsiniz:

### Bağlantı listesini MQSERVER ile ayarla

Komut istemiyle bağlantı listesini ayarlayabilirsiniz.

Komut isteminde, aşağıdaki komutu ayarlayın:

```
MQSERVER=SYSTEM.DEF.SVRCONN/TCP/Hostname1(Port1),Hostname2(Por2),Hostname3(Port3)
```

Örneğin:

```
MQSERVER=SYSTEM.DEF.SVRCONN/TCP/fred.mq.com(5266),nick.mq.com(6566),jack.mq.com(8413)
```

Bağlantıyı MQSERVER ' da ayarlarsanız, uygulamaya ayarlamayın.

Uygulama içindeki bağlantı listesini ayarladıysanız, uygulama MQSERVER ortam değişkeninde tanımlı olan her şeyi geçersiz kılar.

### Uygulama aracılığıyla bağlantı listesini ayarla

Anasistem adını ve kapı özelliklerini belirterek, uygulama içindeki bağlantı listesini ayarlayabilirsiniz.

```
String connName = "fred.mq.com(2344), nick.mq.com(3746), chris.mq.com(4288)";  
MQQueueManager qm = new MQQueueManager("QM1", "TestChannel", connName);
```

### Bağlantı listesini app.config ile ayarlayın

App.config, anahtar-değer çiftlerini belirttiğiniz bir XML dosyasıdır.

Bağlantı listesinde şu değeri belirtin:

```
<app.Settings>  
<add key="Connection1" value="Hostname1(Port1)"/>  
<add key="Connection2" value="Hostname2(Port2)"/>  
</app.Settings>
```

Örneğin:

```
<app.Settings>  
<add key="Connection1" value="fred.mq.com(2966)"/>  
<add key="Connection2" value="alex.mq.com(6533)"/>  
</app.Settings>
```

You can directly change the connection list in the app.config file.

### Bağlantı listesini MQEnvironment ile ayarlayın

Bağlantı listesini MQEnvironment aracılığıyla ayarlamak için *ConnectionName* özelliğini kullanın.

```
MQEnvironment.ConnectionName = "fred.mq.com(4288),alex.mq.com(5211);
```

*ConnectionName* özelliği, MQEnvironment' ta ayarlanan anasistem adını ve kapı özelliklerini geçersiz kılar.

### İstemci bağlantısı yaratılması

Aşağıdaki örnek, bir kuyruk yöneticisiyle istemci bağlantısının nasıl yaratılacağını göstermektedir. Yeni bir MQQueueManager Nesnesi yaratmadan önce, MQEnvironment değişkenlerini ayarlayarak bir istemci bağlantısı oluşturabilirsiniz.

```

MQEnvironment.Hostname = "fred.mq.com"; // host to connect to
MQEnvironment.Port     = 1414;         // port to connect to
                                     // If not explicitly set,
                                     // defaults to 1414
                                     // (the default IBM MQ port)
MQEnvironment.Channel  = "channel.name"; // the case sensitive
                                     // name of the
                                     // SVR CONN channel on
                                     // the queue manager
MQQueueManager qMgr    = new MQQueueManager("MYQM");

```

Şekil 11. İstemci bağlantısı

### MQEnvironment özelliklerini geçersiz kıl

Aşağıdaki örnek, bir HASH çizelgesinde tanımlı kullanıcı kimliği ve parolasıyla bir kuyruk yöneticisinin nasıl yaratılacağı gösterilmiştir.

```

Hashtable properties = new Hashtable();

properties.Add( MQC.USER_ID_PROPERTY, "ExampleUserId" );
properties.Add( MQC.PASSWORD_PROPERTY, "ExamplePassword" );

try
{
    MQQueueManager qMgr = new MQQueueManager("qmgrname", properties);
}
catch (MQException mqe)
{
    System.Console.WriteLine("Connect failed with " + mqe.Message);
    return((int)mqe.Reason);
}

```

Şekil 12. Overriding MQEnvironment properties

### Yeniden bağlanabilir bağlantı yarat

Aşağıdaki örnek, bir istemciyi bir Kuyruk Yöneticisine otomatik olarak nasıl yeniden bağladığınızı gösterir.

```

Hashtable properties = new Hashtable(); // The queue manager name and the
                                     // properties how it has to be connected

properties.Add(MQC.CONNECT_OPTIONS_PROPERTY, MQC.MQCNO_RECONNECT); // Options
                                     // through which reconnection happens

properties.Add(MQC.CONNECTION_NAME_PROPERTY, "fred.mq.com(4789),nick.mq.com(4790)"); // The list
                                     // of queue managers through which reconnection happens

MQ QueueManager qmgr = new MQQueueManager("qmgrname", properties);

```

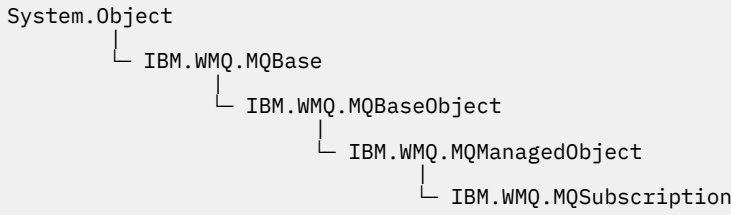
Şekil 13. İstemcinin kuyruk yöneticisine otomatik olarak yeniden bağlanması

## MQSubscription.NET sınıfı

Alıkonaan yayınların aboneye gönderildiğini istemek için MQSubscription kullanın. MQSubscription, abonelik için açılan bir MQTopic nesnesinin bir özeliğidir.

### Sınıf





```
public class IBM.WMQ.MQSubscription extends IBM.WMQ.MQManagedObject;
```

- “Özellikler” sayfa 1749
- “Yöntemler” sayfa 1749
- “Oluşturucular” sayfa 1749

## Özellikler

MQManagedObject sınıfını kullanarak abonelik özelliklerine erişin; bkz. [“Özellikler” sayfa 1708](#).

## Yöntemler

Access subscription Inquire, Set and Get methods using the MQManagedObject class; see [“Yöntemler” sayfa 1709](#).

```
public int RequestPublicationUpdate(int options);
```

MQException yayınlıyor.

Geçerli konu için güncelleştirilmiş bir yayın isteyin. Kuyruk yöneticisinin konu için alıkonan yayınları varsa, bu yayınlar aboneye gönderilir.

RequestPublicationUpdate' i aramadan önce, MQSubscription nesnesini almak üzere abonelik için bir konu açın.

Genellikle, aboneliği MQC.MQSO\_PUBLICATIONS\_ON\_REQUEST seçeneği ile açın. Konu dizgisinde genel arama karakteri yoksa, bu çağrıya ilişkin olarak yalnızca bir yayın gönderilir. Konu dizgisinde joker karakterler varsa, birçok yayın gönderilebilir. Yöntem, abonelik kuyruğuna gönderilen alıkonan yayınların sayısını döndürür. özellikle bu kadar çok yayınının alındığına dair bir garanti yoktur, özellikle de kalıcı olmayan mesajlar.

### seçenekler

#### **MQC.MQSRO\_FAIL\_IF QUIESCING**

Kuyruk yöneticisi susturma durumundaysa yöntem başarısız olur. z/OS üzerinde, CICS ya da IMS uygulaması için MQC.MQSRO\_FAIL\_IF QUIESCING , bağlantı bir susturma durumundaysa, bu yöntemi başarısız olarak zorlar.

#### **MQC.MQSRO\_NONE**

Seçenek belirlenmez.

## Oluşturucular

Genel oluşturucusu yok.

Abonelik için açılan bir MQTopic nesnesinin SubscriptionReference özelliğinde bir MQSubscription nesnesi döndürülür.

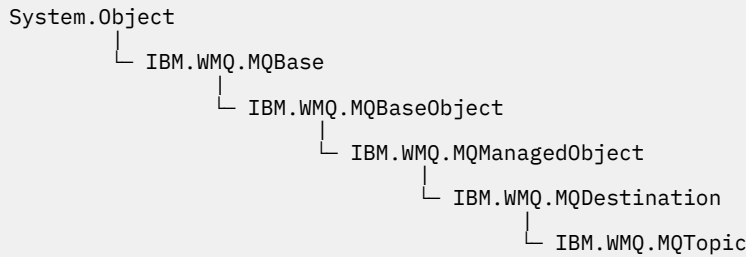
RequestPublicationUpdate yöntemini çağırın. MQSubscription , MQManagedObject sınıfının alt sınıfıdır. MQManagedObject' un özelliklerine ve yöntemlerine erişmek için başvuruyu kullanın.

## MQTopic.NET sınıfı

Bir konuyla ilgili iletileri yayınlamak ya da bir konuya abone olmak ya da bir konunun özniteliklerini sorgulamak ya da ayarlamak için MQTopic ögesini kullanın. Bir oluşturucu ya da

MQQueueManager.AccessTopic yöntemi kullanarak yayınlama ya da abone olma için bir MQTopic nesnesi yaratın.

## Sınıf



```
public class IBM.WMQ.MQTopic extends IBM.WMQ.MQDestination;
```

- [“Özellikler” sayfa 1750](#)
- [“Yöntemler” sayfa 1750](#)
- [“Oluşturucular” sayfa 1752](#)

## Özellikler

Özellikler alınırken MQException için test ediliyor.

### **public Boolean IsDurable {get;}**

Yalnızca abonelik dayanıklı ise True , aksi takdirde False değerini döndüren özelliği okuyun. Konu yayınlanmak üzere açıldıysa, özellik yoksayılr ve her zaman Falsedeğerini döndürür.

### **public Boolean IsManaged {get;};**

Abonelik kuyruk yöneticisi tarafından yönetiliyorsa True ya da tersi durumda False değerini döndüren özelliği okuyun. Konu yayınlanmak üzere açıldıysa, özellik yoksayılr ve her zaman False (Yanlış) değerini döndürür.

### **public Boolean IsSubscribed {get;};**

Konu, abonelik için açıldıysa True değerini döndüren ve konu yayınlanmak üzere açıldıysa False değerini döndürür.

### **public MQSubscription SubscriptionReference {get;};**

Abonelik için açılan bir konu nesnesiyle ilişkili MQSubscription nesnesini döndüren yalnızca okuma özelliği. Bu başvuru, kapatma seçeneklerini değiştirmek ya da nesne yöntemlerinden herhangi birini başlatmak için kullanılabilir.

### **public MQDestination UnmanagedDestinationReference {get;};**

Yönetilmeyen bir abonelikle ilişkili MQQueue değerini döndüren salt okunur özellik. Bu hedef, konu nesnesi yaratıldığında belirlenen hedeftir. Özellik, yayın için açılan herhangi bir konu nesnesi için ya da yönetilen abonelikle boş değer döndürür.

## Yöntemler

### **public void Put(MQMessage message);**

### **public void Put(MQMessage message, MQPutMessageOptions putMessageOptions);**

MQException yayınlıyor.

Konuya bir ileti yayınlar.

Put çağrısından sonra MQMessage nesnesinde yapılan değişiklikler, IBM MQ kuyruğunda ya da yayınındaki gerçek iletiyi etkilemez.

Put , MQMessage nesnesinin MessageId ve CorrelationId özelliklerini güncelleştirir ve ileti verilerini temizlememektedir. Daha fazla Put ya da Get çağrısı, MQMessage nesnesindeki

güncellenmiş bilgilere gönderme yapıyor. Örneğin, aşağıdaki kod parçasısında ilk ileti a ve ikinci abiletisini içerir.

```
msg.WriteString("a");
q.Put(msg, pmo);
msg.WriteString("b");
q.Put(msg, pmo);
```

### ileti

İleti tanımlayıcı verilerini içeren bir MQMessage nesnesi ve gönderilecek ileti. İleti tanımlayıcısı bu yöntemin bir sonucu olarak değiştirilebilir. Bu yöntemin tamamlanmasından hemen sonra ileti tanımlayıcısındaki değerler, kuyruğa konulan ya da konu üzerinde yayınlanan değerlerdir.

Yeniden bağlanabilir bir istemciye döndürülen neden kodları şunlardır:

- MQRC\_CALL\_INTERRUPTED if the connection is broken while running a Put call on a persistent message and the reconnection is successful.
- Bağlantı, kalıcı olmayan bir iletiyle (bkz. [Uygulama Kurtarma](#)) bir put çağrısı çalıştırırken bağlantı başarılı olursa MQRC\_NONE .

### putMessageSeçenekleri

Put işleminin denetlenmesini denetleyen seçenekler.

*putMessageOptions* belirtilmemişse, varsayılan MQPutMessageOptions eşgörünümü kullanılır.

Yeniden bağlanabilir bir istemcide MQPMO\_LOGICAL\_ORDER seçeneğini kullanırsanız, MQRC\_RECONNECT\_INCOMPATIBLE neden kodu döndürülür.

**Not:** Basitlik ve performans için, kuyruğa tek bir ileti koymak isterseniz, MQQueueManager.Put nesnesini kullanın. Bunun için bir MQQueue nesnesine sahip olmanız.

```
public void Get(MQMessage message);  
public void Get(MQMessage message, MQGetMessageOptions getMessageOptions);  
public void Get(MQMessage message, MQGetMessageOptions getMessageOptions, int  
MaxMsgSize);
```

MQException yayınlıyor.

Konudan bir ileti alır.

Bu yöntem, alma işlemi için varsayılan bir MQGetMessageOptions yönetim ortamı kullanır. Kullanılan ileti seçeneği MQGMO\_NOWAIT.

Alma işlemi başarısız olursa, MQMessage nesnesi değişmeden kalır. Başarılı olursa, MQMessage ' un ileti tanımlayıcısı ve ileti verileri kısımları, gelen iletiden ileti tanımlayıcısı ve ileti verileriyle değiştirilir.

Belirli bir MQQueueManager 'den IBM MQ ' e yapılan tüm çağrılar zamanuyumlu olur. Bu nedenle, bir bekleme işlemi gerçekleştirdiğinizde, aynı MQQueueManager ' u kullanan diğer tüm iş parçacıklarının, Alma işlemi gerçekleştirilinceye kadar IBM MQ çağrılarının yapılması engellenir. IBM MQ ' a aynı anda erişmek için birden çok iş parçacığının olması gerekiyorsa, her bir iş parçacığının kendi MQQueueManager nesnesi yaratması gerekir.

### ileti

İleti tanımlayıcısını ve döndürülen ileti verilerini içerir. İleti tanımlayıcısındaki bazı alanlar giriş değiştirgeleri olur. MessageId ve CorrelationId giriş parametrelerinin gerektiği şekilde ayarlandığından emin olmak önemlidir.

Yeniden bağlantılanabilir istemci, MQGM\_SYNCPOINT altında alınan iletiler için, başarılı yeniden bağlantı sonrasında MQRC\_BACKED\_OUT neden kodunu döndürür.

### getMessageSeçenekleri

Alma işleminin denetlenmesini denetleyen seçenekler.

MQC.MQGMO\_CONVERT seçeneğinin kullanılması, tek byte 'lık karakter kodlarından çift byte kodlarına dönüştürülürken MQC.MQRC\_CONVERTED\_STRING\_TOO\_BIG neden koduyla bir kural dışı duruma neden olabilir. Bu durumda, ileti dönüştürmeden arabelleğe kopyalanır.

*getMessageOptions* belirtilmemişse, kullanılan ileti seçeneği MQGMO\_NOWAIT olur.

Yeniden bağlanabilir bir istemcide MQGMO\_LOGICAL\_ORDER seçeneğini kullanırsanız, MQRC\_RECONNECT\_INCOMPATIBLE neden kodu döndürülür.

### **MaxMsgBoyutu**

Bu ileti nesnesinin alınabileceği en büyük ileti. Kuyruktaki ileti bu boyuttan büyükse, iki şeyden biri oluşur:

- If the MQGMO\_ACCEPT\_TRUNCATED\_MSG flag is set in the MQGetMessageOptions object, the message is filled with as much of the message data as possible. MQCC\_WARNING tamamlanma kodu ve MQRC\_TRUNCATED\_MSG\_ACCEPTED neden koduyla bir kural dışı durum yayınlandı.
- MQGMO\_ACCEPT\_TRUNCATED\_MSG işareti ayarlanmamış ise, ileti kuyrukta kalır. MQCC\_WARNING tamamlanma kodu ve MQRC\_TRUNCATED\_MSG\_FAILED neden koduyla bir kural dışı durum yayınlandı.

*MaxMsgSize* belirtilmemişse, tüm ileti alınır.

## **Oluşturucular**

```
public MQTopic(MQQueueManager queueManager, MQDestination destination, string
topicName, string topicObject, int options);
public MQTopic(MQQueueManager queueManager, MQDestination destination, string
topicName, string topicObject, int options, string alternateUserId);
public MQTopic(MQQueueManager queueManager, MQDestination destination, string
topicName, string topicObject, int options, string alternateUserId, string
subscriptionName);
public MQTopic(MQQueueManager queueManager, MQDestination destination, string
topicName, string topicObject, int options, string alternateUserId, string
subscriptionName, System.Collections.Hashtable properties);
public MQTopic(MQQueueManager queueManager, string topicName, string
topicObject, int openAs, int options);
public MQTopic(MQQueueManager queueManager, string topicName, string
topicObject, int openAs, int options, string alternateUserId);
public MQTopic(MQQueueManager queueManager, string topicName, string
topicObject, int options, string alternateUserId, string subscriptionName);
public MQTopic(MQQueueManager queueManager, string topicName, string
topicObject, int options, string alternateUserId, string subscriptionName,
System.Collections.Hashtable properties);
```

*queueManager* üzerindeki bir konuya erişin.

MQTopic nesnelere, bazen konu nesnelere olarak adlandırılan denetim konusu nesnelere yakından ilişkilidir. Girişte, *topicObject* bir yönetici konu nesnesini işaret eder. MQTopic oluşturucusu, konu nesnesinden bir konu dizgisini alır ve bir konu adı yaratmak için bunu *topicName* ile birleştirir. *topicObject* ya da *topicName* boş değerli olabilir. Konu adı, konu ağacına eşleştirilir ve *topicObject* içinde en yakın eşleşen denetim konusu nesnesinin adı döndürülür.

MQTopic nesnesiyle ilişkili konular iki konu dizisini birleştirmenin sonudur. Birinci konu dizisi, *topicObject* ile tanımlanan yönetimle ilgili konu nesnesi tarafından tanımlanır. İkinci konu dizisi: *topicString*. The resulting topic string associated with the MQTopic object can identify multiple topics by including wildcards.

Konunun yayınlama ya da abone olma için açılıp açılmadığına bağlı olarak, konu başlıklarını yayınlamak için MQTopic .Put yöntemlerini ya da konulara ilişkin yayınları almak için MQTopic .Get yöntemlerini kullanabilirsiniz. Aynı konuyu yayınlamak ve bu konuya abone olmak istiyorsanız, yayınlamak için bir kez ve bir kez abone olmak üzere iki kez konuya erişmeniz gerekir.

Abonelik için bir MQTopic nesnesi oluşturursanız, bir MQDestination nesnesi sağlamadan yönetilen bir abonelik varsayılan olarak kabul edilir. Bir kuyruğu MQDestination nesnesi olarak geçerseniz, yönetilmeyen bir abonelik varsayılır. Ayarladığınız abonelik seçeneklerinin yönetilen ya da yönetilmeyen abonelik tutarlı olduğundan emin olmanız gerekir.

## queueManager

Bir konuya erişmek için kuyruk yöneticisi.

## Hedef

*destination* , bir MQQueue yönetim ortağıdır. *destination* sağlanarak, MQTopic yönetilmeyen bir abonelik olarak açılır. Konuyla ilgili yayınlar, *destination* . olarak erişilen kuyruğa teslim edilir.

## topicName

Konu adının ikinci bölümü olan bir konu dizgisi. *topicName* , *topicObject* yönetim konusu nesnesinde tanımlanan konu dizgisiyle bitleştirilir. *topicName* ' ı boş değer olarak ayarlayabilirsiniz; bu durumda konu adı *topicObject* içindeki konu dizgisiyle tanımlanır.

## topicObject

Giriş sırasında, *topicObject* , konu adının ilk bölümünü oluşturan konu dizgisini içeren konu nesnesinin adıdır. The topic string in *topicObject* is concatenated with *topicName* . Konu dizgileri oluşturulmasına ilişkin kurallar, Konu dizgilerinin birleştirilmesi altında tanımlanır.

On output, *topicObject* contains the name of the administrative topic object that is the closest match in the topic tree to the topic identified by the topic string.

## openAs

Yayınlamak ya da abone olmak için konuya erişin. Parametre, şu seçeneklerden yalnızca birini içerebilir:

- MQC.MQTOPIC\_OPEN\_AS\_SUBSCRIPTION
- MQC.MQTOPIC\_OPEN\_AS\_PUBLICATION

## seçenekler

Yayın ya da abonelik için konunun açıklarını denetleyen seçenekleri birleştirin. Abonelik ve MQC.MQOO\_\* değişmezleri için bir konuya erişmek üzere bir konuya erişmek üzere MQC.MQSO\_\* sabitlerini kullanın.

If more than one option is required, add the values together, or combine the option values using the bitwise OR operator.

## alternateUserTanıtıcısı

İşlemi bitirmek için gereken yetkiyi denetlemek için kullanılan diğer kullanıcı kimliğini belirtin. You must specify *alternateUserId* , if either MQC.MQOO\_ALTERNATE\_USER\_AUTHORITY or MQC.MQSO\_ALTERNATE\_USER\_AUTHORITY is set in the options parameter.

## subscriptionName

MQC.MQSO\_DURABLE ya da MQC.MQSO\_ALTER seçenekleri sağlandıysa, *subscriptionName* gereklidir. Her iki durumda da MQTopic , abonelik için örtük olarak açılmıştır.

MQC.MQSO\_DURABLE ayarlıysa ve abonelik varsa ya da MQC.MQSO\_ALTER ayarlandıysa ve abonelik yoksa, kural dışı durum oluşur.

## özellikler

HASH çizelgesi kullanılarak listelenen özel abonelik özelliklerinden herhangi birini ayarlayın. HASH çizelgesindeki belirlenen girişler çıkış değerleriyle güncellenir. Çıkış değerlerini bildirmek için, HASH çizelgesine girişler eklenmez.

- MQC.MQSUB\_PROP\_ALTERNATE\_SECURITY\_ID
- MQC.MQSUB\_PROP\_SUBSCRIPTION\_EXPIRY
- MQC.MQSUB\_PROP\_SUBSCRIPTION\_USER\_DATA
- MQC.MQSUB\_PROP\_SUBSCRIPTION\_CORRELATION\_ID
- MQC.MQSUB\_PROP\_PUBLICATION\_PRIORITY
- MQC.MQSUB\_PROP\_PUBLICATION\_ACCOUNTING\_TOKEN
- MQC.MQSUB\_PROP\_PUBLICATION\_APPLICATIONID\_DATA

```

public MQTopic MQQueueManager.AccessTopic(MQDestination destination, string
topicName, string topicObject, int options);
public MQTopic MQQueueManager.AccessTopic(MQDestination destination, string
topicName, string topicObject, int options, string alternateUserId);
public MQTopic MQQueueManager.AccessTopic(MQDestination destination, string
topicName, string topicObject, int options, string alternateUserId, string
subscriptionName);
public MQTopic MQQueueManager.AccessTopic(MQDestination destination, string
topicName, string topicObject, int options, string alternateUserId, string
subscriptionName, System.Collections.Hashtable properties);
public MQTopic MQQueueManager.AccessTopic(string topicName, string topicObject,
int openAs, int options);
public MQTopic MQQueueManager.AccessTopic(string topicName, string topicObject,
int openAs, int options, string alternateUserId);
public MQTopic MQQueueManager.AccessTopic(string topicName, string topicObject,
int options, string alternateUserId, string subscriptionName);
public MQTopic MQQueueManager.AccessTopic(string topicName, string
topicObject, int options, string alternateUserId, string subscriptionName,
System.Collections.Hashtable properties);

```

Bu kuyruk yöneticisinde bir konuya erişin.

MQTopic nesnelere, bazen konu nesnelere olarak adlandırılan denetim konusu nesnelere yakından ilişkilidir. Girişte, *topicObject* bir yönetici konu nesnesini işaret eder. MQTopic oluşturucusu, konu nesnesinden bir konu dizgisini alır ve bir konu adı yaratmak için bunu *topicName* ile birleştirir. *topicObject* ya da *topicName* boş değerli olabilir. Konu adı, konu ağacına eşleştirilir ve *topicObject* içinde en yakın eşleşen denetim konusu nesnesinin adı döndürülür.

MQTopic nesnesiyle ilişkili konular iki konu dizisini birleştirmenin sonudur. Birinci konu dizisi, *topicObject* ile tanımlanan yönetimle ilgili konu nesnesi tarafından tanımlanır. İkinci konu dizisi: *topicString*. The resulting topic string associated with the MQTopic object can identify multiple topics by including wildcards.

Konunun yayınlama ya da abone olma için açılıp açılmadığına bağlı olarak, konu başlıklarını yayınlamak için MQTopic .Put yöntemlerini ya da konulara ilişkin yayınları almak için MQTopic .Get yöntemlerini kullanabilirsiniz. Aynı konuyu yayınlamak ve bu konuya abone olmak istiyorsanız, yayınlamak için bir kez ve bir kez abone olmak üzere iki kez konuya erişmeniz gerekir.

Abonelik için bir MQTopic nesnesi oluşturursanız, bir MQDestination nesnesi sağlamadan yönetilen bir abonelik varsayılan olarak kabul edilir. Bir kuyruğu MQDestination nesnesi olarak geçerseniz, yönetilmeyen bir abonelik varsayılır. Ayarladığınız abonelik seçeneklerinin yönetilen ya da yönetilmeyen abonelik tutarlı olduğundan emin olmanız gerekir.

#### Hedef

*destination* , bir MQQueue yönetim ortağıdır. *destination* sağlanarak, MQTopic yönetilmeyen bir abonelik olarak açılır. Konuyla ilgili yayınlar, *destination* . olarak erişilen kuyruğa teslim edilir.

#### topicName

Konu adının ikinci bölümü olan bir konu dizisi. *topicName* , *topicObject* yönetim konusu nesnesinde tanımlanan konu dizisiyle bitirilir. *topicName* ' ı boş değer olarak ayarlayabilirsiniz; bu durumda konu adı *topicObject* içindeki konu dizisiyle tanımlanır.

#### topicObject

Giriş sırasında, *topicObject* , konu adının ilk bölümünü oluşturan konu dizisini içeren konu nesnesinin adıdır. The topic string in *topicObject* is concatenated with *topicName*. Konu dizgileri oluşturulmasına ilişkin kurallar, [Konu dizgilerinin birleştirmesi](#) altında tanımlanır.

On output, *topicObject* contains the name of the administrative topic object that is the closest match in the topic tree to the topic identified by the topic string.

#### openAs

Yayınlamak ya da abone olmak için konuya erişin. Parametre, şu seçeneklerden yalnızca birini içerebilir:

- MQC.MQTOPIC\_OPEN\_AS\_SUBSCRIPTION
- MQC.MQTOPIC\_OPEN\_AS\_PUBLICATION

### seenekler

Yayın ya da abonelik için konunun açıklarını denetleyen seenekleri birleştirin. Abonelik ve MQC.MQOO\_\* deęişmezleri için bir konuya erişmek üzere bir konuya erişmek üzere MQC.MQSO\_\* sabitlerini kullanın.

If more than one option is required, add the values together, or combine the option values using the bitwise OR operator.

### alternateUserTanıtıcısı

İşlemi bitirmek için gereken yetkiyi denetlemek için kullanılan dięer kullanıcı kimliğini belirtin. You must specify *alternateUserId*, if either MQC.MQOO\_ALTERNATE\_USER\_AUTHORITY or MQC.MQSO\_ALTERNATE\_USER\_AUTHORITY is set in the options parameter.

### subscriptionName

MQC.MQSO\_DURABLE ya da MQC.MQSO\_ALTER seenekleri sağlandıysa, *subscriptionName* gereklidir. Her iki durumda da MQTopic , abonelik için örtük olarak açılmıştır. MQC.MQSO\_DURABLE ayarlıysa ve abonelik varsa ya da MQC.MQSO\_ALTER ayarlandıysa ve abonelik yoksa, kural dışı durum oluşur.

### özellikler

HASH çizelgesi kullanılarak listelenen özel abonelik özelliklerinden herhangi birini ayarlayın. HASH çizelgesindeki belirlenen girişler çıkış deęerleriyle güncellenir. Çıkış deęerlerini bildirmek için, HASH çizelgesine girişler eklenmez.

- MQC.MQSUB\_PROP\_ALTERNATE\_SECURITY\_ID
- MQC.MQSUB\_PROP\_SUBSCRIPTION\_EXPIRY
- MQC.MQSUB\_PROP\_SUBSCRIPTION\_USER\_DATA
- MQC.MQSUB\_PROP\_SUBSCRIPTION\_CORRELATION\_ID
- MQC.MQSUB\_PROP\_PUBLICATION\_PRIORITY
- MQC.MQSUB\_PROP\_PUBLICATION\_ACCOUNTING\_TOKEN
- MQC.MQSUB\_PROP\_PUBLICATION\_APPLICATIONID\_DATA

## IMQObjectTrigger.NET arabirimi

**runmqdmn.NET** izleme programı tarafından geirilen iletileri işlemek için IMQObjectTrigger komutunu uygulayın.

### Arabirim

```
public interface IBM.WMQMonitor.IMQObjectTrigger();
```

Depending on whether sync point control is specified in the **runmqdmn** command the message is removed from the queue before or after the Execute method returns.

### Yöntemler

**void Execute (MQQueueManager queueManager, MQQueue queue, MQMessage message, string param);**

#### queueManager

İzlenmekte olan kuyruğun bulunduğu kuyruk yöneticisi.

#### kuyruk

İzlenmekte olan kuyruk.

## ileti

Kuyruktan ileti okundu.

## değiştirge

Data passed from UserParameter.

## MQC.NET arabirimi

Refer to an MQI constant by prefixing the constant name with MQC.. MQC , MQI tarafından kullanılan tüm değişmezleri tanımlar.

### Arabirim

```
System.Object
└─ IBM.WMQ.MQC
```

```
public interface IBM.WMQ.MQC extends System.Object;
```

### Örnek

```
MQQueue queue;  
queue.closeOptions = MQC.MQCO_DELETE;
```

## .NET uygulamaları için karakter kümesi tanıtıcıları

.NET IBM MQ iletilerini kodlamak için seçebileceğiniz karakter kümelerinin açıklamaları

Karakter kümesi	Tanım
37	ibm037
437	ibm437 /PC Özgün
500	ibm500
819	iso-8859-1 / latin1 / ibm819
1200	Unicode
1208	UTF-8
273	ibm273
277	ibm277
278	ibm278
280	ibm280
284	ibm284
285	ibm285
297	ibm297
420	ibm420
424	ibm424
737	ibm737 /PC Yunanca
775	ibm775 /PC Baltık



<b>Karakter kümesi</b>	<b>Tanım</b>
813	iso-8859-7 /greek/ ibm813
838	ibm838
850	ibm850 /PC Latin 1
852	ibm852 /PC Latin 2
855	ibm855 /PC Kiril
856	ibm856
857	ibm857 /PC Türkçe
860	ibm860 /PC Portekizcesi
861	ibm861 /PC İzlandaca
862	ibm862 /PC İbranice
863	ibm863 /PC Kanada Fransızcası
864	ibm864 /PC Arapça
865	ibm865 /PC Nordic
866	ibm866 /PC Rusça
868	ibm868
869	ibm869 /PC Modern Yunanca
870	ibm870
871	ibm871
874	ibm874
875	ibm875
912	iso-8859-2 / latin2 / ibm912
913	iso-8859-3 / latin3 / ibm913
914	iso-8859-4 / latin4 / ibm914
915	iso-8859-5 /cyrillik/ ibm915
916	iso-8859-8 /hebrew/ ibm916
918	ibm918
920	iso-8859-9 / latin5 / ibm920
921	ibm921
922	ibm922
930	ibm930
932	PC Japonca
933	ibm933
935	ibm935
937	ibm937
939	ibm939

Karakter kümesi	Tanım
942	ibm942
943	ibm943
948	ibm948
949	ibm949
950	ibm950 /Büyük 5 Geleneksel Çince
954	EUCJIS
964	ibm964 /CNS 11643 Geleneksel Çince
970	ibm970
1006	ibm1006
1025	ibm1025
1026	ibm1026
1089	iso-8859-6 /arapça/ ibm1089
1097	ibm1097
1098	ibm1098
1112	ibm1112
1122	ibm1122
1123	ibm1123
1124	ibm1124
1250	Windows Latin 2
1251	Windows Kiril
1252	Windows Latin 1
1253	Windows Yunanca
1254	Windows Türkçe
1255	Windows İbranice
1256	Windows Arapça
1257	Windows Baltık
1258	Windows Vietnam dili
1381	ibm1381
1383	ibm1383
2022	JIS
5601	ksc-5601 Korece
33722	ibm33722

## IBM MQ C++ sınıfları

IBM MQ C++ sınıfları, IBM MQ Message Queue Interface (MQI) ' ı sarsalıyor. Bu sınıfların tümünü kapsayan tek bir C++ üstbilgi dosyası ( **imqi.hpp** ) vardır.

Her sınıf için aşağıdaki bilgiler görüntülenir:

### **Sınıf sıradüzeni çizgesi**

Varsa, birincil üst sınıflarıyla olan kalıtım ilişkisinde sınıfı gösteren bir sınıf çizgesi.

### **Diğer ilgili sınıflar**

Belge bağlantıları, üst sınıflar gibi diğer ilgili sınıflara ve yöntem imzalarında kullanılan nesnelerin sınıflarına bağlar.

### **Nesne öznitelikleri**

Sınıfın öznitelikleri. Bunlar, herhangi bir üst sınıf için tanımlanmış özniteliklere ek olarak eklenmiştir. Birçok öznitelik IBM MQ veri yapısı üyelerini yansıtır (bkz. [“C++ ve MQI çapraz başvurusu” sayfa 1760](#)). Ayrıntılı açıklamalar için bkz. [“Nesnelerin öznitelikleri” sayfa 785](#).

### **Oluşturucular**

Sınıf nesnesi yaratmak için kullanılan özel yöntemlerin imzaları.

### **Nesne yöntemleri (genel)**

Kendi çalışması için sınıfın bir eşgörünümünü gerektiren ve kullanım kısıtlaması olmayan yöntemlerin imzaları.

Geçerli olduğu yerlerde, aşağıdaki bilgiler de gösterilir:

### **Sınıf yöntemleri (genel)**

Sınıfın bir eşgörünümü gerektirmeyen ve kullanım kısıtlamalarına sahip olmayan yöntemlerin imzaları.

### **Aşırı yüklü (üst sınıf) yöntemler**

Üst sınıflarda tanımlanan bu sanal yöntemlerin imzaları, ancak bu sınıf için farklı, polimorfik, davranış sergiler.

### **Nesne yöntemleri (korunan)**

Kendi işlemleri için sınıfın bir eşgörünümünü gerektiren ve türetilmiş sınıfların somutlamaları tarafından kullanılmak üzere ayrılmış yöntemlerin imzaları. Bu bölüm, sınıf kullanıcılarının tersine yalnızca sınıf yazarlarına ilgi göstermektedir.

### **Nesne verileri (korunan)**

Türetilmiş sınıfların somutlamaları için kullanılacak nesne eşgörünümü verilerine ilişkin somutlama ayrıntıları. Bu bölüm, sınıf kullanıcılarının tersine yalnızca sınıf yazarlarına ilgi göstermektedir.

### **Neden kodları**

MQRC\_ \* değerleri (bkz. API tamamlama ve neden kodları) başarısız olan yöntemlerden beklenebilir. Sınıf nesnesi için ortaya çıkabilecek neden kodlarının ayrıntılı bir listesi için üst sınıf belgelerine bakın. Bir sınıfa ilişkin neden kodlarının belgelenmiş listesinde, üst sınıflara ilişkin neden kodları yer almıyorsa.

### **Not:**

1. Bu sınıfların nesnelere iş parçacığı korumalı değildir. Bu, en iyi başarıyı sağlar, ancak birden çok iş parçacığından herhangi bir nesneye erişmeye özen göstermenizi sağlar.
2. Çok iş parçacıklı bir program için, her iş parçacığı için ayrı bir ImqQueueManager nesnesi kullanılması önerilir. Her bir yönetici nesnesi, farklı iş parçacıklarındaki nesnelerin birbirlerinden yalıtıldığından emin olmak için, kendi bağımsız derlemine sahip olmalıdır.

Sınıflar şunlardır:

- [“ImqAuthenticationKayıt C++ sınıfı” sayfa 1775](#)
- [“ImqBinary C++ sınıfı” sayfa 1777](#)
- [“ImqCache C++ sınıfı” sayfa 1779](#)
- [“ImqChannel C++ sınıfı” sayfa 1782](#)
- [“ImqCICSBridgeHeader C++ sınıfı” sayfa 1787](#)
- [“ImqDeadLetterHeader C++ sınıfı” sayfa 1793](#)
- [“ImqDistributionListe C++ sınıfı” sayfa 1795](#)
- [“ImqError C++ sınıfı” sayfa 1797](#)

- [“ImqGetMessageOptions C++ sınıfı” sayfa 1798](#)
- [“ImqHeader C++ sınıfı” sayfa 1801](#)
- [“ImqIMSBridgeHeader C++ sınıfı” sayfa 1803](#)
- [“ImqItem C++ sınıfı” sayfa 1806](#)
- [“ImqMessage C++ sınıfı” sayfa 1807](#)
- [“ImqMessageTracker C++ sınıfı” sayfa 1814](#)
- [“ImqNamelist C++ sınıfı” sayfa 1816](#)
- [“ImqObject C++ sınıfı” sayfa 1818](#)
- [“ImqProcess C++ sınıfı” sayfa 1823](#)
- [“ImqPutMessageOptions C++ sınıfı” sayfa 1825](#)
- [“ImqQueue C++ sınıfı” sayfa 1827](#)
- [“ImqQueueManager C++ sınıfı” sayfa 1837](#)
- [“ImqReferenceHeader C++ sınıfı” sayfa 1853](#)
- [“ImqString C++ sınıfı” sayfa 1856](#)
- [“ImqTrigger C++ sınıfı” sayfa 1861](#)
- [“ImqWorkÜstbilgi C++ sınıfı” sayfa 1864](#)

## C++ ve MQI çapraz başvurusu

Bu konu derlemi, C++ ile MQI 'a ilişkin bilgileri içerir.

Bu bilgileri [“MQI ' da kullanılan veri tipleri” sayfa 235](#) ile birlikte okuyun.

Bu çizelge, MQI veri yapılarını C++ sınıflarına ilişkilendirir ve dosyaları içerir. Aşağıdaki konularda, her C++ sınıfına ilişkin çapraz başvuru bilgileri gösterilir. Bu çapraz başvurular, temel IBM MQ yordamsal arabirimlerinin kullanımıyla ilgilidir. The classes ImqBinary, ImqDistributionList, and ImqString have no attributes that fall into this category and are excluded.

<i>Çizelge 846. Veri yapısı, sınıf ve include-file çapraz başvuru</i>		
<b>Veri yapısı</b>	<b>Sınıf</b>	<b>İçerme dosyası</b>
MQAIR	ImqAuthenticationKaydı	imqair.hpp
	ImqBinary	imqbin.hpp
	ImqCache	imqcac.hpp
MQCD	ImqChannel	imqchl.hpp
MQCIH	ImqCICSBridgeHeader	imqcih.hpp
MQDLH	ImqDeadLetterHeader	imqdlh.hpp
MQYA	ImqDistributionListesi	imqdst.hpp
	ImqError	imqerr.hpp
MQGMO	ImqGetMessageOptions	imqgmo.hpp
	ImqHeader	imqhdr.hpp
MQIH	ImqIMSBridgeHeader	imqiih.hpp
	ImqItem	imqitm.hpp
Mqmd	ImqMessage	imqmsg.hpp
	ImqMessageİzleyicisi	imqmtr.hpp

Çizelge 846. Veri yapısı, sınıf ve include-file çapraz başvuru (devamı var)

Veri yapısı	Sınıf	İçerme dosyası
	ImqNamelist	imqnml.hpp
MQOD, MQRR	ImqObject	imqobj.hpp
MQPMO, MQPMR, MQRR	ImqPutMessageOptions	imqpmo.hpp
	ImqProcess	imqpro.hpp
	ImqQueue	imqque.hpp
MQBO, MQCNO, MQCSP	ImqQueueYöneticisi	imqmgr.hpp
MQRMH	ImqReferenceÜstbilgisi	imqrfh.hpp
	ImqString	imqstr.hpp
MQTM	ImqTrigger	imqtrg.hpp
MQTMCM		
MQTMCM2	ImqTrigger	imqtrg.hpp
MQXQH		
MQWIH	ImqWorkÜstbilgisi	imqwih.hpp

## ImqAuthenticationKayıt çapraz başvurusu

Öznitelikler, veri yapıları, alanlar ve ImqAuthenticationKayıt C++ sınıfı çağrılarını çapraz başvuru.

Çizelge 847. Öznitelikler, veri yapıları, alanlar ve aramalar

Öznitelik	Veri yapısı	Alan	Ara
Bağlantı adı	MQAIR	AuthInfoConnName	MQCONN
parola	MQAIR	LDAPPassword	MQCONN
tip	MQAIR	AuthInfoTipi	MQCONN
kullanıcı adı	MQAIR	LDAPUserNamePtr	MQCONN
	MQAIR	LDAPUserNameGörelisi Konumu	MQCONN
	MQAIR	LDAPUserNameUzunluğu	MQCONN

## ImqCache çapraz başvuru

Cross-reference of attributes and calls for the ImqCache C++ class.

Çizelge 848. Öznitelikler ve çağrılar

Öznitelik	Ara
otomatik arabellek	MQGet
arabellek uzunluğu	MQGet
arabellek göstergesi	MQGET, MQPUT
veri uzunluğu	MQGet
veri görelisi konumu	MQGet

<i>Çizelge 848. Öznitelikler ve çağrılar (devamı var)</i>	
<b>Öznitelik</b>	<b>Ara</b>
veri göstergesi	MQGet
ileti uzunluğu	MQGET, MQPUT

## **ImqChannel çapraz başvuru**

Cross-reference of attributes, data structures, fields, and calls for the ImqChannel C++ class.

<i>Çizelge 849. Öznitelikler, veri yapıları, alanlar ve aramalar</i>			
<b>Öznitelik</b>	<b>Veri yapısı</b>	<b>Alan</b>	<b>Ara</b>
toplu kalp ritmi	MQCD	BatchHeartbeat	MQCONN
Kanal Adı	MQCD	ChannelName	MQCONN
Bağlantı adı	MQCD	ConnectionName	MQCONN
	MQCD	ShortConnectionAdı	MQCONN
üstbilgi sıkıştırması	MQCD	HdrCompListesi	MQCONN
kalp ritmi aralığı	MQCD	HeartbeatInterval	MQCONN
canlı tutma aralığı	MQCD	KeepAliveAralığı	MQCONN
Yerel adres	MQCD	LocalAddress	MQCONN
ileti uzunluğu üst sınırı	MQCD	MaxMsgUzunluğu	MQCONN
ileti sıkıştırma	MQCD	MsgCompListesi	MQCONN
kip adı	MQCD	ModeName	MQCONN
parola	MQCD	Parola	MQCONN
çıkış sayısı alma	MQCD		MQCONN
çıkış adlarını al	MQCD	ReceiveExit	MQCONN
	MQCD	ReceiveExitsTanımlı	MQCONN
	MQCD	ReceiveExitPtr	MQCONN
kullanıcı verilerini al	MQCD	ReceiveUserVerileri	MQCONN
	MQCD	ReceiveUserDataPtr	MQCONN
güvenlik çıkış adı	MQCD	SecurityExit	MQCONN
güvenlik kullanıcı verileri	MQCD	SecurityUserVerileri	MQCONN
çıkış sayısı gönder	MQCD		MQCONN
çıkış adlarını gönder	MQCD	SendExit	MQCONN
	MQCD	SendExitsTanımlı	MQCONN
	MQCD	SendExitPtr	MQCONN
kullanıcı verilerini gönder	MQCD	SendUserVerileri	MQCONN
	MQCD	SendUserDataPtr	MQCONN
SSL CipherSpec	MQCD	sslCipherBelirtimi	MQCONN
SSL istemcisi kimlik denetimi tipi	MQCD	sslClientKimlik Doğrulaması	MQCONN

<i>Çizelge 849. Öznitelikler, veri yapıları, alanlar ve aramalar (devamı var)</i>			
<b>Öznitelik</b>	<b>Veri yapısı</b>	<b>Alan</b>	<b>Ara</b>
SSL eş adı	MQCD	sslPeerAdı	MQCONN
hareket programı adı	MQCD	TpName	MQCONN
taşıma tipi	MQCD	TransportType	MQCONN
kullanıcı kimliği	MQCD	UserIdentifier	MQCONN

## **ImqCICSBridgeHeader çapraz başvurusu**

ImqCICSBridgeHeader C++ sınıfına ilişkin özniteliklere, veri yapılarına ve alanlara çapraz başvuru.

<i>Çizelge 850. Özniteliklerin, veri yapılarının ve alanların eşlenmesi</i>		
<b>Öznitelik</b>	<b>Veri yapısı</b>	<b>Alan</b>
köprü olağandışı bitiş kodu	MQCIH	AbendCode
ADS tanımlayıcı	MQCIH	AdsDescriptor
uyarı tanıtıcısı	MQCIH	AttentionId
kimlik doğrulayıcı	MQCIH	Kimlik doğrulayıcı
köprü tamamlama kodu	MQCIH	BridgeCompletionKodu
köprü hata göreli konumu	MQCIH	ErrorOffset
köprü neden kodu	MQCIH	BridgeReason
köprü iptal kodu	MQCIH	CancelCode
etkileşimli görev	MQCIH	ConversationalTask
imleç konumu	MQCIH	CursorPosition
tesis simgesi	MQCIH	Tesis
tesis zaman tutma	MQCIH	FacilityKeepSüresi
gibi bir tesis	MQCIH	FacilityLike
işlev	MQCIH	İşlev
bekleme süresi aralığı	MQCIH	GetWaitAralığı
Bağlantı tipi	MQCIH	LinkType
sonraki işlem tanıtıcısı	MQCIH	NextTransactionTanıtıcısı
çıkış veri uzunluğu	MQCIH	OutputDataUzunluğu
yanıtlama biçimi	MQCIH	ReplyToBiçimi
köprü dönüş kodu	MQCIH	ReturnCode
başlatma kodu	MQCIH	StartCode
görev bitiş durumu	MQCIH	TaskEndDurumu
işlem tanıtıcısı	MQCIH	TransactionId
uow denetimi	MQCIH	UowControl
sürüm	MQCIH	S\u00fcr\u00fcm

## ImqDeadLetterHeader çapraz başvurusu

ImqDeadLetterHeader C++ sınıfına ilişkin özniteliklerin, veri yapılarının ve alanların çapraz başvurusu.

<i>Çizelge 851. Özniteliklerin, veri yapılarının ve alanların eşlenmesi</i>		
Öznitelik	Veri yapısı	Alan
ölü harf neden kodu	MQDLH	Neden
Hedef kuyruk yöneticisi adı	MQDLH	DestQMgrAdı
hedef kuyruk adı	MQDLH	DestQName
Koyma Uygulaması Adı	MQDLH	PutApplAdı
Koyma Uygulaması Tipi	MQDLH	PutApplTipi
bitiş tarihi	MQDLH	PutDate
sürme zamanı	MQDLH	PutTime

## ImqError çapraz başvurusu

Cross-reference of attributes and calls for the ImqError C++ class.

<i>Çizelge 852. Öznitelikler ve çağrılar</i>	
Öznitelik	Ara
tamamlanma kodu	MQBACK, MQBEGIN, MQCLOSE, MQCMIT, MQCONN, MQCONNX, MQDISC, MQGET, MQINQ, MQOPEN, MQPUT, MQSET
Neden Kodu	MQBACK, MQBEGIN, MQCLOSE, MQCMIT, MQCONN, MQCONNX, MQDISC, MQGET, MQINQ, MQOPEN, MQPUT, MQSET

## ImqGetMessageOptions çapraz başvuru

ImqGetMessageOptions C++ sınıfına ilişkin özniteliklerin, veri yapılarının ve alanların çapraz başvurusu.

<i>Çizelge 853. Özniteliklerin, veri yapılarının ve alanların eşlenmesi</i>		
Öznitelik	Veri yapısı	Alan
grup durumu	MQGMO	GroupStatus
eşleşme seçenekleri	MQGMO	MatchOptions
ileti simgesi	MQGMO	MessageToken
seçenekler	MQGMO	Seçenekler
çözömlenen kuyruk adı	MQGMO	ResolvedQName
döndürölen uzunluk	MQGMO	ReturnedLength
kesimlere ayırma	MQGMO	Bölümlleme
bölüm durumu	MQGMO	SegmentStatus
	MQGMO	Signal1
	MQGMO	Signal2
syncpoint katılımı	MQGMO	Seçenekler
Bekleme Aralığı	MQGMO	WaitInterval



## ImqHeader çapraz başvurusu

ImqHeader C++ sınıfına ilişkin öznitelikler, veri yapıları ve alanlar çapraz başvuru.

<i>Çizelge 854. Özniteliklerin, veri yapılarının ve alanların eşlenmesi</i>		
Öznitelik	Veri yapısı	Alan
karakter kümesi	MQDLH, MQIIH	CodedCharSetId
Kodlama	MQDLH, MQIIH	Kodlama
biçim	MQDLH, MQIIH	Biçim
üstbilgi işaretleri	MQIIH, MQRMH	İşaretler

## ImqIMSBridgeHeader çapraz başvurusu

ImqAuthenticationRecord C++ sınıfına ilişkin özniteliklere, veri yapılarına ve alanlara çapraz başvuru.

<i>Çizelge 855. Özniteliklerin, veri yapılarının ve alanların eşlenmesi</i>		
Öznitelik	Veri yapısı	Alan
kimlik doğrulayıcı	MQIH	Kimlik doğrulayıcı
kesinleştirme kipi	MQIH	CommitMode
mantıksal uçbirim geçersiz kılma	MQIH	LTermOverride
ileti biçimi hizmetleri eşlemi adı	MQIH	MFSMapName
yanıtlama biçimi	MQIH	ReplyToBiçimi
güvenlik kapsamı	MQIH	SecurityScope
işlem eşgörünümü tanıtıcısı	MQIH	TranInstanceTanıtıcısı
hareket durumu	MQIH	TranState

## ImqItem çapraz başvuru

Cross-reference of attributes and calls for the ImqItem C++ class.

<i>Çizelge 856. Öznitelikler ve çağrılar</i>	
Öznitelik	Ara
yapı tanıtıcısı	MQGet

## ImqMessage çapraz başvuru

Cross-reference of attributes, data structures, fields, and calls for the ImqMessage C++ class.

<i>Çizelge 857. Öznitelikler, veri yapıları, alanlar ve aramalar</i>			
Öznitelik	Veri yapısı	Alan	Ara
Uygulama Tnt Verisi	Mqmd	ApplIdentityVerileri	
Uygulama Kökeni Verisi	Mqmd	ApplOriginVerileri	
Geriletme Sayısı	Mqmd	BackoutCount	
karakter kümesi	Mqmd	CodedCharSetId	
Kodlama	Mqmd	Kodlama	

<i>Çizelge 857. Öznitelikler, veri yapıları, alanlar ve aramalar (devamı var)</i>			
<b>Öznitelik</b>	<b>Veri yapısı</b>	<b>Alan</b>	<b>Ara</b>
Son kullanma tarihi	Mqmd	Son kullanma tarihi	
biçim	Mqmd	Biçim	
İleti İşaretleri	Mqmd	MsgFlags	
ileti tipi	Mqmd	MsgType	
offset	Mqmd	Görelî Konum	
Özgün Uzunluk	Mqmd	OriginalLength	
Kalıcılık	Mqmd	Kalıcılık	
öncelik	Mqmd	Öncelik	
Koyma Uygulaması Adı	Mqmd	PutApplAdı	
Koyma Uygulaması Tipi	Mqmd	PutApplTipi	
bitiş tarihi	Mqmd	PutDate	
sürme zamanı	Mqmd	PutTime	
yanıtlama-kuyruk yöneticisi adı	Mqmd	ReplyToQMgr	
yanıt kuyruğu adı	Mqmd	ReplyToQ	
rapor	Mqmd	Rapor	
Sıra Numarası	Mqmd	MsgSeqNumarası	
toplam ileti uzunluğu		DataLength	MQGet
kullanıcı kimliği	Mqmd	UserIdentifier	

### **ImqMessageİzleyici çapraz başvurusu**

ImqMessageTracker C++ sınıfına ilişkin özniteliklere, veri yapılarına ve alanlara çapraz başvuru.

<i>Çizelge 858. Özniteliklerin, veri yapılarının ve alanların eşlenmesi</i>		
<b>Öznitelik</b>	<b>Veri yapısı</b>	<b>Alan</b>
muhasebe simgesi	Mqmd	AccountingToken
İlinti Tanıtıcısı	Mqmd	CorrelId
Geribildirim	Mqmd	Geribildirim
grup tanıtıcısı	Mqmd	GroupId
İleti Tnt	Mqmd	MsgId

### **ImqNamelist çapraz başvurusu**

Cross-reference of attributes, inquiries, and calls for the ImqNamelist C++ class.

<i>Çizelge 859. Öznitelikler, sorgular ve aramalar</i>		
<b>Öznitelik</b>	<b>Sorgu</b>	<b>Ara</b>
ad sayısı	MQIA_NAME_COUNT	MQINQ
ad listesi adı	MQCA_NAMELIK_ADı	MQINQ

## ImqObject çapraz başvurusu

Öznitelikler, veri yapıları, alanlar, sorgular ve ImqObject C++ sınıfı çağrılarını çapraz başvuru.

<i>Çizelge 860. Öznitelikler, veri yapıları, alanlar, sorgular ve aramalar</i>				
Öznitelik	Veri yapısı	Alan	Sorgu	Ara
değiştirme tarihi			MQCA_ALTERATION_DEĞERİ	MQINQ
değiştirme zamanı			MQCA_ALTERATION_KEZ	MQINQ
diğer kullanıcı kimliği	MQOD	AlternateUserTanıtıcısı		
diğer güvenlik tanıtıcısı				
kapatma seçenekleri				MQCLOSE
açıklama			MQCA_Q_DESC, MQCA_Q_MGR_DESC, MQCA_PROCESS_DESC	MQINQ
ad	MQOD	ObjectName	MQCA_Q_MGR_NAME, MQCQ_Q_NAME, MQCA_PROCESS_NAME	MQINQ
açma seçenekleri				MQOPEN
açık durum				MQAç, MQCLOSE
kuyruk yöneticisi tanıtıcısı	kuyruk yöneticisi tanıtıcısı		MQCA_Q_MGR_IDENTIFIER	MQINQ

## ImqProcess çapraz başvurusu

Öznitelikler, sorgular ve ImqAuthenticationKayıt C++ sınıfı çağrılarını çapraz başvuru.

<i>Çizelge 861. Öznitelikler, sorgular ve aramalar</i>		
Öznitelik	Sorgu	Ara
Uygulama tanıtıcısı	MQCA_APPL_ID	MQINQ
Uygulama Tipi	MQIA_APPL_TYPE	MQINQ
ortam verileri	MQCA_ENV_DATA	MQINQ
Kullanıcı verileri	MQCA_USER_DATA	MQINQ

## ImqPutMessageOptions çapraz başvurusu

ImqAuthenticationRecord C++ sınıfına ilişkin özniteliklere, veri yapılarına ve alanlara çapraz başvuru.

<i>Çizelge 862. Özniteliklerin, veri yapılarının ve alanların eşlenmesi</i>		
Öznitelik	Veri yapısı	Alan
bağlam başvurusu	MQPMO	Bağlam
	MQPMO	InvalidDestSayısı
	MQPMO	KnownDestSayı

Çizelge 862. Özniteliklerin, veri yapılarının ve alanların eşlenmesi (devamı var)

Öznitelik	Veri yapısı	Alan
seçenekler	MQPMO	Seçenekler
kayıt alanları	MQPMO	PutMsgRecFields
çözömlenen kuyruk yöneticisi adı	MQPMO	ResolvedQMgrAdı
çözömlenen kuyruk adı	MQPMO	ResolvedQName
	MQPMO	Zamanaşımı
	MQPMO	UnknownDestSayısı
syncpoint katılımı	MQPMO	Seçenekler

## ImqQueue çapraz başvurusu

Cross-reference of attributes, data structures, fields, inquiries, and calls for the ImqQueue C++ class.

Çizelge 863. ImqQueue çapraz başvurusu

Öznitelik	Veri yapısı	Alan	Sorgu	Ara
backout requesue adı			MQCA_BACKUP_REQ_Q_NAME	MQINQ
Geriletme eşiğı			MQIA_BACKOUT_THRESHOLD	MQINQ
temel kuyruk adı			MQCA_BASE_Q_ADı	MQINQ
Küme adı			MQCA_CLUSTER_NAME	MQINQ
küme adı listesi adı			MQCA_CLUSTER_NAMELIST	MQINQ
küme iş yükü sırası			MQIA_CLWL_Q_RANK	MQINQ
küme iş yükü önceliğı			MQIA_CLWL_Q_PRIORITY	MQINQ
küme iş yükü kullanım kuyruğı			MQIA_CLWL_USEQ	MQINQ
yaratma tarihi			MQCA_CREATION_DEğERI	MQINQ
Oluşturulma zamanı			MQCA_CREATION_TIME	MQINQ
yürürlükteki derinlik			MQIA_CURRENT_Q_DEPTH	MQINQ
varsayılan bağlama			MQIA_DEF_BIND	MQINQ
varsayılan giriş açma seçeneğı			MQIA_DEF_INPUT_OPEN_OPTION	MQINQ
varsayılan kalıcılık			MQIA_DEF_PERSISTENCE	MQINQ
varsayılan öncelik			MQIA_DEF_PRIORITY	MQINQ
Tanımlama Tipi			MQIA_TANIMLI TIPI	MQINQ
derinlik yüksek olayı			MQIA_Q_DEPTH_HIGH_EVENT	MQINQ
derinlik üst sınırı			MQIA_Q_DEPTH_HIGH_LIMIT	MQINQ
derinlik düşük olay			MQIA_Q_DEPTH_LOW_EVENT	MQINQ
derinlik alt sınırı			MQIA_Q_DEPTH_LOW_LIMIT	MQINQ
derinlik üst sınırı olayı			MQIA_Q_DEPTH_MAX_LIMIT	MQINQ

Çizelge 863. İmqQueue çapraz başvurusu (devamı var)				
Öznitelik	Veri yapısı	Alan	Sorgu	Ara
dağıtım listeleri			MQIA_DIST_LISTS	MQINQ, MQSET
dinamik kuyruk adı	MQOD	DynamicQName		
harden geri tepme			MQIA_HARDEN_GET_BACKOUT	MQINQ
dizin tipi			MQIA_INDEX_TYPE	MQINQ
alma engelini			MQIA_INHIBIT_GET	MQINQ, MQSET
engelleyici			MQIA_INHIBIT_PUT	MQINQ, MQSET
başlatma kuyruğu adı			MQCA_INTIATION_Q_NAME	MQINQ
derinlik üst sınırı			MQIA_MAX_Q_DEPTH	MQINQ
ileti uzunluğu üst sınırı			MQIA_MAX_MSG_LENGTH	MQINQ
ileti teslim sırası			MQIA_MSG_DELIVERY_SEQUENCE	MQINQ
sonraki dağıtılmış kuyruk				
kalıcı olmayan ileti sınıfı			MQIA_NPM_SINIFI	MQINQ
açık giriş sayısı			MQIA_OPEN_INPUT_COUNT	MQINQ
açık çıkış sayısı			MQIA_OPEN_OUTPUT_COUNT	MQINQ
önceki dağıtılmış kuyruk				
İşlem adı			MQCA_PROCESS_NAME	MQINQ
kuyruk muhasebesi			MQIA_ACCOUNTING_Q	MQINQ
Kuyruk yöneticisi adı	MQOD	ObjectQMgrAdı		
kuyruk izleme			MQIA_MONITORING_Q	MQINQ
kuyruk istatistikleri			MQIA_STATISTICS_Q	MQINQ
kuyruk tipi			MQIA_Q_TYPE	MQINQ
uzak kuyruk yöneticisi adı			MQCA_REMOTE_Q_MGR_NAME	MQINQ
uzak kuyruk adı			MQCA_REMOTE_Q_NAME	MQINQ
çözümlenen kuyruk yöneticisi adı	MQOD	ResolvedQMgrAdı		
çözümlenen kuyruk adı	MQOD	ResolvedQName		
alıkoyma aralığı			MQIA_RETENTION_INTERVAL	MQINQ
kapsam			MQIA_SCOPE	MQINQ

<i>Çizelge 863. ImqQueue çapraz başvurusu (devamı var)</i>				
<b>Öznitelik</b>	<b>Veri yapısı</b>	<b>Alan</b>	<b>Sorgu</b>	<b>Ara</b>
hizmet aralığı			MQIA_Q_SERVICE_INTERVAL	MQINQ
hizmet aralığı olayı			MQIA_Q_SERVICE_INTERVAL_OLAY	MQINQ
paylaşılabilirlik			MQIA_SHAREYETENEĞİ	MQINQ
depolama sınıfı			MQCA_STORAGE_CLASS	MQINQ
iletim kuyruğu adı			MQCA_XMIT_Q_ADI	MQINQ
tetikleyici denetimi			MQIA_TRIGGER_CONTROL	MQINQ, MQSET
tetikleyici verileri			MQCA_TRIGGER_DATA	MQINQ, MQSET
tetikleyici derinliği			MQIA_TRIGGER_DEPTH	MQINQ, MQSET
tetikleyici ileti önceliği			MQIA_TRIGGER_MSG_PRIORITY	MQINQ, MQSET
Tetikleme Tipi			MQIA_TRIGGER_TYPE	MQINQ, MQSET
Kullanım			MQIA_KULLANICI	MQINQ

### **ImqQueueManager çapraz başvurusu**

Cross-reference of attributes, data structures, fields, inquiries, and calls for the ImqQueueManager C++ class.

<i>Çizelge 864. Öznitelikler, veri yapıları, alanlar, sorgular ve aramalar</i>				
<b>Öznitelik</b>	<b>Veri yapısı</b>	<b>Alan</b>	<b>Sorgu</b>	<b>Ara</b>
muhasabe bağlantıları geçersiz kılma			MQIA_ACCOUNTING_CONN_OVERRIDE	MQINQ
muhasabe aralığı			MQIA_ACCOUNTING_INTERVAL	MQINQ
etkinlik kaydı			MQIA_ACTIVITY_RECORD	MQINQ
yeni mca denetimi kabul et			MQIA_ADOPTNEWMCA_CHECK	MQINQ
yeni mca tipini benimseme			MQIA_ADOPTNEWMCA_TYPE	MQINQ
Kimlik Doğrulama Tipi	MQCSP	AuthenticationType		MQCONN
yetki olayı			MQIA_AUTHORITY_OLAY	MQINQ
başlangıç seçenekleri	MQBO	Seçenekler		MQBEGIN
köprü olayı			MQIA_BRIDGE_OLAY	MQINQ

Çizelge 864. Öznitelikler, veri yapıları, alanlar, sorgular ve aramalar (devamı var)

Öznitelik	Veri yapısı	Alan	Sorgu	Ara
kanal otomatik tanımı			MQIA_CHANNEL_AUTO_DEF	MQINQ
kanal otomatik tanımlama olayı			MQIA_CHANNEL_AUTO_OLAY	MQIA
kanal otomatik tanımlama çıkışı			MQIA_CHANNEL_AUTO_EXIT	MQIA
kanal olayı			MQIA_CHANNEL_OLAY	MQINQ
kanal başlatıcı bağdaştırıcıları			MQIA_CHINIT_ADAPTERS	MQINQ
kanal başlatıcı denetimi			MQIA_CHINIT_CONTROL	MQINQ
kanal başlatan dağıtıcıları			MQIA_CHINIT_DAĞITICILAR	MQINQ
kanal başlatıcı izleme otomatik başlatma			MQIA_CHINIT_TRACE_AUTO_START	MQINQ
kanal başlatıcı izleme çizelgesi büyüklüğü			MQIA_CHINIT_TRACE_TABLE_SIZE	MQINQ
kanal izleme			MQIA_MONITORING_KANAL	MQINQ
kanal başvurusu	MQCD	ChannelType		MQCONN
kanal istatistikleri			MQIA_STATISTICS_CHANNEL	MQINQ
karakter kümesi			MQIA_CODED_CHAR_SET_ID	MQINQ
küme gönderen izleme			MQIA_MONITORING_AUTO_CLUSSDR	MQINQ
küme gönderen istatistikleri			MQIA_STATISTICS_AUTO_CLUSSDR	MQINQ
küme iş yükü verileri			MQCA_CLUSTER_WORKLOAD_DATA	MQINQ
küme iş yükü çıkışı			MQCA_CLUSTER_WORKLOAD_EXIT	MQINQ
küme iş yükü uzunluğu			MQIA_CLUSTER_WORKLOAD_LENGTH	MQINQ
küme iş yükü mru			MQIA_CLWL_MRU_KANALS	MQINQ
küme iş yükü kullanım kuyruğu			MQIA_CLWL_USEQ	MQINQ
komut olayı			MQIA_COMMAND_OLAY	MQINQ
komut giriş kuyruğu adı			MQCA_COMMAND_INPUT_Q_NAME	MQINQ
komut düzeyi			MQIA_KOMUT_DÜZEYI	MQINQ

Çizelge 864. Öznitelikler, veri yapıları, alanlar, sorgular ve aramalar (devamı var)

Öznitelik	Veri yapısı	Alan	Sorgu	Ara
komut sunucusu denetimi			MQIA_CMD_SERVER_CONTROL	MQINQ
bağlantı seçenekleri	MQCNO	Seçenekler		MQCONN, MQCONNX
Bağlantı Tanıtıcısı	MQCNO	ConnectionId		MQCONNX
Bağlantı durumu				MQCONN, MQCONNX, MQDISC
bağlantı etiketi	MQCD	ConnTag		MQCONNX
şifreleme donanımı	MQSCO	CryptoHardware		MQCONNX
-Ölü harf kuyruğu adı			MQCA_DEAD_LETTER_Q_NAME	MQINQ
varsayılan iletim kuyruğu adı			MQCA_DEF_XMIT_Q_ADı	MQINQ
dağıtım listeleri			MQIA_DIST_LISTS	MQINQ
dns grubu			MQCA_DNS_GROUP	MQINQ
dns wlm			MQIA_DNS_WLM	MQINQ
ilk kimlik doğrulama kaydı	MQSCO	AuthInfoRecOffset		MQCONNX
	MQSCO	AuthInfoRecPtr		MQCONNX
etkinliği engelle			MQIA_INHIBIT_EVENT	MQINQ
ip adresi sürümü			MQIA_IP_ADDRESS_VERSION	MQINQ
anahtar havuzu	MQSCO	KeyRepository		MQCONNX
anahtar sıfırlama sayısı	MQSCO	KeyResetSayı		MQCONNX
dinleyici zamanlayıcısı			MQIA_LISTINER_TIMER	MQINQ
yerel olay			MQIA_LOCAL_OLAY	MQINQ
günlüğe kaydetme olayı			MQIA_LOGGER_OLAY	MQINQ
lu grup adı			MQCA_LU_GROUP_NAME	MQINQ
lu adı			MQCA_LU_NAME	MQINQ
lu62 kol soneki			MQCA_LU62_ARM_SUFFIX	MQINQ
lu62 kanalları			MQIA_LU62_CHANNELS	MQINQ
etkin kanal üst sınırı			MQIA_ACTIVE_KANALS	MQINQ
kanal üst sınırı			MQIA_MAX_KANALS	MQINQ
işleyici üst sınırı			MQIA_MAX_HANDHANS	MQINQ



Çizelge 864. Öznitelikler, veri yapıları, alanlar, sorgular ve aramalar (devamı var)

Öznitelik	Veri yapısı	Alan	Sorgu	Ara
ileti uzunluğu üst sınırı			MQIA_MAX_MSG_LENGTH	MQINQ
en yüksek öncelik			MQIA_MAX_PRIORITY	MQINQ
kesinleştirilmemiş iletiler üst sınırı			MQIA_MAX_UNCOMMITTED_MSGS	MQINQ
mqi muhasebe			MQIA_ACCOUNTING_MQI	MQINQ
mqi istatistikleri			MQIA_STATISTICS_MQI	MQINQ
giden kapı üst sınırı			MQIA_OUTBOUND_PORT_MAX	MQINQ
giden kapı alt sınırı			MQIA_OUTBOUND_PORT_MIN	MQINQ
parola	MQCSP	CSPPasswordPtr		MQCONN
	MQCSP	CSPPasswordOffset		MQCONN
	MQCSP	CSPPasswordLength		MQCONN
başarım olayı			MQIA_PERFORMANCE_OLAY	MQINQ
Platform			MQIA_PLATFORM	MQINQ
kuyruk muhasebesi			MQIA_ACCOUNTING_Q	MQINQ
kuyruk izleme			MQIA_MONITORING_Q	MQINQ
kuyruk istatistikleri			MQIA_STATISTICS_Q	MQINQ
alma zamanaşımı			MQIA_RECEIVE_TIMEOUT	MQINQ
alma zamanaşımı alt sınırı			MQIA_RECEIVE_TIMEOUT_MIN	MQINQ
alma zamanaşımı tipi			MQIA_RECEIVE_TIMEOUT_TYPE	MQINQ
uzak olay			MQIA_REMOTE_EVENT	MQINQ
Havuz adı			MQCA_REPOSITORY_NAME	MQINQ
havuz ad listesi			MQCA_REPOSITORY_NAMELIST	MQINQ
paylaşılan kuyruk kuyruk yöneticisi adı			MQIA_SHARED_Q_Q_MGR_ADI	MQINQ
ssl olayı			MQIA_SSL_OLAYI	MQINQ
ssl fips			MQIA_SSL_FIPS_REQUIRED	MQINQ
ssl anahtarı sıfırlama sayısı			MQIA_SSL_RESET_COUNT	MQINQ
start-stop olayı			MQIA_START_STOP_OLAY	MQINQ
istatistik aralığı			MQIA_STATISTICS_INTERVAL	MQINQ
syncpoint kullanılabilirliği			MQIA_SYNCPOINT	MQINQ

Çizelge 864. Öznitelikler, veri yapıları, alanlar, sorgular ve aramalar (devamı var)

Öznitelik	Veri yapısı	Alan	Sorgu	Ara
tcp kanalları			MQIA_TCP_KANALS	MQINQ
tcp canlı tutma			MQIA_TCP_KEEP_ALVE	MQINQ
tcp adı			MQCA_TCP_ADı	MQINQ
tcp yığın tipi			MQIA_TCP_STACK_TYPE	MQINQ
izleme rotası kaydı			MQIA_TRACE_ROUTE_RECORD	MQINQ
tetikleme aralığı			MQIA_TRIGGER_INTERVAL	MQINQ
kullanıcı kimliği	MQCSP	CSPUserIdPtr		MQCONN
	MQCSP	CSPUserIdGörelı Konumu		MQCONN
	MQCSP	CSPUserIduzunluđu		MQCONN

### ImqReferenceÜstbilgi çapraz başvurusu

ImqAuthenticationRecord C++ sınıfına ilişkin özniteliklere, veri yapılarına ve alanlara çapraz başvuru.

Çizelge 865. Özniteliklerin, veri yapılarının ve alanların eşlenmesi

Öznitelik	Veri yapısı	Alan
hedef ortam	MQRMH	DestEnvUzunluđu, DestEnvGörelı Konumu
Hedef Adı	MQRMH	DestNameUzunluđu, DestNameGörelı Konumu
Eşgörünüm Tanıtıcısı	MQRMH	ObjectInstanceTanıtıcısı
mantıksal uzunluk	MQRMH	DataLogicalLength
mantıksal görelı konum	MQRMH	DataLogicalGörelı Konumu
mantıksal görelı konum 2	MQRMH	DataLogicalOffset2
Başvuru tipi	MQRMH	ObjectType
Kaynak Ortamı	MQRMH	SrcEnvUzunluđu, SrcEnvGörelı Konumu
KAYNAK ADI	MQRMH	SrcNameUzunluđu, SrcNameGörelı Konumu

### ImqTrigger çapraz başvurusu

ImqAuthenticationRecord C++ sınıfına ilişkin özniteliklere, veri yapılarına ve alanlara çapraz başvuru.

Çizelge 866. Özniteliklerin, veri yapılarının ve alanların eşlenmesi

Öznitelik	Veri yapısı	Alan
Uygulama tanıtıcısı	MQTM	ApplId
Uygulama Tipi	MQTM	ApplType
ortam verileri	MQTM	EnvData
İşlem adı	MQTM	ProcessName

Çizelge 866. Özniteliklerin, veri yapılarının ve alanların eşlenmesi (devamı var)		
Öznitelik	Veri yapısı	Alan
Kuyruk adı	MQTM	QName
tetikleyici verileri	MQTM	TriggerData
Kullanıcı verileri	MQTM	UserData

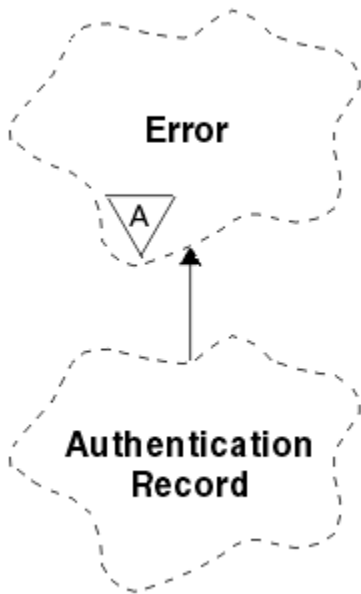
## ImqWorkÜstbilgi çapraz başvurusu

ImqAuthenticationRecord C++ sınıfına ilişkin özniteliklere, veri yapılarına ve alanlara çapraz başvuru.

Çizelge 867. Özniteliklerin, veri yapılarının ve alanların eşlenmesi		
Öznitelik	Veri yapısı	Alan
ileti simgesi	MQWIH	MessageToken
Hizmet Adı	MQWIH	ServiceName
hizmet adımı	MQWIH	ServiceStep

## ImqAuthenticationKayıt C++ sınıfı

Bu sınıf, özel TLS istemci bağlantıları için, ImqQueueManager: :connect yönteminin yürütülmesi sırasında kullanılmak üzere bir kimlik doğrulama bilgisi kaydını (MQAIR) sarmalanır.



Şekil 14. ImqAuthenticationKayıt sınıfı

Daha fazla ayrıntı için ImqQueueManager: :connect yönteminin açıklamalarına bakın. Bu sınıf z/OS platformunda kullanılamaz.

- [“Nesne öznitelikleri” sayfa 1776](#)
- [“Oluşturucular” sayfa 1776](#)
- [“Nesne yöntemleri \(genel\)” sayfa 1776](#)
- [“Nesne yöntemleri \(korunan\)” sayfa 1777](#)

## Nesne öznitelikleri

### Bağlantı adı

LDAP CRL sunucusuyla bağlantı adı. Bu, IP adresi ya da DNS adıdır, isteğe bağlı olarak kapı numarasına göre parantez içinde olur.

### bağlantı başvurusu

Bir (yerel) kuyruk yöneticisine gerekli bağlantıyı sağlayan bir `ImqQueueManager` nesnesine yönelik başvuru. Başlangıç değeri sıfır. Bu değeri, bir kuyruk yöneticisini (büyük olasılıkla uzak) adlandırılmış bir kuyruk için tanıtan kuyruk yöneticisi adıyla karıştırmayın.

### sonraki kimlik doğrulama kaydı

Bu sınıfın bir sonraki nesnesi, belirli bir sırada, bu nesne ile aynı **bağlantı başvurusu** sahip olmalıdır. Başlangıç değeri sıfır.

### parola

LDAP CRL sunucusuna bağlantı kimlik doğrulaması için sağlanan bir parola.

### önceki kimlik doğrulama kaydı

Bu sınıfın önceki nesnesi (belirli bir sırada), bu nesne ile aynı **bağlantı başvurusu** olmasına neden olur. Başlangıç değeri sıfır.

### tip

Kayıtta bulunan kimlik doğrulama bilgilerinin tipi.

### kullanıcı adı

LDAP CRL sunucusu için yetkilendirme için sağlanan bir kullanıcı kimliği.

## Oluşturucular

### `ImqAuthenticationRecord ()`;

Varsayılan oluşturucu.

## Nesne yöntemleri (genel)

### `void operator = (const ImqAuthenticationRecord & air )`;

Varolan yönetim ortamı verilerini değiştirerek, yönetim ortamı verilerini *hava'* dan kopyalar.

### `const ImqString & connectionName () const`;

**Bağlantı adı**değerini döndürür.

### `void setConnectionAd (const ImqString & ad )`;

**Bağlantı adı'** yı ayarlar.

### `void setConnectionAd (const char * ad = 0)`;

**Bağlantı adı'** yı ayarlar.

### `ImqQueueManager * connectionReference () const`;

**Bağlantı başvurusu**değerini döndürür.

### `void setConnectionReferansı ( ImqQueueManager & yöneticisi )`;

**Bağlantı başvurusunu**ayarlar.

### `void setConnectionReferansı ( ImqQueueManager * yönetici = 0)`;

**Bağlantı başvurusunu**ayarlar.

### `void copyOut (MQAIR * pAir )`;

Eşgörünüm verilerini, var olan eşgörünüm verilerini değiştirerek *pAir*olarak kopyalar. Bu, bağımlı depolamanın ayrılmasını içerebilir.

### `void clear (MQAIR * pAir )`;

Clears the structure and releases dependent storage referenced by *pAir*.

### `ImqAuthenticationKayıt * nextAuthenticationRecord () const`;

**Sonraki kimlik doğrulama kaydı**değerini döndürür.

### `const ImqString & password () const`;

**Parola**değerini döndürür.

**void setPassword (const ImqString & password );**

Parolaseçeneğini belirler.

**void setPassword (const char \* parola = 0);**

Parolaseçeneğini belirler.

**ImqAuthenticationKayıt \* previousAuthenticationRecord () const;**

Önceki kimlik doğrulama kaydını döndürür.

**MQTHOE tipi () const;**

tip değerini döndürür.

**void setType (const MQHOBE tip );**

tip' i ayarlar.

**const ImqString & userName () const;**

Kullanıcı adı değerini döndürür.

**void setUserAdı (const ImqString & ad );**

Kullanıcı adı' yı ayarlar.

**void setUserAd (const char \* ad = 0);**

Kullanıcı adı' yı ayarlar.

### Nesne yöntemleri (korunan)

**void setNextAuthenticationRecord ( ImqAuthenticationRecord \* pAir = 0);**

Sonraki kimlik doğrulama kaydını ayarlar.

**Dikkat:** Bu işlevi yalnızca, kimlik doğrulama kaydı listesini bozmayacağından eminseniz kullanın.

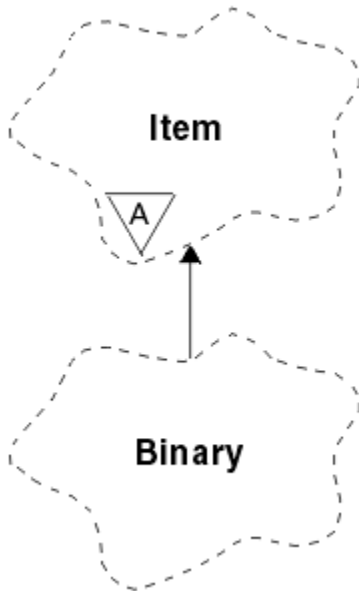
**void setPreviousAuthenticationRecord ( ImqAuthenticationRecord \* pAir = 0);**

Önceki kimlik doğrulama kaydını ayarlar.

**Dikkat:** Bu işlevi yalnızca, kimlik doğrulama kaydı listesini bozmayacağından eminseniz kullanın.

## ImqBinary C++ sınıfı

Bu sınıf, ImqMessage **muhasebe simgesi**, **ilinti tanıtıcısı** ve **ileti tanıtıcısı** değerleri için kullanılacak bir ikili bayt dizisini sarsalıyor. Kolay atama, kopyalama ve karşılaştırma olanağı sağlar.



Şekil 15. ImqBinary sınıfı

- “Nesne öznitelikleri” sayfa 1778
- “Oluşturucular” sayfa 1778

- [“Aşırı yüklenmiş ImqItem yöntemleri” sayfa 1778](#)
- [“Nesne yöntemleri \(genel\)” sayfa 1778](#)
- [“Nesne yöntemleri \(korunan\)” sayfa 1779](#)
- [“Neden kodları” sayfa 1779](#)

## Nesne öznitelikleri

### veri

İkili veri baytları dizisi. Başlangıç değeri boş değerli.

### veri uzunluğu

Bayt sayısı. Başlangıç değeri sıfır.

### veri göstergesi

**Veri'** in ilk baytının adresi. Başlangıç değeri sıfır.

## Oluşturucular

### ImqBinary();

Varsayılan oluşturucu.

### ImqBinary( const ImqBinary & binary );

Kopya oluşturucusu.

### ImqBinary( const void \* data, const size\_t length );

Copies *uzunluk* bytes from *veri*.

## Aşırı yüklenmiş ImqItem yöntemleri

### sanal ImqBoolean copyOut ( ImqMessage & msg );

Varolan içeriğin yerine, **veri** ' u ileti arabelleğiyle kopyalar. *msg biçimi* ' i MQFMT\_NONE olarak ayarlar.

Ek ayrıntılar için ImqItem sınıfı yöntem açıklamasına bakın.

### sanal ImqBoolean pasteIn ( ImqMessage & msg );

Var olan **verilerideğiştirerek**, kalan verileri ileti arabelleğinden aktararak **veri** ' i ayarlar.

Başarılı olmak için, ImqMessage **format** , MQFMT\_NONE olmalıdır.

Ek ayrıntılar için ImqItem sınıfı yöntem açıklamasına bakın.

## Nesne yöntemleri (genel)

### void operator = ( const ImqBinary & binary );

Bayt 'ları *ikili*' den kopyalar.

### ImqBoolean operator == ( const ImqBinary & binary );

Bu nesneyi *binary* ile karşılaştırır. Eşit değilse FALSE değerini ve tersi durumda TRUE değerini döndürür. Nesnelere aynı **veri uzunluğuna** ve byte eşleşmesine sahip olduğunda eşitlenir.

### ImqBoolean copyOut ( void \* buffer, const size\_t length, const char pad = 0 );

Copies up to *uzunluk* bytes from the **veri göstergesi** to *arabellek*. **Veri uzunluğu** yetersizse, *arabellekteki* kalan alan *doldurma* baytlarıyla doldurulur. *length* değeri de sıfırsa, *buffer* sıfır olabilir. *uzunluk* negatif olmamalıdır. Başarılı olursa TRUE değerini döndürür.

### size\_t dataLength () const ;

**Veri uzunluğudeğerini** döndürür.

### ImqBoolean setDataLength ( const size\_t length );

**Veri uzunluğunubelirler**. Bu yöntemin bir sonucu olarak **veri uzunluğu** değişirse, nesnedeki veriler kullanıma hazırlanmamış olur. Başarılı olursa TRUE değerini döndürür.

### void \* dataPointer () const ;

**Veri işaretçisini** döndürür.

### **ImqBoolean isNull () const ;**

**Veri uzunluęu** sıfır ya da tüm **verilerde** byte 'lar sıfırda TRUE deęerini döndürür. Ters durumda FALSE deęerini döndürür.

### **ImqBoolean set ( const void \* buffer, const size\_t length );**

*length* byte 'ı *buffer*' tan kopyalar. Başarılı olursa TRUE deęerini döndürür.

## **Nesne yöntemleri (korunan)**

### **void clear ();**

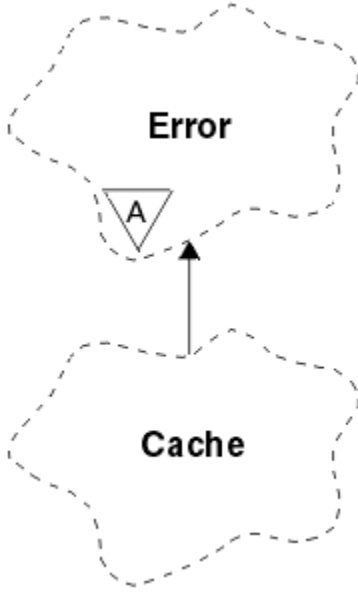
**Veri uzunluęu** deęerini sıfıra indirir.

## **Neden kodları**

- MQRC\_NO\_BUFFER
- MQRC\_STORAGE\_NOT\_AVAM
- MQRC\_INCONSTENT\_BIÇIMI

## **ImqCache C++ sınıfı**

Belleęindeki verileri tutmak ya da verileri bellek için bu sınıfı kullanın.



Şekil 16. ImqCache sınıfı

Belleęindeki verileri tutmak ya da verileri bellek için bu sınıfı kullanın. Sabit büyüklükte bir bellek arabelleęi gösterebilirsiniz ya da sistem, otomatik olarak esnek bir bellek miktarı sağlayabilir. Bu sınıf, "ImqCache çapraz başvuru" sayfa 1761 içinde listelenen MQI çağrılarını ile ilgilidir.

- "[Nesne öznitelikleri](#)" sayfa 1779
- "[Oluşturucular](#)" sayfa 1780
- "[Nesne yöntemleri \(genel\)](#)" sayfa 1780
- "[Neden kodları](#)" sayfa 1781

## **Nesne öznitelikleri**

### **otomatik arabellek**

Arabellek belleęinin sistem tarafından otomatik olarak mı (TRUE), yoksa kullanıcı tarafından mı (FALSE) sağlanıp sağlanmadığını belirtir. Başlangıçta TRUE olarak ayarlanır.

Bu öznitelik doğrudan belirlenmez. Bu, **useEmptyBuffer** ya da **useFullBuffer** yöntemini dolaylı olarak ayarlıdır.

Kullanıcı saklama alanı sağlanırsa, bu öznitelik FALSE olur, arabellek belleği büyümmez ve arabellek taşması hataları oluşabilir. Arabelleğin adresi ve uzunluğu sabit kalır.

Kullanıcı depolama alanı sağlanmıyorsa, bu öznitelik TRUE olur ve arabellek belleği, isteğe bağlı miktarda ileti verisi barındırmak için artımsal olarak büyüyebilir. Ancak, arabellek büyüdüğünde, arabelleğin adresi değişebilir; bu nedenle, **arabellek göstergesi** ve **veri göstergesi** kullanırken dikkatli olun.

#### **arabellek uzunluğu**

Arabelleğindeki bellek miktarı. Başlangıç değeri sıfır.

#### **arabellek göstergesi**

Arabellek belleğinin adresi. Başlangıç değeri boş değerli.

#### **veri uzunluğu**

**Veri işaretçisi**'nin başarılı olduğu bayt sayısı. Bu, **ileti uzunluğu** değerine eşit ya da bu değere eşit olmalıdır. Başlangıç değeri sıfır.

#### **veri görelî konumu**

**Veri işaretçisi**'nin önündeki bayt sayısı. Bu, **ileti uzunluğu** değerine eşit ya da bu değere eşit olmalıdır. Başlangıç değeri sıfır.

#### **veri göstergesi**

Bir sonraki yazıya yazılacak ya da yazılacak olan arabelleğin bir kısmının adresi. Başlangıç değeri boş değerli.

#### **ileti uzunluğu**

Arabellekteki önemli verilerin bayt sayısı. Başlangıç değeri sıfır.

### **Oluşturucular**

#### **ImqCache();**

Varsayılan oluşturucu.

#### **ImqCache( const ImqCache & cache );**

Kopya oluşturucusu.

### **Nesne yöntemleri (genel)**

#### **void operator = ( const ImqCache & cache );**

*önbellek* nesnesinden nesneye **ileti uzunluğu** byte 'a kadar veri kopyalar. **Otomatik arabellek** değeri FALSE ise, kopyalanan verileri barındırmak için **arabellek uzunluğu** 'nın yeterli olması gerekir.

#### **ImqBoolean automaticBuffer () const ;**

**Otomatik arabellek** değerini döndürür.

#### **size\_t bufferSize () const ;**

**arabellek uzunluğu** değerini döndürür.

#### **char \* bufferPointer () const ;**

**Arabellek işaretçisi** değerini döndürür.

#### **void clearMessage ();**

**İleti uzunluğu** ve **veri görelî konumu** değerini sıfır olarak ayarlar.

#### **size\_t dataLength () const ;**

**Veri uzunluğu** değerini döndürür.

#### **size\_t dataOffset () const ;**

**Veri görelî konumu** değerini döndürür.

#### **ImqBoolean setDataOffset ( const size\_t görelî konum );**

**Veri görelî konumu**'ne ayarlar. The **ileti uzunluğu** is increased if necessary to ensure that it is no less than the **veri görelî konumu**. Bu yöntem başarılı olursa TRUE değerini döndürür.



**char \* dataPointer () const ;**

Veri işaretçisi' nin bir kopyasını döndürür.

**size\_t messageLength () const ;**

İleti uzunluğudeğerini döndürür.

**ImqBoolean setMessageLength ( const size\_t length );**

İleti uzunluğunu belirler. Increases the **arabellek uzunluğu** if necessary to ensure that the **ileti uzunluğu** is no greater than the **arabellek uzunluğu**. Reduces the **veri görelî konumu** if necessary to ensure that it is no greater than the **ileti uzunluğu**. Başarılı olursa TRUE değerini döndürür.

**ImqBoolean moreBytes ( const size\_t byte-requirems );**

Assures that *bayt-gerekli* more bytes are available (for writing) between the **veri göstergesi** and the end of the buffer. Başarılı olursa TRUE değerini döndürür.

**Automatic buffer** (Otomatik arabellek) değeri TRUE (Doğru) ise, gerektiği gibi daha fazla bellek elde edilir; tersi durumda, **buffer length** (arabellek uzunluğu) yeterli olmalıdır.

**ImqBoolean read (okuma) ( const size\_t length, char \* & external-buffer );**

Copies *uzunluk* bytes, from the buffer starting at the **veri göstergesi** position, into the *dış arabellek*. Veriler kopyalandıktan sonra, **veri görelî konumu** *uzunluk*uzunluğuna göre artırılır. Bu yöntem başarılı olursa TRUE değerini döndürür.

**ImqBoolean resizeBuffer ( const size\_t length );**

**Automatic buffer** (otomatik arabellek) değeri TRUE (Doğru) olduğunda **arabellek uzunluğunu** gösterir. Bu, arabellek belleğindeki yerini yeniden ayırarak elde edilir. Varolan arabellekten **ileti uzunluğu** byte 'a kadar veri, yeni bir veri kaynağı arabelleğinden kopyalanır. Kopya sayısı üst sınırı *uzunluk* byte 'tır. **Arabellek göstergesi** değiştirilir. **ileti uzunluğu** ve **veri görelî konumu** , yeni arabelleğin sınırları içinde mümkün olan en yakın şekilde korunur. Başarılı olursa TRUE değerini, **otomatik arabellek** değeri FALSE ise FALSE değerini döndürür.

**Not:** Sistem kaynaklarıyla ilgili bir sorun varsa, bu yöntem MQRG\_STORAGE\_NOT\_AVAM ile kullanılabilir durumda olabilir.

**ImqBoolean useEmptyArabelleği ( const char \* external-buffer, const size\_t length );**

Boş bir kullanıcı arabelleğinden birini tanımlar; **arabellek göstergesi** için *dış-arabelleği*, **arabellek uzunluğu** değerini *uzunluk*değerine ve **ileti uzunluğu** değerini sifıra doğru ayarlar. Bir **clearMessage** gerçekleştirir. Arabellek verilerle tam olarak yazılmışsa, bunun yerine **useFullBuffer** yöntemini kullanın. Arabellek veriler ile kısmen yeniden başlandıysa, doğru miktarı belirtmek için **setMessageLength** yöntemini kullanın. Bu yöntem başarılı olursa TRUE değerini döndürür.

Bu yöntem, önceden açıklandığı gibi, sabit bellek miktarını tanımlamak için kullanılabilir ( *dış-arabellek* boş değerli değildir ve *uzunluk* sıfır değerli), bu durumda **otomatik arabellek** değeri FALSE olarak ayarlanır ya da sistem tarafından yönetilen esnek belleğe geri dönmek için kullanılabilir ( *dış-arabellek* boş değerli ve *uzunluk* sıfır), bu durumda **otomatik arabellek** TRUE olarak ayarlanır.

**ImqBoolean useFullArabellek ( const char \* externalBuffer, const size\_t length );**

**useEmptyArabellek**için olduğu gibi, **ileti uzunluğu** değeri *uzunluk*olarak ayarlıdır. Başarılı olursa TRUE değerini döndürür.

**ImqBoolean write ( const size\_t length, const char \* external-buffer );**

Copies *uzunluk* bytes, from the *dış arabellek*, into the buffer starting at the **veri göstergesi** position. After the data has been copied, the **veri görelî konumu** is increased by *uzunluk*, and the **ileti uzunluğu** is increased if necessary to ensure that it is no less than the new **veri görelî konumu** value. Bu yöntem başarılı olursa TRUE değerini döndürür.

**Otomatik arabellek** değeri TRUE ise, yeterli miktarda bellek garanti edilir; tersi durumda, en son **veri görelî konumu** , **arabellek uzunluğunu**aşmamalıdır.

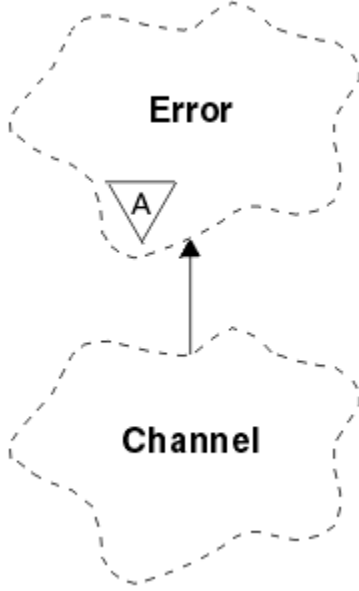
## Neden kodları

- MQRG\_BUFFER\_NOT\_AUTOMAKRED
- MQRG\_DATA\_TRUNCATED
- MQRG\_INSUFFICIENT\_BUFFER

- MQRC\_INSUFFICIENT\_DATA
- MQRC\_NULL\_POINTER
- MQRC\_STORAGE\_NOT\_AVAM
- MQRC\_ZERO\_LENGTH

## ImqChannel C++ sınıfı

Bu sınıf, özel istemci bağlantıları için, Yönetici: :connect yöntemi yürütülürken kullanılacak bir kanal tanımlamasını (MQCD) kapsüller.



Şekil 17. ImqChannel sınıfı

Daha fazla ayrıntı için, Yönetici: :connect yöntemi ve [Örnek program HELLO WORLD \(imqwrl.cpp\)](#) açıklamasına bakın.

Listelenen tüm yöntemlerin tümü, tüm altyapılar için geçerli değildir. Ek bilgi için [DEFINE CHANNEL](#) ve [ALTER CHANNEL](#) komutlarının açıklamalarına bakın.

ImqChannel sınıfı z/OSüzerinde desteklenmez.

- “Nesne öznitelikleri” sayfa 1782
- “Oluşturucular” sayfa 1783
- “Nesne yöntemleri (genel)” sayfa 1784
- “Neden kodları” sayfa 1787

## Nesne öznitelikleri

### toplu kalp ritmi

Bir uzak kanalın etkin olduğu denetimler arasındaki milisaniye sayısı. Başlangıç değeri 0 olur.

### Kanal Adı

Kanalın adı. Başlangıç değeri boş değerli.

### Bağlantı adı

Bağlantının adı. Örneğin, bir anasistem bilgisayarının IP adresi. Başlangıç değeri boş değerli.

### üstbilgi sıkıştırması

Kanal tarafından desteklenen üstbilgi veri sıkıştırma tekniklerinin listesi. İlk değerler, tümü MQCOMPRESS\_NOT\_AVAM değerine ayarlanır.

**kalp ritmi aralığı**

Bir bağlantının halen çalışmakta olduğu denetimlerin arasındaki saniye sayısı. Başlangıç değeri 300 'dür.

**canlı tutma aralığı**

Kanala ilişkin canlı tutma zamanlamasını belirten iletişim yığınının geçirilen saniye sayısı. İlk değer MÇKAI\_AUTO ' dur.

**Yerel adres**

Kanala ilişkin yerel iletişim adresi.

**ileti uzunluğu üst sınırı**

Tek bir iletişimde kanal tarafından desteklenen ileti uzunluğu üst sınırı. Başlangıç değeri 4 194 304 'tür.

**ileti sıkıştırma**

Kanal tarafından desteklenen ileti veri sıkıştırma tekniklerinin listesi. İlk değerler, tümü MÇCOMPRESS\_NOT\_AVAM değerine ayarlanır.

**kip adı**

Kipin adı. Başlangıç değeri boş değerli.

**parola**

Bağlantı kimlik doğrulaması için sağlanan parola. Başlangıç değeri boş değerli.

**çıkış sayısı alma**

Giriş çıkış sayısı. Başlangıç değeri sıfır. Bu öznitelik salt okunurdur.

**çıkış adlarını al**

Alma çıkışlarının adları.

**kullanıcı verilerini al**

Alma çıkışlarıyla ilişkili veriler.

**güvenlik çıkış adı**

Bağlantının sunucu tarafında çağrılacak güvenlik çıkışa ilişkin ad. Başlangıç değeri boş değerli.

**güvenlik kullanıcı verileri**

Güvenlik çıkışa geçirilecek veriler. Başlangıç değeri boş değerli.

**çıkış sayısı gönder**

Çıkış çıkış sayısı. Başlangıç değeri sıfır. Bu öznitelik salt okunurdur.

**çıkış adlarını gönder**

Çıkış gönderilerinin adları.

**kullanıcı verilerini gönder**

Gönderme çıkışlarıyla ilişkili veriler.

**SSL CipherSpec**

TLS ile kullanmak içinCipherSpec .

**SSL istemcisi kimlik denetimi tipi**

TLS ile kullanılacak istemci kimlik denetimi tipi.

**SSL eş adı**

TLS ile kullanım için eşdüzey ad.

**hareket programı adı**

Hareket programının adı. Başlangıç değeri boş değerli.

**taşıma tipi**

Bağlantının iletim tipi. İlk değer MÇXPT\_LU62' dir.

**kullanıcı kimliği**

Yetki için sağlanan bir kullanıcı kimliği. Başlangıç değeri boş değerli.

**Oluşturucular****ImqChannel( ) ;**

Varsayılan oluşturucu.

**ImqChannel( const ImqChannel & kanal );**

Kopya oluşturucusu.

## Nesne yöntemleri (genel)

**void operator = (const ImqChannel & channel );**

Var olan yönetim ortamı verilerini değiştirerek, yönetim ortamı verilerini *kanaldankopyalar*.

**MQlong batchHeartBeat () const;**

Toplu kalp ritmi değerini döndürür.

**ImqBoolean setBatchHeartBeat(const MQlong heartbeat = 0L );**

Toplu kalp ritmi ayarını ayarlar. Bu yöntem başarılı olursa TRUE değerini döndürür.

**ImqString channelName() const;**

Kanal adı değerini döndürür.

**ImqBoolean setChannelAd (const char \* ad = 0);**

Kanal adı' yı ayarlar. Bu yöntem başarılı olursa TRUE değerini döndürür.

**ImqString connectionName() const;**

Bağlantı adı değerini döndürür.

**ImqBoolean setConnectionAd (const char \* ad = 0);**

Bağlantı adı' yı ayarlar. Bu yöntem başarılı olursa TRUE değerini döndürür.

**size\_t headerCompressionSayı () const;**

Desteklenen üstbilgi veri sıkıştırma tekniklerini döndürür.

**ImqBoolean headerCompression(const size\_t count, MQlong compress []) const;**

sıkıştırıcıdaki desteklenen üstbilgi veri sıkıştırma tekniklerinin kopyalarını döndürür. Bu yöntem başarılı olursa TRUE değerini döndürür.

**ImqBoolean setHeaderCompression (const size\_t count, const MQlong compress []);**

Desteklenen üstbilgi veri sıkıştırma tekniklerini sıkıştırılarak ayarlar.

Desteklenen üstbilgi veri sıkıştırma tekniklerini sayı olarak belirler.

Bu yöntem başarılı olursa TRUE değerini döndürür.

**MQlong heartBeatInterval () const;**

heart-beat interval değerini döndürür.

**ImqBoolean setHeartBeatInterval(const MQlong interval = 300L );**

Kalp ritmi aralığı belirler. Bu yöntem başarılı olursa TRUE değerini döndürür.

**MQlong keepAliveInterval () const;**

Canlı tutma aralığı değerini döndürür.

**ImqBoolean setKeepAliveInterval(const MQlong interval = MQKAI\_AUTO);**

Canlı tutma aralığı ayarını ayarlar. Bu yöntem başarılı olursa TRUE değerini döndürür.

**ImqString localAddress() const;**

Yerel adres değerini döndürür.

**ImqBoolean setLocalAddress (const char \* address = 0);**

Yerel adres' i ayarlar. Bu yöntem başarılı olursa TRUE değerini döndürür.

**MQlong maximumMessageUzunluk () const;**

İleti uzunluğu üst sınırı değerini döndürür.

**ImqBoolean setMaximumMessageLength(const MQROLD length = 4194304L );**

İleti uzunluğu üst sınırı değerini ayarlar. Bu yöntem başarılı olursa TRUE değerini döndürür.

**size\_t messageCompressionSayı () const;**

Desteklenen ileti verisi sıkıştırma tekniklerini döndürür.

**ImqBoolean messageCompression(const size\_t count, MQlong compress []) const;**

sıkıştırıcıdaki desteklenen ileti veri sıkıştırma tekniklerinin kopyalarını döndürür. Bu yöntem başarılı olursa TRUE değerini döndürür.

**ImqBoolean setMessageCompression (const size\_t count, const Mqlong compress []);**

Sıkıştırılacak desteklenen ileti verisi sıkıştırma tekniklerini ayarlar.

Desteklenen ileti veri sıkıştırma tekniklerini sayılacak şekilde ayarlar.

Bu yöntem başarılı olursa TRUE değerini döndürür.

**ImqString modeName() const;**

**mode name**(Kip adı) değerini döndürür.

**ImqBoolean setModeAdı (const char \* ad = 0);**

**mode name**(Kip adı) ögesini ayarlar. Bu yöntem başarılı olursa TRUE değerini döndürür.

**ImqString parolası () const;**

**Paroladeğerini** döndürür.

**ImqBoolean setPassword(const char \* password = 0);**

**Parolaseçeneğini** belirler. Bu yöntem başarılı olursa TRUE değerini döndürür.

**size\_t receiveExitSayı () const;**

**Alma çıkış sayısı**değerini döndürür.

**ImqString receiveExitAd ();**

Varsa, **çıkış çıkış adlarını** alışlevinin ilk ögesini döndürür. **Çıkış sayısını al** sıfırda, boş bir dizgi döndürür.

**ImqBoolean receiveExitAdlar (const size\_t count, ImqString \* names []);**

**adiçindeki çıkış adlarının al** kopyalarının kopyalarını döndürür. Sets any **adlar** in excess of **çıkış sayısı alma** to null strings. Bu yöntem başarılı olursa TRUE değerini döndürür.

**ImqBoolean setReceiveExitName(const char \* ad = 0);**

**Alma çıkış adlarını** tek **ad**olarak ayarlar. **ad** boş değerli ya da boş olabilir. **çıkış sayısı alma** ' 1 ya da sıfır olarak ayarlar. **Kullanıcı verilerini al** seçeneğini temizler. Bu yöntem başarılı olursa TRUE değerini döndürür.

**ImqBoolean setReceiveExitNames(const size\_t count, const char \* names []);**

**çıkış adlarını al** ' ı **adlar**olarak ayarlar. Tek tek **adlar** değerleri boş bırakılmamalı ya da boş bırakılmamalı. **Alma çıkış sayısını count**(sayı) olarak ayarlar. **Kullanıcı verilerini al** seçeneğini temizler. Bu yöntem başarılı olursa TRUE değerini döndürür.

**ImqBoolean setReceiveExitNames(const size\_t count, const ImqString \* names []);**

**çıkış adlarını al** ' ı **adlar**olarak ayarlar. Tek tek **adlar** değerleri boş bırakılmamalı ya da boş bırakılmamalı. **Alma çıkış sayısını count**(sayı) olarak ayarlar. **Kullanıcı verilerini al** seçeneğini temizler. Bu yöntem başarılı olursa TRUE değerini döndürür.

**ImqString receiveUserData ();**

Varsa, **kullanıcı verilerini al** öğelerinin ilk ögesini döndürür. **Çıkış sayısını al** sıfırda, boş bir dizgi döndürür.

**ImqBoolean receiveUserData (const size\_t count, ImqString \* data []);**

**Veri**' de **kullanıcı verilerini al** öğelerinin kopyalarını döndürür. Sets any **veri** in excess of **çıkış sayısı alma** to null strings. Bu yöntem başarılı olursa TRUE değerini döndürür.

**ImqBoolean setReceiveUserData(const char \* data = 0);**

**kullanıcı verilerini al** ögesini tek öğeye ( **veri**) ayarlar. **veri** boş değerli değilse, **çıkış sayısını al** en az 1 olmalıdır. Bu yöntem başarılı olursa TRUE değerini döndürür.

**ImqBoolean setReceiveUserData(const size\_t count, const char \* data []);**

**Kullanıcı verilerini al** seçeneğini **veri**olarak ayarlar. **count** , **alma çıkış sayımından**büyük olmamalıdır. Bu yöntem başarılı olursa TRUE değerini döndürür.

**ImqBoolean setReceiveUserData(const size\_t count, const ImqString \* data []);**

**Kullanıcı verilerini al** seçeneğini **veri**olarak ayarlar. **count** , **alma çıkış sayımından**büyük olmamalıdır. Bu yöntem başarılı olursa TRUE değerini döndürür.

**ImqString securityExitAd () const;**

**Güvenlik çıkışı adı**değerini döndürür.

**ImqBoolean setSecurityExitName(const char \* ad = 0);**

Güvenlik çıkışı adını ayarlar. Bu yöntem başarılı olursa TRUE değerini döndürür.

**ImqString securityUserData () const;**

Güvenlik kullanıcı verileri değerini döndürür.

**ImqBoolean setSecurityUserData(const char \* data = 0);**

Güvenlik kullanıcı verilerini ayarlar. Bu yöntem başarılı olursa TRUE değerini döndürür.

**size\_t sendExitSayı () const;**

çıkış sayısını gönder değerini döndürür.

**ImqString sendExitAd ();**

Varsa, çıkış adlarını gönder' in ilk ögesini döndürür. Çıkış sayısı gönder sıfırda, boş bir dizgi döndürür.

**ImqBoolean sendExitAdlar (const size\_t count, ImqString \* names []);**

adi içindeki çıkış adlarının gönderilmesi işlevinin kopyalarını döndürür. Sets any *adlar* in excess of çıkış sayısı gönder to null strings. Bu yöntem başarılı olursa TRUE değerini döndürür.

**ImqBoolean setSendExitName(const char \* ad = 0);**

Çıkış adlarını gönder ögesini tek bir ad değerine ayarlar. *ad* boş değerli ya da boş olabilir. Çıkış sayısını gönder ' i 1 ya da sıfır olarak ayarlar. Kullanıcı verilerini gönder seçeneğini temizler. Bu yöntem başarılı olursa TRUE değerini döndürür.

**ImqBoolean setSendExitNames(const size\_t count, const char \* names []);**

çıkış adlarını gönder ' ı *adlar* olarak ayarlar. Tek tek *adlar* değerleri boş bırakılmamalı ya da boş bırakılmamalı. Çıkış sayısını gönder ' i *count* ' a ayarlar. Kullanıcı verilerini gönder seçeneğini temizler. Bu yöntem başarılı olursa TRUE değerini döndürür.

**ImqBoolean setSendExitNames(const size\_t count, const ImqString \* names []);**

çıkış adlarını gönder ' ı *adlar* olarak ayarlar. Tek tek *adlar* değerleri boş bırakılmamalı ya da boş bırakılmamalı. Çıkış sayısını gönder ' i *count* ' a ayarlar. Kullanıcı verilerini gönder seçeneğini temizler. Bu yöntem başarılı olursa TRUE değerini döndürür.

**ImqString sendUserData ();**

Varsa, kullanıcı verilerini gönder öğelerinin ilk ögesini döndürür. Çıkış sayısı gönder sıfırda, boş bir dizgi döndürür.

**ImqBoolean sendUserData (const size\_t count, ImqString \* data []);**

*Veri* ' de kullanıcı verilerinin gönderilmesi öğelerinin kopyalarını döndürür. Herhangi bir *veriyi* , çıkış sayısını gönder fazlasından boş değer dizgilerine ayarlar. Bu yöntem başarılı olursa TRUE değerini döndürür.

**ImqBoolean setSendUserData(const char \* data = 0);**

kullanıcı verilerini gönder ögesini tek öğeye ( *veri* ) ayarlar. *veri* boş değerli değilse, çıkış sayısı gönder en az 1 olmalıdır. Bu yöntem başarılı olursa TRUE değerini döndürür.

**ImqBoolean setSendUserData(const size\_t count, const char \* data []);**

Kullanıcı verilerini gönder ' i *veri* olarak ayarlar. *sayı* , çıkış sayısı gönder değerinden büyük olmamalıdır. Bu yöntem başarılı olursa TRUE değerini döndürür.

**ImqBoolean setSendUserData(const size\_t count, const ImqString \* data []);**

Kullanıcı verilerini gönder ' i *veri* olarak ayarlar. *sayı* , çıkış sayısı gönder değerinden büyük olmamalıdır. Bu yöntem başarılı olursa TRUE değerini döndürür.

**ImqString sslCipherSpecification () const;**

TLS şifre belirtimini döndürür.

**ImqBoolean setSslCipherSpecification(const char \* ad = 0);**

TLS şifre belirtimini ayarlar. Bu yöntem başarılı olursa TRUE değerini döndürür.

**MQHOT sslClientKimlik Doğrulaması () const;**

TLS istemcisi kimlik doğrulama tipini döndürür.

**ImqBoolean setSslClientAuthentication(const MQlong auth = MQSCA\_REQUIREMOND);**

TLS istemcisi kimlik doğrulama tipini ayarlar. Bu yöntem başarılı olursa TRUE değerini döndürür.

**ImqString sslPeerAd () const;**

TLS eş adını döndürür.

**ImqBoolean setSslPeerName(const char \* ad = 0);**

TLS eş adını ayarlar. Bu yöntem başarılı olursa TRUE değerini döndürür.

**ImqString transactionProgramAd () const;**

hareket programı adı değerini döndürür.

**ImqBoolean setTransactionProgramName(const char \* ad = 0);**

Hareket programı adı' na ayarlar. Bu yöntem başarılı olursa TRUE değerini döndürür.

**MQHOT transportType() const;**

İletim tipi değerini döndürür.

**ImqBoolean setTransportType (const MQHOMET tip = MQXPT\_LU62 );**

İletim tipini belirler. Bu yöntem başarılı olursa TRUE değerini döndürür.

**ImqString userId() const;**

Kullanıcı kimliği değerini döndürür.

**ImqBoolean setUserId (const char \* id = 0);**

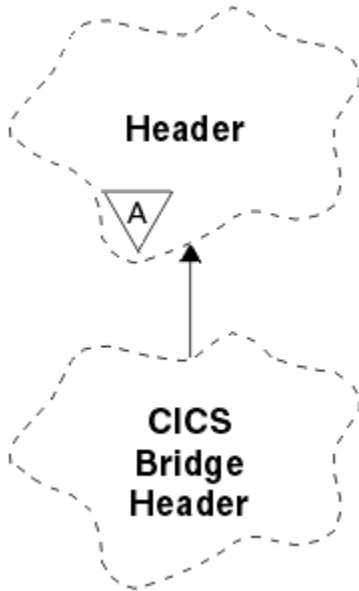
Kullanıcı kimliği' yi ayarlar. Bu yöntem başarılı olursa TRUE değerini döndürür.

### Neden kodları

- MQRC\_DATA\_LENGTH\_ERROR
- MQRC\_ITETİĞ\_COUNT\_ERROR
- MQRC\_NULL\_POINTER
- MQRC\_SOURCE\_BUFFER\_ERROR

## ImqCICSBridgeHeader C++ sınıfı

Bu sınıf, MQCIH veri yapısının belirli özelliklerini sarsalıyor.



Şekil 18. ImqCICSBridgeHeader sınıfı

Objects of this class are used by applications that send messages to the CICS bridge through IBM MQ for z/OS.

- [“Nesne öznitelikleri” sayfa 1788](#)
- [“Oluşturucular” sayfa 1790](#)
- [“Aşırı yüklenmiş ImqItem yöntemleri” sayfa 1790](#)
- [“Nesne yöntemleri \(genel\)” sayfa 1790](#)

- “Nesne verileri (korunan)” sayfa 1792
- “Neden kodları” sayfa 1793
- “Dönüş kodları” sayfa 1793

## Nesne öznitelikleri

### ADS tanımlayıcı

Gönderme/alma ADS tanımlayıcısı. Bu, MQCADSD\_NONE kullanılarak ayarlanır. İlk değer MQCADSD\_NONE olur. Aşağıdaki ek değerler kullanılabilir:

- MQCADSD\_NONE
- MQCADSD\_SEND
- MQCADSD\_RECV
- MQCADSD\_MSGSAYI

### uyarı tanıtıcısı

AID anahtarı. Alan, MQ\_ATTEN\_ID\_LENGTH uzunluğunda olmalıdır.

### kimlik doğrulayıcı

RACF parolası ya da passticket. Başlangıçtaki değer boşluk, MQ\_AUTHENTICATOR\_LENGTH uzunluğunda boşluk içeriyor.

### köprü olağandışı bitiş kodu

Köprü olağandışı bitiş kodu, MQ\_ABEND\_CODE\_LENGTH uzunluğunda. Başlangıç değeri dört boş karakterdir. Bu alana döndürülen değer, dönüş koduna bağlıdır. Daha ayrıntılı bilgi için bkz. [Çizelge 868 sayfa 1793](#).

### köprü iptal kodu

Köprü olağandışı bitiş hareket kodu. Alan ayrılmıştır, boşluk içermelidir ve MQ\_CANCEL\_CODE\_LENGTH uzunluğuna sahip olmalıdır.

### köprü tamamlama kodu

Completion code, which can contain either the IBM MQ completion code or the CICS EIBRESP value. Alan, MQCC\_OK ' un başlangıç değerine sahiptir. Bu alana döndürülen değer, dönüş koduna bağlıdır. Daha ayrıntılı bilgi için bkz. [Çizelge 868 sayfa 1793](#).

### köprü hata görelili konumu

Köprü hata görelili konumu. Başlangıç değeri sıfır. Bu öznitelik salt okunurdur.

### köprü neden kodu

Neden kodu. Bu alan, IBM MQ nedenini ya da CICS EIBRESP2 değerini içerebilir. Alan, MQRC\_NONE ilk değerine sahiptir. Bu alana döndürülen değer, dönüş koduna bağlıdır. Daha ayrıntılı bilgi için bkz. [Çizelge 868 sayfa 1793](#).

### köprü dönüş kodu

CICS bridge' den dönüş kodu. İlk değer MQCRC\_OK ' dir.

### etkileşimli görev

Görevin etkileşimli olup olmayacağını. İlk değer MQCCT\_NO olur. Aşağıdaki ek değerler kullanılabilir:

- MQCCT\_YES
- MQCCT\_NO

### imleç konumu

İmleç konumu. Başlangıç değeri sıfır.

### tesis zaman tutma

CICS bridge olanağı yayın süresi.

### gibi bir tesis

Uçbirim öykünülen özneliği. Alan, MQ\_FACILITY\_LIKE\_LENGTH uzunluğunda olmalıdır.

### tesis simgesi

BVT simge değeri. Alan, MQ\_FACILITY\_LENGTH uzunluğunda olmalıdır. İlk değer MQCFAC\_NONE olur.



## **işlev**

IBM MQ çağrı adını ya da CICS EIBFN işlevini içerebilen işlev. Alanda MQCFUNC\_NONE değeri, MQ\_FUNCTION\_LENGTH uzunluğuna sahip bir değer. Bu alana döndürülen değer, dönüş koduna bağlıdır. Daha ayrıntılı bilgi için bkz. [Çizelge 868 sayfa 1793](#).

Aşağıdaki ek değerler, **işlev** bir IBM MQ arama adı içerdiğinde mümkündür:

- MQCFUNC\_MQCONN
- MQCFUNC\_MQGET
- MQCFUNC\_MQINQ
- MQCFUNC\_NONE
- MQCFUNC\_MQAÇ
- MQCFUNC\_PUT
- MQCFUNC\_MQPUT1

## **bekleme süresi aralığı**

CICS bridge görevi tarafından yayınlanan bir MQGET çağrısına ilişkin bekleme aralığı. İlk değer MQCGWI\_VARSAYILANDIR. Alan yalnızca, **uow control** değeri MQCUOWC\_FIRST değerine sahip olduğunda geçerlidir. Aşağıdaki ek değerler kullanılabilir:

- MQCGWI\_VARSAYILAN
- MQWI\_UNSNMA

## **Bağlantı tipi**

Link tipi. İlk değer MQCLT\_PROGRAR olur. Aşağıdaki ek değerler kullanılabilir:

- MQCLT\_PROGRAM
- MQCLT\_TRANSACTION

## **sonraki işlem tanıtıcısı**

Eklenecek sonraki işlemin tanıtıcısı. Alan, MQ\_TRANSACTION\_ID\_LENGTH uzunluğunda olmalıdır.

## **çıkış veri uzunluğu**

COMMAREA veri uzunluğu. İlk değer MQCODL\_AS\_INPUT değeridir.

## **yanıtlama biçimi**

Yanıt iletisinin biçim adı. İlk değer MQFMT\_NONE, MQ\_FORMAT\_LENGTH uzunluğuna sahip.

## **başlatma kodu**

İşlem başlatma kodu. Alan, MQ\_START\_CODE\_LENGTH uzunluğunda olmalıdır. İlk değer MQCSC\_NONE değeridir. Aşağıdaki ek değerler kullanılabilir:

- MQCSC\_START
- MQCSC\_STARTDATA
- MQCSC\_TERMINPUT
- MQCSC\_NONE

## **görev bitiş durumu**

Görevin bitiş durumu. İlk değer MQCTES\_NOSYNC ' dir. Aşağıdaki ek değerler kullanılabilir:

- MQCTES\_COMMIT
- MQCTES\_BACKUT
- MQCTES\_ENDTASK
- MQCTES\_NOSYNC

## **işlem tanıtıcısı**

Bağlanılacak işlemin tanıtıcısı. İlk değer boşluk içermemeli ve MQ\_TRANSACTION\_ID\_LENGTH uzunluğunda olmalıdır. Bu alan yalnızca, **uow control** değeri MQCUOWC\_FIRST ya da MQCUOWC\_ONLY değerine sahip olduğunda geçerlidir.

## UOW denetimi

-OOW kontrolü. İlk deęer MQCUOWC\_ONLY deęeridir. Ařaęıdaki ek deęerler kullanılabilir:

- İLK MQCUOWC\_FIRST
- MQCUOWC\_ORTA
- MQCUOWC\_SON
- YALNIZCIK
- MQCUOWC\_COMMIT
- MQCUOWC\_BACKOUT
- MQCUOWC\_CONTINY

## sürüm

MQCIH sürüm numarası. İlk deęer MQCIH\_VERSION\_2' dir. Desteklenen dięer deęer MQCIH\_VERSION\_1' dir.

## Oluřturucular

### ImqCICSBridgeHeader();

Varsayılan oluřturucu.

### ImqCICSBridgeHeader(const ImqCICSBridgeHeader & header );

Kopya oluřturucusu.

## Ařırı yüklenmiř ImqItem yöntemleri

### sanal ImqBoolean copyOut( ImqMessage & msg );

Bir MQCIH veri yapısını bařlangıçta ileti arabelleęine ekler, var olan ileti verilerini daha ileriye taşıyarak bu ileti biçimini MQFMT\_CICS ' e ayarlar.

Daha fazla ayrıntı için üst sınıf yöntemi açıklamasına bakın.

### sanal ImqBoolean pasteIn( ImqMessage & msg );

İleti arabelleęinden bir MQCIH veri yapısını okur. Bařarılı olmak için, msg nesnesinin kodlamasının MQENC\_NATIVE olması gerekir. İletileri MQGMO\_CONVERT ile MQENC\_NATIVE ile alın. Bařarılı olmak için ImqMessage biçiminin MQFMT\_CICS olması gerekir.

Daha fazla ayrıntı için üst sınıf yöntemi açıklamasına bakın.

## Nesne yöntemleri (genel)

### void operator = (const ImqCICSBridgeHeader & header );

Eřgörünüm verilerini, var olan eřgörünüm verilerini deęiřtirerek *üstbilgisinden* kopyalar.

### MQUQT ADSDescriptor () const;

**ADS tanımlayıcısı'** nın bir kopyasını döndürür.

### void setADSDescriptor(const MQHOMED tanımlayıcısı = MQCADSD\_NONE);

**ADS tanımlayıcısı'** i ayarlar.

### ImqString attentionIdentifier() const;

MQ\_ATTENTION\_ID\_LENGTH uzunluęuna sahip sondaki boşluklarla doldurulmuř olan **uyarı tanıtıcısı** kopyasını döndürür.

### void setAttentionTanıtıcısı (const char \* data = 0);

Sondaki boşluklarla doldurulan **dikkat tanıtıcısını** MQ\_ATTENTION\_ID\_LENGTH uzunluęuna ayarlar. Veri sağlanmıyorsa, **dikkat tanıtıcısı** deęerini ilk deęerine döndürür.

### ImqString kimlik doęrulayıcı () const;

MQ\_AUTHENTICATOR\_LENGTH uzunluęuna sahip sondaki boşluklarla doldurulan **kimlik doęrulayıcı'** nın bir kopyasını döndürür.

**void setAuthenticator(const char \* data = 0);**

Sondaki boşluklarla doldurulan **kimlik doğrulayıcı** MQ\_authenticator\_length uzunluğuna ayarlar. *Veri* verilmezse, **kimlik doğrulayıcı** değerini ilk değerle ilk durumuna getirir.

**ImqString bridgeAbendKodu () const;**

MQ\_ABEND\_CODE\_LENGTH uzunluğuna sahip sondaki boşluklarla doldurulan **köprü olağandışı bitiş kodu** kopyasının bir kopyasını döndürür.

**ImqString bridgeCancelKodu () const;**

MQ\_CANCEL\_CODE\_LENGTH uzunluğuna sahip sondaki boşluklarla doldurulan **köprü iptal kodu**' nun bir kopyasını döndürür.

**void setBridgeCancelCode(const char \* data = 0);**

İzleyen boşluklarla doldurulan **köprü iptal kodunu** MQ\_CANCEL\_CODE\_LENGTH uzunluğuna kadar ayarlar. *Veri* verilmezse, **köprü iptal kodunu** başlangıç değerine sıfırlar.

**Mqlong bridgeCompletionKodu () const;**

**Köprü tamamlanma kodu**' nun bir kopyasını döndürür.

**MQHOT bridgeErrorGörelili Konumu () const;**

**Köprü hata görelili konumu**' nin bir kopyasını döndürür.

**MQHOT bridgeReasonKodu () const;**

**Köprü neden kodu**' nun bir kopyasını döndürür.

**Mqlong bridgeReturnKodu () const;**

**Köprü dönüş kodu** değerini döndürür.

**Mqlong conversationalTask() const;**

**conversasyonel task** ögesinin bir kopyasını döndürür.

**void setConversationalGörev (const MQWOLD task = MQCCT\_NO);**

**etkileşimli görevi** ayarlar.

**MQLONG cursorPosition() const;**

**İmleç konumu**' nun bir kopyasını döndürür.

**void setCursorPosition (const MQORT konumu = 0);**

**İmleç konumunu** ayarlar.

**MQUZT facilityKeepTime () const;**

**Tesise tutma süresi**' nin bir kopyasını döndürür.

**void setFacilityKeepTime(const MQHOTR zaman = 0);**

**Tesise ilişkin tutma süresi**' ne ayarlar.

**ImqString facilityLike() const;**

İzleyen boşluklarla doldurulan **tesis gibi** bir kopyasını MQ\_FACILITY\_LIKE\_LENGTH uzunluğuna döndürür.

**void setFacilityLike (const char \* ad = 0);**

Sondaki boşluklarla doldurulan **gibi bir tesis** değerini, MQ\_FACILITY\_LIKE\_LENGTH uzunluğuna ayarlar. *ad* belirtilmezse, ilk değeri **gibi bir tesis** sıfırlayın.

**ImqBinary facilityToken() const;**

**olanak simgesi** ögesinin bir kopyasını döndürür.

**ImqBoolean setFacilitySimgesi (const ImqBinary & simge );**

**tesis simgesi**' i ayarlar. *token* ile ilgili **veri uzunluğu** sıfır ya da MQ\_FACILITY\_LENGTH olmalıdır. Başarılı olursa TRUE değerini döndürür.

**void setFacilityToken (const MQBYTE8 simge = 0);**

**tesis simgesi**' i ayarlar. *simge* sıfır olabilir, bu da MQCFAC\_NONE belirtilerek aynıdır. *simge* sıfırda, bu değer MQ\_FACILITY\_LENGTH byte 'ı ikili verileri ele almalıdır. MQCFAC\_NONE gibi önceden tanımlanmış değerleri kullanırken, imza eşleştirmesini sağlamak için bir tip dönüşümü yapmanız gerekebilir. Örneğin, (MQBYTE \*) MQCFAC\_NONE.

**ImqString işlevi () const;**

MQ\_FUNCTION\_LENGTH uzunluğuna sahip sondaki boşluklarla doldurulan **işlevin** bir kopyasını döndürür.

**MQlong getWaitInterval () const;**

**get wait interval** öğesinin bir kopyasını döndürür.

**void setGetWaitInterval(const MQUBE *aralık* = MQCGWI\_DEFA**

**Bekleme süresi aralığını** al seçeneğini belirler.

**MQLONG linkType() const;**

**link type**' in bir kopyasını döndürür.

**void setLinkTip (const MQORT *tip* = MQCLT\_PROPRON);**

**Bağlantı tipini** ayarlar.

**ImqString nextTransactionTanıtıcısı () const;**

MQ\_TRANSACTION\_ID\_LENGTH uzunluğuna sahip sondaki boşluklarla doldurulan **sonraki işlem tanıtıcısı** verilerinin bir kopyasını döndürür.

**MQlong outputDataUzunluk () const;**

**Çıkış verileri uzunluğu**' nın bir kopyasını döndürür.

**void setOutputDataLength(const MQOUT *length* = MQCODL\_AS\_INPUT);**

**Çıkış veri uzunluğunu** belirler.

**ImqString replyToBiçimi () const;**

İzleyen boşluklarla doldurulan **yanıtlama biçimi** adının bir kopyasını MQ\_FORMAT\_LENGTH uzunluğuna döndürür.

**void setReplyToFormat(const char \* *ad* = 0);**

Sondaki boşluklarla doldurulan **yanıt-biçimi**' yi MQ\_FORMAT\_LENGTH uzunluğuna ayarlar. *ad* sağlanmıyorsa, **yanıt-biçimi** değerini başlangıç değerine sıfırlar.

**ImqString startCode() const;**

MQ\_START\_CODE\_LENGTH uzunluğuna sahip sondaki boşluklarla doldurulan **başlangıç kodunun** bir kopyasını döndürür.

**void setStartKodu (const char \* *data* = 0);**

İzleyen boşluklarla doldurulan **başlangıç kodu** verilerini MQ\_START\_CODE\_LENGTH uzunluğuna ayarlar. *Veri* sağlanmazsa, **başlangıç kodu** ' na ilk değeri ilk durumuna getirir.

**MQHOT taskEndDurumu () const;**

**Görev bitiş durumu**' nun bir kopyasını döndürür.

**ImqString transactionIdentifier() const;**

İzleyen boşluklarla doldurulan **işlem tanıtıcısı** verilerinin bir kopyasını MQ\_TRANSACTION\_ID\_LENGTH uzunluğuna geri döndürür.

**void setTransactionTanıtıcısı (const char \* *veri* = 0);**

İzleyen boşluklarla doldurulan **işlem tanıtıcısını** MQ\_TRANSACTION\_ID\_LENGTH uzunluğuna ayarlar. *Veri* sağlanmıyorsa, **işlem tanıtıcısı** değerini ilk değerine döndürür.

**MQLONG UOWControl () const;**

**UOW denetimi**' nin bir kopyasını döndürür.

**void setUOWControl(const MQORT *control* = MQCUOWC\_ONLY);**

**UOW denetimi**' ni ayarlar.

**MQUZE sürümü () const;**

**version** numarasını döndürür.

**ImqBoolean setVersion(const MQHOMET *version* = MQCIH\_VERSION\_2 );**

**sürüm** numarasını ayarlar. Başarılı olursa TRUE değerini döndürür.

## Nesne verileri (korunan)

**MQUZT olVersion**

*Opci* için ayrılan depolamada konaklayabilecek MQCIH sürüm numarası üst sınırı.

**PMQCIH *opcih***

MQCIH veri yapısının adresi. Ayrılan depolama miktarı *olVersion* ile gösterilir.

## Neden kodları

- MQRC\_BINARY\_DATA\_LENGTH\_ERROR
- MQRC\_WRONG\_SÜRÜM

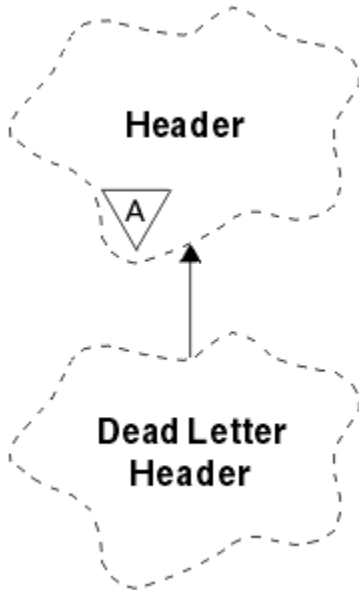
## Dönüş kodları

Çizelge 868. ImqCICSBridgeHeader sınıf dönüş kodları

Dönüş Kodu	İşlev	CompCode	Neden	Olağandışı Son Kod
MQCRC_OK				
MQCRC_BRIDGE_ERROR			MQFB_CICS	
MQCRC_MQ_API_ERROR	IBM MQ arama adı	IBM MQ CompCode	IBM MQ Neden	
MQCRC_BRIDGE_TIMEOUT	IBM MQ arama adı	IBM MQ CompCode	IBM MQ Neden	
MQCRC_CICS_EXEC_ERROR	CICS EIBFN	CICS EIBRESP	CICS EIBRESP2	
MQCRC_SECURITY_ERROR	CICS EIBFN	CICS EIBRESP	CICS EIBRESP2	
MQCRC_PROGRAM_NOT_YOK	CICS EIBFN	CICS EIBRESP	CICS EIBRESP2	
MQCRC_TRANSID_NOT_VAR	CICS EIBFN	CICS EIBRESP	CICS EIBRESP2	
MQCRC_BRIDGE_ABEND				CICS ABCODE
MQCRC_APPLICATION_ABEND				CICS ABCODE

## ImqDeadLetterHeader C++ sınıfı

Bu sınıf, MQDLH veri yapısının özelliklerini sarsalıyor.



Şekil 19. ImqDeadLetterHeader sınıfı

Bu sınıfın nesneleri genellikle, işlenemeyen bir iletiyle karşılaştıran bir uygulama tarafından kullanılır. Ölü-mektup üstbilgisini oluşturan yeni bir ileti ve ileti içeriği, ölü-mektup kuyruğuna yerleştirilir ve ileti atılır.

- [“Nesne öznitelikleri” sayfa 1794](#)
- [“Oluşturucular” sayfa 1794](#)
- [“Aşırı yüklenmiş ImqItem yöntemleri” sayfa 1794](#)
- [“Nesne yöntemleri \(genel\)” sayfa 1795](#)
- [“Nesne verileri \(korunan\)” sayfa 1795](#)
- [“Neden kodları” sayfa 1795](#)

## Nesne öznitelikleri

### ölü harf neden kodu

İletinin, ölü mektup kuyruğuna ulaşmasının nedeni. İlk değer MQRC\_NONE olur.

### Hedef kuyruk yöneticisi adı

Özgün hedef kuyruk yöneticisinin adı. Bu ad, MQ\_Q\_MGR\_NAME\_LENGTH uzunluğunun dizilimini içerir. Başlangıç değeri boş değerli.

### hedef kuyruk adı

Özgün hedef kuyruğun adı. Ad, MQ\_Q\_NAME\_LENGTH uzunluğunun dizgisidir. Başlangıç değeri boş değerli.

### Koyma Uygulaması Adı

İletiyi ölü-mektup kuyruğuna koyan uygulamanın adı. Ad, MQ\_PUT\_APPL\_NAME\_LENGTH uzunluğunun dizgisidir. Başlangıç değeri boş değerli.

### Koyma Uygulaması Tipi

İletiyi ölü-mektup kuyruğuna koyan uygulama tipi. Başlangıç değeri sıfır.

### bitiş tarihi

İletinin, ölü-mektup kuyruğuna konduğu tarih. Tarih, MQ\_PUT\_DATE\_LENGTH uzunluğunda bir dizilimdir. Başlangıç değeri, boş değerli bir dizgidir.

### sürme zamanı

İletinin, ölü-mektup kuyruğuna konulduğu zaman. Saat, MQ\_PUT\_TIME\_LENGTH uzunluğunda bir dizilimdir. Başlangıç değeri, boş değerli bir dizgidir.

## Oluşturucular

### ImqDeadLetterHeader();

Varsayılan oluşturucu.

### ImqDeadLetterHeader(const ImqDeadLetterHeader & header );

Kopya oluşturucusu.

## Aşırı yüklenmiş ImqItem yöntemleri

### sanal ImqBoolean copyOut ( ImqMessage & msg );

Başlangıçta ileti arabelleğiyle birlikte bir MQDLH veri yapısı ekler ve var olan ileti verilerini daha da ileriye taşımayı sağlar. msg biçimini MQFMT\_DEAD\_LETTER\_HEADER olarak ayarlar.

See the ImqHeader class method description on page [“ImqHeader C++ sınıfı” sayfa 1801](#) for further details.

### sanal ImqBoolean pasteIn ( ImqMessage & msg );

İleti arabelleğinden bir MQDLH veri yapısını okur.

Başarılı olmak için, ImqMessage biçimi MQFMT\_DEAD\_LETTER\_HEADER olmalıdır.

See the ImqHeader class method description on page [“ImqHeader C++ sınıfı” sayfa 1801](#) for further details.

## Nesne yöntemleri (genel)

**void operator = (const ImqDeadLetterHeader & header );**

Eşgörünüm verilerinin kopyalanması, var olan eşgörünüm verilerinin yerine *üstbilgisinden* kopyalanır.

**MQlong deadLetterReasonCode () const;**

Ölü harfin neden kodunu döndürür.

**void setDeadLetterReasonKodu (const MQHOBE neden );**

Ölüm mektubu neden kodunu ayarlar.

**ImqString destinationQueueManagerName () const;**

Sondaki boşluklardan oluşan hedef kuyruk yöneticisi adını döndürür.

**void setDestinationQueueManagerAdı (const char \* ad );**

Hedef kuyruk yöneticisi adını ayarlar. MQ\_Q\_MGR\_NAME\_LENGTH ' den (48 karakter) daha uzun veri keser.

**ImqString destinationQueueAd () const;**

İzleyen boşluklardan alınan hedef kuyruk adının bir kopyasını döndürür.

**void setDestinationQueueName (const char \* ad );**

Hedef kuyruk adını belirler. MQ\_Q\_NAME\_LENGTH ' den (48 karakter) daha uzun veri keser.

**ImqString putApplicationAdı () const;**

Herhangi bir sondaki boşluklardan oluşan, koyma uygulaması adının bir kopyasını döndürür.

**void setPutApplicationName (const char \* ad = 0);**

Koyma uygulama adını belirler. MQ\_PUT\_APPL\_NAME\_LENGTH ' den (28 karakter) daha uzun veri keser.

**MQOUR putApplicationTip () const;**

Koyma uygulama tipini döndürür.

**void setPutApplicationType (const MQHOMET tip = MQAT\_NO\_CONTEXT);**

Koyma uygulama tipini ayarlar.

**ImqString putDate () const;**

Sondaki boşluklardan alınan, koyma tarihinin bir kopyasını döndürür.

**void setPutDate (const char \* date = 0);**

Koyma tarihini belirler. MQ\_PUT\_DATE\_LENGTH (8 karakter) değerinden daha uzun veri keser.

**ImqString putTime () const;**

Sondaki boşluklardan oluşan bir koyma süresinin bir kopyasını döndürür.

**void setPutTime (const char \* time = 0);**

Koyma süresini ayarlar. MQ\_PUT\_TIME\_LENGTH (8 karakter) uzunluğundan daha uzun verileri keser.

## Nesne verileri (korunan)

**MQDLH omqdlh**

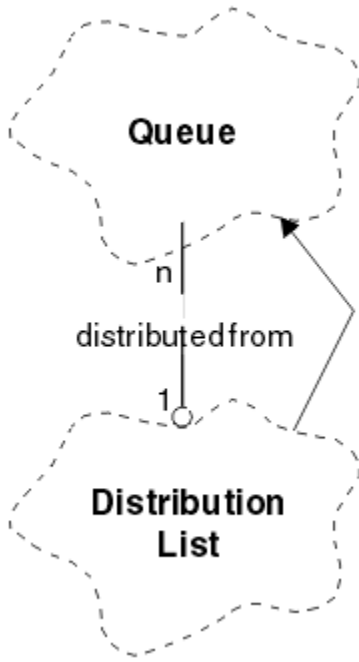
MQDLH veri yapısı.

## Neden kodları

- MQRC\_INCONSTENT\_BIÇIMI
- MQRC\_STRUC\_ID\_ERROR
- MQRCENCODING\_ERROR

## ImqDistributionListe C++ sınıfı

Bu sınıf, birden çok hedefe bir ileti ya da ileti gönderme amacıyla bir ya da daha fazla kuyruğa gönderme yapan dinamik bir dağıtım listesini sarsalıyor.



Şekil 20. *ImqDistributionListe Sınıfı*

- [“Nesne öznitelikleri” sayfa 1796](#)
- [“Oluşturucular” sayfa 1796](#)
- [“Nesne yöntemleri \(genel\)” sayfa 1796](#)
- [“Nesne yöntemleri \(korunan\)” sayfa 1797](#)

## Nesne öznitelikleri

### ilk dağıtılmış kuyruk

Sınıfın ilk ya da daha çok nesnesi, **dağıtım listesi başvurusu** ' un bu nesneyi ele aldığı belirli bir sırada yer alır.

Başlangıçta böyle bir nesne yoktur. Bir *ImqDistributionListesini* başarıyla açmak için en az bir nesne olması gerekir.

**Not:** Bir *ImqDistributionList* nesnesi açıldığında, bu nesneye başvuran açık nesnelere otomatik olarak kapatılır.

## Oluşturucular

### **ImqDistributionListesi ();**

Varsayılan oluşturucu.

### **ImqDistributionListesi ( const ImqDistributionList & list );**

Kopya oluşturucusu.

## Nesne yöntemleri (genel)

### **void operator = ( const ImqDistributionList & list );**

All objects that reference **bu** object are dereferenced before copying. Bu yöntemin çağrılmasından sonra hiçbir nesne **bu** nesneye başvuruda bulunmayacak.

### **\* firstDistributedKuyruğu () const ;**

**ilk dağıtılmış kuyruk** değerini döndürür.



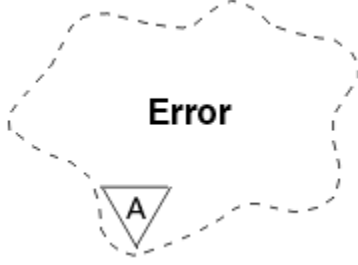
## Nesne yöntemleri (korunan)

`void setFirstDistributedQueue ( * queue = 0);`

İlk dağıtılmış kuyruğu ayarlar.

## ImqError C++ sınıfı

Bu soyut sınıf, bir nesneyle ilişkili hatalarla ilgili bilgi sağlar.



Şekil 21. ImqError sınıfı

- “Nesne öznitelikleri” sayfa 1797
- “Oluşturucular” sayfa 1797
- “Nesne yöntemleri (genel)” sayfa 1797
- “Nesne yöntemleri (korunan)” sayfa 1798
- “Neden kodları” sayfa 1798

## Nesne öznitelikleri

### tamamlanma kodu

En son tamamlanma kodu. Başlangıç değeri sıfır. Aşağıdaki ek değerler kullanılabilir:

- MQCC\_OK
- MQCC\_UYARI
- MQCC\_FAILED

### Neden Kodu

En son neden kodu. Başlangıç değeri sıfır.

## Oluşturucular

### ImqError();

Varsayılan oluşturucu.

### ImqError( const ImqError & error );

Kopya oluşturucusu.

## Nesne yöntemleri (genel)

### void operator = ( const ImqError & error );

Eşgörünüm verilerini, var olan eşgörünüm verilerini değiştirerek *hatadankopyalar*.

### void clearErrorCodes ();

**tamamlanma kodu** ve **neden kodu** ' yi her ikisi de sıfır olarak ayarlar.

### MQlong completionCode () const ;

**tamamlanma kodu** değerini döndürür.

### MQUZN reasonCode () const ;

**Neden kodu** değerini döndürür.

## Nesne yöntemleri (korunan)

**ImqBoolean checkReadPointer ( const void \* pointer, const size\_t length );**

İşaretçi ve uzunluk bileşiminin salt okunur erişim için geçerli olduğunu doğrular ve başarılı olursa TRUE değerini döndürür.

**ImqBoolean checkWritePointer ( const void \* pointer, const size\_t length );**

İşaretçi ve uzunluk bileşiminin okuma yazma erişimi için geçerli olduğunu doğrular ve başarılı olursa TRUE değerini döndürür.

**void setCompletionKodu ( const MQlong kod = 0);**

tamamlanma kodunu ayarlar.

**void setReasonKodu ( const MQlong kod = 0);**

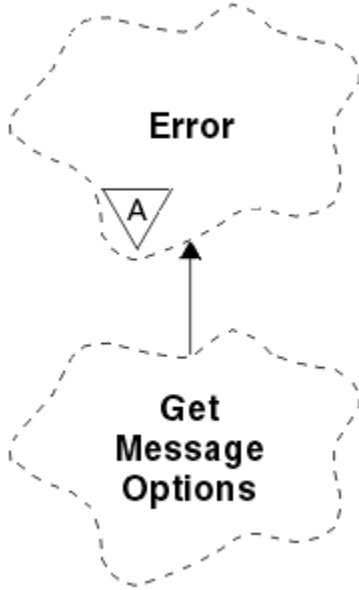
Neden kodu' yı belirler.

## Neden kodları

- MQRC\_BUFFER\_ERROR

## ImqGetMessageOptions C++ sınıfı

Bu sınıf MQGMO veri yapısını sarsalıyor



Şekil 22. ImqGetMessageOptions sınıfı

- “Nesne öznitelikleri” sayfa 1798
- “Oluşturucular” sayfa 1800
- “Nesne yöntemleri (genel)” sayfa 1800
- “Nesne yöntemleri (korunan)” sayfa 1801
- “Nesne verileri (korunan)” sayfa 1801
- “Neden kodları” sayfa 1801

## Nesne öznitelikleri

### grup durumu

Bir ileti grubuna ilişkin iletinin durumu. İlk değer MQGS\_NOT\_IN\_GROUP değeridir. Aşağıdaki ek değerler kullanılabilir:

- MQGS\_MSG\_IN\_GROUP

- MQGS\_LAST\_MSG\_IN\_GROUP

#### **eşleşme seçenekleri**

Gelen iletilerin seçilmesine ilişkin seçenekler. Başlangıç değeri: MQMO\_MATCH\_MSG\_ID | MQMO\_MATCH\_COREL\_ID. Aşağıdaki ek değerler kullanılabilir:

- MQMO\_GROUP\_ID
- MQMO\_MATCH\_MSG\_SEQ\_NUMBER
- MQMO\_MATCH\_OFFSET
- MQMO\_MSG\_TOKEN
- MQMO\_NONE

#### **ileti simgesi**

İleti simgesi. MQ\_MSG\_TOKEN\_LENGTH uzunluğuna ilişkin bir ikili değer (MQBYTE16). İlk değer MQMTOK\_NONE değeridir.

#### **seçenekler**

Bir ileti için geçerli seçenekler. Başlangıçtaki değer MQGMO\_NO\_BEK. değeridir. Aşağıdaki ek değerler kullanılabilir:

- MQGMO\_BEKLE
- MQGMO\_SYNCPOINT
- MQGMO\_SYNCPOINT\_IF\_PERSISTENT
- MQGMO\_NO\_SYNCPOINT
- MQGMO\_MARK\_SKIP\_BACKUT
- MQGMO\_BROWSE\_FIRST
- MQGMO\_BROWSE\_NEXT
- MQGMO\_BROWSE\_MSG\_UNDER\_CURSOR
- MQGMO\_MSG\_UNDER\_CURSOR
- MQGMO\_LOCK
- MQGMO\_UNLOCK
- MQGMO\_ACCEPT\_TRUNCATED\_MSG
- MQGMO\_SET\_SIGNAL
- MQGMO\_FAIL\_IF QUIESCING
- MQGMO\_CONVERT
- MQGMO\_LOGICAL\_ORDER
- MQGMO\_COMPLE\_MSG
- MQGMO\_ALL\_MSGS\_AVALABILIR
- MQGMO\_ALL\_SEGMENTS\_AVALABILIR
- MQGMO\_NONE

#### **çözümlenen kuyruk adı**

Çözülmüş kuyruk adı. Bu öznitelik salt okunurdur. Adlar hiçbir zaman 48 karakterden uzun değildir ve boş değerler ile bu uzunlukla doldurulabilir. Başlangıç değeri, boş değerli bir dizgidir.

#### **döndürülen uzunluk**

Döndürülen uzunluk. İlk değer MQRL\_UNDEFED değerine sahip. Bu öznitelik salt okunurdur.

#### **kesimlere ayırma**

Bir iletiyi bölümlenebilme yeteneği. İlk değer MQSEG\_INHIMATED olur. Ek değer olan MQSEG\_ALLOWASY olanaklıdır.

#### **bölüm durumu**

İletin kesimlere ayırma durumu. İlk değer MQSS\_NOT\_A\_SEGMENTI 'dir. Aşağıdaki ek değerler kullanılabilir:

- MQSS\_SEGMENT
- MQSS\_LAST\_SEGMENT

### syncpoint katılımı

İletilerin syncpoint denetimi altında alındığında TRUE (Doğru) değeri alınır.

### Bekleme Aralığı

Varsa, uygun bir iletinin gelmesini beklerken sınıfın yöntem duraksaması alma süresi (varsa). İlk değer sıfır (sıfır), belirsiz bir bekleme durumu etkiler. MQWI\_UNUSISINI ek değeri mümkün. Seçenekler MQGMO\_WEKLE içermediği sürece bu öznelik yoksayıdır.

## Oluşturucular

### ImqGetMessageOptions( );

Varsayılan oluşturucu.

### ImqGetMessageOptions(const ImqGetMessageOptions & gmo );

Kopya oluşturucusu.

## Nesne yöntemleri (genel)

### void işleci = (const ImqGetMessageOptions & gmo );

Var olan eşgörünüm verilerini değiştirerek, yönetim ortamı verilerini gmo' den kopyalar.

### MQCHAR groupStatus () const;

Grup durumunu döndürür.

### void setGroupStatus (const MQCHAR status );

Grup durumunu ayarlar.

### MQHOT matchOptions () const;

Eşleştirme seçeneklerini döndürür.

### void setMatchSeçenekleri (const MQHOMET seçenekleri );

Eşleştirme seçeneklerini belirler.

### ImqBinary messageToken() const;

İleti simgesini döndürür.

### ImqBoolean setMessageSimgesi (const ImqBinary & simge );

İleti simgesini ayarlar. *simge* veri uzunluğu sıfır ya da MQ\_MSG\_TOKEN\_LENGTH olmalıdır. Bu yöntem başarılı olursa TRUE değerini döndürür.

### void setMessageSimgesi (const MQBYTE16 simge = 0);

İleti simgesini ayarlar. *simge* sıfır olabilir, bu da MQMTOK\_NONE belirtilerek aynıdır. *simge* sıfır değilse, bu, ikili verilerin MQ\_MSG\_TOKEN\_LENGTH byte 'ları ile adreslenmelidir.

MQMTOK\_NONE gibi önceden tanımlanmış değerleri kullanırken, imza eşleştirmesini sağlamak için tip dönüşümü yapmanız gerekmeyebilir; örneğin (MQBYTE \*) MQMTOK\_NONE gibi.

### MQTHOE seçenekleri () const;

Seçenekleri döndürür.

### void setOptions (const MQHOMET seçenekleri );

Syncpoint katılım değeri de dahil olmak üzere seçenekleri belirler.

### ImqString resolvedQueueAdı () const;

Çözülen kuyruk adının bir kopyasını döndürür.

### MQHOT returnedLength() const;

Döndürülen uzunluğu döndürür.

### MQCHAR bölümlenmesi () const;

Bölümlenme değerini döndürür.

### void setSegmentation (const MQCHAR value );

Kesimlere ayırma belirler.

**MQCHAR segmentStatus () const;**

Bölüm durumunu döndürür.

**void setSegmentStatus (const MQCHAR status );**

Kesim durumunu ayarlar.

**ImqBoolean syncPointKatılım () const;**

Seçenekler MQGMO\_SYNCPOINT ya da MQGMO\_SYNCPOINT\_IF\_PERSISTENT içerdiğinde, TRUE değerini içeren syncpoint katılım değerini döndürür.

**void setSyncPointParticipation (const ImqBoolean sync );**

Syncpoint katılım değerini ayarlar. *Eşitleme* TRUE ise, seçenekleri MQGMO\_SYNCPOINT içermeye ve MQGMO\_NO\_SYNCPOINT ve MQGMO\_SYNCPOINT\_IF\_PERSISTENT öğesini dışlamak için seçenekleri değiştirir. *Eşitleme* FALSE ise, seçenekleri MQGMO\_NO\_SYNCPOINT içermeye ve hem MQGMO\_SYNCPOINT, hem de MQGMO\_SYNCPOINT\_IF\_PERSISTENT seçeneklerini içerecek şekilde değiştirir.

**MQHOT waitInterval () const;**

Bekleme aralığını döndürür.

**void setWaitInterval (const MQUBE aralık );**

Bekleme aralığını ayarlar.

**Nesne yöntemleri (korunan)****static void setVersionSupported (const MQLong);**

MQGMO sürümünü ayarlar. Varsayılan değer olarak MQGMO\_VERSION\_3 değerine ayarlanır.

**Nesne verileri (korunan)****MQGMO omqgmo**

Bir MQGMO Sürüm 2 veri yapısı. Yalnızca MQGMO\_VERSION\_2 için desteklenen MQGMO alanlarına erişir.

**PMQGMO opgmo**

MQGMO veri yapısının adresi. Bu adrese ilişkin sürüm numarası *olVersion* içinde belirtilir. MQGMO alanlarına erişmeden önce bunların var olduğundan emin olmak için sürüm numarasını inceleyin.

**MQUZE olVersion**

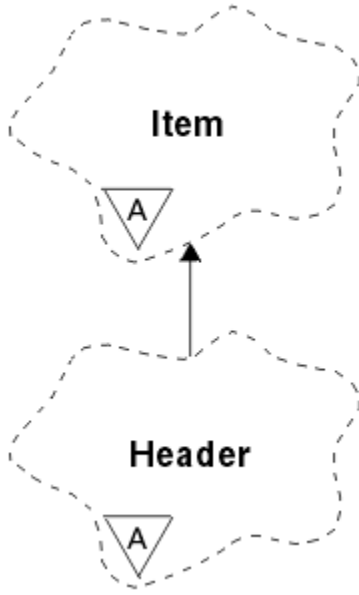
*opgmo* tarafından adreslenen MQGMO veri yapısının sürüm numarası.

**Neden kodları**

- MQRC\_BINARY\_DATA\_LENGTH\_ERROR

**ImqHeader C++ sınıfı**

Bu soyut sınıf, MQDLH veri yapısının ortak özelliklerini kapsüller.



Şekil 23. *ImqHeader* sınıfı

- “Nesne öznitelikleri” sayfa 1802
- “Oluşturucular” sayfa 1802
- “Nesne yöntemleri (genel)” sayfa 1802

## Nesne öznitelikleri

### karakter kümesi

Özgün kodlanmış karakter takımı tanıtıcısı. Başlangıçta MQCCSI\_Q\_MGR.

### Kodlama

Özgün kodlama. Başlangıçta MQENC\_NATIVE.

### biçim

Özgün biçim. İlk olarak MQFMT\_NONE.

### üstbilgi işaretleri

Başlangıç değerleri şunlardır:

- *ImqDeadLetterHeader* sınıfının nesnelere ilişkin sıfır
- MQIIH\_NONE, *ImqIMSBridgeHeader* sınıfının nesnelere için
- MQRMHF\_SON *ImqReferenceüstbilgi* sınıfı nesnelere için
- MQCIH\_NONE, *ImqCICSBridgeHeader* sınıfının nesnelere için
- MQWIIH\_NONE, *ImqWorküstbilgi* sınıfı nesnelere için

## Oluşturucular

### **ImqHeader( );**

Varsayılan oluşturucu.

### **ImqHeader( const ImqHeader & header );**

Kopya oluşturucusu.

## Nesne yöntemleri (genel)

### **void operator = ( const ImqHeader & header );**

Var olan eşgörünüm verilerini değiştirerek, yönetim ortamı verilerini *üstbilgi*<sup>1</sup> den kopyalar.

**sanal MQTHOE characterSet () const ;**  
Karakter kümesideğerini döndürür.

**virtual void setCharacterSet ( const MQLong ccsid = MQCCSI\_Q\_MGR);**  
Karakter kümesi' yi ayarlar.

**sanal MQORT encoding () const ;**  
encodingdeğerini döndürür.

**virtual void setEncoding ( const MQLong encoding = MQENC\_NATIVE);**  
kodlama' yı ayarlar.

**sanal ImqString biçim () const ;**  
Sondaki boşluklar da içinde olmak üzere, **biçim'** in bir kopyasını döndürür.

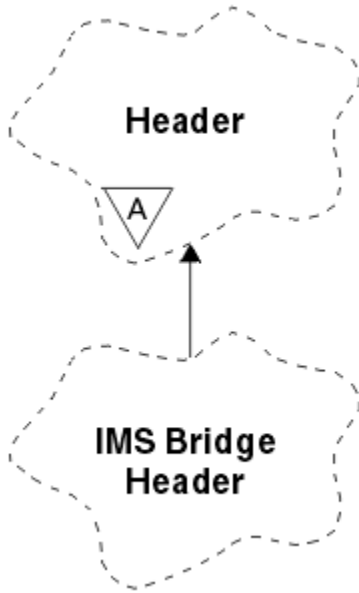
**virtual void setFormat ( const char \* ad = 0);**  
Sondaki boşluklar içeren 8 karaktere kadar doldurulmuş **biçim'** i ayarlar.

**sanal MQLong headerFlags () const ;**  
Üstbilgi işaretlerini döndürür.

**virtual void setHeaderFlags ( const MQLong flags = 0);**  
Üstbilgi işaretlerini ayarlar.

## ImqIMSBridgeHeader C++ sınıfı

Bu sınıf, MQIIH veri yapısının özelliklerini sarsalıyor.



Şekil 24. ImqIMSBridgeHeader sınıfı

Objects of this class are used by applications that send messages to the IMS bridge through IBM MQ for z/OS.

**Not:** ImqHeader karakter kümesi ve kodlamasının varsayılan değerleri olmalıdır ve başka bir değer ayarlanmamış olmalıdır.

- [“Nesne öznitelikleri” sayfa 1804](#)
- [“Oluşturucular” sayfa 1804](#)
- [“Aşırı yüklenmiş ImqItem yöntemleri” sayfa 1804](#)
- [“Nesne yöntemleri \(genel\)” sayfa 1804](#)
- [“Nesne verileri \(korunan\)” sayfa 1805](#)
- [“Neden kodları” sayfa 1805](#)

## Nesne öznitelikleri

### kimlik doğrulayıcı

RACF parolası ya da passticket, MQ\_AUTHENTICATOR\_LENGTH uzunluğundan. İlk değer MQIAUT\_NONE olur.

### kesinleştirme kipi

Kesinleştirme kipi. IMS kesinleştirme kiplerine ilişkin ek bilgi için *OTMA User's Guide* belgesine bakın. İlk değer MQICM\_COMMIT\_THEN\_SEND değeridir. Ek değer olan MQICM\_SEND\_THEN\_COMMIT olanaklıdır.

### mantıksal uçbirim geçersiz kılma

Mantıksal uçbirim geçersiz kılma değeri, MQ\_LTERM\_OVERRIDE\_LENGTH uzunluğundan. Başlangıç değeri, boş değerli bir dizgidir.

### ileti biçimi hizmetleri eşlemi adı

MFS eşlem adı, MQ\_MFS\_MAP\_NAME\_LENGTH uzunluğunda. Başlangıç değeri, boş değerli bir dizgidir.

### yanıtlama biçimi

Herhangi bir yanıtın biçimi, MQ\_FORMAT\_LENGTH uzunluğundan. İlk değer MQFMT\_NONE olur.

### güvenlik kapsamı

IMS güvenlik işlemlerinin kapsamı. Başlangıçtaki değer MQISS\_REKVER değeridir. Ek değer olan MQISS\_FULL, olanaklı.

### işlem eşgörünümü tanıtıcısı

Hareket eşgörünümü kimliği, ikili (MQBYTE16) değeri MQ\_TRAN\_INSTANCE\_ID\_LENGTH değeri. İlk değer MQITII\_NONE olur.

### hareket durumu

IMS sohbetinin durumu. İlk değer MQITS\_NOT\_IN\_CONVERSACE değeridir. Ek değer olan MQITS\_IN\_CONVERSACE olanaklıdır.

## Oluşturucular

### ImqIMSBridgeHeader();

Varsayılan oluşturucu.

### ImqIMSBridgeHeader(const ImqIMSBridgeHeader & header );

Kopya oluşturucusu.

## Aşırı yüklenmiş ImqItem yöntemleri

### sanal ImqBoolean copyOut ( ImqMessage & msg );

Başlangıçta ileti arabelleğindeki bir MQIIH veri yapısı ekler ve var olan ileti verilerini daha da ileriye taşımayı sağlar. *msg* biçimini MQFMT\_IMS olarak ayarlar.

Ek ayrıntılar için üst sınıf yöntemi açıklamasına bakın.

### sanal ImqBoolean pasteIn ( ImqMessage & msg );

İleti arabelleğinden bir MQIIH veri yapısını okur.

Başarılı olmak için, *msg* nesnesinin kodlamasının MQENC\_NATIVE olması gerekir. İletileri MQGMO\_CONVERT ile MQENC\_NATIVE ile alın.

Başarılı olmak için, ImqMessage biçimi MQFMT\_IMS olmalıdır.

Ek ayrıntılar için üst sınıf yöntemi açıklamasına bakın.

## Nesne yöntemleri (genel)

### void operator = (const ImqIMSBridgeHeader & header );

Var olan eşgörünüm verilerini değiştirerek, yönetim ortamı verilerini *üstbilgi'* den kopyalar.

### ImqString kimlik doğrulayıcı () const;

MQ\_AUTHENTICATOR\_LENGTH uzunluğuna sondaki boşluklarla doldurulan kimlik doğrulayıcının bir kopyasını döndürür.



**void setAuthenticator (const char \* ad );**

Kimlik doğrulayıcıyı ayarlar.

**MQCHAR commitMode () const;**

Kesinleştirme kipini döndürür.

**void setCommitKipi (const MQCHAR kipi );**

Kesinleştirme kipini ayarlar.

**ImqString logicalTerminalGeçersiz Kılma () const;**

Mantıksal uçbirim geçersiz kılma işlevinin bir kopyasını döndürür.

**void setLogicalTerminalOverride (const char \* override );**

Mantıksal uçbirim geçersiz kılma değerini ayarlar.

**ImqString messageFormatServicesMapName () const;**

İleti biçimi hizmetleri eşlemi adının bir kopyasını döndürür.

**void setMessageFormatServicesMapName (const char \* ad );**

İleti biçimi hizmetleri eşlemi adını ayarlar.

**ImqString replyToBiçimi () const;**

MQ\_FORMAT\_LENGTH uzunluğuna kadar boşluk içeren, yanıtlama biçiminin bir kopyasını döndürür.

**void setReplyToFormat (const char \* biçim );**

Yanıtlanacak yanıtı belirler; sondaki boşluklarla doldurulmuş olarak, MQ\_FORMAT\_LENGTH uzunluğuna kadar doldurulur.

**MQCHAR securityScope () const;**

Güvenlik kapsamını döndürür.

**void setSecurityKapsamı (const MQCHAR scope );**

Güvenlik kapsamını belirler.

**ImqBinary transactionInstanceId () const;**

İşlem eşgörünümü tanıtıcısının bir kopyasını döndürür.

**ImqBoolean setTransactionInstanceId (const ImqBinary & id );**

İşlem eşgörünümü tanıtıcısını ayarlar. *simge* veri uzunluğunun sıfır ya da MQ\_TRAN\_INSTANCE\_ID\_LENGTH olması gerekir. Bu yöntem başarılı olursa TRUE değerini döndürür.

**void setTransactionInstanceId (const MQBYTE16 id = 0);**

İşlem eşgörünümü tanıtıcısını ayarlar. *id* sıfır (sıfır) olabilir; bu, MQITII\_NONE belirtilerek aynıdır. *id* sıfır dışında bir değer ise, bu, MQ\_TRAN\_INSTANCE\_ID\_LENGTH of binary (ikili veri) byte 'ı adresli olmalıdır. MQITII\_NONE gibi önceden tanımlanmış değerleri kullanırken, imza eşleştirmesini sağlamak için bir tip dönüşümü yapmanız gerekebilir; örneğin (MQBYTE \*) MQITII\_NONE.

**MQCHAR transactionState () const;**

Hareket durumunu döndürür.

**void setTransactionState (const MQCHAR state );**

Hareket durumunu ayarlar.

## Nesne verileri (korunan)

**MQIIH oqiih**

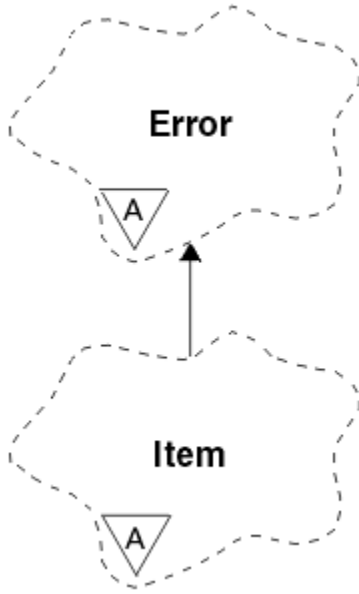
MQIIH veri yapısı.

## Neden kodları

- MQRC\_BINARY\_DATA\_LENGTH\_ERROR
- MQRC\_INCONSTENT\_BIÇIMI
- MQRCENCODING\_ERROR
- MQRC\_STRUC\_ID\_ERROR

## ImqItem C++ sınıfı

Bu soyut sınıf, bir ileti içindeki bir öğeyi (belki de birden çok) temsil eder.



Şekil 25. ImqItem sınıfı

Öğeler, bir ileti arabelleğindeki birleştirilir. Her bir uzmanlaşma, bir yapı tanıtıcısıyla başlayan belirli bir veri yapısıyla ilişkilendirilir.

Bu soyut sınıftaki çokmorfik yöntemler, öğelerin iletilere ve iletilerden kopyalanmasına olanak sağlar. ImqMessage sınıfı **readItem** ve **writeItem** yöntemleri, uygulama programları için daha doğal olan bu çokmorfik yöntemlerin çağrılmasına ilişkin başka bir stil sağlar.

- [“Nesne öznitelikleri” sayfa 1806](#)
- [“Oluşturucular” sayfa 1806](#)
- [“Sınıf yöntemleri \(genel\)” sayfa 1806](#)
- [“Nesne yöntemleri \(genel\)” sayfa 1807](#)
- [“Neden kodları” sayfa 1807](#)

### Nesne öznitelikleri

#### yapı tanıtıcısı

Veri yapısının başlangıcındaki dört karakterden oluşan bir dizgi. Bu öznitelik salt okunurdur. Türetilmiş sınıflar için bu özniteliği dikkate alın. Otomatik olarak içerilmez.

### Oluşturucular

#### ImqItem();

Varsayılan oluşturucu.

#### ImqItem( const ImqItem & item );

Kopya oluşturucusu.

### Sınıf yöntemleri (genel)

#### statik ImqBoolean structureIdis ( const char \* structure-id-to-test, const ImqMessage & msg );

Gelen *msg* içindeki sonraki ImqItem öğesinin **yapı tanıtıcısı** *yapı-tanıtıcısı-test* ile aynıysa TRUE (Doğru) değerini döndürür. Sonraki öğe, şu anda ImqCache **veri işaretçisi** tarafından adreslenen ileti arabelleğinin bir parçası olarak tanımlanır. Bu yöntem **yapı tanıtıcısı** 'na dayanır ve bu nedenle, tüm ImqItem türetilmiş sınıflara ilişkin çalışma garantilenmez.

## Nesne yöntemleri (genel)

**void işleç = ( const ImqItem & item );**

Eşgörünüm verilerini, var olan eşgörünüm verilerini değiştirerek *öğedenkopyalar*.

**virtual ImqBoolean copyOut ( ImqMessage & msg ) = 0;**

Bu nesneyi, giden ileti arabelleğindeki sonraki öge olarak yazar ve var olan ögelere ekler. Yazma işlemi başarılı olursa, ImqCache **veri uzunluğunu** artırır. Bu yöntem başarılı olursa TRUE değerini döndürür.

Belirli bir alt sınıfla çalışmak için bu yöntemi geçersiz kılın.

**sanal ImqBoolean pasteIn ( ImqMessage & msg ) = 0;**

Bu nesneyi, gelen ileti arabelleğinden *yok edici olarak* okur. Okunan değer, ImqCache **veri işaretçisi** 'nin üzerine taşındığı için yıkıcı bir şekilde ortaya çıkarılır. Ancak, arabellek içeriği aynı kalır, bu nedenle ImqCache **veri işaretçisini** sıfırlayarak veriler yeniden okunabilir.

Bu nesnenin (alt) sınıfı, *msg* nesnesinin ileti arabelleğindeki bir sonraki sırada bulunan **yapı tanıtıcısı** ile tutarlı olmalıdır.

*msg* nesnesinin **encoding** değeri MQENC\_NATIVE olmalıdır. İletilerin ImqMessage **kodlama** kümesiyle MQENC\_NATIVE ile ve MQGMO\_CONVERT de içinde olmak üzere ImqGetMessageOptions **seçenekler** ile alınması önerilir.

Okuma işlemi başarılı olursa, ImqCache **veri uzunluğu** azaltılır. Bu yöntem başarılı olursa TRUE değerini döndürür.

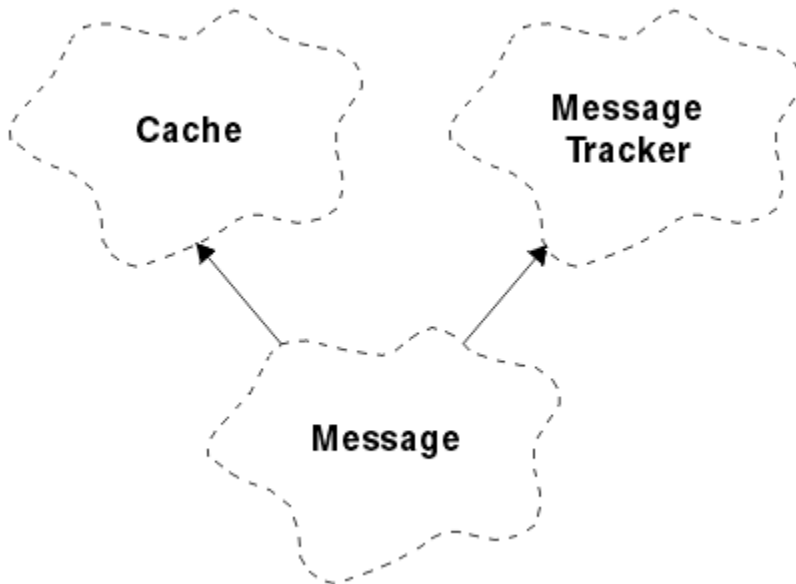
Belirli bir alt sınıfla çalışmak için bu yöntemi geçersiz kılın.

## Neden kodları

- MQRCENCODING\_ERROR
- MQRC\_STRUC\_ID\_ERROR
- MQRC\_INCONSTANT\_BİÇİMİ
- MQRC\_INSUFFICIENT\_BUFFER
- MQRC\_INSUFFICIENT\_DATA

## ImqMessage C++ sınıfı

Bu sınıf bir MQMD veri yapısını kapsüller ve ileti verilerinin yapımını ve yeniden yapılandırımını da işler.



Şekil 26. ImqMessage sınıfı

- [“Nesne öznitelikleri” sayfa 1808](#)
- [“Oluşturucular” sayfa 1811](#)
- [“Nesne yöntemleri \(genel\)” sayfa 1811](#)
- [“Nesne yöntemleri \(korunan\)” sayfa 1813](#)
- [“Nesne verileri \(korunan\)” sayfa 1813](#)

## Nesne öznitelikleri

### Uygulama Tnt Verisi

Bir iletiyle ilişkili kimlik bilgileri. Başlangıç değeri, boş değerli bir dizgidir.

### Uygulama Kökeni Verisi

İletiyle ilişkilendirilen başlangıç bilgileri. Başlangıç değeri, boş değerli bir dizgidir.

### Geriletme Sayısı

Bir iletinin geçici olarak alındığı ve daha sonra geriletilmesinin kaç kez olduğu. Başlangıç değeri sıfır. Bu öznitelik salt okunurdur.

### karakter kümesi

Kodlanmış Karakter Takımı Tanıtıcısı. Başlangıçtaki değer MQCCSI\_Q\_MGR ' dir. Aşağıdaki ek değerler kullanılabilir:

- MQCCSI\_INHERIT
- MQCCSI\_EMBEDDED

Ayrıca, seçiminize ilişkin bir Kodlanmış Karakter Takımı Tanıtıcısı da kullanabilirsiniz. Bu konuya ilişkin bilgi için bkz. [“Kod sayfası dönüşümü” sayfa 922](#).

### Kodlama

İleti verilerinin makine kodlaması. İlk değer MQENC\_NATIVE değeridir.

### Son kullanma tarihi

IBM MQ ' un alınmamış bir iletiyi atmadan önce ne kadar süre sakladığını denetleyen, zamana bağlı bir miktar. Başlangıçtaki değer MQII\_UNESSıNıDıdır.

### biçim

Arabelleğindeki verilerin yerleşim düzenini tanımlayan biçim (şablon) adı. Sekiz karakterden daha uzun adlar sekiz karaktere kısaltılır. Adlar her zaman, sekiz karakter uzunluğunda boşluklarla dolduruludur. Başlangıçtaki değişmez değer MQFMT\_NONE olur. Aşağıdaki ek sabitler şunlardır:

- MQFMT\_ADMIN
- MQFMT\_CICS
- MQFMT\_COMMAND\_1
- MQFMT\_COMMAND\_2
- MQFMT\_DEAD\_LETTER\_HEADER
- MQFMT\_DIST\_HEADER
- MQFMT\_OLAY
- MQFMT\_IMS
- MQFMT\_IMS\_VAR\_STRING
- MQFMT\_MD\_EXTENSION
- MQFMT\_PCF
- MQFMT\_REF\_MSG\_HEADER
- MQFMT\_RF\_HEADER
- MQFMT\_STRING
- MQFMT\_TETIKLEYICISI
- MQFMT\_WORK\_INFO\_HEADER

- MQFMT\_XMIT\_Q\_HEADER

Ayrıca, seçiminize özgü bir dizeyi de kullanabilirsiniz. Bu konuda daha fazla bilgi için, ileti tanımlayıcısının (MQMD) [“Biçim \(MQCHAR8\)” sayfa 441](#) alanına bakın.

### İleti İşaretleri

Kesimlere ayırma denetimi bilgileri. İlk değer MQMF\_SEGMENTATION\_INHIBITED değeridir. Aşağıdaki ek değerler kullanılabilir:

- MQMF\_SEGMENTATION\_ALLOW
- MQMF\_MSG\_IN\_GROUP
- MQMF\_LAST\_MSG\_IN\_GROUP
- MQMF\_SEGMENT
- MQMF\_LAST\_SEGMENT
- MQMF\_NONE

### ileti tipi

Bir iletinin geniş kategorilendirme. İlk değer MQMT\_DATAGRAM ' tır. Aşağıdaki ek değerler kullanılabilir:

- MQMT\_SYSTEM\_FIRST
- MQMT\_SYSTEM\_SON
- MQMT\_DATAGRAM
- MQMT\_REQUEST
- MQMT\_REPLY
- MQMT\_REPORT
- MQMT\_APPL\_FIRST
- MQMT\_APPL\_SON

Ayrıca, seçiminize özgü bir uygulamaya özgü bir değer de kullanabilirsiniz. Bu konuda daha fazla bilgi için, ileti tanımlayıcısının (MQMD) [“MsgType \(MQUZE\)” sayfa 431](#) alanına bakın.

### offset

Görelî konum bilgisi. Başlangıç değeri sıfır.

### Özgün Uzunluk

Kesimlere ayrılmış bir iletinin özgün uzunluğu. İlk değer MQOL\_UNDEFINED değeridir.

### Kalıcılık

İletinin önemli olduğunu ve kalıcı saklama alanı kullanılarak tüm sürelerde yedekleneceğini belirtir. Bu seçenek, bir performans cezasının anlamına gelir. İlk değer MQPER\_PERSISTENCE\_AS\_Q\_DEF değeridir. Aşağıdaki ek değerler kullanılabilir:

- MQPER\_PERISPER
- MQPER\_NOT\_PERSISTENT

### öncelik

İletim ve teslimat için görelî öncelik. Aynı önceliğe sahip iletiler, genellikle sağlandığı sırayla teslim edilir (bunu garantilmekten memnun olması gereken birkaç kriter olmasına rağmen). Başlangıçtaki değer MQPRI\_PRIORITY\_AS\_Q\_DEF olur.

### özellik geçerlilik denetimi

İletinin bir özelliği ayarlandığında özellikler doğrulamasının gerçekleşip gerçekleştirilmeyeceğini belirtir. İlk değer MQCMHO\_DEFAULT\_VALIDATION değeridir. Aşağıdaki ek değerler kullanılabilir:

- MQCMHO\_VALIDATE
- MQCMHO\_NO\_VALIDATION

Aşağıdaki yöntemler **özellik geçerlilik denetimi** üzerinde işlem sağlar:

**MQHOT propertyValidation() const;**

Özellik geçerlilik denetimi seçeneğini döndürür.

**void setPropertyValidation (const MQHOMET seçenek );**

Özellik doğrulaması seçeneğini belirler.

**Koyma Uygulaması Adı**

İleti koyan uygulamanın adı. Başlangıç değeri, boş değerli bir dizgidir.

**Koyma Uygulaması Tipi**

Bir iletiyi koyan uygulamanın tipi. Başlangıçtaki değer MQAT\_NO\_CONTEXT ' dir. Aşağıdaki ek değerler kullanılabilir:

- MQAT\_AIX
- MQAT\_CICS
- MQAT\_CICS\_BRIDGE
- MQAT\_DOS
- MQAT\_IMS
- MQAT\_IMS\_BRIDGE
- MQAT\_MVS
- MQAT\_NOTES\_AGENT
- MQAT\_OS2
- MQAT\_OS390
- MQAT\_OS400
- MQAT\_QMGR
- MQAT\_UNIX
- MQAT\_WINDOWS
- MQAT\_WINDOWS\_NT
- MQAT\_XCF
- MQAT\_DEFAULT
- MQAT\_BILINMIYOR
- MQAT\_USER\_FIRST
- MQAT\_USER\_SON

Ayrıca, seçiminize özgü bir dizeyi de kullanabilirsiniz. Bu konuda daha fazla bilgi için, ileti tanımlayıcısının (MQMD) [“PutApplTipi \(MQLONG\)”](#) sayfa 454 alanına bakın.

**bitiş tarihi**

Bir iletinin yerleştirdiği tarih. Başlangıç değeri, boş değerli bir dizgidir.

**sürme zamanı**

Bir iletinin yerleştirdiği saat. Başlangıç değeri, boş değerli bir dizgidir.

**yanıtlama-kuyruk yöneticisi adı**

Herhangi bir yanıtın gönderileceği kuyruk yöneticisinin adı. Başlangıç değeri, boş değerli bir dizgidir.

**yanıt kuyruğu adı**

Herhangi bir yanıtın gönderileceği kuyruğun adı. Başlangıç değeri, boş değerli bir dizgidir.

**rapor**

Bir iletiyle ilişkili geribildirim bilgileri. İlk değer MQRO\_NONE olur. Aşağıdaki ek değerler kullanılabilir:

- MQRO\_EXCEPTION
- MQRO\_EXCEPTION\_WITH\_DATA
- MQRO\_EXCEPTION\_WITH\_FULL\_DATA \*
- MQRO\_EXPIRATION

- MQRO\_EXPIRATION\_WITH\_DATA
- MQRO\_EXPIRATION\_WITH\_FULL\_DATA \*
- MQRO\_COA
- MQRO\_CO\_WITH\_DATA
- MQRO\_COA\_WITHHL\_FULL\_DATA \*
- MQRO\_COD
- MQRO\_COD\_WITH\_DATA
- MQRO\_COD\_WITH\_FULL\_DATA \*
- MQRO\_PAN
- MQRO\_NAN
- MQRO\_NEW\_MSG\_ID
- MQRO\_NEW\_CORREL\_ID
- MQRO\_COPY\_MSG\_ID\_TO\_CORREL\_ID
- MQRO\_PASS\_COREL\_ID
- MQRO\_DEAD\_LETTER\_Q
- MQRO\_DISCARD\_MSG

Burada \*, IBM MQ for z/OSüzerinde desteklenmeyen deęerleri gösterir.

#### **Sıra Numarası**

Grup içindeki bir iletiyi tanımlayan sıra bilgileri. Başlangıç deęeri bir deęerdir.

#### **toplam ileti uzunluęu**

En son ileti okuma giriřimi sırasında kullanılabilir olan bayt sayısı. Son ileti kısaltıldıysa, bu sayı ImqCache **ileti uzunluęu** deęerinden büyük ya da kesme geręekleřmiř olduęu için son ileti okunmuyorsa daha büyük olur. Bu öznitelik salt okunurdur. Başlangıç deęeri sıfır.

Bu öznitelik, kesilmiř iletiler ięeren herhangi bir durumda yararlı olabilir.

#### **kullanıcı kimlięi**

Bir iletiyle iliřkili kullanıcı kimlięi. Başlangıç deęeri, boş deęerli bir dizgidir.

### **Oluřturucular**

#### **ImqMessage( );**

Varsayılan oluřturucu.

#### **ImqMessage( const ImqMessage & msg );**

Kopya oluřturucusu. Ayrıntılar için **operator** = yöntemine bakın.

### **Nesne yöntemleri (genel)**

#### **void operator = ( const ImqMessage & msg );**

MQMD 'yi ve ileti verilerini *msg*' den kopyalar. Kullanıcı tarafından bu nesne için bir arabellek saęlandıysa, kopyalanan veri miktarı, kullanılabilir arabellek boyuyla sınırlandırılır. Ters durumda, sistem, kopyalanan veriler için yeterli büyüklükte bir arabellek saęlanmasını saęlar.

#### **ImqString applicationIdData () const ;**

**Application ID data**(uygulama tanıtıcısı verilerinin) bir kopyasını döndürür.

#### **void setApplicationIdData ( const char \* data = 0);**

**uygulama tanıtıcısı verileri'** i ayarlar.

#### **ImqString applicationOriginVerileri () const ;**

**Application origin data**(uygulama başlangıç verileri) öęesinin bir kopyasını döndürür.

#### **void setApplicationOriginData ( const char \* data = 0);**

**Uygulama kaynaęı verilerini**ayarlar.

**MQUZE backoutCount () const ;**  
Geri alma sayısı değerini döndürür.

**MQLONG characterSet () const ;**  
Karakter kümesi değerini döndürür.

**void setCharacterSet ( const MQlong ccsid = MQCCSI\_Q\_MGR);**  
Karakter kümesi' yi ayarlar.

**MQORT encoding () const ;**  
encoding değerini döndürür.

**void setEncoding ( const MQlong encoding = MQENC\_NATIVE);**  
kodlama' yı ayarlar.

**MQUZE süre bitimi () const ;**  
Süre bitimi değerini döndürür.

**void setExpiry ( const MQlong süre bitimi );**  
Süre Bitimi' yi Ayarlar.

**ImqString biçim () const ;**  
Sondaki boşluklar da içinde olmak üzere, **biçim'** in bir kopyasını döndürür.

**ImqBoolean formatIs ( const char \* format-to-test ) const ;**  
**Biçim** , *biçim-test* ile aynıysa TRUE değerini döndürür.

**void setFormat ( const char \* ad = 0);**  
Sondaki boşluklar içeren sekiz karaktere kadar doldurulmuş **biçim'** i ayarlar.

**MQUZT messageFlags () const ;**  
İleti işaretleri döndürür.

**void setMessageİşaretleri ( const MQlong işaretler );**  
İleti işaretleri' i ayarlar.

**MQUZT messageType () const ;**  
İleti tipini döndürür.

**void setMessageType ( const MQORT tip );**  
İleti tipini ayarlar.

**MQUVE görelî konumu () const ;**  
görelî konum değerini döndürür.

**void setOffset ( const MQlong görelî konum );**  
Görelî Konum' u ayarlar.

**MQLONG originalLength () const ;**  
Özgün uzunluk değerini döndürür.

**void setOriginalLength ( const MQlong uzunluk );**  
Özgün uzunluk değerini belirler.

**MQHOT kalıcılık () const ;**  
Persistence değerini döndürür.

**void setPersistence ( const MQlong persistence );**  
Kalıcılık' ı ayarlar.

**MQLONG öncelik () const ;**  
Öncelik değerini döndürür.

**void setPriority ( const MQlong priority );**  
Öncelik seçeneğini belirler.

**ImqString putApplicationAd () const ;**  
**put application name** (uygulama adının konulması) ögesinin bir kopyasını döndürür.

**void setPutApplicationName ( const char \* ad = 0);**  
Uygulama adını koy ögesini ayarlar.

**MQLONG putApplicationTipi () const ;**  
**put application type** değerini döndürür.



**void setPutApplicationType ( const MQLong type = MQAT\_NO\_CONTEXT);**

Uygulama tipini indirseçeneğini belirler.

**ImqString putDate () const ;**

put date öğesinin bir kopyasını döndürür.

**void setPutDate ( const char \* date = 0);**

put date(put tarihi) seçeneğini belirler.

**ImqString putTime () const ;**

put time' in bir kopyasını döndürür.

**void setPutTime ( const char \* time = 0);**

put time(put time) öğesini ayarlar.

**ImqBoolean readItem ( ImqItem & item );**

Reads into the öğe object from the message buffer, using the ImqItem **pasteIn** method. Başarılı olursa TRUE değerini döndürür.

**ImqString replyToQueueManagerName () const ;**

yanıtlama kuyruğu yöneticisi adı öğesinin bir kopyasını döndürür.

**void setReplyToQueueManagerName ( const char \* ad = 0);**

Yanıtlama kuyruğu yöneticisi adını ayarlar.

**ImqString replyToQueueName () const ;**

Yanıtlama kuyruğu adı' nın bir kopyasını döndürür.

**void setReplyToQueueAdı ( const char \* ad = 0);**

Yanıt kuyruğu adı' nın adını belirler.

**MQUZN rapor () const ;**

Rapor' u döndürür.

**void setReport ( const MQLong report );**

Rapor' u ayarlar.

**MQUZN sequenceNumber () const ;**

Sıra numarasını döndürür.

**void setSequenceNumber ( const MQORT sayı );**

Sıra numarasını ayarlar.

**size\_t totalMessageUzunluk () const ;**

Toplam ileti uzunluğu değerini döndürür.

**ImqString userId () const ;**

Kullanıcı kimliği' nin bir kopyasını döndürür.

**void setUserId ( const char \* id = 0);**

Kullanıcı kimliği' yi ayarlar.

**ImqBoolean writeItem ( ImqItem & item );**

ImqItem **copyOut** yöntemini kullanarak, öğe nesnesinden ileti arabelleğine yazar. Yazma işlemi, ekleme, değiştirme ya da ekleme biçimini alabilir: bu, öğe nesnesinin sınıfına bağlıdır. Bu yöntem başarılı olursa TRUE değerini döndürür.

## Nesne yöntemleri (korunan)

**static void setVersionDestekleniyor ( const MQHOTR );**

MQMD sürümünü ayarlar. Varsayılan değer olarak MQMD\_VERSION\_2 değerine ayarlanır.

## Nesne verileri (korunan)

**z/OS MQMD1 omqmd**

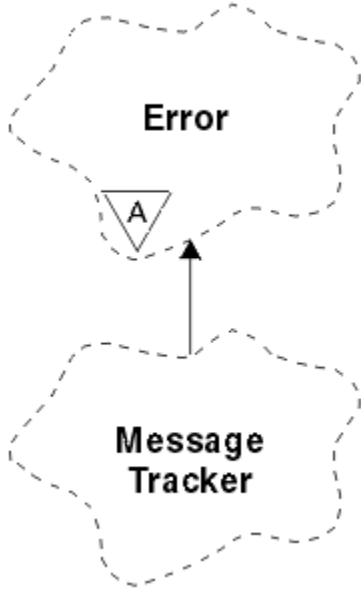
z/OS üzerindeki MQMD veri yapısı.

**Multi MQMD2 omqmd**

Multiplatforms üzerindeki MQMD veri yapısı.

## ImqMessageTracker C++ sınıfı

Bu sınıf, herhangi bir nesneyle ilişkilendirilebilecek bir ImqMessage ya da ImqQueue nesnesinin özniteliklerini sarsar.



Şekil 27. ImqMessageİzleyici sınıfı

Bu sınıf, [“ImqMessageİzleyici çapraz başvurusu”](#) sayfa 1766’inde listelenen MQI çağruları ile ilgilidir.

- [“Nesne öznitelikleri”](#) sayfa 1814
- [“Oluşturucular”](#) sayfa 1815
- [“Nesne yöntemleri \(genel\)”](#) sayfa 1815
- [“Neden kodları”](#) sayfa 1816

### Nesne öznitelikleri

#### muhasebe simgesi

MQ\_ACCOUNTING\_TOKEN\_LENGTH uzunluğunda bir ikili değer (MQBYTE32). İlk değer MQACT\_NONE olur.

#### İlinti Tanıtıcısı

İlintilendirme iletilerine atadığınız MQ\_COREL\_ID\_LENGTH uzunluğunda bir ikili değer (MQBYTE24). İlk değer MQCI\_NONE olur. MQCI\_NEW\_SESSION ek değeri mümkün olabilir.

#### Geribildirim

Bir iletiyle gönderilecek geribildirim bilgileri. İlk değer MQFB\_NONE olur. Aşağıdaki ek değerler kullanılabilir:

- MQFB\_SYSTEM\_FIRST
- MQFB\_SYSTEM\_SON
- MQFB\_APPL\_FIRST
- MQFB\_APPL\_SON
- MQFB\_COA
- MQFB\_COD
- MQFB\_BITIM
- MQFB\_PAN
- MQFB\_NAN

- MQFB\_QUIT
- MQFB\_DATA\_LENGTH\_ZERO
- MQFB\_DATA\_LENGTH\_NEGATIF
- MQFB\_DATA\_LENGTH\_TOO\_BIG
- MQFB\_BUFFER\_OVERFLOW
- MQFB\_LENGTH\_OFF\_BY\_ONE
- MQFB\_IH\_ERROR
- MQFB\_NOT\_AUTHORIZED\_FOR\_IMS
- MQFB\_IMS\_ERROR
- MQFB\_IMS\_FIRST
- MQFB\_IMS\_SON
- MQFB\_CICS\_APPL\_ABENLENDİ
- MQFB\_CICS\_APPL\_NOT\_STARTED
- MQFB\_CICS\_BRIDGE\_FAILURE
- MQFB\_CICS\_CCSID\_ERROR
- MQFB\_CICS\_CIH\_ERROR
- MQFB\_CICS\_COMMAREA\_ERROR
- MQFB\_CICS\_CORREL\_ID\_ERROR
- MQFB\_CICS\_DLQ\_ERROR
- MQFB\_CICS\_ENCODING\_ERROR
- MQFB\_CICS\_INTERNAL\_ERROR
- MQFB\_CICS\_NOT\_YETKILI
- MQFB\_CICS\_UOW\_BACKED\_OUT
- MQFB\_CICS\_UOW\_ERROR

Ayrıca, seçiminize özgü bir dizeyi de kullanabilirsiniz. Bu konuda daha fazla bilgi için, ileti tanımlayıcısının (MQMD) [“Geribildirim \(MQUZE\)”](#) sayfa 435 alanına bakın.

#### **grup tanıtıcısı**

MQ\_GROUP\_ID\_LENGTH uzunluğunda bir ikili değer (MQBYTE24), bir kuyruk içinde benzersiz. İlk değer MQGI\_NONE olur.

#### **İleti Tnt**

Bir kuyruktaki MQ\_MSG\_ID\_LENGTH uzunluğunun ikili değeri (MQBYTE24). İlk değer MQMI\_NONE olur.

### **Oluşturucular**

#### **ImqMessageTracker ();**

Varsayılan oluşturucu.

#### **ImqMessageİzleyici ( const ImqMessageTracker & tracker );**

Kopya oluşturucusu. Ayrıntılar için **operator =** yöntemine bakın.

### **Nesne yöntemleri (genel)**

#### **void operator = ( const ImqMessageTracker & tracker );**

Var olan eşgörünüm verilerini değiştirerek eşgörünüm verilerini *tracker'* dan kopyalar.

#### **ImqBinary accountingToken () const ;**

**muhasebe simgesi** ögesinin bir kopyasını döndürür.

#### **ImqBoolean setAccountingSimgesi ( const ImqBinary & simge );**

**muhasebe belirteci'** yi ayarlar. *token* ile ilgili **veri uzunluğu** sıfır ya da

MQ\_ACCOUNTING\_TOKEN\_LENGTH olmalıdır. Bu yöntem başarılı olursa TRUE değerini döndürür.

**void setAccountingSimgesi ( const MQBYTE32 simge = 0);**

**muhasabe belirteci**' yi ayarlar. *simge* sıfır olabilir, bu da MQACT\_NONE belirtilerek aynıdır. *simge* sıfır değilse, bu değer, ikili verilerin MQ\_ACCOUNTING\_TOKEN\_LENGTH byte 'ı ele almalıdır. MQACT\_NONE gibi önceden tanımlanmış değerleri kullanırken, imza eşleştirmesini sağlamak için bir tip dönüştürme yapmanız gerekebilir; örneğin, (MQBYTE \*) MQACT\_NONE.

**ImqBinary correlationId () const ;**

**İlinti tanıtıcısı**' nin bir kopyasını döndürür.

**ImqBoolean setCorrelationId ( const ImqBinary & token );**

**İlinti tanıtıcısını** ayarlar. *token* ile ilgili **veri uzunluğu** sıfır ya da MQ\_COREL\_ID\_LENGTH olmalıdır. Bu yöntem başarılı olursa TRUE değerini döndürür.

**void setCorrelationId ( const MQBYTE24 id = 0);**

**İlinti tanıtıcısını** ayarlar. *id* sıfır (sıfır) olabilir; bu, MQCI\_NONE belirtilerek aynıdır. *id* sıfır dışında bir değer ise, bu değer, MQ\_COREL\_ID\_LENGTH of binary (ikili) veri byte 'ını adreslemelidir. MQCI\_NONE gibi önceden tanımlanmış değerleri kullanırken, imza eşleştirmesini sağlamak için bir tip dönüştürme yapmanız gerekebilir; örneğin, (MQBYTE \*) MQCI\_NONE.

**MQHOT geribildirim () const ;**

**Geribildirim** değerini döndürür.

**void setFeedback ( const MQLong geribildirim );**

**Geribildirim**' yi ayarlar.

**ImqBinary groupId () const ;**

**Grup tanıtıcısı**' nin bir kopyasını döndürür.

**ImqBoolean setGroupId ( const ImqBinary & token );**

**Grup tanıtıcısı**' yi ayarlar. *token* ile ilgili **veri uzunluğu** sıfır ya da MQ\_GROUP\_ID\_LENGTH olmalıdır. Bu yöntem başarılı olursa TRUE değerini döndürür.

**void setGroupId ( const MQBYTE24 id = 0);**

**Grup tanıtıcısı**' yi ayarlar. *id* sıfır (sıfır) olabilir; bu, MQGI\_NONE belirtme ile aynıdır. *id* sıfır dışında bir değer ise, bu, MQ\_GROUP\_ID\_LENGTH of binary (ikili veri) verilerini ele almalıdır. MQGI\_NONE gibi önceden tanımlanmış değerleri kullanırken, imza eşleştirmesini sağlamak için bir tip dönüştürme yapmanız gerekebilir; örneğin (MQBYTE \*) MQGI\_NONE.

**ImqBinary messageId () const ;**

**İleti tanıtıcısı**' nin bir kopyasını döndürür.

**ImqBoolean setMessageId ( const ImqBinary & token );**

**İleti tanıtıcısını** belirler. *token* ile ilgili **veri uzunluğu** sıfır ya da MQ\_MSG\_ID\_LENGTH olmalıdır. Bu yöntem başarılı olursa TRUE değerini döndürür.

**void setMessageId ( const MQBYTE24 id = 0);**

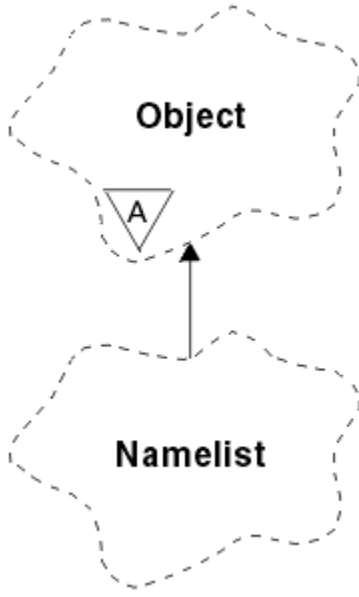
**İleti tanıtıcısını** belirler. *id* sıfır (sıfır) olabilir; bu, MQMI\_NONE belirtme ile aynıdır. *id* sıfır değilse, bu, MQ\_MSG\_ID\_LENGTH ikili verilerinin byte 'ını ele almalıdır. MQMI\_NONE gibi önceden tanımlanmış değerleri kullanırken, imza eşleştirmesini sağlamak için bir tip dönüştürme yapmanız gerekebilir; örneğin (MQBYTE \*) MQMI\_NONE.

## Neden kodları

- MQRC\_BINARY\_DATA\_LENGTH\_ERROR

## ImqNameList C++ sınıfı

Bu sınıf bir ad listesini sarsalıyor.



Şekil 28. *ImqNamelist* sınıfı

Bu sınıf, [“ImqNamelist çapraz başvurusu” sayfa 1766](#) içinde listelenen MQI çağrılarını ile ilgilidir.

- [“Nesne öznitelikleri” sayfa 1817](#)
- [“Oluşturucular” sayfa 1817](#)
- [“Nesne yöntemleri \(genel\)” sayfa 1817](#)
- [“Neden kodları” sayfa 1818](#)

## Nesne öznitelikleri

### ad sayısı

**ad listesi adları** içindeki nesne adları sayısı. Bu öznitelik salt okunurdur.

### ad listesi adları

Object names, the number of which is indicated by the **ad sayısı**. Bu öznitelik salt okunurdur.

## Oluşturucular

### **ImqNamelist();**

Varsayılan oluşturucu.

### **ImqNamelist(const ImqNamelist & list );**

Kopya oluşturucusu. **ImqObject open status** değeri false (yanlış) değeridir.

### **ImqNamelist(const char \* ad );**

**ImqObject** adını **name** olarak ayarlar.

## Nesne yöntemleri (genel)

### **void operator = (const ImqNamelist & list );**

Varolan yönetim ortamı verilerini değiştirerek, yönetim ortamı verilerini *listen* içinden kopyalar. **ImqObject open status** değeri false (yanlış) değeridir.

### **ImqBoolean nameCount(MQUZO & sayı );**

**Ad sayısı**'nin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

### **MQHOT nameCount ();**

Olası hatalara ilişkin herhangi bir belirti olmadan **ad sayısı** değerini döndürür.

### **ImqBoolean namelistName (const MQORD dizin, ImqString & ad );**

Provides a copy of one the **ad listesi adları** by zero based index. Başarılı olursa TRUE değerini döndürür.

### **ImqString namelistName (const MQHOMEN dizin );**

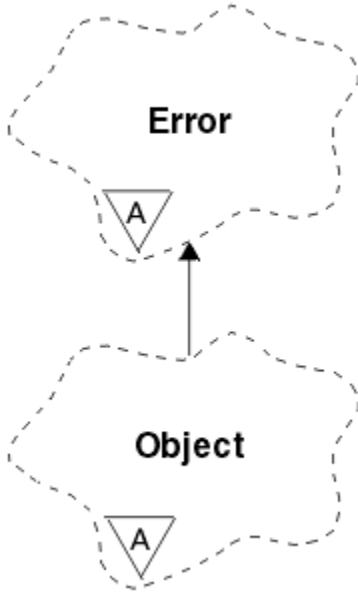
Herhangi bir hata belirtisi göstermeksizin, sıfır tabanlı dizinle **ad listesi adlarından** birini döndürür.

### **Neden kodları**

- MQRC\_INDEX\_ERROR
- MQRC\_INDEX\_NOT\_PRESENT

## **ImqObject C++ sınıfı**

Bu sınıf soyut. Bu sınıfın bir nesnesi yok edildiğinde, otomatik olarak kapatılır ve ImqQueueManager bağlantısı kesilir.



Şekil 29. ImqObject sınıfı

Bu sınıf, [“ImqObject çapraz başvurusu” sayfa 1767](#) içinde listelenen MQI çağrılarını ile ilgilidir.

- [“Sınıf öznitelikleri” sayfa 1818](#)
- [“Nesne öznitelikleri” sayfa 1819](#)
- [“Oluşturucular” sayfa 1820](#)
- [“Sınıf yöntemleri \(genel\)” sayfa 1820](#)
- [“Nesne yöntemleri \(genel\)” sayfa 1820](#)
- [“Nesne yöntemleri \(korunan\)” sayfa 1822](#)
- [“Nesne verileri \(korunan\)” sayfa 1823](#)
- [“Neden kodları” sayfa 1823](#)
- 

### **Sınıf öznitelikleri**

#### **davranış**

Örtülü açıklığın davranışını denetler.

#### **IMQ\_IMPL\_ASK (8L)**

Örtülü açma işlemine izin verilir. Bu varsayılandır.

## Nesne öznitelikleri

### değişirme tarihi

Değişim randevusu. Bu öznitelik salt okunurdur.

### değişirme zamanı

Değişim zamanı. Bu öznitelik salt okunurdur.

### diğer kullanıcı kimliği

Diğer kullanıcı kimliği, MQ\_USER\_ID\_LENGTH karakterlerine kadar. Başlangıç değeri, boş değerli bir dizgidir.

### diğer güvenlik tanıtıcısı

Diğer güvenlik tanıtıcısı. MQ\_SECURITY\_ID\_LENGTH uzunluğunda bir ikili değer (MQBYTE40). İlk değer MQSID\_NONE 'dır.

### kapatma seçenekleri

Bir nesne kapatıldığında geçerli olan seçenekler. İlk değer MQCO\_NONE olur. Bu öznitelik, örtük olarak yeniden açma işlemleri sırasında yoksayılır; burada MQCO\_NONE değeri her zaman kullanılır.

### bağlantı başvurusu

Bir (yerel) kuyruk yöneticisine gerekli bağlantıyı sağlayan bir ImqQueueManager nesnesine yönelik başvuru. Bir ImqQueueManager nesnesi için, nesnenin kendisidir. Başlangıç değeri sıfır.

**Not:** Bu değeri, bir kuyruk yöneticisini (büyük olasılıkla uzak) adlandırılmış bir kuyruk için tanıtan kuyruk yöneticisi adıyla karıştırmayın.

### açıklama

Kuyruk yöneticisi, kuyruk, ad listesi ya da süreç için açıklayıcı ad (en çok 64 karakter). Bu öznitelik salt okunurdur.

### ad

Kuyruk yöneticisi, kuyruk, ad listesi ya da işlem için ad (en çok 48 karakter). Başlangıç değeri, boş değerli bir dizgidir. The name of a model queue changes after an **open** to the name of the resulting dynamic queue.

**Not:** ImqQueueYöneticisi, varsayılan kuyruk yöneticisini temsil eden boş değerli bir ada sahip olabilir. Bu ad, başarılı bir açık açıldıktan sonra gerçek kuyruk yöneticisinde değişir. ImqDistributionListesi dinamiktir ve boş değerli bir ada sahip olmalıdır.

### sonraki yönetilen nesne

Bu, bu sınıfın bir sonraki nesnesidir (belirli bir sırada), bu nesneden aynı bağlantı başvuruuna sahip olmalıdır. Başlangıç değeri sıfır.

### açma seçenekleri

Bir nesne açıldığında geçerli olan seçenekler. İlk değer MQOO\_SORGULAMADA olur. Uygun değerleri ayarlamaya ilişkin iki yöntem vardır:

1. Açık seçenekleri ayarlamaz ve açık yöntemi kullanmaz. IBM MQ , açık seçenekleri otomatik olarak ayarlar ve otomatik olarak açılır, yeniden açar ve nesnelere gerektiği gibi kapatır. Bu, gereksiz yeniden açma işlemleriyle sonuçlanabilir; IBM MQ , openFor yöntemini kullanır ve açık seçenekleri yalnızca artımsal olarak ekler.
2. Bir MQI çağrısına neden olan yöntemleri kullanmadan önce açık seçenekleri ayarlayın (bkz. [“C++ ve MQI çapraz başvurusu” sayfa 1760](#) ). Bu, gereksiz yeniden açma işlemlerinin gerçekleşmemesini sağlar. Olası yeniden açma sorunlarından herhangi biri gerçekleşecekse açık seçenekleri açık bir şekilde ayarlayın (bkz. [Yeniden Aç](#) ).

Açık yöntemi kullanırsanız, önce açık seçeneklerin uygun olduğundan emin olmanız gerekir. Ancak, açık yöntemi kullanmak zorunlu değildir; IBM MQ yine de, 1. vakada aynı davranışı sergiler, ancak bu durumda davranış verimlidir.

Sıfır geçerli bir değer değil; nesneyi açma girişiminde bulunmadan önce uygun değeri ayarlayın. Bu işlem, **setOpenOptions** (*IOpenSeçenekleri*) ve onu izleyen **open** () ya da **openFor** (*IRequiredOpenOption*) kullanılarak yapılabilir.

### Not:

1. Bir dağıtım listesi için **open** yöntemi sırasında MQOO\_OUTPUT yerine MQOO\_OUTPUT yerine kondu; MQOO\_OUTPUT, şu anda geçerli olan tek **open option** . Ancak, **open** yöntemini kullanan uygulama programlarında MQOO\_OUTPUT seçeneğini belirttik olarak ayarlamak her zaman iyi bir uygulamadır.
2. Sınıfın **resolved queue manager name** ve **resolved queue name** özniteliklerini kullanmak istiyorsanız MQOO\_RESOLVE\_NAMES değerini belirtin.

#### **açık durum**

Nesnenin açık (TRUE) ya da kapalı (FALSE) olup olmadığı. Başlangıç değeri FALSE (Yanlış) değeridir. Bu öznitelik salt okunurdur.

#### **önceki yönetilen nesne**

Bu sınıfın önceki nesnesi, belirli bir sırada, bu nesneden aynı bağlantı başvuruuna sahip olmalıdır. Başlangıç değeri sıfır.

#### **kuyruk-yönetici-tanıtıcısı**

Kuyruk yöneticisi tanıtıcısı. Bu öznitelik salt okunurdur.

## **Oluşturucular**

#### **ImqObject();**

Varsayılan oluşturucu.

#### **ImqObject(const ImqObject & object );**

Kopya oluşturucusu. Açık durum FALSE (Yanlış) olur.

## **Sınıf yöntemleri (genel)**

#### **statik MQUZE davranışı ();**

Davranışı döndürür.

#### **void setBehavior(const MQlong davranış = 0);**

Davranışı ayarlar.

## **Nesne yöntemleri (genel)**

#### **void işleci = (const ImqObject & object );**

Gerekirse bir kapatma gerçekleştirir ve eşgörünüm verilerini *nesne*' den kopyalar. Açık durum FALSE (Yanlış) olur.

#### **ImqBoolean alterationDate( ImqString & tarih );**

Değiştirme tarihinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

#### **ImqString alterationDate();**

Olası hatalara ilişkin herhangi bir belirti olmadan değiştirme tarihini döndürür.

#### **ImqBoolean alterationTime( ImqString ve time );**

Değiştirme saatinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

#### **ImqString alterationTime();**

Olası hatalara ilişkin herhangi bir belirti olmadan değiştirme süresini döndürür.

#### **ImqString alternateUserId () const;**

Diğer kullanıcı kimliğinin bir kopyasını döndürür.

#### **ImqBoolean setAlternateUserId (const char \* id );**

Diğer kullanıcı kimliğini ayarlar. Diğer kullanıcı kimliği yalnızca, açık durumu FALSE (Yanlış) olduğunda ayarlanabilir. Bu yöntem başarılı olursa TRUE değerini döndürür.

#### **ImqBinary alternateSecurityId () const;**

Diğer güvenlik tanıtıcısının bir kopyasını döndürür.

#### **ImqBoolean setAlternateSecurityId(const ImqBinary & simge );**

Diğer güvenlik tanıtıcısını ayarlar. Diğer güvenlik tanıtıcısı yalnızca, açık durumu FALSE (Yanlış) olduğunda ayarlanabilir. *simge* veri uzunluğunun sıfır ya da MQ\_SECURITY\_ID\_LENGTH değeri olması gerekir. Başarılı olursa TRUE değerini döndürür.



**ImqBoolean setAlternateSecurityId(const MQBYTE\* *simge* = 0);**

Diğer güvenlik tanıtıcısını ayarlar. *simge* sıfır olabilir; bu, MQSID\_NONE belirtilerek aynı olur. *simge* sıfır değilse, bu, MQ\_SECURITY\_ID\_LENGTH baytlık ikili verileri ele almalıdır. MQSID\_NONE gibi önceden tanımlanmış değerleri kullanırken, imza eşleştirmesini sağlamak için bir tip dönüştürme yapmanız gerekebilir; örneğin, (MQBYTE \*) MQSID\_NONE.

Diğer güvenlik tanıtıcısı yalnızca, açık durum TRUE olduğunda ayarlanabilir. Başarılı olursa TRUE değerini döndürür.

**ImqBoolean setAlternateSecurityId(const unsigned char \* *id* = 0);**

Diğer güvenlik tanıtıcısını ayarlar.

**ImqBoolean close ();**

Açık durumu FALSE olarak ayarlar. Başarılı olursa TRUE değerini döndürür.

**MQlong closeOptions () const;**

Kapatma seçeneklerini döndürür.

**void setCloseSeçenekleri (const MQHOMET *seçenekleri* );**

Kapatma seçeneklerini belirler.

**ImqQueueManager \* connectionReference () const;**

Bağlantı başvurusunu döndürür.

**void setConnectionReferansı ( ImqQueueManager & *yöneticisi* );**

Bağlantı başvurusunu ayarlar.

**void setConnectionReferansı ( ImqQueueManager \* *yönetici* = 0);**

Bağlantı başvurusunu ayarlar.

**sanal ImqBoolean tanımlaması ( ImqString & *description* ) = 0;**

Tanımın bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**ImqString açıklaması ();**

Olası hatalara ilişkin herhangi bir belirti göstermeden, tanımın bir kopyasını döndürür.

**sanal ImqBoolean adı ( ImqString & *ad* );**

Adın bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**ImqString adı ();**

Olası hatalara ilişkin herhangi bir belirti göstermeden, adın bir kopyasını döndürür.

**ImqBoolean setName (const char \* *ad* = 0);**

Adı belirler. Ad yalnızca FALSE olduğunda ve ImqQueueManager için bağlantı durumu FALSE olduğunda bu ad ayarlanabilir. Başarılı olursa TRUE değerini döndürür.

**ImqObject \* nextManagedNesnesi () const;**

Sonraki yönetilen nesneyi döndürür.

**ImqBoolean open ();**

Açık durumu gereken şekilde açarak açık durumu TRUE olarak değiştirir; diğer öznitelikler arasında, açık seçeneklerin ve adın kullanılması gerekir. This method uses the connection reference information and the ImqQueueManager connect method if necessary to ensure that the ImqQueueManager connection status is TRUE. Açık durumu döndürür.

**ImqBoolean openFor (const MQlong *gerekli-seçenekler* = 0);**

Nesnenin açık seçeneklerle ya da *gerekli-seçenekler* parametre değerinin belirttiği davranışı garanti eden açık seçeneklerle açık olmasını sağlamaya çalışır.

*requires-options* (zorunlu seçenekler) sıfırsa, giriş gereklidir ve giriş seçeneği yeterli olur. Bu nedenle, açık seçenekler önceden aşağıdakilerden birini içeriyorsa:

- MQOO\_INPUT\_AS\_Q\_DEF
- MQOO\_INPUT\_SHARED
- MQOO\_INPUT\_EXCLUSIVE

açık seçenekler zaten tatmin edici ve değişmiyor; eğer açık seçenekler zaten bu seçeneklerden herhangi birini içermiyorsa, açık seçeneklerde MQOO\_INPUT\_AS\_Q\_DEF ayarlıdır.

*Zorunlu-seçenekler* sıfırda, gerekli seçenekler açık seçeneklere eklenir. *gerekli-seçenekler* bu seçeneklerden herhangi biri değilse, diğerleri sıfırlanır.

Açık seçeneklerden herhangi biri değiştirilirse ve nesne zaten açıksa, açık seçenekleri ayarlamak için nesne geçici olarak kapatılır ve yeniden açılır.

Başarılı olursa TRUE değerini döndürür. Başarı, nesnenin uygun seçeneklerle açık olduğunu gösterir.

#### **MQHOT openOptions () const;**

Açma seçeneklerini döndürür.

#### **ImqBoolean setOpenSeçenekleri (const MQHOMET seçenekleri );**

Açık seçenekleri belirler. Açık olan seçenekler yalnızca açık durum FALSE olduğunda ayarlanabilir. Başarılı olursa TRUE değerini döndürür.

#### **ImqBoolean openStatus () const;**

Açık durumu döndürür.

#### **ImqObject \* previousManagedNesnesi () const;**

Önceki yönetilen nesneyi döndürür.

#### **ImqBoolean queueManagerTanıtıcısı ( ImqString & id );**

Kuyruk yöneticisi tanıtıcısının bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

#### **ImqString queueManagerTanıtıcısı ();**

Olası hatalara ilişkin herhangi bir belirti olmadan kuyruk yöneticisi tanıtıcısını döndürür.

### **Nesne yöntemleri (korunan)**

#### **sanal ImqBoolean closeTemporarily ();**

Yeniden açmadan önce bir nesneyi güvenli bir şekilde kapatır. Başarılı olursa TRUE değerini döndürür. Bu yöntem, açık durumun TRUE olduğunu varsayar.

#### **MQHCONN connectionHandle () const;**

Bağlantı başvurusuyla ilişkili MQHCONN değerini döndürür. Herhangi bir bağlantı başvurusu yoksa ya da Yönetici bağlı değilse, bu değer sıfır olur.

#### **ImqBoolean sorgula (const MQlong int-attr, MQORTO & değer );**

Bir tamsayı değeri döndürür; bunun dizini bir MQIA\_ \* değeri olur. Hata durumunda, değer MQIAV\_UNDEFINED olarak ayarlanır.

#### **ImqBoolean sorgula (const MQlong char-attr, char \* & buffer, const size\_t length );**

Bir MQCA\_ \* değeri olan bir karakter dizisi döndürür.

**Not:** Bu yöntemlerin her ikisi de yalnızca tek bir öznitelik değeri döndürür. Bir *anlık görüntü* birden çok değer gerektiriyorsa, değerler bir anında birbiriyle tutarlı olduğu durumlarda, IBM MQ C++ bu olanağı sağlamaz ve MQINQ çağırısını uygun parametrelerle kullanmanız gerekir.

#### **virtual void openInformationDasperse ();**

MQOPED çağırısından hemen sonra MQOD veri yapısının değişken bölümündeki bilgileri dağıtır.

#### **sanal ImqBoolean openInformationPrepare ();**

MQOD veri yapısının değişken kısmına ilişkin bilgileri MQOPED çağırısından hemen önce hazırlar ve başarılı olursa TRUE değerini döndürür.

#### **ImqBoolean set (const MQORT int-attr, const MQHOMET değer );**

Bir IBM MQ tamsayı özniteliğini ayarlar.

#### **ImqBoolean set (const MQlong char-attr, const char \* buffer, const size\_t requirems-length );**

Bir IBM MQ karakter özniteliğini ayarlar.

#### **void setNextManagedObject (const ImqObject \* object = 0);**

Sonraki yönetilen nesneyi ayarlar.

Dikkat: Bu işlevi, yönetilen nesne listesinin bozulmayacağından eminseniz kullanın.

#### **void setPreviousManagedObject (const ImqObject \* object = 0);**

Önceki yönetilen nesneyi ayarlar.

Dikkat: Bu işlevi, yönetilen nesne listesinin bozulmayacağından eminseniz kullanın.

## Nesne verileri (korunan)

### **MQHOBj** *ohobj*

IBM MQ nesne tanıtıcısı (yalnızca açık durum TRUE olduğunda geçerlidir).

### **MQOD** *omqod*

Gömülü MQOD veri yapısı. Bu veri yapısı için ayrılan depolama miktarı, MQOD Sürüm 2 için gereklidir. (*omqod.Version*) numaralı sürüm numarasını inceleyin ve diğer alanlara aşağıdaki gibi erişin:

#### **MQOD\_VERSION\_1**

*omqod* içinde yer alan diğer tüm alanlara erişilebilir.

#### **MQOD\_VERSION\_2**

*omqod* içinde yer alan diğer tüm alanlara erişilebilir.

#### **MQOD\_VERSION\_3**

*omqod.pmqod* , dinamik olarak ayrılmış, daha büyük bir MQOD ' ye gösterici bir işaretidir. *omqod* içinde başka alan erişilemez. *omqod.pmqod* ile adreslenen tüm alanlara erişilebilir.

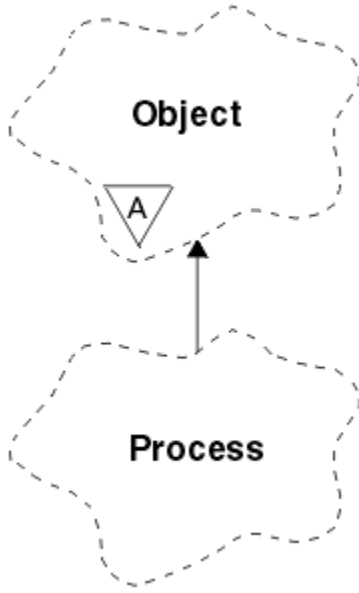
**Not:** *omqod.pmqod.Version* , *omqod.Version* değerinden küçük olabilir; bu, IBM MQ MQI client ' in IBM MQ sunucusundan daha fazla işleve sahip olduğunu gösterir.

## Neden kodları

- MQRC\_ATTRIBUTE\_LOCKED
- MQRC\_INCONSISTENT\_OBJECT\_STATE
- MQRC\_NO\_CONNECTION\_REFERENCE
- MQRC\_STORAGE\_NOT\_AVAM
- MQRC\_REOPEN\_SAVED\_CONTEXT\_ERR
- (MQCLOSE ' in neden kodları)
- (MQCONN ' den neden kodları)
- (MQINQ ' un neden kodları)
- (MQOPEN ' dan neden kodları var)
- (MQSET ' ten neden kodları)

## ImqProcess C++ sınıfı

Bu sınıf, bir tetikleme izleme programı tarafından tetiklenebilecek bir uygulama sürecini (MQOT\_PROCESS tipinde bir IBM MQ nesnesi) sarmalıyor.



Şekil 30. *ImqProcess* sınıfı

- “Nesne öznitelikleri” sayfa 1824
- “Oluşturucular” sayfa 1824
- “Nesne yöntemleri (genel)” sayfa 1824

## Nesne öznitelikleri

### Uygulama tanıtıcısı

Uygulama işleminin tanıtıcısı. Bu öznitelik salt okunurdur.

### Uygulama Tipi

Uygulama işleminin tipi. Bu öznitelik salt okunurdur.

### ortam verileri

Sürece ilişkin ortam bilgileri. Bu öznitelik salt okunurdur.

### Kullanıcı verileri

Sürece ilişkin kullanıcı verileri. Bu öznitelik salt okunurdur.

## Oluşturucular

### **ImqProcess();**

Varsayılan oluşturucu.

### **ImqProcess( const ImqProcess & process );**

Kopya oluşturucusu. **ImqObject open status** , FALSE (Yanlış).

### **ImqProcess( const char \* ad );**

**ImqObject adını**ayarlar.

## Nesne yöntemleri (genel)

### **void işleç = ( const ImqProcess & process );**

Gerekirse bir kapatma gerçekleştirir ve eşgörünüm verilerini *süreçten*kopyalar. **ImqObject open status** , FALSE olur.

### **ImqBoolean applicationId ( ImqString & id );**

**Uygulama kimliği'** nin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

### **ImqString applicationId ( );**

Olası hatalara ilişkin herhangi bir belirti olmadan **uygulama tanıtıcısı** değerini döndürür.

**ImqBoolean applicationType ( MQORT & tip );**

**Uygulama tipi'** nin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQUZE applicationType ();**

Olası hatalara ilişkin herhangi bir belirti olmadan **uygulama tipini** döndürür.

**ImqBoolean environmentData ( ImqString & data );**

**Ortam verileri'** nin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**ImqString environmentData ();**

Olası hatalara ilişkin herhangi bir belirti olmadan **ortam verilerini** döndürür.

**ImqBoolean userData ( ImqString & data );**

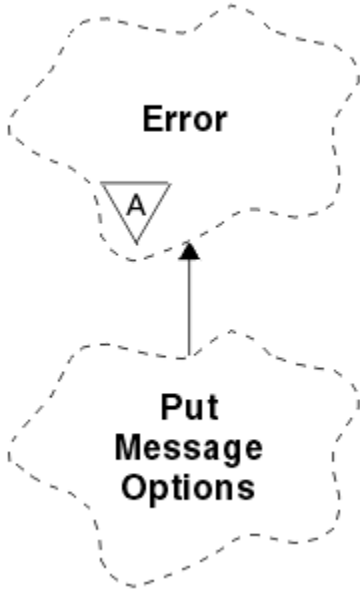
**Kullanıcı verileri'** nin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**ImqString userData ();**

Olası hatalara ilişkin herhangi bir belirti olmaksızın **kullanıcı verilerini** döndürür.

## ImqPutMessageOptions C++ sınıfı

Bu sınıf MQPMO veri yapısını sarsalıyor.



Şekil 31. ImqPutMessageOptions sınıfı

- [“Nesne öznitelikleri” sayfa 1825](#)
- [“Oluşturucular” sayfa 1826](#)
- [“Nesne yöntemleri \(genel\)” sayfa 1826](#)
- [“Nesne verileri \(korunan\)” sayfa 1827](#)
- [“Neden kodları” sayfa 1827](#)

### Nesne öznitelikleri

#### bağlam başvurusu

İletiler için bağlam sağlayan bir ImqQueue . Başlangıçta başvuru yok.

#### seçenekler

Put iletisi seçenekleri. İlk değer MQPMO\_NONE olur. Aşağıdaki ek değerler kullanılabilir:

- MQPMO\_SYNCPOINT
- MQPMO\_NO\_SYNCPOINT
- MQPMO\_NEW\_MSG\_ID

- MQPMO\_NEW\_CORREL\_ID
- MQPMO\_LOGICAL\_ORDER
- MQPMO\_NO\_BAĞLAMı
- MQPMO\_DEFAULT\_CONTEXT
- MQPMO\_PASS\_IDENTITY\_CONTEXT
- MQPMO\_PASS\_ALL\_CONTEXT
- MQPMO\_SET\_IDENTITY\_CONTEXT
- MQPMO\_SET\_ALL\_CONTEXT
- MQPMO\_ALTERNATE\_USER\_AUTHORITY
- MQPMO\_FAIL\_IF QUIESCING

### **kayıt alanları**

Bir ileti konduğunda ileti kayıtlarının eklenmesini denetleyen işaretler. İlk değer MQPMRF\_NONE olur. Aşağıdaki ek değerler kullanılabilir:

- MQPMRF\_MSG\_ID
- MQPMRF\_CORREL\_ID
- MQPMRF\_GROUP\_ID
- MQPMRF\_FEEDBACK
- MQPMRF\_ACCOUNTING\_TOKEN

ImqMessageTracker öznitelikleri, belirtilen herhangi bir alan için nesneden alınır. ImqMessageİzleyici öznitelikleri, belirtilmeyen herhangi bir alan için ImqMessage nesnesinden alınır.

### **çözümlenen kuyruk yöneticisi adı**

Put sırasında belirlenen hedef kuyruk yöneticisinin adı. Başlangıç değeri boş değerli. Bu öznitelik salt okunurdur.

### **çözümlenen kuyruk adı**

Put sırasında belirlenen hedef kuyruğun adı. Başlangıç değeri boş değerli. Bu öznitelik salt okunurdur.

### **syncpoint katılımı**

İletiler syncpoint denetimi altına konduğunda TRUE (Doğru).

## **Oluşturucular**

### **ImqPutMessageOptions( );**

Varsayılan oluşturucu.

### **ImqPutMessageOptions(const ImqPutMessageOptions & pmo );**

Kopya oluşturucusu.

## **Nesne yöntemleri (genel)**

### **void operator = (const ImqPutMessageOptions & pmo );**

Var olan eşgörünüm verilerini değiştirerek, yönetim ortamı verilerini *Pmo*' den kopyalar.

### **ImqQueue \* contextReference () const;**

Bağlam başvurusunu döndürür.

### **void setContextBaşvuru (const ImqQueue & queue );**

Bağlam başvurusunu ayarlar.

### **void setContextBaşvuru (const ImqQueue \* queue = 0);**

Bağlam başvurusunu ayarlar.

### **MQTHOE seçenekleri () const;**

Seçenekleri döndürür.

### **void setOptions (const MQHOMET seçenekleri );**

Syncpoint katılım değeri de dahil olmak üzere seçenekleri belirler.

**MQlong recordFields () const;**

Kayıt alanlarını döndürür.

**void setRecordFields (const MQHOT alanları );**

Kayıt alanlarını ayarlar.

**ImqString resolvedQueueManagerName () const;**

Çözülen kuyruk yöneticisi adının bir kopyasını döndürür.

**ImqString resolvedQueueAdı () const;**

Çözülen kuyruk adının bir kopyasını döndürür.

**ImqBoolean syncPointKatılım () const;**

Seçenekler MQPMO\_SYNCPOINT içeriliyorsa, TRUE olan syncpoint katılım değerini döndürür.

**void setSyncPointParticipation (const ImqBoolean sync );**

Syncpoint katılım değerini ayarlar. *Eşitle* TRUE ise, seçenekler MQPMO\_SYNCPOINT ögesini içerecek şekilde değiştirilir ve MQPMO\_NO\_SYNCPOINT ögesini dışlamak için değiştirilir. *sync* FALSE ise, seçenekler MQPMO\_NO\_SYNCPOINT içerecek şekilde değiştirilir ve MQPMO\_SYNCPOINT ögesini dışlamak için değiştirilir.

**Nesne verileri (korunan)****MQPMO omqpmo**

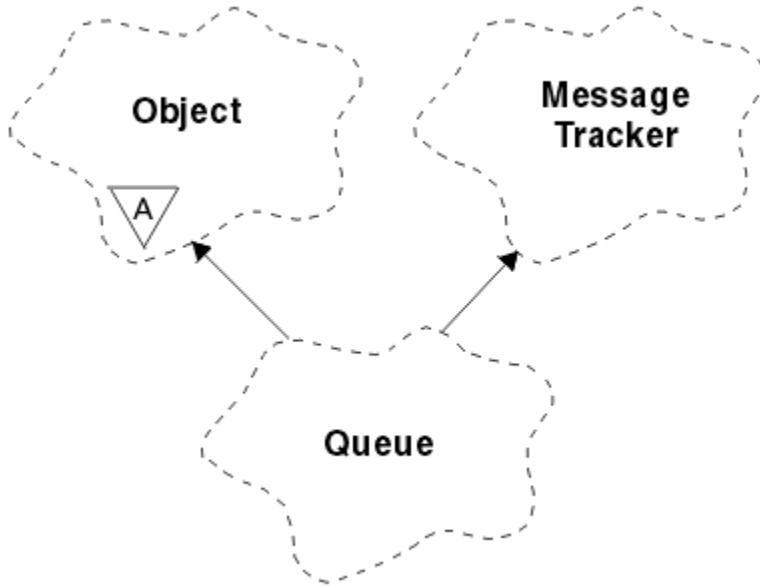
MQPMO veri yapısı.

**Neden kodları**

- MQRC\_STORAGE\_NOT\_AVAM

**ImqQueue C++ sınıfı**

Bu sınıf bir ileti kuyruğunu sarsalıyor (MQOT\_Q tipinde bir IBM MQ nesnesi).



Şekil 32. ImqQueue sınıfı

Bu sınıf, Çizelge 863 sayfa 1768’inde listelenen MQI çağrılarını ile ilgilidir.

- “Nesne öznitelikleri” sayfa 1828
- “Oluşturucular” sayfa 1830
- “Nesne yöntemleri (genel)” sayfa 1831
- “Nesne yöntemleri (korunan)” sayfa 1837

- [“Neden kodları” sayfa 1837](#)

## **Nesne öznitelikleri**

### **backout requesue adı**

Gereğinden fazla geri alma isteği adı. Bu öznitelik salt okunurdur.

### **Geriletme eşiği**

Geriletme eşiği. Bu öznitelik salt okunurdur.

### **temel kuyruk adı**

Diğer adın çözümlendiği kuyruğun adı. Bu öznitelik salt okunurdur.

### **Küme adı**

Küme adı. Bu öznitelik salt okunurdur.

### **küme adı listesi adı**

Küme adı listesi adı. Bu öznitelik salt okunurdur.

### **küme iş yükü sırası**

Küme iş yükü sırası. Bu öznitelik salt okunurdur.

### **küme iş yükü önceliği**

Küme iş yükü önceliği. Bu öznitelik salt okunurdur.

### **küme iş yükü kullanım kuyruğu**

Küme iş yükü kullanım kuyruğu değeri. Bu öznitelik salt okunurdur.

### **yaratma tarihi**

Kuyruk yaratma verileri. Bu öznitelik salt okunurdur.

### **Oluşturulma zamanı**

Kuyruk yaratma zamanı. Bu öznitelik salt okunurdur.

### **yürürlükteki derinlik**

Kuyruktaki ileti sayısı. Bu öznitelik salt okunurdur.

### **varsayılan bağlama**

Varsayılan bağ tanımlama. Bu öznitelik salt okunurdur.

### **varsayılan giriş açma seçeneği**

Giriş için varsayılan açma seçeneği. Bu öznitelik salt okunurdur.

### **varsayılan kalıcılık**

Varsayılan ileti kalıcılığı. Bu öznitelik salt okunurdur.

### **varsayılan öncelik**

Varsayılan ileti önceliği. Bu öznitelik salt okunurdur.

### **Tanımlama Tipi**

Kuyruk tanımlaması tipi. Bu öznitelik salt okunurdur.

### **derinlik yüksek olayı**

Kuyruk derinliği yüksek olayları için denetim özniteliği. Bu öznitelik salt okunurdur.

### **derinlik üst sınırı**

Kuyruk derinliği üst sınırı. Bu öznitelik salt okunurdur.

### **derinlik düşük olay**

Kuyruk derinliği düşük olaylarına ilişkin denetim özniteliği. Bu öznitelik salt okunurdur.

### **derinlik alt sınırı**

Kuyruk derinliği için alt sınır. Bu öznitelik salt okunurdur.

### **derinlik üst sınırı olayı**

Kuyruk derinliği olayları üst sınırı için denetim özniteliği. Bu öznitelik salt okunurdur.

### **dağıtım listesi başvurusu**

İletileri birden çok kuyruğa dağıtmak için kullanılacak bir ImqDistributionListesi için isteğe bağlı başvuru, bu liste de içinde olmak üzere. Başlangıç değeri boş değerli.



**Not:** Bir ImqQueue nesnesi açıldığında, başvurduğu açık ImqDistributionListe nesnesi otomatik olarak kapatılır.

### **dağıtım listeleri**

Dağıtım listelerini desteklemek için bir iletim kuyruğu yeteneği. Bu öznitelik salt okunurdur.

### **dinamik kuyruk adı**

Dinamik kuyruk adı. İlk değer AMQ.\* değeridir. Tüm AIX, Linux, and Windows platformları için.

### **harden geri tepme**

Geri sayım için harden de olsun. Bu öznitelik salt okunurdur.

### **dizin tipi**

Dizin tipi. Bu öznitelik salt okunurdur.

### **alma engelini**

Alma işlemlerine izin verilir verilmediği. Başlangıç değeri, kuyruk tanımlamasına bağlıdır. Bu öznitelik yalnızca bir diğer ad ya da yerel kuyruk için geçerlidir.

### **engelleyici**

Koyma işlemlerine izin verilir verilmediği. Başlangıç değeri, kuyruk tanımlamasına bağlıdır.

### **başlatma kuyruğu adı**

Başlatma kuyruğunun adı. Bu öznitelik salt okunurdur.

### **derinlik üst sınırı**

Kuyruktan izin verilen ileti sayısı üst sınırı. Bu öznitelik salt okunurdur.

### **ileti uzunluğu üst sınırı**

Bu kuyruktaki herhangi bir ileti için uzunluk üst sınırı; bu, ilişkili kuyruk yöneticisi tarafından yönetilen herhangi bir kuyruk için üst sınırdan az olabilir. Bu öznitelik salt okunurdur.

### **ileti teslim sırası**

İleti önceliğinin ilgili olup olmadığı. Bu öznitelik salt okunurdur.

### **sonraki dağıtılmış kuyruk**

Bu sınıfın bir sonraki nesnesi, belirli bir sırada, bu nesne olarak aynı **dağıtım listesi başvurusuna** sahip olmak. Başlangıç değeri sıfır.

Zincirdeki bir nesne silinirse, önceki nesnede ve sonraki nesne, dağıtılmış kuyruk bağlantılarının silinen nesneyi artık göstermeyecek şekilde güncellenmektedir.

### **kalıcı olmayan ileti sınıfı**

Bu kuyruğa koyulan kalıcı olmayan iletiler için güvenilirlik düzeyi. Bu öznitelik salt okunurdur.

### **açık giriş sayısı**

Giriş için açık olan ImqQueue nesnelerinin sayısı. Bu öznitelik salt okunurdur.

### **açık çıkış sayısı**

Çıkış için açık olan ImqQueue nesneleri sayısı. Bu öznitelik salt okunurdur.

### **önceki dağıtılmış kuyruk**

Bu sınıfın önceki nesnesi (belirli bir düzende), bu nesne olarak aynı **dağıtım listesi başvurusuna** sahip olmalıdır. Başlangıç değeri sıfır.

Zincirdeki bir nesne silinirse, önceki nesnede ve sonraki nesne, dağıtılmış kuyruk bağlantılarının silinen nesneyi artık göstermeyecek şekilde güncellenmektedir.

### **İşlem adı**

Süreç tanımlamasının adı. Bu öznitelik salt okunurdur.

### **kuyruk muhasebesi**

Kuyruklara ilişkin muhasebe bilgilerinin düzeyi. Bu öznitelik salt okunurdur.

### **kuyruk-yönetici-adi**

Kuyruğun bulunduğu kuyruk yöneticisinin adı (büyük olasılıkla uzak). Burada adı belirtilen kuyruk yöneticisini ImqObject **bağlantı başvurusu** ile karıştırmayın; bu, (yerel) bir bağlantı sağlayan (yerel) kuyruk yöneticisine gönderme yapar. Başlangıç değeri boş değerli.

### **kuyruk izleme**

Kuyruğa ilişkin izleme verileri toplamının düzeyi. Bu öznitelik salt okunurdur.

**kuyruk istatistikleri**

Kuyruğa ilişkin istatistik verileri düzeyi. Bu öznitelik salt okunurdur.

**kuyruk tipi**

Kuyruk tipi. Bu öznitelik salt okunurdur.

**uzak kuyruk yöneticisi adı**

Uzak kuyruk yöneticisinin adı. Bu öznitelik salt okunurdur.

**uzak kuyruk adı**

Uzak kuyruğun uzak kuyruk yöneticisinde bulunduğu gibi, uzak kuyruk adı. Bu öznitelik salt okunurdur.

**çözümlenen kuyruk yöneticisi adı**

Çözülmüş kuyruk yöneticisi adı. Bu öznitelik salt okunurdur.

**çözümlenen kuyruk adı**

Çözülmüş kuyruk adı. Bu öznitelik salt okunurdur.

**alıkoyma aralığı**

Kuyruk alıkoyma aralığı. Bu öznitelik salt okunurdur.

**kapsam**

Kuyruk tanımlamasının kapsamı. Bu öznitelik salt okunurdur.

**hizmet aralığı**

Hizmet aralığı. Bu öznitelik salt okunurdur.

**hizmet aralığı olayı**

Hizmet aralığı olayları için denetim özniteliği. Bu öznitelik salt okunurdur.

**paylaşılabilirlik**

Kuyruğun paylaşılabilirlik paylaşamayacağı. Bu öznitelik salt okunurdur.

**depolama sınıfı**

Depolama sınıfı. Bu öznitelik salt okunurdur.

**iletim kuyruğu adı**

İletim kuyruğunun adı. Bu öznitelik salt okunurdur.

**tetikleyici denetimi**

Tetik kontrolü. Başlangıçtaki değer kuyruk tanımlamasına bağlıdır. Bu öznitelik yalnızca yerel bir kuyruk için geçerlidir.

**tetikleyici verileri**

Verileri tetikler. Başlangıçtaki değer kuyruk tanımlamasına bağlıdır. Bu öznitelik yalnızca yerel bir kuyruk için geçerlidir.

**tetikleyici derinliği**

Tetik derinliği. Başlangıçtaki değer kuyruk tanımlamasına bağlıdır. Bu öznitelik yalnızca yerel bir kuyruk için geçerlidir.

**tetikleyici ileti önceliği**

Tetikleyiciler için eşik iletisi önceliği. Başlangıçtaki değer kuyruk tanımlamasına bağlıdır. Bu öznitelik yalnızca yerel bir kuyruk için geçerlidir.

**Tetikleme Tipi**

Tetikleyici tipi. Başlangıçtaki değer kuyruk tanımlamasına bağlıdır. Bu öznitelik yalnızca yerel bir kuyruk için geçerlidir.

**Kullanım**

Kullanım. Bu öznitelik salt okunurdur.

**Oluşturucular****ImqQueue();**

Varsayılan oluşturucu.

**ImqQueue( const ImqQueue & queue );**

Kopya oluşturucusu. ImqObject **open status** , FALSE olur.

**ImqQueue( const char \* ad );**  
ImqObject adınıayarlar.

## Nesne yöntemleri (genel)

**void işleç = ( const ImqQueue & queue );**

Gerekirse bir kapatma gerçekleştirir ve eşgörünüm verilerini *kuyruktankopyalar*. ImqObject **open status** , FALSE olur.

**ImqBoolean backoutRequeueName ( ImqString & ad );**

**backout require name**ögesinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**ImqString backoutRequeueAd ();**

Olası hatalara ilişkin herhangi bir belirti olmadan **backout requeue name** değerini döndürür.

**ImqBoolean backoutThreshold ( MQORT & eşik );**

**Geri çıkış eşiği**' nin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQUZE backoutThreshold ();**

Olası hatalara ilişkin herhangi bir belirti olmadan **backout threshold** değerini döndürür.

**ImqBoolean baseQueueName ( ImqString & ad );**

**Temel kuyruk adı**' nin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**ImqString baseQueueName ();**

Olası hatalara ilişkin herhangi bir belirti olmadan **temel kuyruk adı** değerini döndürür.

**ImqBoolean clusterName( ImqString & ad );**

**Küme adı**' nin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**ImqString clusterName();**

Olası hatalara ilişkin herhangi bir belirti olmadan **küme adını** döndürür.

**ImqBoolean clusterNamelistAdı ( ImqString & ad );**

**Küme adı listesi adı**' nun bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**ImqString clusterNamelistName ();**

Herhangi bir hata belirtisi olmadan **küme adilistesi adını** döndürür.

**ImqBoolean clusterWorkLoadPriority (MQOLD & priority);**

Küme iş yükü önceliği değerinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQlong clusterWorkLoadPriority ();**

Olası hatalara ilişkin herhangi bir belirti olmadan küme iş yükü önceliği değerini döndürür.

**ImqBoolean clusterWorkLoadRank (MQOLD & rank);**

Küme iş yükü sıra değerinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQlong clusterWorkLoadRank ();**

Olası hatalara ilişkin herhangi bir belirti olmadan küme iş yükü sıra değerini döndürür.

**ImqBoolean clusterWorkLoadUseQ (MQUBY & useq);**

Küme iş yükü kullanım kuyruğu değerini içeren bir kopya sağlar. Başarılı olursa TRUE değerini döndürür.

**MQlong clusterWorkLoadUseQ ();**

Olası hatalara ilişkin herhangi bir belirti olmadan küme iş yükü kullanım kuyruğu değerini döndürür.

**ImqBoolean creationDate ( ImqString & date );**

**Yaratma tarihi**' nin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**ImqString creationDate ();**

Olası hatalara ilişkin herhangi bir belirti olmadan **creation date** (yaratma tarihi) değerini döndürür.

**ImqBoolean creationTime ( ImqString & time );**

**Yaratma süresi**' nin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**ImqString creationTime ();**

Olası hatalara ilişkin herhangi bir belirti olmadan **creation time** (yaratma zamanı) değerini döndürür.

**ImqBoolean currentDepth ( MQORT & derinlik );**

**Geçerli derinlik**' in bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQLONG currentDepth ();**

Olası hatalara ilişkin herhangi bir belirti olmadan **yürürlükteki derinlik** değerini döndürür.

**ImqBoolean defaultInputOpenOption ( MQORT & seçeneği );**

**Varsayılan giriş açık seçeneği**' nin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQLONG defaultInputOpenOption ();**

Olası hatalara ilişkin herhangi bir belirti olmadan **varsayılan giriş açma seçeneğini** döndürür.

**ImqBoolean defaultPersistence ( MQLong & persistence );**

**Varsayılan kalıcılık**' in bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQLONG defaultPersistence ();**

Olası hatalara ilişkin herhangi bir belirti olmadan **varsayılan kalıcılık** değerini döndürür.

**ImqBoolean defaultPriority ( MQORT & priority );**

**Varsayılan öncelik**' in bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQLONG defaultPriority ();**

Olası hatalara ilişkin herhangi bir belirti olmadan **varsayılan öncelik** değerini döndürür.

**ImqBoolean defaultBind ( MQORT & bind );**

**Varsayılan bağlama**' nin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQUZT defaultBind ();**

Olası hatalara ilişkin herhangi bir belirti olmadan **default bind** değerini döndürür.

**ImqBoolean definitionType ( MQORT & tip );**

**Tanım tipi**' nin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQLONG definitionType ();**

Olası hatalara ilişkin herhangi bir belirti olmadan **definition type** (tanımlama tipi) değerini döndürür.

**ImqBoolean depthHighEvent ( MQORT & olay );**

**derinlik yüksek etkinliği**' nin etkinleştirme durumunun bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQLONG depthHighOlayı ();**

Olası hatalara ilişkin herhangi bir belirti olmadan **derinlik yüksek olayı** ' nin etkinleştirme durumunu döndürür.

**ImqBoolean depthHighLimit ( MQLong & limit );**

**derinlik yüksek sınırı**' nin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQUZT depthHighLimit ();**

Olası hatalara ilişkin herhangi bir belirti olmadan **derinlik üst sınırı** değerini döndürür.

**ImqBoolean depthLowOlayı ( MQBOLD & olay );**

Provides a copy of the enablement state of the **derinlik düşük olay**. Başarılı olursa TRUE değerini döndürür.

**MQLONG depthLowOlayı ();**

Olası hatalara ilişkin herhangi bir belirti olmadan **derinlik düşük olayı** ' nin etkinleştirme durumunu döndürür.

**ImqBoolean depthLowLimit ( MQORT & sınır );**

**derinlik alt sınırı**' nin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQLONG depthLowSınır ();**

Olası hatalara ilişkin herhangi bir belirti olmadan **derinlik düşük sınırı** değerini döndürür.

**ImqBoolean depthMaximumEvent ( MQORT & olay );**

Provides a copy of the enablement state of the **derinlik üst sınırı olayı**. Başarılı olursa TRUE değerini döndürür.

**MQLONG depthMaximumOlayı ();**

Olası hatalara ilişkin herhangi bir belirti olmaksızın, **derinlik üst sınırı olayı** ' in etkinleştirme durumunu döndürür.

**ImqDistributionList \* distributionListReference () const ;**

**Dağıtım listesi başvurusunu** döndürür.

**void setDistributionListReference ( ImqDistributionList & list );**

**Dağıtım listesi başvurusunu**ayarlar.

**void setDistributionListReference ( ImqDistributionList \* list = 0);**

**Dağıtım listesi başvurusunu**ayarlar.

**ImqBoolean distributionLists ( MQORT & destek );**

**Dağıtım listeleri** değerinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQlong distributionLists ();**

Olası hatalara ilişkin herhangi bir belirti olmadan **dağıtım listeleri** değerini döndürür.

**ImqBoolean setDistributionListeleri ( const MQlong support );**

**Dağıtım listeleri** değerini ayarlar. Başarılı olursa TRUE değerini döndürür.

**ImqString dynamicQueueAd () const ;**

**Dinamik kuyruk adı'** nun bir kopyasını döndürür.

**ImqBoolean setDynamicQueueName ( const char \* ad );**

**Dinamik kuyruk adı'** nin adını belirler. **Dinamik kuyruk adı** yalnızca ImqObject **open status** değeri FALSE olduğunda ayarlanabilir. Başarılı olursa TRUE değerini döndürür.

**ImqBoolean get ( ImqMessage & msg, ImqGetMessageOptions & options );**

Belirtilen *seçenekler'* i kullanarak kuyruktan bir ileti alır. Invokes the ImqObject **openFor** method if necessary to ensure that the ImqObject **açma seçenekleri** include either one of the MQOO\_INPUT\_\* values, or the MQOO\_BROWSE value, depending on the *seçenekler*. *msg* nesnesinde ImqCache **otomatik arabellek** varsa, arabellek alınan tüm iletileri barındıracak şekilde büyür. The **clearMessage** method is invoked against the *It* object before retrieval.

Bu yöntem başarılı olursa TRUE değerini döndürür.

**Not:** Bu **neden kodu** uyarı olarak sınıflandırılrsa da, ImqObject **neden kodu**

MQRC\_TRUNCATED\_MSG\_FAILED ise, yöntem çağırımı sonucu FALSE olur. Kesilen bir ileti kabul edildiyse, ImqCache **ileti uzunluğu** kısaltılmış uzunluğa yansıtır. Her iki olayda da, ImqMessage **toplam ileti uzunluğu** , kullanılabilir byte sayısını gösterir.

**ImqBoolean get ( ImqMessage & msg );**

Önceki yönteme ilişkin olarak, varsayılan get (varsayılan alma) ileti seçenekleri kullanılır.

**ImqBoolean get ( ImqMessage & msg, ImqGetMessageOptions & options, const size\_t arabellek-size );**

Önceki iki yöntemde olduğu gibi, geçersiz kılma değeri *arabellek-büüklüğü* belirtilmektedir. *msg* nesnesi bir ImqCache **otomatik arabellek** çalıştırıyorsa, ileti alınmadan önce *msg* nesnesinde **resizeBuffer** yöntemi çağırılır ve arabellek daha büyük bir iletiyi sığdırmak için daha fazla büyümeyiz.

**ImqBoolean get ( ImqMessage & msg, const size\_t buffer-size );**

Önceki yönteme ilişkin olarak, varsayılan get (varsayılan alma) ileti seçenekleri kullanılır.

**ImqBoolean hardenGetBackout ( MQORT & harden );**

**Harden get backout** (Harden get backout) değerinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQlong hardenGetGeri Al ();**

Olası hatalara ilişkin herhangi bir belirti olmadan **harden get backout** değerini döndürür.

**ImqBoolean indexType(MQBOLD & tip );**

**Dizin tipi'** nin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQLONG indexType();**

Olası hatalara ilişkin herhangi bir belirti olmadan **dizin tipini** döndürür.

**ImqBoolean inhibitGet ( MQORT & engelleyci );**

**Inhibit get** değerinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQLONG inhibitGet ();**

Olası hatalara ilişkin herhangi bir belirti olmadan **inhibit get** değerini döndürür.

**ImqBoolean setInhibitGet ( const MQlong inhibit );**

**Inhibit get** değerini ayarlar. Başarılı olursa TRUE değerini döndürür.

**ImqBoolean inhibitPut ( MQORT & engelleyici );**  
**Inhibit put** değerinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQLONG inhibitPut ();**  
Olası hatalara ilişkin herhangi bir belirti olmadan **inhibit put** değerini döndürür.

**ImqBoolean setInhibitCut ( const MQlong inhibit );**  
**Inhibit put** değerini ayarlar. Başarılı olursa TRUE değerini döndürür.

**ImqBoolean initiationQueueAdı ( ImqString & ad );**  
**Kullanıma hazırlama kuyruğu adı'** nin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**ImqString initiationQueueAd ();**  
Olası hatalara ilişkin herhangi bir belirti olmadan **başlangıç kuyruğu adını** döndürür.

**ImqBoolean maximumDepth ( MQlong & derinlik );**  
**derinlik üst sınırı'** nin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQlong maximumDepth ();**  
Olası hataların herhangi bir göstergesi olmadan **derinlik üst sınırı** değerini döndürür.

**ImqBoolean maximumMessageLength ( MQORT & uzunluk );**  
**İleti uzunluğu üst sınırı'** nin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQlong maximumMessageLength ();**  
Olası hatalara ilişkin herhangi bir belirti olmadan **ileti uzunluğu üst sınırı** değerini döndürür.

**ImqBoolean messageDeliverySequence ( MQlong & sequence );**  
**İleti teslim sırası'** nin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQUQT messageDeliverySırası ();**  
Olası hatalara ilişkin herhangi bir belirti olmadan **ileti teslimi sırası** değerini döndürür.

**ImqQueue \* nextDistributedKuyruk () const ;**  
**Sonraki dağıtılmış kuyruk** değerini döndürür.

**ImqBoolean nonPersistentMessageClass (MQUBE & monq);**  
Kalıcı olmayan ileti sınıfı değerinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQLONG nonPersistentMessageClass ();**  
Kalıcı olmayan ileti sınıfı değerini, olası hatalara ilişkin herhangi bir gösterge olmadan döndürür.

**ImqBoolean openInputCount ( MQORT & sayı );**  
**Açık giriş sayısı'** nin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQHOT openInputCount ();**  
Olası hataların herhangi bir göstergesi olmadan **open input count** (giriş sayısını aç) değerini döndürür.

**ImqBoolean openOutputCount ( MQLONG & sayı );**  
**Açık çıkış sayısı'** nin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQHOT openOutputCount ();**  
Olası hatalara ilişkin herhangi bir belirti olmadan **open output count** değerini döndürür.

**ImqQueue \* previousDistributedKuyruk () const ;**  
**Önceki dağıtılmış kuyruğu** döndürür.

**ImqBoolean processName ( ImqString & ad );**  
**Süreç adı'** nin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**ImqString processName ();**  
Olası hatalara ilişkin herhangi bir belirti olmadan **işlem adını** döndürür.

**ImqBoolean put ( ImqMessage & msg );**  
Varsayılan koyma iletisi seçeneklerini kullanarak, kuyruğa ileti yerleştirir. Uses the ImqObject **openFor** method if necessary to ensure that the ImqObject **açma seçenekleri** include MQOO\_OUTPUT.  
Bu yöntem başarılı olursa TRUE değerini döndürür.

**ImqBoolean put ( ImqMessage & msg, ImqPutMessageOptions & pmo );**  
Belirtilen *pmo*' yı kullanarak kuyruğa bir ileti yerleştirir. Uses the ImqObject **openFor** method as necessary to ensure that the ImqObject **açma seçenekleri** include MQOO\_OUTPUT, and (if the *Pmo* **seçenekler** include any of MQPMO\_PASS\_IDENTITY\_CONTEXT,

MQPMO\_PASS\_ALL\_CONTEXT, MQPMO\_SET\_IDENTITY\_CONTEXT, or MQPMO\_SET\_ALL\_CONTEXT) corresponding MQOO\_\*\_CONTEXT values.

Bu yöntem başarılı olursa TRUE değerini döndürür.

**Not:** *pmo* , bir **bağlam başvurusu** içeriyorsa, bir bağlam sağlamak için başvuru nesne açılır.

**ImqBoolean queueAccounting (MQUZO & accktq);**

Kuyruk hesap değerinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQLong queueAccounting ();**

Olası hatalara ilişkin herhangi bir belirti olmadan kuyruk muhasebesi değerini döndürür.

**ImqString queueManagerAd () const ;**

**kuyruk yöneticisi adı** değerini döndürür.

**ImqBoolean setQueueManagerName ( const char \* ad );**

**Kuyruk yöneticisi adını** ayarlar. **Kuyruk yöneticisi adı** yalnızca ImqObject **open status** değeri FALSE olduğunda ayarlanabilir. Bu yöntem başarılı olursa TRUE değerini döndürür.

**ImqBoolean queueMonitoring (MQUUR & monq);**

Kuyruk izleme değerinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQLONG queueMonitoring ();**

Olası hatalara ilişkin herhangi bir belirti olmadan kuyruk izleme değerini döndürür.

**ImqBoolean queueStatistics (MQOLD & statq);**

Kuyruk istatistiği değerinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQHOT queueStatistics ();**

Olası hatalara ilişkin herhangi bir belirti olmadan kuyruk istatistikleri değerini döndürür.

**ImqBoolean queueType ( MQORT & tip );**

**Kuyruk tipi** değerinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQLONG queueType ();**

Olası hatalara ilişkin herhangi bir belirti olmadan **kuyruk tipini** döndürür.

**ImqBoolean remoteQueueManagerName ( ImqString & ad );**

**Uzak kuyruk yöneticisi adı**' nun bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**ImqString remoteQueueManagerName ( );**

Olası hatalara ilişkin herhangi bir belirti olmadan **uzak kuyruk yöneticisi adını** döndürür.

**ImqBoolean remoteQueueName ( ImqString & ad );**

**Uzak kuyruk adı**' nun bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**ImqString remoteQueueAd ();**

Olası hatalara ilişkin herhangi bir belirti olmadan **uzak kuyruk adı** değerini döndürür.

**ImqBoolean resolvedQueueManagerName( ImqString & ad );**

**çözülen kuyruk yöneticisi adı**' nın bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**Not:** MQOO\_RESOLVE\_NAMES ImqObject **açma seçenekleri** arasında yer almıyorsa, bu yöntem başarısız olur.

**ImqString resolvedQueueManagerName( );**

Olası hatalara ilişkin herhangi bir belirti olmadan **çözümlenen kuyruk yöneticisi adı** değerini döndürür.

**ImqBoolean resolvedQueueAd( ImqString & ad );**

**Çözümlenen kuyruk adı**' nun bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**Not:** MQOO\_RESOLVE\_NAMES ImqObject **açma seçenekleri** arasında yer almıyorsa, bu yöntem başarısız olur.

**ImqString resolvedQueueName ();**

Olası hatalara ilişkin herhangi bir belirti olmadan **çözülen kuyruk adı** değerini döndürür.

**ImqBoolean retentionInterval ( MQLONG & aralık );**

**Alıkoyma aralığı**' nın bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQlong retentionInterval ();**

Olası hatalara ilişkin herhangi bir belirti olmadan **alıkoyma aralığı** değerini döndürür.

**ImqBoolean scope ( MQORT & kapsam );**

**Kapsam**' in bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQUZN kapsam ();**

Olası hatalara ilişkin herhangi bir belirti olmadan **scope** ögesini döndürür.

**ImqBoolean serviceInterval ( MQORT & aralık );**

**Hizmet aralığı**' nin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQUZE serviceInterval ();**

Olası hatalara ilişkin herhangi bir belirti olmadan **hizmet aralığı** değerini döndürür.

**ImqBoolean serviceIntervalEvent ( MQORT & olay );**

**Hizmet aralığı olayı**' nin etkinleştirme durumunun bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQUZE serviceIntervalOlayı ();**

Olası hatalara ilişkin herhangi bir belirti olmadan **hizmet aralığı olayının** etkinleştirme durumunu döndürür.

**ImqBoolean paylaşılabilirlik ( MQORT & paylaşılabilirlik );**

**paylaşılabilirlik** değerinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQHOT paylaşılabilirlik ();**

Olası hatalara ilişkin herhangi bir belirti olmadan **sharebility** değerini döndürür.

**ImqBoolean storageClass( ImqString & class );**

**storage class**' in bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**ImqString storageClass();**

Olası hatalara ilişkin herhangi bir belirti olmadan **storage class** (depolama sınıfı) değerini döndürür.

**ImqBoolean transmissionQueueAdı ( ImqString & ad );**

**İletim kuyruğu adı**' nin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**ImqString transmissionQueueAd ();**

Olası hatalara ilişkin herhangi bir belirti olmadan **iletim kuyruğu adı** değerini döndürür.

**ImqBoolean triggerControl ( MQORT & denetimi );**

**Tetikleme denetimi** değerinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQUZE triggerControl ();**

Olası hatalara ilişkin herhangi bir belirti olmadan **tetikleme denetimi** değerini döndürür.

**ImqBoolean setTriggerControl ( const MQlong control );**

**Tetikleme denetimi** değerini ayarlar. Başarılı olursa TRUE değerini döndürür.

**ImqBoolean triggerData ( ImqString & data );**

**tetikleyici verileri**' nin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**ImqString triggerData ( );**

Olası hatalara ilişkin herhangi bir belirti olmaksızın, **tetikleyici verileri** ' in bir kopyasını döndürür.

**ImqBoolean setTriggerData ( const char \* data );**

**Tetikleme verilerini** ayarlar. Başarılı olursa TRUE değerini döndürür.

**ImqBoolean triggerDepth ( MQORT & derinlik );**

**Tetikleme derinliği**' nin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQUZE triggerDepth ();**

Olası hatalara ilişkin herhangi bir belirti olmadan **tetikleyici derinliğini** döndürür.

**ImqBoolean setTriggerDerinlik ( const MQlong derinlik );**

**Tetikleme derinliği**' ne ayarlar. Başarılı olursa TRUE değerini döndürür.

**ImqBoolean triggerMessagePriority ( MQORT & priority );**

**Tetikleme iletisi önceliği**' nin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQLONG triggerMessageÖnceliği ();**

Olası hatalara ilişkin herhangi bir belirti olmadan **tetikleyici ileti önceliği** değerini döndürür.



**ImqBoolean setTriggerMessagePriority ( const MQlong priority );**

**İleti önceliğini tetikle** seçeneğini belirler. Başarılı olursa TRUE değerini döndürür.

**ImqBoolean triggerType ( MQORT & tip );**

**Tetikleme tipi'** nin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQLONG triggerType ();**

Olası hatalara ilişkin herhangi bir belirti olmadan **tetikleme tipini** döndürür.

**ImqBoolean setTriggerType ( const MQlong tip );**

**Tetikleme tipini** belirler. Başarılı olursa TRUE değerini döndürür.

**ImqBoolean usage ( MQBOLD & kullanımı );**

**Kullanım** değerinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQUZE kullanımı ();**

Olası hatalara ilişkin herhangi bir belirti olmadan **usage** değerini döndürür.

**Nesne yöntemleri (korunan)****void setNextDistributedQueue ( ImqQueue \* queue = 0);**

**Sonraki dağıtılmış kuyruğu** ayarlar.

**Dikkat:** Bu işlevi yalnızca, dağıtılmış kuyruk listesini bozmayacağından eminseniz kullanın.

**void setPreviousDistributedQueue ( ImqQueue \* queue = 0);**

**Önceki dağıtılmış kuyruğu** ayarlar.

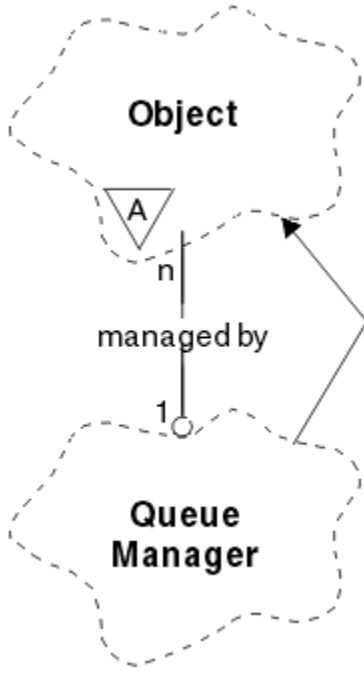
**Dikkat:** Bu işlevi yalnızca, dağıtılmış kuyruk listesini bozmayacağından eminseniz kullanın.

**Neden kodları**

- MQRC\_ATTRIBUTE\_LOCKED
- MQRC\_CONTEXT\_OBJECT\_NOT\_VALID
- MQRC\_CONTEXT\_OPEN\_ERROR
- MQRC\_CURSOR\_NOT\_VALID
- MQRC\_NO\_BUFFER
- MQRC\_REOPEN\_EXCL\_INPUT\_ERROR
- MQRC\_REOPEN\_INQUIRE\_ERROR
- MQRC\_REOPEN\_TEMPORARY\_Q\_ERROR
- (MQGET ' den neden kodları)
- (MQPUT ' un neden kodları)

**ImqQueueManager C++ sınıfı**

Bu sınıf bir kuyruk yöneticisini sarmalıyor (MQOT\_Q\_MGR tipinde bir IBM MQ nesnesi).



Şekil 33. *ImqQueueManager* sınıfı

Bu sınıf, “[ImqQueueManager çapraz başvurusu](#)” sayfa 1770’inde listelenen MQI çağrılarını ile ilgilidir. Listelenen tüm yöntemler tüm altyapılar için geçerli değildir; ek ayrıntılar için [ALTER QMGR](#) konusuna bakın.

- “Sınıf öznitelikleri” sayfa 1838
- “Nesne öznitelikleri” sayfa 1839
- “Oluşturucular” sayfa 1844
- “Yok ediciler” sayfa 1844
- “Sınıf yöntemleri (genel)” sayfa 1844
- “Nesne yöntemleri (genel)” sayfa 1844
- “Nesne yöntemleri (korunan)” sayfa 1853
- “Nesne verileri (korunan)” sayfa 1853
- “Neden kodları” sayfa 1853

## Sınıf öznitelikleri

### davranış

Örtülü bağlantının ve bağlantının kesilmesinin davranışını denetler.

#### **IMQ\_EXPL\_DISC\_BACKOUT (0L)**

Bağlantı kesme yöntemine açık bir çağrı, geriletme yöntemini belirtir. Bu öznitelik IMQ\_EXPL\_DISC\_COMMIT ile karşılıklı olarak birbirini dışlar.

#### **IMQ\_EXPL\_DISC\_COMMIT (1L)**

Bağlantı kesme yöntemine ilişkin belirttik bir çağrı kesinleştirme anlamına gelir (varsayılan değer). Bu öznitelik IMQ\_EXPL\_DISC\_BACKOUT ile karşılıklı olarak birbirini dışlar.

#### **IMQ\_IMPL\_CONN (2L)**

Örtük bağlantıya izin verilir (varsayılan değer).

#### **IMQ\_IMPL\_DISC\_BACKOUT (0L)**

Nesne yok etme sırasında oluşabilecek bağlantı kesme yöntemine yönelik örtük bir çağrı, geriletme anlamına gelir. Bu öznitelik, IMQ\_IMPL\_DISC\_COMMIT ile karşılıklı olarak birbirini dışlar.

## **IMQ\_IMPL\_DISC\_COMMIT (4L)**

Nesne imhası sırasında oluşabilecek bağlantı kesme yöntemine yönelik örtük bir çağrı, kesinleştirmeyi belirtir (varsayılan). Bu öznitelik IMQ\_IMPL\_DISC\_BACKUT ile karşılıklı olarak birbirini dışlar.

At IBM MQ V7.0 and above, C++ applications that make use of an implicit connection, need to specify IMQ\_IMPL\_CONN along with any other options provided in the `setBehavior()` method on an object of class `ImqQueueManager`. Uygulamanız, davranış seçeneklerini belirttik olarak ayarlamak için `setBehavior()` yöntemini kullanmıyorsa, örneğin,

```
ImqQueueManager_object.setBehavior(IMQ_IMPL_DISC_COMMIT)
```

Bu değişiklik, varsayılan olarak MQ\_IMPL\_CONN etkinleştirildiğinden bu değişiklik sizi etkilemez.

Uygulamanız belirttik olarak davranış seçeneklerini ayarlarsa, örneğin,

```
ImqQueueManager_object.setBehavior(IMQ_IMPL_DISC_COMMIT)
```

Uygulamanızın örtük bir bağlantıyı tamamlamasına izin vermek için, `setBehavior()` yönteminde IMQ\_IMPL\_CONN 'yi aşağıdaki gibi eklemeniz gerekir:

```
ImqQueueManager_object.setBehavior(IMQ_IMPL_CONN | IMQ_IMPL_DISC_COMMIT)
```

## **Nesne öznitelikleri**

### **muhasabe bağlantıları geçersiz kılma**

Uygulamaların MQI muhasabe ve kuyruk hesabı values.This özniteliğinin ayarını geçersiz kılmasına izin verir.

### **muhasabe aralığı**

Ara muhasabe kayıtlarının yazıldığı süre (saniye cinsinden). Bu öznitelik salt okunurdur.

### **etkinlik kaydı**

Etkinlik raporlarının oluşturulmasını denetler. Bu öznitelik salt okunurdur.

### **yeni mca denetimi kabul et**

Bir MCA 'nın, önceden etkin bir MCA ile aynı adı taşıyan bir yeni gelen kanalı algılandığında, MCA 'nın kabul edilmesi gerekip gerekmediğini saptamak için denetlenen öğeler denetlenir. Bu öznitelik salt okunurdur.

### **yeni mca tipini benimseme**

Yeni bir kanal tipindeki bir MCA örneğinin artık yeni bir gelen kanal isteği ile eşleştiğinde otomatik olarak yeniden başlatılıp başlatılmayacağını, yeni mca denetimi değıştirtgeleri algılandığında otomatik olarak yeniden başlatılmalıdır. Bu öznitelik salt okunurdur.

### **Kimlik Doğrulama Tipi**

Gerçekleştirilmekte olan kimlik doğrulamasının tipini belirtir.

### **yetki olayı**

Yetki olaylarını denetler. Bu öznitelik salt okunurdur.

### **başlangıç seçenekleri**

Başlangıç yöntemi için geçerli olan seçenekler. İlk değer MQBO\_NONE olur.

### **köprü olayı**

IMS köprüsü olaylarının oluşturulup oluşturulmayacağı. Bu öznitelik salt okunurdur.

### **kanal otomatik tanımı**

Kanal otomatik tanımlama değeri. Bu öznitelik salt okunurdur.

### **kanal otomatik tanımlama olayı**

Kanal otomatik tanımlama olay değeri. Bu öznitelik salt okunurdur.

### **kanal otomatik tanımlama çıkışı**

Kanal otomatik tanımlama çıkış adı. Bu öznitelik salt okunurdur.

**kanal olayı**

Kanal olaylarının oluşturulup oluşturulmayacağı. Bu öznitelik salt okunurdu.

**kanal başlatıcı bağdaştırıcıları**

IBM MQ çağrılarını işlemek için kullanılacak bağdaştırıcı alt görevlerinin sayısı. Bu öznitelik salt okunurdu.

**kanal başlatıcı denetimi**

Kuyruk Yöneticisi başlatıldığında Kanal Başlatıcı 'nın otomatik olarak başlatılıp başlatılmayacağını belirleyin. Bu öznitelik salt okunurdu.

**kanal başlatan dağıtıcıları**

Kanal başlatıcısı için kullanılacak dağıtıcıların sayısı. Bu öznitelik salt okunurdu.

**kanal başlatıcı izleme otomatik başlatma**

Kanal başlatıcı izinin otomatik olarak başlamalı ya da başlatılmayacağı. Bu öznitelik salt okunurdu.

**kanal başlatıcı izleme çizelgesi büyüklüğü**

Kanal başlatıcısının izleme verileri alanının büyüklüğü (MB). Bu öznitelik salt okunurdu.

**kanal izleme**

Kanallar için çevrimiçi izleme verilerinin toplanmasını denetler. Bu öznitelik salt okunurdu.

**kanal başvurusu**

İstemci bağlantısı sırasında kullanılacak kanal tanımlamasına yönelik başvuru. Bağlıken, bu öznitelik boş değere ayarlanabilir, ancak başka bir değere çevrilemiyor. Başlangıç değeri boş değerli.

**kanal istatistikleri**

Kanallar için istatistik verilerinin toplanmasını denetler. Bu öznitelik salt okunurdu.

**karakter kümesi**

Kodlanmış karakter takımı tanıtıcısı (CCSID). Bu öznitelik salt okunurdu.

**küme gönderen izleme**

Otomatik olarak tanımlanmış küme gönderen kanallarına ilişkin çevrimiçi izleme verilerinin toplanmasını denetler. Bu öznitelik salt okunurdu.

**küme gönderen istatistikleri**

Otomatik olarak tanımlanmış küme gönderen kanallarına ilişkin istatistik verileri toplamının denetlenmesini sağlar. Bu öznitelik salt okunurdu.

**küme iş yükü verileri**

Küme iş yükü çıkış verileri. Bu öznitelik salt okunurdu.

**küme iş yükü çıkışı**

Küme iş yükü çıkış adı. Bu öznitelik salt okunurdu.

**küme iş yükü uzunluğu**

Küme iş yükü uzunluğu. Bu öznitelik salt okunurdu.

**küme iş yükü mru**

Küme iş yükü en son kullanılan kanallar değeri. Bu öznitelik salt okunurdu.

**küme iş yükü kullanım kuyruğu**

Küme iş yükü kullanım kuyruğu değeri. Bu öznitelik salt okunurdu.

**komut olayı**

Komut olaylarının oluşturulup oluşturulmayacağı. Bu öznitelik salt okunurdu.

**komut giriş kuyruğu adı**

Sistem komutu giriş kuyruğu adı. Bu öznitelik salt okunurdu.

**komut düzeyi**

Kuyruk yöneticisi tarafından desteklenen komut düzeyi. Bu öznitelik salt okunurdu.

**komut sunucusu denetimi**

Kuyruk Yöneticisi başlatıldığında Komut Sunucusu 'nın da otomatik olarak başlatılıp başlatılmayacağını belirleyin. Bu öznitelik salt okunurdu.

**bağlantı seçenekleri**

Bağlanma yöntemi için geçerli olan seçenekler. Başlangıçtaki değer MQCNO\_NONE olur. Platforma bağlı olarak aşağıdaki ek değerler mümkün olabilir:

- MQCNO\_STANDARD\_BINDING
- MQCNO\_FASTPATH\_BINDING
- MQCNO\_HANDLE\_SHARE\_NONE
- MQCNO\_HANDLE\_SHARE\_BLOCK
- MQCNO\_HANDLE\_SHARE\_NO\_BLOCK
- MQCNO\_SERIALIZE\_CONN\_TAG\_Q\_MGR
- MQCNO\_SERIALIZE\_CONN\_TAG\_QSG
- MQCNO\_RESTRICT\_CONN\_TAG\_Q\_MGR
- MQCNO\_RESTRICT\_CONN\_TAG\_QSG

#### **Bağlantı Tanıtıcısı**

Bir uygulamayı güvenilir olarak tanımlamak için MQ ' ya izin veren benzersiz bir tanıtıcı.

#### **Bağlantı durumu**

Kuyruk yöneticisine bağlıyken TRUE (Doğru). Bu öznitelik salt okunurdur.

#### **bağlantı etiketi**

Bağlantıyla ilişkilendirilecek bir etiket. Bu öznitelik yalnızca bağlantı kurulmadığında ayarlanabilir. Başlangıç değeri boş değerli.

#### **şifreleme donanımı**

Şifreleme donanımı için yapılandırma ayrıntıları. MQ MQI istemci bağlantıları için.

#### **-Ölü harf kuyruğu adı**

Ölü-mektup kuyruğunun adı. Bu öznitelik salt okunurdur.

#### **varsayılan iletim kuyruğu adı**

Varsayılan iletim kuyruğu adı. Bu öznitelik salt okunurdur.

#### **dağıtım listeleri**

Dağıtım listelerini desteklemek için kuyruk yöneticisinin yeteneği.

#### **dns grubu**

Kuyruk paylaşım grubu için gelen iletimleri işleyen TCP dinleyicisinin, İş Yüğü Yöneticisi Dinamik Etki Alanı Adı Hizmetleri desteği kullanılırken katılması gereken grubun adı. Bu öznitelik salt okunurdur.

#### **dns wlm**

Kuyruk paylaşım grubu için gelen iletimleri işleyen TCP dinleyicisinin, Dinamik Etki Alanı Adı Hizmetleri için İş Yüğü Yöneticisi ile kaydolması gerekip gerekmediğini belirleyin. Bu öznitelik salt okunurdur.

#### **ilk kimlik doğrulama kaydı**

The first of one or more objects of class ImqAuthenticationRecord, in no particular order, in which the ImqAuthenticationRecord connection reference addresses this object. MQ MQI istemci bağlantıları için.

#### **ilk yönetilen nesne**

The first of one or more objects of class ImqObject, in no particular order, in which the ImqObject connection reference addresses this object. Başlangıç değeri sıfır.

#### **etkinliği engelle**

Olayları engelleyici olayları denetler. Bu öznitelik salt okunurdur.

#### **ip adresi sürümü**

Kanal bağlantısı için kullanılacak IP protokolü (IPv4 ya da IPv6). Bu öznitelik salt okunurdur.

#### **anahtar havuzu**

Anahtarların ve sertifikaların saklandığı anahtar veri tabanı dosyasının yeri. IBM MQ MQI client bağlantıları için.

#### **anahtar sıfırlama sayısı**

Gizli anahtar yeniden anlaşılmadan önce bir TLS iletişimde gönderilen ve alınan şifrelenmemiş bayt sayısı. Bu öznitelik yalnızca MQCONNX kullanan istemci bağlantıları için geçerlidir. Ayrıca bkz. [ssl](#) anahtarı sıfırlama sayısı.

**dinleyici zamanlayıcısı**

Bir APPC ya da TCP/IP hatası varsa, dinleyiciyi yeniden başlatmak için IBM MQ ile denemeler arasındaki zaman aralığı (saniye olarak). Bu öznitelik salt okunurdur.

**yerel olay**

Yerel olayları denetler. Bu öznitelik salt okunurdur.

**günlüğe kaydetme olayı**

Kurtarma günlüğü olaylarının oluşturulup oluşturulmayacağını denetler. Bu öznitelik salt okunurdur.

**lu grup adı**

Kuyruk paylaşım grubu için gelen iletimleri işleyen LU 6.2 dinleyicisinin kullanması gereken soysal LU adı. Bu öznitelik salt okunurdur.

**lu adı**

Giden LU 6.2 iletimleri için kullanılacak LU ' nun adı. Bu öznitelik salt okunurdur.

**lu62 kol soneki**

SYS1.PARMLIB üyesi APPCPMxx, bu kanal başlatıcısı için LUADD ' yi (LUADD) atamaz. Bu öznitelik salt okunurdur.

**lu62 kanalları**

LU 6.2 iletim protokolünü kullanan, bağlanabilecek en çok kanal sayısı ya da istemci sayısı üst sınırı. Bu öznitelik salt okunurdur.

**etkin kanal üst sınırı**

Herhangi bir zamanda etkin olabilecek kanal sayısı üst sınırı. Bu öznitelik salt okunurdur.

**kanal üst sınırı**

Güncel olabilecek kanal sayısı üst sınırı (bağlı istemcilerle sunucu bağlantısı kanalları da içinde olmak üzere). Bu öznitelik salt okunurdur.

**işleyici üst sınırı**

Tanıtıcı sayısı üst sınırı. Bu öznitelik salt okunurdur.

**ileti uzunluğu üst sınırı**

Bu kuyruk yöneticisi tarafından yönetilen kuyruktaki herhangi bir ileti için olası uzunluk üst sınırı. Bu öznitelik salt okunurdur.

**en yüksek öncelik**

İleti önceliği üst sınırı. Bu öznitelik salt okunurdur.

**kesinleştirilmemiş iletiler üst sınırı**

Bir birim ya da iş içinde kesinleştirilmemiş ileti sayısı üst sınırı. Bu öznitelik salt okunurdur.

**mqi muhasebe**

MQI verilerine ilişkin muhasebe bilgilerinin toplanmasını denetler. Bu öznitelik salt okunurdur.

**mqi istatistikleri**

Kuyruk yöneticisine ilişkin istatistik izleme bilgilerinin derlemine denetler. Bu öznitelik salt okunurdur.

**giden kapı üst sınırı**

Giden kanalların bağ tanımlanırken kullanılacak kapı numaraları aralığının daha üst bir ucu. Bu öznitelik salt okunurdur.

**giden kapı alt sınırı**

Giden kanallar için bağ tanımlanırken kullanılacak kapı numaraları aralığının alt ucu. Bu öznitelik salt okunurdur.

**parola**

kullanıcı kimliği ile ilişkili parola

**başarım olayı**

Performans olaylarını denetler. Bu öznitelik salt okunurdur.

**Platform**

Kuyruk yöneticisinin bulunduğu altyapı. Bu öznitelik salt okunurdur.

**kuyruk muhasebesi**

Kuyruklar için muhasebe bilgilerinin toplanmasını denetler. Bu öznitelik salt okunurdur.

**kuyruk izleme**

Kuyruklar için çevrimiçi izleme verilerinin toplanmasını denetler. Bu öznelik salt okunurdu.

**kuyruk istatistikleri**

Kuyruklar için istatistik verilerinin toplanmasını denetler. Bu öznelik salt okunurdu.

**alma zamanaşımı**

Bir TCP/IP ileti kanalının, etkin olmayan duruma dönmeden önce iş ortasından kalp atışları da içinde olmak üzere verileri almak için ne kadar bekleyeceđi (yaklaşık olarak). Bu öznelik salt okunurdu.

**alma zamanaşımı alt sınırı**

Bir TCP/IP kanalının, etkin olmayan duruma dönmeden önce iş ortasından kalp atışları da içinde olmak üzere verileri almak için bekleyeceđi minimum süre. Bu öznelik salt okunurdu.

**alma zamanaşımı tipi**

Alma zamanaşımını almak için uygulanan bir niteleyici. Bu öznelik salt okunurdu.

**uzak olay**

Uzak olayları denetler. Bu öznelik salt okunurdu.

**Havuz adı**

Havuz adı. Bu öznelik salt okunurdu.

**havuz ad listesi**

Havuz adı listesi adı. Bu öznelik salt okunurdu.

**paylaşılan kuyruk yöneticisi adı**

Whether MQOPENS of a shared queue where the ObjectQMGrName is another queue manager in the queue sharing group should resolve to an open of the shared queue on the local queue manager. Bu öznelik salt okunurdu.

**ssl olayı**

SSL olaylarının oluşturulup oluşturulmayacağı. Bu öznelik salt okunurdu.

**ssl FIPS**

Whether only FIPS-certified algorithms should be used if the cryptography is executed in IBM MQ software. Bu öznelik salt okunurdu.

**ssl anahtarı sıfırlama sayısı**

Gizli anahtar yeniden anlaşılmadan önce, bir SSL etkileşimi içinde gönderilen ve alınan şifrelenmemiş baytların sayısı. Bu öznelik salt okunurdu.

**start-stop olayı**


Start-stop olaylarını denetler. Bu öznelik salt okunurdu.

**istatistik aralığı**

İstatistik izleme verilerinin izleme kuyruđuna ne sıklıkta yazıldığı. Bu öznelik salt okunurdu.

**syncpoint kullanılabilirliği**

Eşitleme noktası katılımının kullanılabilirliği. Bu öznelik salt okunurdu.

**Not:** Kuyruk yöneticisi-eşgüdümlü genel iş birimleri IBM i altyapısında desteklenmez.  **IBM i**  
\_Rcommit ve \_Rback yerel sistem çağrılarını kullanarak, IBM itarafından dışarıdan eşgüdümlü bir çalışma birimi programlayabilirsiniz. STRCMTCTL komutunu kullanarak, iş düzeyinde kesinleştirme denetimi altında IBM MQ uygulamasını başlatarak bu tip çalışma birimini başlatın. Ek ayrıntılar için IBM i dış eşitleme noktası yöneticisine yönelik arabirimler başlıklı konuya bakın. Geriletme ve kesinleştirme, bir kuyruk yöneticisi tarafından koordine edilen yerel iş birimleri için IBM i platformunda desteklenir.

**tcp kanalları**

TCP/IP iletim protokolünü kullanan, bağlanabilecek, yürürlükteki ya da istemci olabilecek kanal sayısı üst sınırı. Bu öznelik salt okunurdu.

**tcp keepalive**

Bağlantının diğer ucunun hala kullanılabilir olup olmadığını denetlemek için TCP KEEPALIVE olanağının kullanılıp kullanılmayacağını denetleyin. Bu öznelik salt okunurdu.

**tcp adı**

Tcp yığını tipine bağlı olarak, kullanılacak tek ya da varsayılan TCP/IP sisteminin adı. Bu öznitelik salt okunurdur.

**tcp yığın tipi**

Kanal başlatıcının, yalnızca tcp adında belirtilen TCP/IP adres alanını kullanmasına izin verilip verilmediğini ya da seçilen herhangi bir TCP/IP adresine bağ tanımlayıp bağlanmadığını belirleyin. Bu öznitelik salt okunurdur.

**izleme rotası kaydı**

Rota izleme bilgilerinin kaydedilmesini denetler. Bu öznitelik salt okunurdur.

**tetikleme aralığı**

Tetikleme aralığı. Bu öznitelik salt okunurdur.

**kullanıcı kimliği**

AIX and Linux platformlarında, uygulamanın gerçek kullanıcı kimliği. Windows platformlarında, uygulamanın kullanıcı kimliği.

**Oluşturucular****ImqQueueManager ();**

Varsayılan oluşturucu.

**ImqQueueManager (const ImqQueueManager & yönetici);**

Kopya oluşturucusu. Bağlantı durumu FALSE (Yanlış) olur.

**ImqQueueManager (const char \* ad);**

ImqObject adını *name* olarak ayarlar.

**Yok ediciler**

Bir ImqQueueManager nesnesi yok edildiğinde, otomatik olarak kesilir.

**Sınıf yöntemleri (genel)****statik MQUZE davranışı ();**

Davranışı döndürür.

**void setBehavior(const MQlong davranış = 0);**

Davranışı ayarlar.

**Nesne yöntemleri (genel)****void operator = (const ImqQueueManager & mgr);**

Gerekirse bağlantıyı keser ve eşgörünüm verilerini *mgr*' dan kopyalar. Bağlantı durumu FALSE (Yanlış) olur.

**ImqBoolean accountingConnOverride (MQUSED & statint);**

Hesap bağlantıları geçersiz kılma değerinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQHOT accountingConnOverride ();**

Olası hatalara ilişkin herhangi bir belirti olmadan, hesap bağlantıları geçersiz kılma değerini döndürür.

**ImqBoolean accountingInterval (MQUVE & statint);**

Muhasebe aralığı değerinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQLONG accountingInterval ();**

Olası hatalara ilişkin herhangi bir belirti olmadan, hesap aralığı değerini döndürür.

**ImqBoolean activityRecording (MQOLD & rec);**

Etkinlik kaydı değerinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQHOT activityRecording ();**

Olası hatalara ilişkin herhangi bir belirti olmadan etkinlik kaydı değerini döndürür.



**ImqBoolean adoptNewMCACheck (MQBOLD & check);**

Yeni MCA denetim deęerini benimsemenin bir kopyasını saęlar. Başarılı olursa TRUE deęerini döndürür.

**MQlong adoptNewMCACheck ();**

Olası hatalara ilişkin herhangi bir belirti olmadan yeni MCA denetim deęerini kabul eder.

**ImqBoolean adoptNewMCAType (MQBOLD & tip);**

Yeni MCA tipini benimsemenin bir kopyasını saęlar. Başarılı olursa TRUE deęerini döndürür.

**MQlong adoptNewMCAType ();**

Olası hatalara ilişkin herhangi bir belirti olmadan yeni MCA tipini benimseme olanaęı saęlar.

**QXX\_ENCODE\_CASE\_ONE long authenticationType () const;**

Kimlik doęrulama tipini döndürür.

**void setAuthenticationTip (const MQTORT tipi = MQCSP\_AUTH\_NONE);**

Kimlik doęrulama tipini ayarlar.

**ImqBoolean authorityEvent(MQUZE & olay );**

Yetki olayına ilişkin etkinleřtirme durumunun bir kopyasını saęlar. Başarılı olursa TRUE deęerini döndürür.

**MQHOT authorityEvent();**

Olası hatalara ilişkin herhangi bir belirti olmaksızın, yetki olayının etkinleřtirme durumunu döndürür.

**ImqBoolean geriletme ();**

Kesinleřtirilmemiř deęişiklikleri yedekler. Başarılı olursa TRUE deęerini döndürür.

**ImqBoolean begin ();**

Bir çalışma birimi başlatır. Başlangıç seęenekleri, bu yöntemin davranışını etkiler. Başarılı olursa TRUE deęerini döndürür; ancak, temeldeki MQBEGIN çağırısı MQRC\_NO\_EXTERNAL\_PARTICILAR ya da MQRC\_PARTICIPANT\_NOT\_AVAM (her ikisi de MQCC\_UYARI ile ilişkilidir) döndürse de TRUE deęerini döndürür.

**MQlong beginOptions() const;**

Başlangıç seęeneklerini döndürür.

**void setBeginSeęenekleri (const MQlong options = MQBO\_NONE);**

Başlangıç seęeneklerini belirler.

**ImqBoolean bridgeEvent (MQORT & event);**

Köprü olay deęerinin bir kopyasını saęlar. Başarılı olursa TRUE deęerini döndürür.

**MQHOT bridgeEvent ();**

Olası hatalara ilişkin herhangi bir belirti olmadan köprü olayı deęerini döndürür.

**ImqBoolean channelAutoTanımlaması (MQUZE & deęer );**

Kanal otomatik tanımlama deęerinin bir kopyasını saęlar. Başarılı olursa TRUE deęerini döndürür.

**MQOUR channelAutoTanımlaması ();**

Olası hatalara ilişkin herhangi bir belirti olmadan kanal otomatik tanımlama deęerini döndürür.

**ImqBoolean channelAutoDefinitionEvent(MQORTO & deęer );**

Kanal otomatik tanımlama olay deęerinin bir kopyasını saęlar. Başarılı olursa TRUE deęerini döndürür.

**MQHOT channelAutoDefinitionEvent();**

Olası hatalara ilişkin herhangi bir belirti olmadan kanal otomatik tanımlama olayı deęerini döndürür.

**ImqBoolean channelAutoDefinitionExit( ImqString & ad );**

Kanal otomatik tanımlama çıkış adının bir kopyasını saęlar. Başarılı olursa TRUE deęerini döndürür.

**ImqString channelAutoDefinitionExit();**

Olası hatalara ilişkin herhangi bir belirti olmadan kanal otomatik tanımlama çıkış adını döndürür.

**ImqBoolean channelEvent (MQORT & event);**

Kanal olay deęerinin bir kopyasını saęlar. Başarılı olursa TRUE deęerini döndürür.

**MQHOMEN channelEvent();**

Olası hatalara ilişkin herhangi bir belirti olmadan kanal olay deęerini döndürür.

**MQlong channelInitiatorAdapters ();**

Kanal başlatıcı bağdaştırıcılarını, olası hatalara ilişkin herhangi bir gösterge olmadan döndürür.

**ImqBoolean channelInitiatorAdapters (MQLONG & bağdaştırıcılar);**

Kanal başlatıcı bağdaştırıcılarının bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQlong channelInitiatorControl ();**

Olası hatalara ilişkin herhangi bir belirti olmadan kanal başlatıcı başlatma değerini döndürür.

**ImqBoolean channelInitiatorControl (MQORT & init);**

Kanal başlatıcı denetimi başlangıç değerinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQlong channelInitiatorDağıtıcılar ();**

Olası hatalara ilişkin herhangi bir belirti olmadan kanal başlatıcı dağıtıcıları değerini döndürür.

**ImqBoolean channelInitiatorDağıtıcılar (MQORT & dağıtıcılar);**

Kanal başlatıcısı dağıtıcılarının bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQlong channelInitiatorTraceAutoStart ();**

Olası hatalara ilişkin herhangi bir belirti olmadan kanal başlatıcı izleme otomatik başlatma değerini döndürür.

**ImqBoolean channelInitiatorTraceAutoStart (MQHOT & auto);**

Kanal başlatıcı izleme otomatik başlatma değerinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQOUR channelInitiatorTraceTableBoyutu ();**

Olası hatalara ilişkin herhangi bir belirti olmadan kanal başlatıcı izleme çizelgesi büyüklüğü değerini döndürür.

**ImqBoolean channelInitiatorTraceTableBoyutu (MQBOLD & boyut);**

Kanal başlatıcı izleme çizelgesi büyüklüğünün bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**ImqBoolean channelMonitoring (MQUVE & monchl);**

Kanal izleme değerinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQOUR channelMonitoring ();**

Olası hatalara ilişkin herhangi bir belirti olmadan kanal izleme değerini döndürür.

**ImqBoolean channelReference( ImqChannel \* & pchannel );**

Kanal başvurularının bir kopyasını sağlar. Kanal başvurusu geçersizse, *pchannel* 'ı boş değer olarak ayarlar. Bu yöntem başarılı olursa TRUE değerini döndürür.

**ImqChannel \* channelReference( );**

Olası hatalara ilişkin herhangi bir belirti olmadan kanal başvurusunu döndürür.

**ImqBoolean setChannelReferansı ( ImqChannel & kanal );**

Kanal başvurusunu ayarlar. Bu yöntem başarılı olursa TRUE değerini döndürür.

**ImqBoolean setChannelReferansı ( ImqChannel \* kanal = 0 );**

Kanal başvurusunu ayarlar ya da sıfırlar. Bu yöntem başarılı olursa TRUE değerini döndürür.

**ImqBoolean channelStatistics (MQUSED & statchl);**

Kanal istatistiği değerinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQHOT channelStatistics ();**

Olası hatalara ilişkin herhangi bir belirti olmadan kanal istatistik değerini döndürür.

**ImqBoolean characterSet(MQORT & ccsid);**

Karakter kümesinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQlong characterSet();**

Olası hatalara ilişkin herhangi bir belirti olmadan, karakter kümesinin bir kopyasını döndürür.

**MQHOT clientSslKeyResetCount () const;**

İstemci bağlantılarında kullanılan SSL anahtarı sıfırlama sayısı değerini döndürür.

**void setClientSslKeyResetCount(const MQBORT sayısı);**

İstemci bağlantılarında kullanılan SSL anahtarı sıfırlama sayısını ayarlar.

**ImqBoolean clusterSenderİzlemesi (MQUVE & monacIs);**

Küme gönderen izleme varsayılan deęerinin bir kopyasını saęlar. Başarılı olursa TRUE deęerini döndürür.

**MQHOT clusterSenderMonitoring ();**

Olası hatalara ilişkin herhangi bir gösterge olmadan, küme gönderen izleme varsayılan deęerini döndürür.

**ImqBoolean clusterSenderStatistics (MQUVE & statacls);**

Küme gönderen istatistik deęerinin bir kopyasını saęlar. Başarılı olursa TRUE deęerini döndürür.

**MQlong clusterSenderStatistics ();**

Olası hatalara ilişkin herhangi bir gösterge olmadan, küme gönderen istatistik deęerini döndürür.

**ImqBoolean clusterWorkloadVerileri ( ImqString & data );**

Küme iş yükü çıkış verisinin bir kopyasını saęlar. Başarılı olursa TRUE deęerini döndürür.

**ImqString clusterWorkloadData ();**

Olası hatalara ilişkin herhangi bir belirti olmadan küme iş yükü çıkış verilerini döndürür.

**ImqBoolean clusterWorkloadExit ( ImqString & ad );**

Küme iş yükü çıkış adının bir kopyasını saęlar. Başarılı olursa TRUE deęerini döndürür.

**ImqString clusterWorkloadExit ();**

Olası hatalara ilişkin herhangi bir belirti olmadan küme iş yükü çıkış adını döndürür.

**ImqBoolean clusterWorkloadLength (MQBOLD & uzunluk );**

Küme iş yükü uzunluęunun bir kopyasını saęlar. Başarılı olursa TRUE deęerini döndürür.

**MQlong clusterWorkloadLength ();**

Olası hatalara ilişkin herhangi bir belirti olmadan küme iş yükü uzunluęunu döndürür.

**ImqBoolean clusterWorkLoadMRU (MQUBE & mru);**

Küme iş yükünün en son kullanılan kanal deęerinin bir kopyasını saęlar. Başarılı olursa TRUE deęerini döndürür.

**MQUZE clusterWorkLoadMRU ();**

Olası hatalara ilişkin herhangi bir gösterge olmadan, en son kullanılan kanal iş yükünü en son kullanılan kanallar deęerini döndürür.

**ImqBoolean clusterWorkLoadUseQ (MQUBY & useq);**

Küme iş yükü kullanım kuyruęu deęerini içeren bir kopya saęlar. Başarılı olursa TRUE deęerini döndürür.

**MQlong clusterWorkLoadUseQ ();**

Olası hatalara ilişkin herhangi bir belirti olmadan küme iş yükü kullanım kuyruęu deęerini döndürür.

**ImqBoolean commandEvent (MQOLD & event);**

Komut olay deęerinin bir kopyasını saęlar. Başarılı olursa TRUE deęerini döndürür.

**MQlong commandEvent ();**

Olası hatalara ilişkin herhangi bir belirti olmadan komut olay deęerini döndürür.

**ImqBoolean commandInputQueueName( ImqString & ad );**

Komut giriş kuyruęu adının bir kopyasını saęlar. Başarılı olursa TRUE deęerini döndürür.

**ImqString commandInputQueueName( );**

Olası hatalara ilişkin herhangi bir belirti olmadan komut giriş kuyruęu adını döndürür.

**ImqBoolean commandLevel(MQUZE & düzey);**

Komut düzeyinin bir kopyasını saęlar. Başarılı olursa TRUE deęerini döndürür.

**MQUZN commandLevel();**

Olası hatalara ilişkin herhangi bir belirti olmadan komut düzeyini döndürür.

**MQlong commandServerControl ();**

Olası hatalara ilişkin herhangi bir belirti olmadan komut sunucusu başlatma deęerini döndürür.

**ImqBoolean commandServerControl (MQBOLD & server);**

Komut sunucusu denetimi başlangıç deęerinin bir kopyasını saęlar. Başarılı olursa TRUE deęerini döndürür.

**ImqBoolean commit ();**

Kesinleştirilmemiş değişiklikleri kesinleştirin. Başarılı olursa TRUE değerini döndürür.

**ImqBoolean connect ();**

Belirtilen ImqObject adı ile kuyruk yöneticisine bağlanır, varsayılan değer yerel kuyruk yöneticisidir. Belirli bir kuyruk yöneticisine bağlanmak istiyorsanız, bağlantıdan önce ImqObject setName yöntemini kullanın. Bir kanal başvurusu varsa, bu uygulama kanal tanımlamasıyla ilgili bilgileri MQCD ' deki MQCONNX ' e aktarmak için kullanılır. MQCD ' deki ChannelType , MQCHT\_CLNTCONN olarak ayarlıdır. Sunucu bağlantıları için yalnızca anlamlı olan, istemci bağlantıları için anlamlı olan kanal başvuru bilgileri dikkate alınmaz. Bağlanma seçenekleri bu yöntemin işleyişini etkiler. Bu yöntem, başarılı olursa, bağlantı durumunu TRUE olarak ayarlar. Yeni bağlantı durumunu döndürür.

İlk kimlik doğrulama kaydı varsa, güvenli istemci kanallarına ilişkin sayısal sertifikaların kimliğini doğrulamak için kimlik doğrulama kayıtları zinciri kullanılır.

Birden çok ImqQueueManager nesnesini aynı kuyruk yöneticisine bağlayabilirsiniz. Tümünü aynı MQHCONN bağlantı tanıtıcısını kullanır ve iş parçacığıyla ilişkilendirilen bağlantı için UOW işlevselliğini paylaşır. Bağlanmak için ilk ImqQueueYöneticisi, MQHCONN tanıtıcısını alır. Bağlantıyı kesmek için son ImqQueueManager programı MQDISC ' yi gerçekleştirir.

Çok iş parçacıklı bir program için, her iş parçacığı için ayrı bir ImqQueueManager nesnesi kullanılması önerilir.

**ImqBinary connectionId () const;**

Bağlantı tanıtıcısını döndürür.

**ImqBinary connectionTag () const;**

Bağlantı etiketini döndürür.

**ImqBoolean setConnectionEtiketi (const MQBYTE128 etiketi = 0);**

Bağlantı etiketini ayarlar. *etiket* sıfırsa, bağlantı etiketini temizler. Bu yöntem başarılı olursa TRUE değerini döndürür.

**ImqBoolean setConnectionEtiketi (const ImqBinary & tag );**

Bağlantı etiketini ayarlar. *tag* veri uzunluğunun sıfır olması gerekir (bağlantı etiketini temizlemek için) ya da MQ\_CONN\_TAG\_LENGTH. Bu yöntem başarılı olursa TRUE değerini döndürür.

**MQHOT connectOptions() const;**

Bağlantı seçeneklerini döndürür.

**void setConnectSeçenekleri (const MQLong options = MQCNO\_NONE);**

Bağlantı seçeneklerini belirler.

**ImqBoolean connectionStatus() const;**

Bağlantı durumunu döndürür.

**ImqString cryptographicHardware ();**

Şifreleme donanımını döndürür.

**ImqBoolean setCryptographicDonanımı (const char \* hardware = 0);**

Şifreleme donanımını ayarlar. Bu yöntem başarılı olursa TRUE değerini döndürür.

**ImqBoolean deadLetterQueueName( ImqString & ad );**

Ölü harf kuyruğu adının bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**ImqString deadLetterQueueName ();**

Olası hatalara ilişkin herhangi bir belirti olmadan, ölü harf kuyruğu adının bir kopyasını döndürür.

**ImqBoolean defaultTransmissionQueueName( ImqString & ad );**

Varsayılan iletim kuyruğu adının bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**ImqString defaultTransmissionQueueName ();**

Olası hatalara ilişkin herhangi bir belirti olmadan, varsayılan iletim kuyruğu adını döndürür.

**ImqBoolean bağlantı kesme ();**

Kuyruk yöneticisinden bağlantıyı keser ve bağlantı durumunu FALSE olarak ayarlar. Bu nesneyle ilişkili tüm ImqProcess ve ImqQueue nesnelerini kapatır ve bağlantıyı kesmeden önce bağlantı başvurularını keser. Aynı kuyruk yöneticisine birden çok ImqQueueManager nesnesi bağlıysa, yalnızca son bağlantı

kesme işlemi fiziksel bir bağlantı kesme işlemi gerçekleştirir; diğerleri mantıksal bir bağlantı kesme gerçekleştirir. Kesinleştirilmemiş değişiklikler yalnızca fiziksel disconnection üzerinde kesinleştirilir.

Bu yöntem başarılı olursa TRUE değerini döndürür. Var olan bir bağlantı olmadığında çağrılırsa, dönüş kodu da doğrudur.

**ImqBoolean distributionLists(MQUBLE & destek );**

Dağıtım listelerinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQlong distributionLists();**

Olası hatalara ilişkin herhangi bir belirti olmadan, dağıtım listelerini döndürür.

**ImqBoolean dnsGroup ( ImqString & group);**

DNS grubu adının bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**ImqString dnsGroup ();**

Olası hatalara ilişkin herhangi bir belirti olmadan DNS grubu adını döndürür.

**ImqBoolean dnsWlm (MQORT & wlm);**

DNS WLM değerinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQLONG dnsWlm ();**

DNS WLM değerini, olası hatalara ilişkin herhangi bir belirti olmadan döndürür.

**ImqAuthenticationKayıt \* firstAuthenticationKaydı () const;**

İlk kimlik doğrulama kaydını döndürür.

**void setFirstAuthenticationRecord (const ImqAuthenticationRecord \* hava = 0);**

İlk kimlik doğrulama kaydını ayarlar.

**ImqObject \* firstManagedNesnesi () const;**

İlk yönetilen nesneyi döndürür.

**ImqBoolean inhibitEvent(MQUZE & olay );**

Engelleyici olayın etkinleştirme durumunun bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQLONG inhibitEvent();**

Olası hatalara ilişkin herhangi bir belirti olmaksızın, engelleyici olayın etkinleştirme durumunu döndürür.

**ImqBoolean ipAddressVersion (MQBOLD & version);**

IP adresi sürüm değerinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQHOT ipAddressVersion ();**

Olası hatalara ilişkin herhangi bir belirti olmadan IP adresi sürüm değerini döndürür.

**ImqBoolean keepAlive (MQlong & keepalive);**

Canlı tutma değerinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQlong keepAlive ();**

Olası hatalara ilişkin herhangi bir belirti olmadan canlı tutma değerini döndürür.

**ImqString keyRepository ();**

Anahtar havuzunu döndürür.

**ImqBoolean setKeyRepository (const char \* havuz = 0);**

Anahtar havuzunu ayarlar. Başarılı olursa TRUE değerini döndürür.

**ImqBoolean listenerTimer (MQORT & timer);**

Dinleyici süreölçeri değerinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQUZT listenerTimer ();**

Olası hatalara ilişkin herhangi bir belirti olmadan dinleyici zamanlayıcısı değerini döndürür.

**ImqBoolean localEvent(MQBOLD & olay );**

Yerel olayın etkinleştirilme durumunun bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQLONG localEvent();**

Olası hatalara ilişkin herhangi bir belirti olmaksızın, yerel olayın etkinleştirme durumunu döndürür.

**ImqBoolean loggerEvent (MQBOLD & count);**

Günlüğe kaydedici olayı değerinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQlong loggerEvent ();**

Olası hatalara ilişkin herhangi bir belirti olmadan günlüğe kaydedici olay değerini döndürür.

**ImqBoolean luGroupAdı ( ImqString & name);**

LU grubu adının bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**ImqString luGroupAdı ();**

Olası hatalara ilişkin herhangi bir belirti olmadan LU grubu adını döndürür.

**ImqBoolean lu62ARMSuffix ( ImqString & sonek);**

LU62 ARM sonekinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**ImqString lu62ARMSuffix ();**

Olası hatalara ilişkin herhangi bir belirti olmadan LU62 ARM sonekini döndürür.

**ImqBoolean luName ( ImqString & name);**

LU adının bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**ImqString luName ();**

Olası hatalara ilişkin herhangi bir belirti olmadan LU adını döndürür.

**ImqBoolean maximumActiveKanalları (MQGOLD ve kanallar);**

Etkin kanal sayısı üst sınırının bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQlong maximumActiveKanalları ();**

Olası hatalara ilişkin herhangi bir belirti olmadan, etkin kanal sayısı üst sınırını döndürür.

**ImqBoolean maximumCurrentKanalları (MQBOLD ve kanallar);**

Yürürlükteki kanal değeri üst sınırının bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQlong maximumCurrentChannels ();**

Olası hatalara ilişkin herhangi bir gösterge olmadan, geçerli kanal sayısı üst sınırını döndürür.

**ImqBoolean maximumHandles(MQLONG & sayı );**

Çekme noktalarının bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQLONG maximumHandles();**

Olası hatalara ilişkin herhangi bir gösterge olmadan, tutamaçlar üst sınırını döndürür.

**ImqBoolean maximumLu62Channels (MQGOLD ve kanallar);**

Maksimum LU62 kanal değeri için bir kopya sağlar. Başarılı olursa TRUE değerini döndürür.

**MQlong maximumLu62Channels ();**

Olası hatalara ilişkin herhangi bir gösterge olmadan LU62 kanal değeri üst sınırını döndürür.

**ImqBoolean maximumMessageUzunluk (MQUZE & uzunluk );**

İleti uzunluğu üst sınırının bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQlong maximumMessageUzunluk ();**

Olası hatalara ilişkin herhangi bir belirti olmadan ileti uzunluğu üst sınırını döndürür.

**ImqBoolean maximumPriority(MQHOMET & priority );**

Öncelik üst sınırının bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQLONG maximumPriority();**

Olası hatalara ilişkin herhangi bir gösterge olmadan, öncelik üst sınırının bir kopyasını döndürür.

**ImqBoolean maximumTcpKanalları (MQGOLD ve kanallar);**

TCP kanal sayısı üst sınırının bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQlong maximumTcpKanalları ();**

Olası hatalara ilişkin herhangi bir gösterge olmadan, TCP kanal sayısı üst sınırını döndürür.

**ImqBoolean maximumUncommittedİletileri (MQBOLD & sayı );**

Kesinleştirilmemiş iletilerin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQlong maximumUncommittedİletileri ();**

Olası hatalara ilişkin herhangi bir gösterge olmadan, kesinleştirilmemiş en fazla ileti sayısını döndürür.

**ImqBoolean mqAccounting (MQUVE & statint);**

MQI hesap değerinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQHOT mqAccounting ();**

Olası hatalara ilişkin herhangi bir belirti olmadan MQI hesap değerini döndürür.

**ImqBoolean mqiStatistics (MQUZO & statqi);**

MQI istatistik deęerinin bir kopyasını saęlar. Başarılı olursa TRUE deęerini döndürür.

**MQHOT mqiStatistics ();**

Olası hatalara ilişkin herhangi bir belirti olmadan MQI istatistik deęerini döndürür.

**ImqBoolean outboundPortMaks (MQlong & max);**

Giden kapı deęeri üst sınırının bir kopyasını saęlar. Başarılı olursa TRUE deęerini döndürür.

**MQHOT outboundPortMax ();**

Olası hatalara ilişkin herhangi bir gösterge olmadan, giden kapı deęeri üst sınırını döndürür.

**ImqBoolean outboundPortMin (MQUVE & min);**

Giden kapı deęeri alt sınırının bir kopyasını saęlar. Başarılı olursa TRUE deęerini döndürür.

**MQlong outboundPortMin ();**

Olası hatalara ilişkin herhangi bir gösterge olmadan, giden en düşük kapı deęerini döndürür.

**ImqBinary parolası () const;**

İstemci bağlantılarında kullanılan parolayı döndürür.

**ImqBoolean setPassword (const ImqString & password);**

İstemci bağlantılarında kullanılan parolayı ayarlar.

**ImqBoolean setPassword (const char \* = 0 password);**

İstemci bağlantılarında kullanılan parolayı ayarlar.

**ImqBoolean setPassword (const ImqBinary & password);**

İstemci bağlantılarında kullanılan parolayı ayarlar.

**ImqBoolean performanceEvent(MQLONG & olay );**

Başarım olayının etkinleştirilme durumunun bir kopyasını saęlar. Başarılı olursa TRUE deęerini döndürür.

**MQLONG performanceEvent();**

Olası hatalara ilişkin herhangi bir belirti göstermeden, performans olayının etkinleştirme durumunu döndürür.

**ImqBoolean altyapısı (MQUZE & altyapı );**

Altyapıya ilişkin bir kopya saęlar. Başarılı olursa TRUE deęerini döndürür.

**MQLONG platformu ();**

Olası hatalara ilişkin herhangi bir belirti göstermeden platformu döndürür.

**ImqBoolean queueAccounting (MQUZO & acctq);**

Kuyruk hesap deęerinin bir kopyasını saęlar. Başarılı olursa TRUE deęerini döndürür.

**MQlong queueAccounting ();**

Olası hatalara ilişkin herhangi bir belirti olmadan kuyruk muhasebesi deęerini döndürür.

**ImqBoolean queueMonitoring (MQUUR & monq);**

Kuyruk izleme deęerinin bir kopyasını saęlar. Başarılı olursa TRUE deęerini döndürür.

**MQLONG queueMonitoring ();**

Olası hatalara ilişkin herhangi bir belirti olmadan kuyruk izleme deęerini döndürür.

**ImqBoolean queueStatistics (MQOLD & statq);**

Kuyruk istatistięi deęerinin bir kopyasını saęlar. Başarılı olursa TRUE deęerini döndürür.

**MQHOT queueStatistics ();**

Olası hatalara ilişkin herhangi bir belirti olmadan kuyruk istatistikleri deęerini döndürür.

**ImqBoolean receiveTimeout (MQUZE & timeout);**

Alma zamaşıımı deęerinin bir kopyasını saęlar. Başarılı olursa TRUE deęerini döndürür.

**MQHOMEN receiveTimeout ();**

Olası hatalara ilişkin herhangi bir belirti olmadan alma zamaşıımı deęerini döndürür.

**ImqBoolean receiveTimeoutMin (MQUVE & min);**

Alma zamaşıımı alt sınır deęerinin bir kopyasını saęlar. Başarılı olursa TRUE deęerini döndürür.

**MQHOMEN receiveTimeoutMin ();**

Olası hatalara ilişkin herhangi bir gösterge olmadan minimum alma zamaşıımı deęerini döndürür.

**ImqBoolean receiveTimeoutTip (MQBOLD & tip);**

Alma zamanasını tipine ilişkin bir kopya sağlar. Başarılı olursa TRUE değerini döndürür.

**MQMET receiveTimeoutTip ();**

Olası hatalara ilişkin herhangi bir belirti olmadan alma zamanasını tipini döndürür.

**ImqBoolean remoteEvent(MQBOLD & olay );**

Uzak olaya ilişkin etkinleştirme durumunun bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQHOT remoteEvent();**

Olası hatalara ilişkin herhangi bir belirti olmadan, uzak olayın etkinleştirme durumunu döndürür.

**ImqBoolean repositoryName( ImqString & ad );**

Havuz adının bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**ImqString repositoryName( );**

Olası hatalara ilişkin herhangi bir belirti olmadan havuz adını döndürür.

**ImqBoolean repositoryNameListAdı ( ImqString & ad );**

Havuz adı listesi adının bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**ImqString repositoryNameListName ();**

Olası hatalara ilişkin herhangi bir belirti göstermeden havuz adı listesi adının bir kopyasını döndürür.

**ImqBoolean sharedQueueQueueManagerAdı (MQBOLD & name);**

Paylaşılan kuyruk kuyruk yöneticisi adı değerinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQlong sharedQueueQueueManagerName ();**

Olası hatalara ilişkin herhangi bir gösterge olmadan, paylaşılan kuyruk kuyruk yöneticisi adı değerini döndürür.

**ImqBoolean sslEvent (MQORT & event);**

SSL olay değerinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQHOT sslEvent ();**

Olası hatalara ilişkin herhangi bir belirti olmadan SSL olay değerini döndürür.

**ImqBoolean sslFips (MQUSED & sslfips);**

SSL FIPS değerinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQLONG sslFips ();**

Olası hatalara ilişkin herhangi bir belirti olmadan SSL FIPS değerini döndürür.

**ImqBoolean sslKeyResetCount (MQORT & count);**

SSL anahtar ilk duruma getirme sayısı değerinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQHOT sslKeyResetCount ();**

Olası hatalara ilişkin herhangi bir belirti olmadan SSL anahtarı sıfırlama sayısı değerini döndürür.

**ImqBoolean startStopOlayı (MQUZE & olay );**

Başlatma olayının etkinleştirilme durumunun bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQlong startStopEvent ();**

Olası hatalara ilişkin herhangi bir belirti olmaksızın, başlangıç durdurma olayının etkinleştirme durumunu döndürür.

**ImqBoolean statisticsInterval (MQUSED & statint);**

İstatistik aralığı değerinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQLONG statisticsInterval ();**

Olası hatalara ilişkin herhangi bir belirti olmadan istatistik aralığı değerini döndürür.

**ImqBoolean syncPointAvailability (MQORT & eşitleme );**

Syncpoint kullanılabilirlik değerinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**MQOUR syncPointUygunluğu ();**

Olası hatalara ilişkin herhangi bir belirti olmadan, syncpoint kullanılabilirlik değerinin bir kopyasını döndürür.



**ImqBoolean tcpName ( ImqString & name);**

TCP sistem adının bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**ImqString tcpName ( );**

Olası hatalara ilişkin herhangi bir belirti olmadan TCP sistem adını döndürür.

**ImqBoolean tcpStackTipi (MQBOLD & tip);**

TCP yığın tipinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**Mqlong tcpStackTip ( );**

Olası hatalara ilişkin herhangi bir belirti olmadan TCP yığın tipini döndürür.

**ImqBoolean traceRouteKaydı (MQUUR & routerec);**

İzleme rotası kayıt değerinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

**Mqlong traceRouteKaydı ( );**

Olası hatalara ilişkin herhangi bir belirti olmadan izleme rotası kayıt değerini döndürür.

**ImqBoolean triggerInterval(MQUZE & aralık );**

Tetikleme aralığından bir kopya sağlar. Başarılı olursa TRUE değerini döndürür.

**MQLONG triggerInterval();**

Olası hatalara ilişkin herhangi bir belirti olmadan tetikleme aralığını döndürür.

**ImqBinary userId ( ) const;**

İstemci bağlantılarında kullanılan kullanıcı kimliğini döndürür.

**ImqBoolean setUserId (const ImqString & id);**

İstemci bağlantılarında kullanılan kullanıcı kimliğini ayarlar.

**ImqBoolean setUserId (const char \* = 0 id);**

İstemci bağlantılarında kullanılan kullanıcı kimliğini ayarlar.

**ImqBoolean setUserId (const ImqBinary & id);**

İstemci bağlantılarında kullanılan kullanıcı kimliğini ayarlar.

**Nesne yöntemleri (korunan)****void setFirstManagedObject (const ImqObject \* object = 0);**

İlk yönetilen nesneyi ayarlar.

**Nesne verileri (korunan)****MQHCONN ohconn**

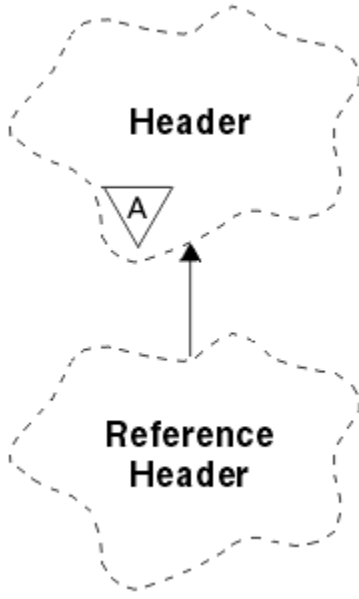
IBM MQ bağlantısı tanıtıcısı (yalnızca bağlantı durumu TRUE olduğunda anlamlıdır).

**Neden kodları**

- MQRC\_ATTRIBUTE\_LOCKED
- MQRC\_ENVIRONMENT\_ERROR
- MQRC\_FUNCTION\_NOT\_SUPPORTED
- MQRC\_REFERENCE\_ERROR
- (MQBACK için neden kodları)
- (MQBEGIN için neden kodları)
- (MQCMIT için neden kodları)
- (MQCONN için neden kodları)
- (MQDISC için neden kodları)
- (MQCONN için neden kodları)

**ImqReferenceHeader C++ sınıfı**

Bu sınıf, MQRMH veri yapısının özelliklerini sarsalıyor.



Şekil 34. *ImqReferenceÜstbilgi* sınıfı

Bu sınıf, [“ImqReferenceÜstbilgi çapraz başvurusu” sayfa 1774](#) içinde listelenen MQI çağrılarını ile ilgilidir.

- [“Nesne öznitelikleri” sayfa 1854](#)
- [“Oluşturucular” sayfa 1855](#)
- [“Aşırı yüklenmiş ImqItem yöntemleri” sayfa 1855](#)
- [“Nesne yöntemleri \(genel\)” sayfa 1855](#)
- [“Nesne verileri \(korunan\)” sayfa 1856](#)
- [“Neden kodları” sayfa 1856](#)

## Nesne öznitelikleri

### hedef ortam

Hedefe ilişkin ortam. Başlangıç değeri, boş değerli bir dizgidir.

### Hedef Adı

Veri hedefinin adı. Başlangıç değeri, boş değerli bir dizgidir.

### Eşgörünüm Tanıtıcısı

Eşgörünüm tanıtıcısı. MQ\_OBJECT\_INSTANCE\_ID\_LENGTH uzunluğunda bir ikili değer (MQBYTE24). İlk değer MQOLI\_NONE olur.

### mantıksal uzunluk

Bu üstbilgiyi izleyen ileti verilerinin mantıksal ya da amaçlanan uzunluğu. Başlangıç değeri sıfır.

### mantıksal görelî konum

Aşağıdaki ileti verilerine ilişkin mantıksal görelî konum, verilerin bir bütün olarak bağlamında yorumlanmak üzere nihai varış noktasına göre yorumlanır. Başlangıç değeri sıfır.

### mantıksal görelî konum 2

Mantıksal görelî konuma yüksek sıralı uzantı. Başlangıç değeri sıfır.

### Başvuru tipi

Başvuru tipi. Başlangıç değeri, boş değerli bir dizgidir.

### Kaynak Ortamı

Kaynak için ortam. Başlangıç değeri, boş değerli bir dizgidir.

### KAYNAK ADI

Veri kaynağının adı. Başlangıç değeri, boş değerli bir dizgidir.

## Olusturucular

### **ImqReferenceÜstbilgisi ();**

Varsayılan oluşturuçu.

### **ImqReferenceÜstbilgisi (const ImqReferenceÜstbilgisi & üstbilgisi );**

Kopya oluşturuçusu.

## Aşırı yüklenmiş ImqItem yöntemleri

### **sanal ImqBoolean copyOut ( ImqMessage & msg );**

Başlangıçta ileti arabelleğiyle bir MQRMH veri yapısı ekler, var olan ileti verilerini daha ileriye taşıyarak *msg* biçimini MQFMT\_REF\_MSG\_HEADER olarak ayarlar.

Daha ayrıntılı bilgi için, "[ImqHeader C++ sınıfı](#)" sayfa 1801 ' ta ImqHeader sınıf yöntemi açıklamasına bakın.

### **sanal ImqBoolean pasteIn ( ImqMessage & msg );**

İleti arabelleğinden bir MQRMH veri yapısını okur.

Başarılı olmak için, ImqMessage biçimi MQFMT\_REF\_MSG\_HEADER olmalıdır.

Daha ayrıntılı bilgi için, "[ImqHeader C++ sınıfı](#)" sayfa 1801 ' ta ImqHeader sınıf yöntemi açıklamasına bakın.

## Nesne yöntemleri (genel)

### **void işleci = (const ImqReferenceÜstbilgisi & üstbilgisi );**

Var olan eşgörünüm verilerini değiştirerek, yönetim ortamı verilerini *üstbilgi*' den kopyalar.

### **ImqString destinationEnvironment () const;**

Hedef ortamın bir kopyasını döndürür.

### **void setDestinationOrtamı (const char \* environment = 0);**

Hedef ortamı ayarlar.

### **ImqString destinationName () const;**

Hedef adının bir kopyasını döndürür.

### **void setDestinationAd (const char \* ad = 0);**

Hedef adı belirler.

### **ImqBinary instanceId () const;**

Eşgörünüm tanıtıcısının bir kopyasını döndürür.

### **ImqBoolean setInstanceId (const ImqBinary & id );**

Eşgörünüm tanıtıcısını ayarlar. *simge* veri uzunluğunun 0 ya da MQ\_OBJECT\_INSTANCE\_ID\_LENGTH olması gerekir. Bu yöntem başarılı olursa TRUE değerini döndürür.

### **void setInstanceId (const MQBYTE24 id = 0);**

Eşgörünüm tanıtıcısını ayarlar. *id* sıfır (sıfır) olabilir; bu, MQOLI\_NONE belirtilerek aynıdır. *id* sıfır değilse, bu değer, ikili verilerin MQ\_OBJECT\_INSTANCE\_ID\_LENGTH byte 'ı ele almalıdır. MQOLI\_NONE gibi önceden tanımlanmış değerleri kullanırken, imza eşleştirmesini sağlamak için bir tip dönüştürme yapmanız gerekebilir; örneğin (MQBYTE \*) MQOLI\_NONE.

### **MQHOT logicalLength () const;**

Mantıksal uzunluğu döndürür.

### **void setLogicalLength (const MQUBE uzunluk );**

Mantıksal uzunluğu belirler.

### **MQHOT logicalOffset () const;**

Mantıksal görel konumu döndürür.

### **void setLogicalOffset (const MQHOMET görel konumu );**

Mantıksal görel konumu ayarlar.

### **MQTHOE logicalOffset2 () const;**

Mantıksal görel konum 2 değerini döndürür.

**void setLogicalOffset2 (const MQOUR görelî konumu );**

Mantıksal görelî konum 2 değeri ayarlar.

**ImqString referenceType () const;**

Başvuru tipinin bir kopyasını döndürür.

**void setReferenceType (const char \* ad = 0);**

Başvuru tipini ayarlar.

**ImqString sourceEnvironment () const;**

Kaynak ortamın bir kopyasını döndürür.

**void setSourceOrtam (const char \* environment = 0);**

Kaynak ortamı ayarlar.

**ImqString sourceName () const;**

Kaynak adının bir kopyasını döndürür.

**void setSourceAd (const char \* ad = 0);**

Kaynak adını belirler.

## Nesne verileri (korunan)

**MQRMH omqrmh**

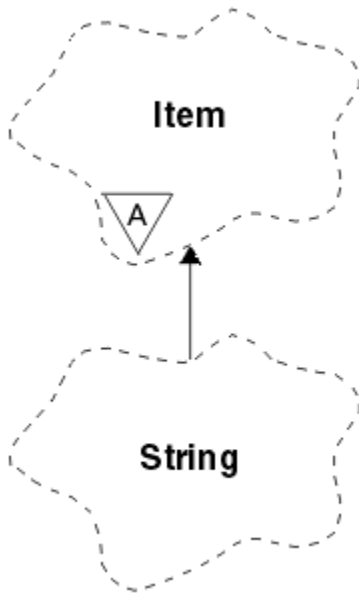
MQRMH veri yapısı.

## Neden kodları

- MQRC\_BINARY\_DATA\_LENGTH\_ERROR
- MQRC\_STRUC\_LENGTH\_ERROR
- MQRC\_STRUC\_ID\_ERROR
- MQRC\_INSUFFICIENT\_DATA
- MQRC\_INCONSTENT\_BIÇIMI
- MQRCENCODING\_ERROR

## ImqString C++ sınıfı

Bu sınıf, boş karakterle biten dizgiler için karakter dizgisi saklama ve işleme olanağı sağlar.



Şekil 35. ImqString sınıfı

Use an `ImqString` in place of a **char \*** in most situations where a parameter calls for a **char \***.

- [“Nesne öznitelikleri” sayfa 1857](#)
- [“Oluşturucular” sayfa 1857](#)
- [“Sınıf yöntemleri \(genel\)” sayfa 1857](#)
- [“Aşırı yüklenmiş `ImqItem` yöntemleri” sayfa 1858](#)
- [“Nesne yöntemleri \(genel\)” sayfa 1858](#)
- [“Nesne yöntemleri \(korunan\)” sayfa 1861](#)
- [“Neden kodları” sayfa 1861](#)

## Nesne öznitelikleri

### karakter

**depolama** içindeki karakterler, sonda boş değer öncesindeki karakterlerden önce gelir.

### uzunluk

**Karakterlerdeki** bayt sayısı. **Saklama alanı** yoksa, **uzunluk** sıfırdır. Başlangıç değeri sıfır.

### depolama

Rasgele boyutların bayt dizisi. Sondaki boş değer her zaman **karakterler**'den sonra **depolama** ' da yer almalı, böylece **karakter** ' in sonunun algılanabileceği şekilde olmalıdır. Yöntemler, bu durumun korunduğundan emin olun, ancak dizideki baytları doğrudan ayarlarken, değişiklikten sonra sonda bir boş değer olduğunu doğrulayın. Başlangıçta, **storage** özneliği yoktur.

## Oluşturucular

### **ImqString( );**

Varsayılan oluşturucu.

### **ImqString(const ImqString & dizgi );**

Kopya oluşturucusu.

### **ImqString(const char c );**

**karakterler** , c' yi oluşturur.

### **ImqString(const char \* metin );**

**karakterler** , *metin*' den kopyalanır.

### **ImqString(const void \* buffer, const size\_t length );**

Copies *uzunluk* bytes starting from *arabellek* and assigns them to the **karakter**. Kopyalanan boş değerli karakterler için yerine koyma değeri yapılır. Yerine koyma karakteri nokta (.) karakteridir. Kopyalanamaz ya da görüntülenemeyen diğer karakterler için özel bir önem verilemez.

## Sınıf yöntemleri (genel)

### **static ImqBoolean copy (char \* hedef-buffer, const size\_t length, const char \* kaynak-buffer, const char pad = 0);**

*kaynak-arabellek* ile *hedef-arabellek* arasındaki *uzunluk* byte 'ı kopyalar. If the number of characters in *kaynak arabellek* is insufficient, fills the remaining space in *hedef-arabellek* with *doldurma* characters. *kaynak-arabellek* 'i sıfır olabilir. *length* değeri de sıfırsa, *hedef-arabellek* 'i sıfır olabilir. Herhangi bir hata kodu kaybedilir. Bu yöntem başarılı olursa TRUE değerini döndürür.

### **statik ImqBoolean copy (char \* hedef-buffer, const size\_t length, const char \* kaynak-buffer, ImqError & error-object, const char pad = 0);**

*kaynak-arabellek* ile *hedef-arabellek* arasındaki *uzunluk* byte 'ı kopyalar. If the number of characters in *kaynak arabellek* is insufficient, fills the remaining space in *hedef-arabellek* with *doldurma* characters. *kaynak-arabellek* 'i sıfır olabilir. *length* değeri de sıfırsa, *hedef-arabellek* 'i sıfır olabilir. *Error-object* (hata-nesne) içinde herhangi bir hata kodu ayarlanır. Bu yöntem başarılı olursa TRUE değerini döndürür.

## Aşırı yüklenmiş ImqItem yöntemleri

### sanal ImqBoolean copyOut ( ImqMessage & msg );

Varolan içeriğin yerine, **karakter** ' u ileti arabelleğiyle kopyalar. **msg biçim** ' i MQFMT\_STRING olarak ayarlar.

Ek ayrıntılar için üst sınıf yöntemi açıklamasına bakın.

### sanal ImqBoolean pasteIn ( ImqMessage & msg );

Var olan **karakterleri**değiştirerek, kalan verileri ileti arabelleğinden aktararak **karakterleri** ayarlar.

Başarılı olmak için, **msg** nesnesinin **kodlama** değeri MQENC\_NATIVE olmalıdır. İletileri MQGMO\_CONVERT ile MQENC\_NATIVE ile alın.

Başarılı olmak için, ImqMessage **biçim** , MQFMT\_STRING olmalıdır.

Ek ayrıntılar için üst sınıf yöntemi açıklamasına bakın.

## Nesne yöntemleri (genel)

### char & operator [] (const size\_t offset ) const;

References the character at offset *görelî konum* in the **depolama**. İlgili baytın var olduğundan ve adreslenebilir olduğundan emin olun.

### ImqString işleci () (const size\_t görelî konum, const size\_t uzunluk = 1) const;

Returns a substring by copying bytes from the **karakter** starting at *görelî konum*. If *uzunluk* is zero, returns the rest of the **karakter**. *Görelî konum* ve *uzunluk* birleşimi, **karakterleri**çinde bir başvuru oluşturmazsa, boş bir ImqStringdöndürür.

### void işleci = (const ImqString & dizgi );

Yönetim ortamı verilerini *dizgi*' den kopyalar ve var olan eşgörünüm verilerini değiştirir.

### ImqString işleci + (const char c ) const;

c karakteri ile **characterlar**arasındaki sonun sonucunu döndürür.

### ImqString işleci + (const char \* metin ) const;

*metin* metninin **karakter**olarak eklenmesinin sonucunu döndürür. Bu da ters çevrilebilir. Örneğin:

```
strOne + "string two" ;  
"string one" + strTwo ;
```

**Not:** Çoğu derleyici **strOne + "string iki"**; Microsoft Visual C++ için **strOne + (char \*) "dizgi iki"** ;gerektirir.

### ImqString işleci + (const ImqString & string1 ) const;

Returns the result of appending *string1* to the **karakter**.

### ImqString işleci + (const çift sayı ) const;

Returns the result of appending *sayı* to the **karakter** after conversion to text.

### ImqString işleci + (const long sayı ) const;

Returns the result of appending *sayı* to the **karakter** after conversion to text.

### void operator + = (const char c );

C 'ı **karakter**' a ekler.

### void operator + = (const char \* text );

*Metin* 'i **karakterler**' e ekler.

### void işleci + = (const ImqString & dizgi );

*dizgi* 'yi **karakterler**' e ekler.

### void operator + = (const double sayı );

Metne dönüştürdükten sonra *sayı* , **karakter** ' a ekler.

### void operator + = (const long number );

Metne dönüştürdükten sonra *sayı* , **karakter** ' a ekler.

### **işleç char \* () const;**

**storage**(depolama) içindeki ilk baytın adresini döndürür. Bu değer sıfır olabilir ve uçucu olabilir. Bu yöntemi yalnızca salt okunur amaçlar için kullanın.

### **ImqBoolean işleci < (const ImqString & dizgi ) const;**

**karakter** yöntemini, **karşılaştır** yöntemini kullanarak *dizgi* ile karşılaştırır. Daha küçük ya da buna eşit ise, sonuç TRUE ve FALSE ise sonuç TRUE olur.

### **ImqBoolean işleci > (const ImqString & string ) const;**

**karakter** yöntemini, **karşılaştır** yöntemini kullanarak *dizgi* ile karşılaştırır. Daha büyük ya da eşit değilse, sonuç TRUE ise FALSE (doğru) olur.

### **ImqBoolean işleci < = (const ImqString & dizgi ) const;**

**karakter** yöntemini, **karşılaştır** yöntemini kullanarak *dizgi* ile karşılaştırır. Bundan küçük ya da bu değere eşit ya da küçükse sonuç TRUE (Doğru) olur.

### **ImqBoolean işleci > = (const ImqString & dizgi ) const;**

**karakter** yöntemini, **karşılaştır** yöntemini kullanarak *dizgi* ile karşılaştırır. Bundan büyük ya da buna eşit ve küçükse FALSE (Yanlış) ve FALSE (yanlış) olur.

### **ImqBoolean işleci == (const ImqString & string ) const;**

**karakter** yöntemini, **karşılaştır** yöntemini kullanarak *dizgi* ile karşılaştırır. TRUE ya da FALSE değerini döndürür.

### **ImqBoolean işleci != (const ImqString & string ) const;**

**karakter** yöntemini, **karşılaştır** yöntemini kullanarak *dizgi* ile karşılaştırır. TRUE ya da FALSE değerini döndürür.

### **Kısa karşılaştırma (const ImqString & dizgi ) const;**

**Karakterlerle** *dizgi* ile karşılaştırır. **karakterler** eşitse, sonuç sıfır olur; daha küçük ve küçükse olumsuzdur (büyüktür). Karşılaştırma büyük ve küçük harfe duyarlıdır. Boş değer ImqString , boş değerli olmayan ImqStringdeğerinden az değer olarak kabul edilir.

### **ImqBoolean copyOut(char \* buffer, const size\_t length, const char pad = 0);**

**karakter** 'ten *uzunluk* ' a kadar olan ve *arabellek* ' a kopyalar. If the number of **karakter** is insufficient, fills the remaining space in *arabellek* with *doldurma* characters. *length* değeri de sıfırsa, *buffer* sıfır olabilir. Başarılı olursa TRUE değerini döndürür.

### **size\_t copyOut(uzun ve sayı ) const;**

Metinden dönüştürme işleminden sonra **karakter** 'den *sayı* ' ı ayarlar ve dönüştürmede yer alan karakter sayısını döndürür. Bu sıfır ise, herhangi bir dönüştürme işlemi gerçekleştirilmez ve *sayı* ayarlanmaz. Dönüştürülebilir bir karakter sırası aşağıdaki değerlerle başlamalıdır:

```
<blank(s)>
<+|->
digit(s)
```

### **size\_t copyOut( ImqString & token, const char c = '' ) const;**

**Karakterler** , *c* ' den farklı bir ya da daha çok karakter içeriyorsa, bir simgeyi bu tür karakterlerin ilk bitişik sırası olarak tanıtır. In this case *simge* is set to that sequence, and the value returned is the sum of the number of leading characters *C* and the number of bytes in the sequence. Tersisi durumda, sıfır değerini döndürür ve *simge*değerini ayarlamaz.

### **size\_t cutOut(uzun ve sayı );**

**kopyala** yöntemi için *sayı* değerini ayarlar, ancak **karakter** değerinden, dönüş değeri tarafından belirtilen bayt sayısını da kaldırır. Örneğin, aşağıdaki örnekte gösterilen *dizgi cutOut ( sayı )* kullanılarak üç sayıdan oluşan bir dizgi olabilir. üç kez:

```
strNumbers = "-1 0 +55 "
while ( strNumbers.cutOut( number ) );
number becomes -1, then 0, then 55
leaving strNumbers == " "
```

### **size\_t cutOut( ImqString & token, const char c = " " )**

Sets *simge* as for the **copyOut** method, and removes from **karakter** the *strToken* characters and also any characters *C* that precede the *simge* characters. *C* boş değilse, *simge* karakterlerini doğrudan doğrulayan *C* karakterlerini kaldırır. Kaldırılan karakter sayısını döndürür. Örneğin, aşağıdaki örnekte gösterilen dizgi **cutOut** ( *simge* ) kullanılarak üç simgeden kesilebilir. üç kez:

```
strText = " Program Version 1.1 "  
while ( strText.cutOut( token ) );  
  
// token becomes "Program", then "Version",  
// then "1.1" leaving strText == " "
```

Aşağıdaki örnekte, bir DOS yol adının nasıl ayrıştırılacağı gösterilmektedir:

```
strPath = "C:\OS2\BITMAP\OS2LOGO.BMP"  
  
strPath.cutOut( strDrive, ':' );  
strPath.stripLeading( ':' );  
while ( strPath.cutOut( strFile, '\\' ) );  
  
// strDrive becomes "C".  
// strFile becomes "OS2", then "BITMAP",  
// then "OS2LOGO.BMP" leaving strPath empty.
```

### **ImqBoolean find (const ImqString & string);**

**karakter** ile tam olarak eşleşen *dizgi* ile eşleşen bir arama yapar. Eşleşme bulunamazsa, FALSE değerini döndürür. Ters durumda, TRUE değerini döndürür. *dizgi* boş değerliyse, TRUE değerini döndürür.

### **ImqBoolean find (const ImqString & string, size\_t & offset );**

*Görelî konum* görelî konumundan itibaren **karakterler** arasında bir yerde *dizgi* ile tam eşleşmeyi arar. *dizgi* boş değerliyse, *görelî konum* güncellenmeden TRUE değerini döndürür. Eşleşme bulunamazsa, FALSE değerini döndürür ( *görelî konum* değeri artırılmış olabilir). Bir eşleşme bulunursa, TRUE değerini döndürür ve *görelî konum* değerini **karakterleri** içindeki *dizgi* görelî konumuna döndürür.

### **size\_t length () const;**

**uzunluk** değerini döndürür.

### **ImqBoolean pasteIn(const double number, const char \* format = "%f");**

Metne dönüştürdükten sonra *sayı* , **karakter** ' a ekler. Başarılı olursa TRUE değerini döndürür.

The specification *biçim* is used to format the floating point conversion. Belirtirirse, **printf** ve kayan nokta numaralarıyla birlikte kullanılmak için uygun bir ad olmalıdır; örneğin, **%3f**.

### **ImqBoolean pasteIn(const long number );**

Metne dönüştürdükten sonra *sayı* , **karakter** ' a ekler. Başarılı olursa TRUE değerini döndürür.

### **ImqBoolean pasteIn(const void \* buffer, const size\_t length );**

*uzunluk* bayt ' ı *arabellek* ' dan **karakter** ' a ekler ve son bir sonda boş değer ekler. Kopyalanan boş karakterleri yerine koyar. Yerine koyma karakteri nokta (.) karakteridir. Başka bir yazdırılmayan ya da görüntülenemeyen karakterlerin kopyalandığı herhangi bir özel durum dikkate alınmaz. Bu yöntem başarılı olursa TRUE değerini döndürür.

### **ImqBoolean set (const char \* buffer, const size\_t length );**

Değişmez uzunluklu bir karakter alanındaki **karakterleri** boş değer içerebilecek şekilde ayarlar. Gerekliyse, değişmez uzunluklu alandaki karakterlere boş değer ekler. Bu yöntem başarılı olursa TRUE değerini döndürür.

### **ImqBoolean setStorage(const size\_t uzunluk );**

**storage** (Depolama) alanını ayırır (ya da yeniden ayırma). Aralarında hala yer varsa, herhangi bir sondaki boş değer de içinde olmak üzere, özgün **karakterleri** korur, ancak ek depolama alanı başlatmaz.

Bu yöntem başarılı olursa TRUE değerini döndürür.



**size\_t storage () const;**

**storage**(depolama) içindeki bayt sayısını döndürür.

**size\_t stripLeading(const char c = " ");**

Strips leading characters C from the **karakter** and returns the number removed.

**size\_t stripTrailing(const char c = " ");**

Strips trailing characters C from the **karakter** and returns the number removed.

**ImqString upperCase() const;**

Returns an uppercase copy of the **karakter**.

## Nesne yöntemleri (korunan)

**ImqBoolean assign ( const ImqString & dizgi );**

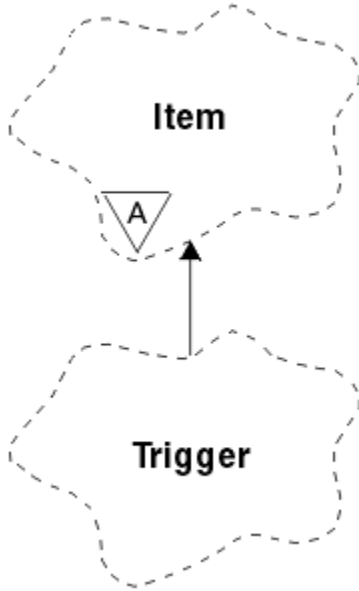
Eşdeğer **işleç** = yöntemine eşdeğer, ancak sanal olmayan bir yöntem. Başarılı olursa TRUE değerini döndürür.

## Neden kodları

- MQRC\_DATA\_TRUNCATED
- MQRC\_NULL\_POINTER
- MQRC\_STORAGE\_NOT\_AVAM
- MQRC\_BUFFER\_ERROR
- MQRC\_INCONSTENT\_BIÇIMI

## ImqTrigger C++ sınıfı

Bu sınıf, MQTM (tetikleme iletisi) veri yapısını sarsalıyor.



Şekil 36. ImqTrigger sınıfı

Bu sınıfın nesnelere genellikle bir tetikleyici izleme programı tarafından kullanılır. Tetikleme izleme programının görevi, bu iletilerin beklenmesini ve iletiler beklenirken diğer IBM MQ uygulamalarının başlatılmasını sağlamak için bu iletilerin üzerinde işlem yapmak olduğunu kabul eder.

Bir kullanım örneği için IMQSTRG örnek programına bakın.

- [“Nesne öznitelikleri” sayfa 1862](#)
- [“Oluşturucular” sayfa 1862](#)

- [“Aşırı yüklenmiş ImqItem yöntemleri” sayfa 1862](#)
- [“Nesne yöntemleri \(genel\)” sayfa 1863](#)
- [“Nesne verileri \(korunan\)” sayfa 1863](#)
- [“Neden kodları” sayfa 1864](#)

## Nesne öznitelikleri

### Uygulama tanıtıcısı

İletiyi gönderen uygulamanın tanıtıcısı. Başlangıç değeri, boş değerli bir dizgidir.

### Uygulama Tipi

İletiyi gönderen uygulama tipi. Başlangıç değeri sıfır. Aşağıdaki ek değerler kullanılabilir:

- MQAT\_AIX
- MQAT\_CICS
- MQAT\_DOS
- MQAT\_IMS
- MQAT\_MVS
- MQAT\_NOTES\_AGENT
- MQAT\_OS2
- MQAT\_OS390
- MQAT\_OS400
- MQAT\_UNIX
- MQAT\_WINDOWS
- MQAT\_WINDOWS\_NT
- MQAT\_USER\_FIRST
- MQAT\_USER\_SON

### ortam verileri

Sürece ilişkin ortam verileri. Başlangıç değeri, boş değerli bir dizgidir.

### İşlem adı

Süreç adı. Başlangıç değeri, boş değerli bir dizgidir.

### Kuyruk adı

Başlatılacak kuyruğun adı. Başlangıç değeri, boş değerli bir dizgidir.

### tetikleyici verileri

Sürece ilişkin verileri tetikler. Başlangıç değeri, boş değerli bir dizgidir.

### Kullanıcı verileri

Sürece ilişkin kullanıcı verileri. Başlangıç değeri, boş değerli bir dizgidir.

## Oluşturucular

### ImqTrigger( );

Varsayılan oluşturucu.

### ImqTrigger(const ImqTrigger & trigger );

Kopya oluşturucusu.

## Aşırı yüklenmiş ImqItem yöntemleri

### sanal ImqBoolean copyOut ( ImqMessage & msg );

Var olan içeriği değiştirerek, ileti arabelleğindeki bir MQTM veri yapısını yazar. msg biçimini MQFMT\_TRIGGER olarak ayarlar.

Daha ayrıntılı bilgi için, "[ImqItem C++ sınıfı](#)" sayfa 1806 adresindeki ImqItem sınıf yöntemi açıklamasına bakın.

**sanal ImqBoolean pasteIn ( ImqMessage & msg );**

İleti arabelleğinden bir MQTM veri yapısını okur.

Başarılı olmak için ImqMessage biçiminin MQFMT\_TRIGGER değeri olmalıdır.

Daha ayrıntılı bilgi için, "[ImqItem C++ sınıfı](#)" sayfa 1806 adresindeki ImqItem sınıf yöntemi açıklamasına bakın.

## Nesne yöntemleri (genel)

**void işleci = (const ImqTrigger & trigger );**

Eşgörünüm verilerini, var olan eşgörünüm verilerini değiştirerek *tetikleyiciden* kopyalar.

**ImqString applicationId () const;**

Uygulama tanıtıcısının bir kopyasını döndürür.

**void setApplicationId (const char \* id );**

Uygulama tanıtıcısını ayarlar.

**MQUZE applicationType () const;**

Uygulama tipini döndürür.

**void setApplicationType (const MQHOMET tip );**

Uygulama tipini ayarlar.

**ImqBoolean copyOut ( MQTMC2 \* ptmc2 );**

Başlatma kuyruklarında alınan MQTM veri yapısını kapsüller. Çağırın tarafından sağlanan eşdeğer bir MQTMC2 veri yapısını doldurur ve QMgrName alanını (MQTM veri yapısında yoktur) tüm boşluklara ayarlar. MQTMC2 veri yapısı, geleneksel olarak bir tetikleme izleyicinin başlattığı uygulamalara parametre olarak kullanılır. Bu yöntem başarılı olursa TRUE değerini döndürür.

**ImqString environmentData () const;**

Ortam verisinin bir kopyasını döndürür.

**void setEnvironmentData (const char \* data );**

Ortam verilerini ayarlar.

**ImqString processName () const;**

Süreç adının bir kopyasını döndürür.

**void setProcessAd (const char \* ad );**

Süreç adını, boşluklarla doldurulacak şekilde 48 karakter olarak ayarlar.

**ImqString queueName () const;**

Kuyruk adının bir kopyasını döndürür.

**void setQueueAdı (const char \* ad );**

Kuyruk adını ayarlar, boşluklarla 48 karakter uzunluğunda doldurulur.

**ImqString triggerData () const;**

Tetikleyici verilerinin bir kopyasını döndürür.

**void setTriggerData (const char \* data );**

Tetikleme verilerini ayarlar.

**ImqString userData () const;**

Kullanıcı verilerinin bir kopyasını döndürür.

**void setUserData (const char \* data );**

Kullanıcı verilerini ayarlar.

## Nesne verileri (korunan)

**MQTM oqtm**

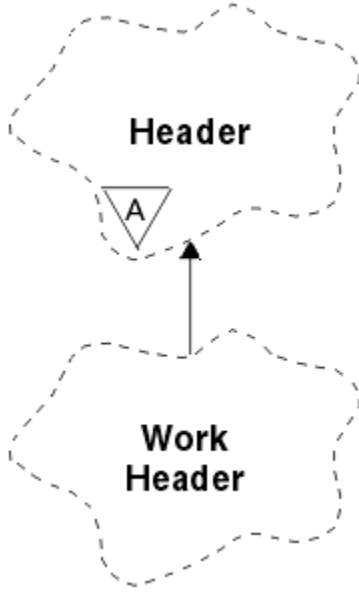
MQTM veri yapısı.

## Neden kodları

- MQRC\_NULL\_POINTER
- MQRC\_INCONSTENT\_BIÇIMI
- MQRCENCODING\_ERROR
- MQRC\_STRUC\_ID\_ERROR

## ImqWorkÜstbilgi C++ sınıfı

Bu sınıf, MQWIH veri yapısının belirli özelliklerini sarsalıyor.



Şekil 37. ImqWorkÜstbilgi sınıfı

Bu sınıfın nesnelere, iletileri z/OS Workload Manager tarafından yönetilen kuyruğa ileti koyan uygulamalar tarafından kullanılır.

- [“Nesne öznitelikleri” sayfa 1864](#)
- [“Oluşturucular” sayfa 1864](#)
- [“Aşırı yüklenmiş ImqItem yöntemleri” sayfa 1865](#)
- [“Nesne yöntemleri \(genel\)” sayfa 1865](#)
- [“Nesne verileri \(korunan\)” sayfa 1865](#)
- [“Neden kodları” sayfa 1865](#)

## Nesne öznitelikleri

### ileti simgesi

z/OS Workload Manager için ileti simgesi, MQ\_MSG\_TOKEN\_LENGTH uzunluğuna sahip. İlk değer MQMTOK\_NONE değeridir.

### Hizmet Adı

Bir işlemin 32 karakterlik adı. Ad, başlangıçta boşluklara neden olur.

### hizmet adımları

İşlem içindeki bir adımın 8 karakterden oluşan adı. Ad, başlangıçta boşluklara neden olur.

## Oluşturucular

### ImqWorkÜstbilgisi ();

Varsayılan oluşturucu.

## **ImqWorkÜstbilgisi (const ImqWorkÜstbilgisi & üstbilgisi );**

Kopya oluşturucusu.

## **Aşırı yüklenmiş ImqItem yöntemleri**

### **sanal ImqBoolean copyOut( ImqMessage & msg );**

İleti arabelleğindeki başlangıca bir MQWIH veri yapısı ekler, var olan ileti verilerini daha ileriye taşıyarak *msg biçimini* MQFMT\_WORK\_INFO\_HEADER ' a ayarlar.

Daha fazla ayrıntı için üst sınıf yöntemi açıklamasına bakın.

### **sanal ImqBoolean pasteIn( ImqMessage & msg );**

İleti arabelleğinden bir MQWIH veri yapısını okur.

Başarılı olmak için, *msg* nesnesinin kodlamasının MQENC\_NATIVE olması gerekir. İletileri MQGMO\_CONVERT ile MQENC\_NATIVE ile alın.

ImqMessage biçimi MQFMT\_WORK\_INFO\_HEADER olmalıdır.

Daha fazla ayrıntı için üst sınıf yöntemi açıklamasına bakın.

## **Nesne yöntemleri (genel)**

### **void işleci = (const ImqWorkÜstbilgisi & üstbilgisi );**

Var olan eşgörünüm verilerini değiştirerek, yönetim ortamı verilerini *üstbilgi*' den kopyalar.

### **ImqBinary messageToken () const;**

**İleti simgesi** değerini döndürür.

### **ImqBoolean setMessageSimgesi (const ImqBinary & simge );**

**İleti belirtecini** belirler. *simge* veri uzunluğu sıfır ya da MQ\_MSG\_TOKEN\_LENGTH olmalıdır. Başarılı olursa TRUE değerini döndürür.

### **void setMessageSimgesi (const MQBYTE16 simge = 0);**

**İleti belirtecini** belirler. *simge* sıfır olabilir, bu da MQMOK\_NONE belirtilerek aynıdır. *simge* sıfırsa, ikili verilerin MQ\_MSG\_TOKEN\_LENGTH byte 'ları ile adreslenmelidir.

MQMOK\_NONE gibi önceden tanımlanmış değerleri kullanırken, imza eşleştirmesini sağlamak için bir tip dönüştürme yapmanız gerekebilir; örneğin, (MQBYTE \*) MQMOK\_NONE.

### **ImqString serviceName () const;**

Sondaki boşluklar da içinde olmak üzere, **hizmet adı** değerini döndürür.

### **void setServiceAd (const char \* ad );**

**Hizmet adı** ' yı ayarlar.

### **ImqString serviceStep () const;**

Sondaki boşluklar da dahil olmak üzere, **hizmet adımı** değerini döndürür.

### **void setServiceStep (const char \* adım );**

**Hizmet adımı** ' yı ayarlar.

## **Nesne verileri (korunan)**

### **MQWIH omqwih**

MQWIH veri yapısı.

## **Neden kodları**

- MQRC\_BINARY\_DATA\_LENGTH\_ERROR

## IBM MQ classes for JMS nesnelerinin özellikleri

IBM MQ classes for JMS içindeki tüm nesnelerin özellikleri vardır. Farklı özellikler farklı nesne tipleri için geçerlidir. Farklı özellikler farklı izin verilen değerlere sahiptir ve simgesel özellik değerleri, denetim aracı ile program kodu arasında farklılık gösterir.

IBM MQ classes for JMS provides facilities to set and query the properties of objects using the IBM MQ JMS administration tool, IBM MQ Explorer, or in an application. Özelliklerin çoğu, yalnızca nesne tiplerinin belirli bir altkümü için anlamlıdır.

IBM MQ JMS yönetim aracını kullanma şekliniz hakkında bilgi için bkz. [Yönetim aracını kullanarak JMS nesnelerini yapılandırma](#).

Çizelge 869 sayfa 1866 , her özelliğin kısa bir açıklamasını verir ve hangi nesne tiplerinin uygulanacağı her özellik için gösterilir. Nesne tipleri anahtar sözcükler kullanılarak tanımlanır; bu nesnelerin açıklaması için [Denetim aracını kullanarak JMS nesnelerinin yapılandırılması](#) başlıklı konuya bakın.

Sayılar, tablonun sonundaki notlara başvuruda bulunur. Ayrıca bkz. [“IBM MQ classes for JMS nesnelere ilişkin özellikler arasındaki bağımlılıklar” sayfa 1869](#).

Bir özellik, bir ad-değer çiftinden biçimden oluşur:

```
PROPERTY_NAME(property_value)
```

Bu bölüm listesindeki konular, her özellik için, özelliğin adı ve kısa bir açıklama ve denetim aracında kullanılan geçerli özellik değerleri gösterilir. ve bir uygulamadaki özelliğin değerini ayarlamak için kullanılan set metodudur. Bu konular ayrıca, araçta kullanılan simgesel özellik değerleri ile programlanabilir eşdeğerleri arasındaki eşlemeyi ve her özelliğe ilişkin geçerli özellik değerlerini de gösterir.

Özellik adları büyük ve küçük harfe duyarlı değildir ve bu konularda gösterilen tanınan adlar kümesine sınırlanır.

Özellik	Kısa Biçim	Nesne tipi							
		CF	QCF	TCF	Q	T	XACF	XAQCF	XATCF
<a href="#">“UYGULAMA ADI” sayfa 1871</a>	appName	Y	Y	Y			Y	Y	Y
<a href="#">“ASYNCEXCEPTION” sayfa 1871</a>	AEX	Y	Y	Y			Y	Y	Y
<a href="#">“BROKERCCDURSUBQ” sayfa 1872 <sup>1</sup></a>	CCDSUB					Y			
<a href="#">“BROKERCCSUQ” sayfa 1873 <sup>1</sup></a>	CCSID ' LER	Y		Y			Y		Y
<a href="#">“BROKERCONQ” sayfa 1873 <sup>1</sup></a>	SIMGE	Y		Y			Y		Y
<a href="#">“BROKERDURSUQ” sayfa 1874 <sup>1</sup></a>	BDSUB					Y			
<a href="#">“BROKERPUBQ” sayfa 1874 <sup>1</sup></a>	PUB	Y		Y		Y	Y		Y
<a href="#">“BROKERPUBQMGR” sayfa 1874 <sup>1</sup></a>	BPQM					Y			
<a href="#">“BROKERQMGR” sayfa 1875 <sup>1</sup></a>	BQM	Y		Y			Y		Y
<a href="#">“BROKERSUBQ” sayfa 1875 <sup>1</sup></a>	BSUB	Y		Y			Y		Y
<a href="#">“BROKERVER” sayfa 1876 <sup>1</sup></a>	BVER	E <sup>2</sup>		E <sup>2</sup>		Y	Y		Y
<a href="#">“CCDTURL” sayfa 1876 <sup>3</sup></a>	CCDT	Y	Y	Y			Y	Y	Y
<a href="#">“CCSID” sayfa 1877</a>	CCS	Y	Y	Y	Y	Y	Y	Y	Y
<a href="#">“Kanal” sayfa 1877 <sup>3</sup></a>	Chan	Y	Y	Y			Y	Y	Y

Çizelge 869. Özellik adları ve geçerli nesne tipleri (devamı var)

Özellik	Kısa Biçim	Nesne tipi							
		CF	QCF	TCF	Q	T	XACF	XAQCF	XATCF
"Temizle" sayfa 1878 <sup>1</sup>	CL	Y		Y			Y		Y
"CLEANUPINT" sayfa 1878 <sup>1</sup>	YNT	Y		Y			Y		Y
"CONNECTIONNAMELIST" sayfa 1879	CNIST	Y	Y	Y					
"İSTEMCILERİYÜZLER" sayfa 1879	ÇAPRAZ	Y	Y	Y					
"CLIENTRECONNECTTIMEOUT" sayfa 1880	CRT	Y	Y	Y					
"CLIENTID" sayfa 1881	CID	E <sup>2</sup>	Y	E <sup>2</sup>			Y	Y	Y
"ÇALIŞTIRMA" sayfa 1881	CLS	Y		Y			Y		Y
"KARMAŞIK" sayfa 1882	HC	Y		Y			Y		Y
"MSG" sayfa 1882	MC	Y	Y	Y			Y	Y	Y
"HAYIR" sayfa 1882	CNOPT	Y	Y	Y			Y	Y	Y
"CONNTAG" sayfa 1883	CNTAG	Y	Y	Y			Y	Y	Y
"AÇIKLAMA" sayfa 1884	ALÇ	E <sup>2</sup>	Y	E <sup>2</sup>	Y	Y	Y	Y	Y
"YÖN YÖNTEMÇİ" sayfa 1884	DAUTH	E <sup>2</sup>		E <sup>2</sup>					
"Kodlama" sayfa 1885	ENC				Y	Y			
"Son kullanma tarihi" sayfa 1886	ÜS				Y	Y			
"FAILIFQUIESCE" sayfa 1886	FIQ	Y	Y	Y	Y	Y	Y	Y	Y
"Anasistem adı" sayfa 1887	Anasistem	E <sup>2</sup>	Y	E <sup>2</sup>			Y	Y	Y
"YEREL ADRES" sayfa 1887	LA	E <sup>2</sup>	Y	E <sup>2</sup>			Y	Y	Y
"MAPNAMESTILI" sayfa 1888	MNST	Y	Y	Y			Y	Y	Y
"MAXBUFFSIZE" sayfa 1889	MBSZ	E <sup>2</sup>		E <sup>2</sup>					
"MDREAD" sayfa 1889	MDR				Y	Y			
"MDWRITE" sayfa 1890	MDC				Y	Y			
"MDMSGCTX" sayfa 1890	MDCTX				Y	Y			
"MSGBATCHSZ" sayfa 1891 <sup>1</sup>	MBS	Y	Y	Y			Y	Y	Y
"MSGONT" sayfa 1891	GÖVDE				Y	Y			
"MSGRECALOG" sayfa 1892	MRET	Y	Y				Y	Y	
"MSGSEÇEK" sayfa 1892 <sup>1</sup>	MSEL	Y		Y			Y		Y
"ÇOK YA" sayfa 1893	MCET	E <sup>2</sup>		E <sup>2</sup>		Y			
"EN İYİ BILDIRIM" sayfa 1893 <sup>1</sup>	OPTPUB	Y		Y					
"KALDIRIM" sayfa 1894 <sup>1</sup>	Bildirim Gönder	Y		Y					
"Kalıcılık" sayfa 1894	HER BİRİ İÇİN				Y	Y			

Çizelge 869. Özellik adları ve geçerli nesne tipleri (devamı var)

Özellik	Kısa Biçim	Nesne tipi							
		CF	QCF	TCF	Q	T	XACF	XAQCF	XATCF
<a href="#">“GİRİŞ TARİHİ” sayfa 1895 <sup>1</sup></a>	YAZI	Y	Y	Y			Y	Y	Y
<a href="#">“PORT” sayfa 1896</a>	PORT	E <sup>2</sup>	Y	E <sup>2</sup>			Y	Y	Y
<a href="#">“Öncelik” sayfa 1896</a>	PRI				Y	Y			
<a href="#">“SüREç” sayfa 1897 <sup>1</sup></a>	PROCDUR	Y		Y					
<a href="#">“SAĞLAMA Sü” sayfa 1897</a>	PVER	Y	Y	Y			Y	Y	Y
<a href="#">“PROXYHOSTNAME” sayfa 1899</a>	PHOST	E <sup>2</sup>		E <sup>2</sup>					
<a href="#">“TANITIM” sayfa 1900</a>	PKAPı	E <sup>2</sup>		E <sup>2</sup>					
<a href="#">“PUBKACT” sayfa 1900 <sup>1</sup></a>	PAI	Y		Y			Y		Y
<a href="#">“PUTASINCALLI” sayfa 1901</a>	PAALD				Y	Y			
<a href="#">“YöNETİM” sayfa 1901</a>	MMGR	Y	Y	Y	Y		Y	Y	Y
<a href="#">“kuyruk” sayfa 1902</a>	QU				Y				
<a href="#">“OKUNABİLENİZİN” sayfa 1902</a>	RALAND				Y	Y			
<a href="#">“READAHEADCLOSEPOLICY” sayfa 1903</a>	RACP				Y	Y			
<a href="#">“RECEIVECCSTnt” sayfa 1903</a>	RCCS				Y	Y			
<a href="#">“ALINMASI” sayfa 1904</a>	RCNV				Y	Y			
<a href="#">“RECEIVEI” sayfa 1904 <sup>1</sup></a>	RCVISOL	Y		Y					
<a href="#">“YENİDEN çıkış” sayfa 1905</a>	RCX	Y	Y	Y			Y	Y	Y
<a href="#">“YENİDEN KALDIRMA” sayfa 1905</a>	RRCXI	Y	Y	Y			Y	Y	Y
<a href="#">“REPLYTOSTYLE” sayfa 1906</a>	RTOST				Y	Y			
<a href="#">“SıFIRA” sayfa 1906 <sup>1</sup></a>	AYİR	Y	Y				Y	Y	
<a href="#">“SEçİLEN” sayfa 1907</a>	SCX	Y	Y	Y			Y	Y	Y
<a href="#">“GüNCELLEMİYİ” sayfa 1907</a>	SCXI	Y	Y	Y			Y	Y	Y
<a href="#">“SENDCHECKCOUNT” sayfa 1908</a>	Gizli	Y	Y	Y			Y	Y	Y
<a href="#">“SENDEXIT” sayfa 1908</a>	SSDX	Y	Y	Y			Y	Y	Y
<a href="#">“SENDEXITINIT” sayfa 1909</a>	SXI	Y	Y	Y			Y	Y	Y
<a href="#">“FARKLILIKLI” sayfa 1909</a>	SALD	Y	Y	Y			Y	Y	Y
<a href="#">“SPARSESUBS” sayfa 1910 <sup>1</sup></a>	SSUBS	Y		Y					
<a href="#">“SSLCIPHERSUIT” sayfa 1911</a>	SCPHS	Y	Y	Y			Y	Y	Y
<a href="#">“SLSCRL” sayfa 1911</a>	SCRL	Y	Y	Y			Y	Y	Y
<a href="#">“SSLFIPSREQUIRED” sayfa 1911</a>	SFIPS	Y	Y	Y			Y	Y	Y
<a href="#">“SLAYICI ADI” sayfa 1912</a>	SPEER	Y	Y	Y			Y	Y	Y
<a href="#">“SSLRESTCOUNT” sayfa 1912</a>	src	Y	Y	Y			Y	Y	Y
<a href="#">“DEYİM YENİDEN” sayfa 1913 <sup>1</sup></a>	SRI	Y		Y			Y		Y



Çizelge 869. Özellik adları ve geçerli nesne tipleri (devamı var)

Özellik	Kısa Biçim	Nesne tipi							
		CF	QCF	TCF	Q	T	XACF	XAQCF	XATCF
"ALT DÜZEN" sayfa 1913 <sup>1</sup>	SS	Y		Y			Y		Y
"SYNCPOINTALLGETS" sayfa 1914	SPAG	Y	Y	Y			Y	Y	Y
"HEDEF İSTEMCI" sayfa 1914	TC				Y	Y			
"TARGCLIENTEŞLEŞTİRME" sayfa 1915	TCM	Y	Y				Y	Y	
"TEMPMODEL" sayfa 1915	TM	Y	Y				Y	Y	
"TEMPQOPREFIX" sayfa 1916	TQP	Y	Y				Y	Y	
"TEMPTOPICFIX" sayfa 1916	TTP	Y		Y			Y		Y
"Konu" sayfa 1917	TOP					Y			
"Aktarım" sayfa 1917	TRAN	E <sup>2</sup>	Y	E <sup>2</sup>			Y	Y	Y
"JOKER BIÇIMI" sayfa 1918	WCFMT	Y		Y			Y		Y

**Not:**

1. Bu özellik, IBM MQ classes for JMS sürüm 70 ile kullanılabilir; ancak, bağlantı üreticisinin PROVIDERVERSION özelliği 7 'den küçük bir sürüm numarası olarak ayarlanmıyorsa, IBM WebSphere MQ 7.0 kuyruk yöneticisine bağlı bir uygulama için hiçbir etkisi yoktur.
2. Bir aracıya gerçek zamanlı bağlantı kullanıldığında, ConnectionFactory ya da TopicConnectionFactory nesnesi için yalnızca BROKERVER, CLIENTID, DESCRIPTION, DIRECTAUTH, HOSTNAME, LOCALADDRESS, MAXBUFFSIZE, MULTICAST, PORT, PROXYHOSTNAME, PROXYPORT ve TRANSPORT özellikleri desteklenmektedir.
3. Bir nesneye ilişkin CCDTURL ve KANAL özellikleri aynı anda ayarlanmamalıdır.

## IBM MQ classes for JMS nesnelere ilişkin özellikler arasındaki bağımlılıklar

Bazı özelliklerin geçerliliği, diğer özelliklerin belirli değerlerine bağlıdır.

Bu bağımlılık aşağıdaki özellik gruplarında oluşabilir:

- İstemci Özellikleri
- Bir aracıya gerçek zamanlı bağlantı için özellikler
- Başlatma dizgilerinden çık

### İstemci Özellikleri

Kuyruk yöneticisine yönelik bir bağlantı için, aşağıdaki özellikler yalnızca TRANSPORT değeri CLIENT ' ı varsa, ilgili özelliklerle ilgilidir:

- Anasistem adı
- PORT
- Kanal
- YEREL ADRES
- CCDTURL
- CCSID
- KARMAŞIK

- MSG
- YENIDEN çıkış
- YENIDEN KALDIRMA
- SEÇİLEN
- GÜNCELLEMİYİ
- SENDEXIT
- SENDEXITINIT
- SENABRALD
- SSLCIPHERSUIT
- SLSCRL
- SSLFIPSREQUIRE
- SLAYICI ADI
- SSLRESTCOUNT
- UYGULAMA ADI

TRANSPORT değeri BIND ' ye sahipse, denetim aracını kullanarak bu özellikler için değer ayarlayamazsınız.

TRANSPORT değeri CLIENT değeri varsa, BROKERVER özelliğinin varsayılan değeri V1 ' dir ve PORT özelliğinin varsayılan değeri 1414 ' tır. BROKERVER ya da PORT değerini belirttik olarak ayarladıysanız, daha sonra TRANSPORT değeri olarak yapılan bir değişiklik seçimlerinizi geçersiz kılmaz.

#### **Bir aracıya gerçek zamanlı bağlantı için özellikler**

TRANSPORT değeri DIRECT YA DA DIRECTHTTP değerine sahipse, yalnızca aşağıdaki özellikler ilişkilidir:

- BROKERVER
- CLIENTID
- AÇIKLAMA
- YÖN YÖNTEMCİ
- Anasistem adı
- YEREL ADRES
- MAXBUFFSIZE
- MULTICAST (yalnızca DIRECT için desteklenir)
- PORT
- PROXYHOSTNAME (yalnızca DIRECT için desteklenir)
- PROXYPET (yalnızca DIRECT için desteklenir)

TRANSPORT değeri DIRECT YA DA DIRECTHTTP değerine sahipse, BROKERVER özelliğinin varsayılan değeri V2' dir ve PORT (Kapı) özelliğinin varsayılan değeri 1506 ' tır. BROKERVER ya da PORT değerini belirttik olarak ayarladıysanız, daha sonra TRANSPORT değeri olarak yapılan bir değişiklik seçimlerinizi geçersiz kılmaz.

#### **Başlatma dizgilerinden çık**

Çıkış başlatma dizgilerinin herhangi birini, karşılık gelen çıkış adını belirtmeden ayarlamayın. Çıkış kullanıma hazırlama özellikleri şunlardır:

- YENIDEN KALDIRMA
- GÜNCELLEMİYİ
- SENDEXITINIT

For example, specifying REEXITINIT(myString) without specifying REEXIT(some.exit.classname) causes an error.

## İlgili başvurular

“Aktarım” sayfa 1917

Bir kuyruk yöneticisine ya da aracıya yönelik bağlantının niteliği.

## UYGULAMA ADI

Bir uygulama, kuyruk yöneticisiyle olan bağlantısını tanıtan bir ad ayarlayabilir. Bu uygulama adı **DISPLAY CONN MQSC/PCF** komutu tarafından gösterilir (alanın adı **APPLTAG** olan yerdir) ya da IBM MQ Explorer **Application Connections** (Uygulama Bağlantıları) görüntüsünde (alanın **App name** olarak adlandırıldığı yerde).

### Uygulanabilir Nesnelere

ConnectionFactory, QueueConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XAQueueConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: APPLICATIONNAME

JMS yönetim aracı kısa adı: APPNAME

### Programlı erişim


Ayarlayıcılar/alıcılar

- MQConnectionFactory.setAppName ()
- MQConnectionFactory.getAppName ()

### Değerler

28 karakterden uzun olmayan geçerli bir dize. Daha uzun adlar, baştaki paket adlarının kaldırılarak sığacağı şekilde ayarlanır. Örneğin, çağrılan sınıf `com.example.MainAppise`, tam ad kullanılır, ancak çağırılan sınıf `com.example.dictionaryAndThesaurus.multilingual.mainAppise`, bu ad, sınıf adının en uzun birleşimi ve kullanılabilir uzunluğa uyan en sağdaki paket adının en uzun birleşimi olduğu için `multilingual.mainApp` kullanılır.

Sınıf adının kendisi 28 karakterden uzunsa, sığaca kesilir. Örneğin, `com.example.mainApplicationForSecondTestCase`, `mainApplicationForSecondTestolur`.

 z/OS üzerinde, APPNAME aşağıdaki gibi olur:

- Ayarlandıysa ve ayarlanırsa, bağ tanımları kipi yoksayılar ve yalnızca boşluk karakterine ayarlanabilir.
- İstemci kipi ayarlanabilir ve kullanılabilir.

## ASYNCEXCEPTION

Bu özellik, IBM MQ classes for JMS 'un bir ExceptionListener (ExceptionListener) bir bağlantı kesildiğinde ya da bir JMS API çağrısıyla zamanuyumsuz olarak ne zaman olağandışı bir durum ortaya çıktığında bildirip bildirmeyeceğini belirtir. Bu, ExceptionListener 'un kayıtlı olduğu bu ConnectionFactory 'den oluşturulan tüm Connections için geçerlidir.

### Uygulanabilir nesnelere

ConnectionFactory, QueueConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XAQueueConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: ASYNCEXCEPTION

JMS yönetim aracı kısa adı: AEX

## Programlı erişim

Ayarlayıcılar/Getter

- MQConnectionFactory.setAsyncKural Dışı Durumlar ()
- MQConnectionFactory.getAsyncKural Dışı Durumlar ()

## Değerler

### ASYNC\_EXCEPTIONS\_ALL

Zamanuyumlu bir API çağrısının kapsamı dışında, zamanuyumsuz olarak algılanan herhangi bir kural dışı durum oluştu ve tüm bağlantı bozuk kural dışı durumları ExceptionListener' e gönderildi.

<i>Çizelge 870. Tüm zamanuyumsuz kural dışı durumlar: ortamlar ve ilgili sabit adlar</i>	
Ortam	Değer
JMS Denetim Aracı	TÜMÜ
Programlı	WMQCONSTANTS.ASYNC_EXCEPTIONS_ALL = -1
IBM MQ Explorer	Tümü

### ASYNC\_EXCEPTIONS\_CONNECTIONBROKEN

Yalnızca, bozuk bir bağlantının ExceptionListener' e gönderildiğini gösteren kural dışı durumlar vardır. Zamanuyumsuz işleme sırasında oluşan diğer kural dışı durumlar ExceptionListener(ExceptionListener) olarak bildirilmez ve bu nedenle, uygulama bu kural dışı durumlarla ilgili bilgilendirilmez. Bu, IBM MQ 8.0.0 Fix Pack 2' un varsayılan değeridir. Bkz. [JMS: IBM MQ 8.0'içinde kural dışı durum dinleyici değişiklikleri](#).

<i>Çizelge 871. Bozuk bir bağlantıyı gösteren kural dışı durumlar: ortamlar ve ilgili sabit adlar</i>	
Ortam	Değer
JMS Denetim Aracı	CONNECTIONBOZUK
Programlı	WMQCONSTANTS.ASYNC_EXCEPTIONS_CONNECTIONBROKEN = 1
IBM MQ Explorer	Bağlantı Kesildi

Aşağıdaki ek sabit değer tanımlıdır:

- IBM MQ 8.0.0 Fix Pack 2: WMQCONSTANTS.ASYNC\_EXCEPTIONS\_DEFAULT = ASYNC\_EXCEPTIONS\_CONNECTIONBROKEN
- IBM MQ 8.0.0 Fix Pack 2 tarihinden önce: WMQCONSTANTS.ASYNC\_EXCEPTIONS\_DEFAULT = ASYNC\_EXCEPTIONS\_ALL

### İlgili kavramlar

[IBM MQ classes for JMS'içinde kural dışı durumlar](#)

## BROKERCCDURSUBQ

ConnectionConsumer(Bağlantı Tüketicisi) için dayanıklı abonelik iletilerinin alındığı kuyruğun adı.

### Uygulanabilir nesnelere

Konu

JMS yönetim aracı uzun adı: BROKERCCDURSUBQ

JMS yönetim aracı kısa adı: CCDSUB

## Programlı erişim

Ayarlayıcılar/alıcılar

- MQTopic.setBrokerCCDurSubQueue()
- MQTopic.getBrokerCCDurSubQueue()

## Değerler

**SYSTEM.JMS.D.CC.SUBSCRIBER.QUEUE**

Bu varsayılan değerdir.

**Herhangi bir geçerli dizgi**

## BROKERCCSUQ

ConnectionConsumer(Bağlantı Tüketicisi) için kalıcı olmayan abonelik iletilerinin alındığı kuyruğun adı.

## Uygulanabilir nesnelere

ConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: BROKERCCSUBQ

JMS yönetim aracı kısa adı: CCSUB

## Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setBrokerCCSubQueue()
- MQConnectionFactory.getBrokerCCSubQueue()

## Değerler

**SYSTEM.JMS.ND.CC.SUBSCRIBER.QUEUE**

Bu varsayılan değerdir.

**Herhangi bir geçerli dizgi**

## BROKERCONQ

Aracıya ilişkin denetim kuyruğu adı.

## Uygulanabilir Nesnelere

ConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: BROKERCONQ

JMS yönetim aracı kısa adı: BCON

## Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setBrokerControlQueue()
- MQConnectionFactory.getBrokerControlQueue()

## Değerler

**SYSTEM.BROKER.CONTROL.QUEUE**

Bu varsayılan değerdir.

**Herhangi bir geçerli dizgi**

## **BROKERDURSUQ**

IBM MQ classes for JMS , IBM MQ ileti alışverişi sağlayıcısı geçiş kipinde kullanılıyorsa, bu özellik, kalıcı abonelik iletilerinin alındığı kuyruğun adını belirtir.

### **Uygulanabilir nesnelere**

Konu

JMS yönetim aracı uzun adı: BROKERDURSUBQ

JMS yönetim aracı kısa adı: BDSUB

### **Programlı erişim**

Ayarlayıcılar/alıcılar

- MQTopic.setBrokerDurSubQueue()
- MQTopic.getBrokerDurSubQueue()

### **Değerler**

#### **SYSTEM.JMS.D.SUBSCRIBER.QUEUE**

Bu varsayılan değerdir.

#### **Herhangi bir geçerli dizgi**

SYSTEM.JMS.D

### **İlgili görevler**

JMS **PROVIDERVERSION** özelliğini yapılandırma

## **BROKERPUBQ**

Yayınlanan iletilerin gönderildiği kuyruğun adı (akış kuyruğu).

### **Uygulanabilir Nesnelere**

ConnectionFactory, TopicConnectionFactory, Topic, XAConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: BROKERPUBQ

JMS yönetim aracı kısa adı: BPUB

### **Programlı erişim**

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setBrokerPubQueue
- MQConnectionFactory.getBrokerPubQueue

### **Değerler**

#### **SYSTEM.BROKER.DEFAULT.STREAM**

Bu varsayılan değerdir.

#### **Herhangi bir geçerli dizgi**

## **BROKERPUBQMGR**

Konu üzerinde yayınlanan iletilerin gönderildiği kuyruğun sahibi olan kuyruk yöneticisinin adı.

## Uygulanabilir Nesneler

Konu

JMS yönetim aracı uzun adı: BROKERPUBQMGR

JMS yönetim aracı kısa adı: BPQM

## Programlı erişim

Ayarlayıcılar/alıcılar

- MQTopic.setBrokerPubQueueManager()
- MQTopic.getBrokerPubQueueManager()

## Değerler

### boş değerli

Bu varsayılan değerdir.

### Herhangi bir geçerli dizgi

## BROKERQMGR

Aracının çalışmakta olduğu kuyruk yöneticisinin adı.

## Uygulanabilir Nesneler

ConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: BROKERQMGR

JMS yönetim aracı kısa adı: BQM

## Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setBrokerQueueManager()
- MQConnectionFactory.getBrokerQueueManager()

## Değerler

### boş değerli

Bu varsayılan değerdir.

### Herhangi bir geçerli dizgi

## BROKERSUBQ

IBM MQ classes for JMS , IBM MQ ileti alışverişi sağlayıcısı geçiş kipinde kullanılıyorsa, bu özellik, kalıcı olmayan abonelik iletilerinin alındığı kuyruğun adını belirtir.

## Uygulanabilir Nesneler

ConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: BROKERSUBQ

JMS yönetim aracı kısa adı: BSUB

## Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setBrokerSubQueue()
- MQConnectionFactory.getBrokerSubQueue()

## Değerler

### SYSTEM.JMS.ND.SUBSCRIBER.QUEUE

Bu varsayılan değerdir.

### Herhangi bir geçerli dizgi

SYSTEM.JMS.ND

## İlgili görevler

JMS **PROVIDERVERSION** özelliğini yapılandırma

## BROKERVER

Kullanılmakta olan aracının sürümü.

## Uygulanabilir Nesnelere

ConnectionFactory, TopicConnectionFactory, Topic, XAConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: BROKERVER

JMS yönetim aracı kısa adı: BVER

## Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setBrokerVersion ()
- MQConnectionFactory.getBrokerVersion ()

## Değerler

### V1

Bir IBM MQ Yayınlama/Abone Olma aracısını kullanmak ya da bir IBM MQ Integrator, WebSphere Event Broker, WebSphere Business Integration Event Broker ya da WebSphere Business Integration Message Broker aracısını uyumluluk kipinde kullanmak için. TRANSPORT, BIND ya da CLIENT olarak ayarlandıysa, varsayılan değer budur.

### V2

To use a broker of IBM MQ Integrator, WebSphere Event Broker, WebSphere Business Integration Event Broker, or WebSphere Business Integration Message Broker in native mode. TRANSPORT, DIRECT ya da DIRECTHTTP olarak ayarlandıysa, bu varsayılan değerdir.

### belirlenmedi

After the broker has migrated from V6 to V7, set this property so that RFH2 headers are no longer used. Geçişten sonra bu özellik artık ilgili değildir.

## CCDTURL

İstemci kanal tanımlama çizelgesini içeren dosyanın adını ve yerini tanıtan ve dosyaya nasıl erişilebileceğini belirten bir URL adresi (URL).



## Uygulanabilir Nesnelere

ConnectionFactory, QueueConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XAQueueConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: CCDTURL

JMS yönetim aracı kısa adı: CCDT

## Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setCCDTURL()
- MQConnectionFactory.getCCDTURL()

## Değerler

### boş değerli

Bu varsayılan değerdir.

### Tek Biçimli Kaynak Yeri Belirleyici (URL)

## CCSID

Bağlantı üreticileri için bu özellik, kuyruk yöneticisiyle iç veri akışları için kullanılacak kodlanmış karakter takımı tanıtcısını (CCSID) belirler. Hedefler için bu özellik, MapMessages, StreamMessages ve TextMessages içindeki dizgi verilerini bu hedefe kodlamak için kullanılacak CCSID 'yi tanımlar.

**Not:** Normalde bağlantı üreticileri için bu özelliğin değiştirilmesi gerekmez.

## Uygulanabilir Nesnelere

ConnectionFactory, QueueConnectionFactory, TopicConnectionFactory, Queue, Topic, XAConnectionFactory, XAQueueConnectionFactory, XATopicConnectionFactory

JMS denetim aracı uzun adı: CCSID

JMS yönetim aracı kısa adı: CCS

## Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setCCSID()
- MQConnectionFactory.getCCSID()

## Değerler

### 819

Bir bağlantı üreticisinin varsayılan değeri.

### 1208

Bir hedefe ilişkin varsayılan değer.

### Pozitif bir tamsayı

### İlgili kavramlar

[JMS ileti dönüştürme](#)

## Kanal

Kullanılmakta olan istemci bağlantı kanalının adı.

## Uygulanabilir Nesneler

ConnectionFactory, QueueConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XAQueueConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: KANAL

JMS yönetim aracı kısa adı: CHAN

## Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setChannel()
- MQConnectionFactory.getChannel()

## Değerler

### SYSTEM.DEF.SVRCONN

Bu varsayılan değerdir.

### Herhangi bir geçerli dizgi

## Temizle

BROKER için Temizleme Düzeyi ya da Abonelik Depoları YENI Düzeye Geçir.

## Uygulanabilir Nesneler

ConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: CLEANUP

JMS yönetim aracı kısa adı: CL

## Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setCleanupDüzey ()
- MQConnectionFactory.getCleanupDüzey ()

## Değerler

### KASA

Güvenli temizlik kullanın. Bu varsayılan değerdir.

### ASPROP

Java komut satırındaki bir özelliğe göre güvenli, güçlü ya da temizleme işlemini kullanmayın.

### YOK

Temizleme işlemini kullanmayın.

### güçlü

Güçlü bir temizlik kullanın.

## CLEANUPINT

Yayınlama/abone olma temizleme yardımcı programının arka plan yürütmeleri arasındaki aralık, milisaniye cinsinden.

## Uygulanabilir Nesneler

ConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: CLEANUPINT

JMS yönetim aracı kısa adı: CLINT

### **Programlı erişim**

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setCleanupInterval ()
- MQConnectionFactory.getCleanupInterval ()

### **Değerler**

**3600000**

Bu varsayılan değerdir.

**Pozitif bir tamsayı**

## **CONNECTIONNAMELIST**

TCP/IP bağlantı adlarının listesi. Liste, her bir yeniden bağlanma yeniden deneme girişimi başına bir kez denenir.

### **Uygulanabilir Nesneler**

ConnectionFactory, QueueConnectionFactory, TopicConnectionFactory

JMS yönetim aracı uzun adı: CONNECTIONNAMELIST

JMS yönetim aracı kısa adı: CNLIST

### **Programlı erişim**

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setconnectionNameList ()
- MQConnectionFactory.getconnectionNameList ()

### **Değerler**

HOSTNAME (PORT) listesinin virgülle ayrılmış listesi. HOSTNAME, bir DNS adı ya da IP adresi olabilir.

PORT varsayılan değeri 1414 olarak ayarlanır.

## **İSTEMCILERİYÜZLER**

Yeniden bağlantıyı yöneten seçenekler.

### **Uygulanabilir Nesneler**

ConnectionFactory, QueueConnectionFactory, TopicConnectionFactory

JMS yönetim aracı uzun adı: CLIENTRECONNECTORTIONS

JMS yönetim aracı kısa adı: CROST

### **Programlı erişim**

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setClientReconnectOptions()
- MQConnectionFactory.getClientReconnectOptions()

## Değerler

### MMGR

Uygulama, başlangıçta bağlı olduğu kuyruk yöneticisine yeniden bağlanabilir.

Uygulamanın bağlantı adı listesinde belirtildiği gibi bağlanmayı denediği kuyruk yöneticisi, başlangıçta bağlı olduğu kuyruk yöneticisinde farklı bir QMID ' ye sahip olduğunda, MQRC\_RECONNECT\_QMID\_MISSATCH neden koduyla bir hata döndürdü.

Bir uygulama yeniden bağlanabiliyorsa bu değeri kullanın, ancak IBM MQ classes for JMS uygulaması arasında bir benzeşim varsa ve ilk olarak bağlantı kurduğu kuyruk yöneticisi arasında bir benzerlik vardır.

Uygulamanın, yüksek kullanılabilirlikli bir kuyruk yöneticisinin yedek yönetim ortamına otomatik olarak yeniden bağlanmasını istiyorsanız bu değeri seçin.

Bu değeri programlı olarak kullanmak için sabit WMQConstants.WMQ\_CLIENT\_RECONNECT\_Q\_MGRdeğerini kullanın.

### HERHANGİ BİRİ

Uygulama, bağlantı adı listesinde belirtilen kuyruk yöneticilerinden herhangi birine yeniden bağlanabiliyor.

Yeniden bağlanma seçeneğini yalnızca, JMS için IBM MQ sınıfları arasında bir benzerlik yoksa ve başlangıçta bağlantı kuran kuyruk yöneticisi arasında bir benzerlik yoksa kullanın.

Bu değeri bir programdan kullanmak için sabit WMQConstants.WMQ\_CLIENT\_RECONNECT' i kullanın.

### DEVRE DIŞI

Uygulama yeniden bağlanmayacak.

Bu değeri programlı olarak kullanmak için sabit WMQConstants.WMQ\_CLIENT\_RECONNECT\_DISABLEDdeğerini kullanın.

### ASDEF

Uygulamanın otomatik olarak yeniden bağlanacağı IBM MQ kanal özneteliğinin değerine bağlı olarak DefReconnect' in değerine bağlıdır.

Bu varsayılan değerdir.

Bu değeri bir programdan kullanmak için sabit WMQConstants.WMQ\_CLIENT\_RECONNECT\_AS\_DEF' i kullanın.

## CLIENTRECONNECTTIMEOUT

Yeniden bağlantı yeniden denemelerinin sona ermesinden önceki süre.

### Uygulanabilir Nesnelere

ConnectionFactory, QueueConnectionFactory, TopicConnectionFactory

JMS yönetim aracı uzun adı: CLIENTRECONNECTTIMEOUT

JMS yönetim aracı kısa adı: CRT

### Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setClientReconnectTimeout()
- MQConnectionFactory.setClientReconnectTimeout()

## Değerler

Saniye cinsinden aralık. Varsayılan 1800 (30 dakika).

## CLIENTID

İstemci tanıtıcısı, kalıcı abonelikler için uygulama bağlantısını benzersiz bir şekilde tanımlamak için kullanılır.

### Uygulanabilir Nesnelere

ConnectionFactory, QueueConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XAQueueConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: CLIENTID

JMS yönetim aracı kısa adı: CID

### Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setClientId ()
- MQConnectionFactory.getClientId ()

## Değerler

### boş değerli

Bu varsayılan değerdir.

### Herhangi bir geçerli dizgi

## ÇALIŞTIRMA

Aynı kalıcı konu abonesinin iki ya da daha fazla örneğinin aynı anda çalıştırılıp çalıştırmayacağı.

### Uygulanabilir Nesnelere

ConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: CLONESUPP

JMS yönetim aracı kısa adı: CLS

### Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setCloneSupport ()
- MQConnectionFactory.getCloneDestegi ()

## Değerler

### DEVRE DIŞI

Bir kerede, dayanıklı bir konu abonesinin yalnızca bir eşgörünümü çalışabilir. Bu varsayılan değerdir.

### Etkinleştirildi

Aynı kalıcı konu abonesinin iki ya da daha çok eşgörünümü aynı anda çalışabilir, ancak her yönetim ortamı ayrı bir Java sanal makinesinde (JVM) çalıştırılmalıdır.

## KARMAŞIK

Bir bağlantıda üstbilgi verilerinin sıkıştırılması için kullanılacak tekniklerin listesi.

### Uygulanabilir Nesnelere

ConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: COMMPHDR

JMS yönetim aracı kısa adı: HC

### Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setHdrCompList()
- MQConnectionFactory.getHdrCompList()

### Değerler

#### YOK

Bu varsayılan değerdir.

#### SYSTEM

RLE iletisi üstbilgisi sıkıştırması gerçekleştirilir.

## MSG

Bir bağlantıda ileti verilerinin sıkıştırılması için kullanılacak tekniklerin listesi.

### Uygulanabilir Nesnelere

ConnectionFactory, QueueConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XAQueueConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: COMMSMSG

JMS yönetim aracı kısa adı: MC

### Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setMsgCompList()
- MQConnectionFactory.getMsgCompList()

### Değerler

#### YOK

Bu varsayılan değerdir.

**Aşağıdaki değerlerden biri ya da birkaçının listesi boş karakterlerle ayrılır:**

RLE ZLIBFAST ZLIBHIGH

## HAYIR

Bağ tanımı iletimi kullanan IBM MQ classes for JMS uygulamalarının kuyruk yöneticisine nasıl bağlanacağını denetler.

## Uygulanabilir Nesneler

ConnectionFactory, QueueConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XAQueueConnectionFactory, XATopicConnectionFactory.

JMS yönetim aracı uzun adı: CONNOPT

JMS yönetim aracı kısa adı: CNOPT

## Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setMQConnectionFactorySeçenekleri ()
- MQConnectionFactory.getMQConnectionFactorySeçenekleri ()

## Değerler

### Standart

Uygulama ile kuyruk yöneticisi arasındaki bağlayıcının niteliği, kuyruk yöneticisinin *DefaultBindType* özniteliğinin değerine bağlıdır. STANDART değer, IBM MQ *ConnectOption* MQCNO\_STANDART\_BINDING ile eşlenir.

### Paylaşılan

Uygulama ve yerel kuyruk yöneticisi aracı ayrı yürütme birimlerinde çalıştırılıyor, ancak bazı kaynakları paylaşıyor. Bu değer, IBM MQ *ConnectOption* MQCNO\_SHARED\_BINDING ile eşlenir.

### Yalıtılmış

Uygulama ve yerel kuyruk yöneticisi aracı ayrı yürütme birimlerinde çalıştırılıyor ve hiçbir kaynağı paylaşmaz. Yalıtılmış değer, IBM MQ *ConnectOption* MQCNO\_ISOLATED\_BINDING ile eşlenir.

### FastPath

Uygulama ve yerel kuyruk yöneticisi aracı aynı yürütme biriminde çalıştırılıyor. Bu değer, IBM MQ *ConnectOption* MQCNO\_FASTPATH\_BINDING ile eşlenir.

### SERIALQM

Uygulama, kuyruk yöneticisi kapsamı içinde bağlantı etiketinin dışlayıcı kullanımını ister. Bu değer, IBM MQ *ConnectOption* MQCNO\_SERIALIZE\_CONN\_TAG\_Q\_MGR ile eşlenir.

### SERIALQSG

Uygulama, bağlantı etiketinin kuyruk yöneticisinin ait olduğu kuyruk paylaşım grubu kapsamında dışlayıcı olarak kullanılmasını ister. SERIALQSG değeri, IBM MQ *ConnectOption* MQCNO\_SERIALIZE\_CONN\_TAG\_QSG ile eşlenir.

### RESTRICTQM

Uygulama, bağlantı etiketinin paylaşılan kullanımını istiyor, ancak kuyruk yöneticisi kapsamı içinde bağlantı etiketinin paylaşılan kullanımıyla ilgili kısıtlamalar var. Bu değer, IBM MQ *ConnectOption* MQCNO\_RESTRICT\_CONN\_TAG\_Q\_MGR ile eşlenir.

### RESTRICTQSG

Uygulama, bağlantı etiketinin paylaşılan kullanımını ister; ancak, kuyruk yöneticisinin ait olduğu kuyruk paylaşım grubu kapsamında, bağlantı etiketinin paylaşılan kullanımına ilişkin kısıtlamalar vardır. Bu değer, IBM MQ *ConnectOption* MQCNO\_RESTRICT\_CONN\_TAG\_QSG ile eşlenir.

IBM MQ bağlantı seçeneklerine ilişkin ek bilgi için [MQCONNX çağrısını kullanarak kuyruk yöneticisine bağlanmabaşlıklı konuya](#) bakın.

## CONNTAG

Kuyruk yöneticisinin, uygulama kuyruk yöneticisine bağlıken, bir iş birimi içinde uygulama tarafından güncellenen kaynaklarla ilişkilendirildiği bir etiket.

## Uygulanabilir Nesnelere

ConnectionFactory, QueueConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XAQueueConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: CONNTAG

JMS yönetim aracı kısa adı: CNTAG

## Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setConnEtiketi ()
- MQConnectionFactory.getConnEtiketi ()

## Değerler

**128 öge içeren bayt dizisi, her ögenin 0 olduğu**

Bu varsayılan değerdir.

**Herhangi bir dizgi**

Değer 128 byte 'tan uzunsa kesilir.

## AÇIKLAMA

Saklanan nesnenin tanımlaması.

## Uygulanabilir Nesnelere

ConnectionFactory, QueueConnectionFactory, TopicConnectionFactory, Kuyruk, Konu, XAConnectionFactory, XAQueueConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: DESCRIPTION

JMS yönetim aracı kısa adı: DESC

## Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setDescription()
- MQConnectionFactory.getDescription()

## Değerler

**boş değerli**

Bu varsayılan değerdir.

**Herhangi bir geçerli dizgi**

## YÖN YÖNTEMCİ

TLS kimlik doğrulamasının bir aracıya gerçek zamanlı bir bağlantıda kullanılıp kullanılmayacağı.

## Uygulanabilir Nesnelere

ConnectionFactory, TopicConnectionFactory

JMS yönetim aracı uzun adı: DIRECTAUTH

JMS yönetim aracı kısa adı: DAUTH



## Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setDirectAuth ()
- MQConnectionFactory.getDirectAuth ()

## Değerler

### Temel Düzey

Kimlik doğrulama, kullanıcı adı kimlik doğrulaması ya da parola kimlik doğrulaması yok. Bu varsayılan değerdir.

### Sertifika

Genel anahtar sertifikası kimlik doğrulaması.

## Kodlama

Bu hedefe gönderilen ileti gönderildiğinde, bir iletinin gövdesindeki sayısal veriler gösterilir. Bu özellik, ikili tamsayıların, paketlenmiş onlu tamsayıların ve kayan noktalı sayıların gösterilmesini belirtir.

## Uygulanabilir Nesnelere

Kuyruk, Konu

JMS yönetim aracı uzun adı: ENCODING

JMS yönetim aracı kısa adı: ENC

## Programlı erişim

Ayarlayıcılar/alıcılar

- MQDestination.setEncoding()
- MQDestination.getEncoding()

## Değerler

### KODLAMA özelliği

ENCODING özelliğinin alabileceği geçerli değerler üç alt özellikten oluşturulur:

#### tamsayı kodlaması

Olağan ya da ters çevrildi

#### ondalık kodlama

Olağan ya da ters çevrildi

#### kayan noktalı kodlama

IEEE normal, IEEE tersine çevrildi ya da z/OS

ENCODING özelliği, aşağıdaki sözdizimine sahip üç karakterlik bir dizgi olarak ifade edilir:

```
{N|R}{N|R}{N|R}3
```

Bu dizgide:

- N olağan anlamına gelir
- R tersine çevrildi
- 3 , z/OS anlamına gelir
- İlk karakter *tamsayı kodlaması*' ı temsil eder
- İkinci karakter *ondalık encoding* değerini gösterir

- Üçüncü karakter *kayan nokta kodlaması* temsil eder.

Bu, ENCODING özelliği için 12 olası değer kümesi sağlar.

Java platformu için uygun kodlama değerlerini belirleyen ek bir değer ( NATIVEDizgisi) vardır.

Aşağıdaki örneklerde, ENCODING için geçerli birleşimler gösterilmektedir:

```
ENCODING (NNR)
ENCODING (NATIVE)
ENCODING (RR3)
```

## Son kullanma tarihi

Bir hedefteki iletilerin süresinin dolduğu saat.

### Uygulanabilir Nesnelere

Kuyruk, Konu

JMS yönetim aracı uzun adı: EXPIRY

JMS yönetim aracı kısa adı: EXP

### Programlı erişim

Ayarlayıcılar/alıcılar

- MQDestination.setExpiry()
- MQDestination.getExpiry()

### Değerler

#### Uygulama

Süre bitimi, JMS uygulaması tarafından tanımlanabilir. Bu varsayılan değerdir.

#### UNLIM

Süre bitimi yok.

#### 0

Süre bitimi yok.

**Milisaniye cinsinden süre bitimini gösteren herhangi bir artı tamsayı.**

## FAILIFQUIESCE

This property determines whether calls to certain methods fail if either the queue manager is in a quiescing state, or an application is connecting to a queue manager using the CLIENT transport and the channel that the application is using has been put into a quiescing state, for example, by using the **STOP CHANNEL** or **STOP CHANNEL MODE (QUIESCE)** MQSC command.

### Uygulanabilir Nesnelere

ConnectionFactory, QueueConnectionFactory, TopicConnectionFactory, Kuyruk, Konu, XAConnectionFactory, XAQueueConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: FAILIFQUIESCE

JMS yönetim aracı kısa adı: FIQ

### Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setFailIfQuiesce()
- MQConnectionFactory.getFailIfQuiesce()

## Değerler

### EVET

Kuyruk yöneticisi susturma durumundaysa ya da bir kuyruk yöneticisine bağlanmak için kullanılan kanal susturuyorsa, bazı yöntemlere çağrılar başarısız olur. Bir uygulama bu koşullardan birini saptarsa, uygulama hemen görevini tamamlayıp bağlantıyı kapatabilir ve kuyruk yöneticisi ya da kanal örneğinin durmasını sağlar. Bu varsayılan değerdir.

### HAYIR

Kuyruk yöneticisi ya da kuyruk yöneticisine bağlanmak için kullanılan kanal susturulmuş durumda olduğundan, yöntem çağrısı başarısız olur. Bu değeri belirtirseniz, bir uygulama kuyruk yöneticisi ya da kanalının susturulmuş olduğunu algılayamaz. Uygulama kuyruk yöneticisine yönelik işlemleri gerçekleştirmeye devam edebilir ve bu nedenle kuyruk yöneticisinin durmasını engelleyebilir.

## Anasistem adı

Kuyruk yöneticisine yönelik bir bağlantı için, kuyruk yöneticisinin çalıştığı sistemin anasistem adı ya da IP adresi ya da aracıya ilişkin gerçek zamanlı bir bağlantı için, aracının çalıştığı sistemin anasistem adı ya da IP adresi.

## Uygulanabilir Nesnelere

ConnectionFactory, QueueConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XAQueueConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: HOSTNAME

JMS yönetim aracı kısa adı: HOST

## Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setHostName ()
- MQConnectionFactory.getHostName ()

## Değerler

### localhost

Bu varsayılan değerdir.

### Herhangi bir geçerli dizgi

## YEREL ADRES

Kuyruk yöneticisine yönelik bir bağlantı için, bu özellik kullanılacak yerel ağ arabirimini ya da kullanılacak yerel kapı ya da yerel kapı aralığını belirtir.

## Uygulanabilir Nesnelere

ConnectionFactory, QueueConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XAQueueConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: LOCALADDRESS

JMS yönetim aracı kısa adı: LA

## Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setLocalAddress ()
- MQConnectionFactory.getLocalAddress ()

## Değerler

### "" (boş dizgi)

Bu varsayılan değerdir.

### Biçimdeki bir dizgi [ ip-addr] [ (low-port [, yüksek kapı])]

Bazı örnekler:

192.0.2.0

The channel binds to address 192.0.2.0 locally.

192.0.2.0(1000)

The channel binds to address 192.0.2.0 locally and uses port 1000.

192.0.2.0(1000,2000)

The channel binds to address 192.0.2.0 locally and uses a port in the range 1000 to 2000.

(1000)

Kanal 1000 numaralı kapıya yerel olarak bağlanıyor.

(1000,2000)

Kanal, 1000 ile 2000 aralığındaki bir kapıya yerel olarak bağlanıyor.

IP adresi yerine bir anasistem adı belirleyebilirsiniz. Bir aracıya gerçek zamanlı bağlantı için, bu özellik yalnızca çoklu yayın kullanıldığında anlamlıdır ve özelliğin değeri bir kapı numarası ya da bir kapı numarası aralığı içermemelidir. Bu durumda özelliğin tek geçerli değerleri boş değerli, bir IP adresi ya da bir anasistem adıdır.

## MAPNAMESTILI

MapMessage ögesi adları için uyumluluk stilinin kullanılmasını sağlar.

## Uygulanabilir Nesnelere

ConnectionFactory, QueueConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XAQueueConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: MAPNAMESTYLE

JMS yönetim aracı kısa adı: MNST

## Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setMapNameStyle()
- MQConnectionFactory.getMapNameStyle()

## Değerler

### Standart

Standart com.ibm.jms.JMSMapMessage ögesi adlandırma biçimi kullanılır. Bu varsayılan değerdir ve yasal olmayan Java tanıtıcılarının öge adı olarak kullanılmasına olanak sağlar.

## Uyumlu

Daha eski com.ibm.jms.JMSMapMessage ögesi adlandırma biçimi kullanılır. Öge adı olarak yalnızca yasal Java tanıtıcıları kullanılabilir. Bu, yalnızca eşleme iletileri 5.3 tarihinden önceki bir IBM MQ classes for JMS sürümünü kullanan bir uygulamaya gönderiliyorsa gereklidir.

## MAXBUFFSIZE

Uygulama tarafından işlenmeyi beklerken bir iç ileti arabelleğinde saklanabilen, alınan ileti sayısı üst sınırı. Bu özellik yalnızca TRANSPORT değeri DIRECT ya da DIRECTHTTP değerine sahip olduğunda geçerlidir.

### Uygulanabilir Nesnelere

ConnectionFactory, TopicConnectionFactory

JMS yönetim aracı uzun adı: MAXBUFFSIZE

JMS yönetim aracı kısa adı: MBSZ

### Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setMaxBufferSize()
- MQConnectionFactory.getMaxBufferSize()

### Değerler

#### 1000

Bu varsayılan değerdir.

#### Pozitif bir tamsayı

## MDREAD

Bu özellik, bir JMS uygulamasının MQMD alanlarının değerlerini ayıklayıp ayıklayamayacağını belirler.

### Uygulanabilir Nesnelere

JMS yönetim aracı uzun adı: MDREAD

JMS yönetim aracı kısa adı: MDR

### Programlı erişim

Ayarlayıcılar/alıcılar

- MQDestination.setMQMDReadEnabled()
- MQDestination.getMQMDReadEnabled()

### Değerler

#### HAYIR

İletileri gönderirken, gönderilen bir iletteki JMS\_IBM\_MQMD\* özellikleri, MQMD 'deki güncellenen alan değerlerini yansıtacak şekilde güncellenmez. İleti alınırken, gönderenin bir kısmını ya da tümünü ayarlasa bile, alınan bir iletide JMS\_IBM\_MQMD\* özelliklerinin hiçbiri kullanılabilir değil. Bu, denetim araçları için varsayılan değerdir.

Programlar için False değerini kullanın.

#### Evet

İletileri gönderirken, gönderilen bir iletteki tüm JMS\_IBM\_MQMD\* özellikleri, gönderenin belirttik olarak ayarlamadığı özellikler de içinde olmak üzere, MQMD 'deki güncellenmiş alan değerlerini

yansıtabak şekilde güncellenir. İleti alınırken, tüm JMS\_IBM\_MQMD\* özellikleri, gönderenin belirtik olarak ayarlamadıđı özellikler de içinde olmak üzere, alınan bir iletilerde kullanılabilir.

Programlar için True (Dođru) seçeneđini kullanın.

## MDWRITE

Bu özellik, bir JMS uygulamasının MQMD alanlarının deđerlerini ayarlayıp ayarlayamayacađını belirler.

### Uygulanabilir Nesnelere

Kuyruk, Konu

JMS yönetim aracı uzun adı: MDWRITE

JMS yönetim aracı kısa adı: MDR

### Programlı erişim

Ayarlayıcılar/alıcılar

- MQDestination.setMQMDWriteEnabled()
- MQDestination.getMQMDWriteEnabled()

### Deđerler

#### HAYIR

Tüm JMS\_IBM\_MQMD\* özellikleri yoksayılr ve deđerleri, temeldeki MQMD yapısıyla kopyalanmaz. Bu, denetim araçları için varsayılan deđerdir.

Programlar için False deđerini kullanın.

#### EVET

JMS\_IBM\_MQMD\* özellikleri işlendi. Deđerleri, temeldeki MQMD yapısına kopyalanır.

Programlar için True (Dođru) seçeneđini kullanın.

## MDMSGCTX

What level of message context is to be set by the JMS application. Uygulamanın, bu özelliđin geçerli olması için uygun bağlam yetkisi ile çalışıyor olması gerekir.

### Uygulanabilir Nesnelere

JMS yönetim aracı uzun adı: MDMSGCTX

JMS yönetim aracı kısa adı: MDCTX

### Programlı erişim

Ayarlayıcılar/alıcılar

- MQDestination.setMQMDMessageContext()
- MQDestination.getMQMDMessageContext()

### Deđerler

#### VARSAYILAN

MQOPER API çağırısı ve MQPMO yapısı belirtik ileti bağlamı seçeneklerini belirtmiyor. Bu, denetim araçları için varsayılan deđerdir.

Programlar için WMQ\_MDCTX\_VARSAYILAN deđerini kullanın.

### **SET\_IDENTITY\_CONTEXT**

MQOPER API çağırısı, MQOO\_SET\_IDENTITY\_CONTEXT ileti bağlamı seçeneğini belirtir ve MQPMO yapısı MQPMO\_SET\_IDENTITY\_CONTEXT belirtisini belirtir.

Programlar için WMQ\_MDCTX\_SET\_IDENTITY\_CONTEXT kullanın.

### **SET\_ALL\_CONTEXT**

MQOPEP API çağırısı, MQOO\_SET\_ALL\_CONTEXT ileti bağlamı seçeneğini belirtir ve MQPMO yapısı MQPMO\_SET\_ALL\_CONTEXT 'yi belirtir.

Programlar için WMQ\_MDCTX\_SET\_ALL\_CONTEXT kullanın.

## **MSGBATCHSZ**

Zamanuyumsuz ileti teslimi kullanılırken bir paketteki bir kuyruktan alınacak ileti sayısı üst sınırı.

### **Uygulanabilir Nesneler**

ConnectionFactory, QueueConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XAQueueConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: MAXBUFFSIZE

JMS yönetim aracı kısa adı: MBSZ

### **Programlı erişim**

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setMsgBatchSize()
- MQConnectionFactory.getMsgBatchSize()

### **Değerler**

**10**

Bu varsayılan değerdir.

**Pozitif bir tamsayı**

## **MSGONT**

Bir JMS uygulamasının, ileti bilgi yükünün bir parçası olarak bir IBM MQ iletisine ilişkin MQRFH2 'ye erişip erişmeyeceğini belirler.

### **Uygulanabilir Nesneler**

Kuyruk, Konu

JMS yönetim aracı uzun adı: WMQ\_MESSAGE\_BODY

JMS yönetim aracı kısa adı: BODY

### **Programlı erişim**

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setMessageBodyStyle()
- MQConnectionFactory.getMessageBodyStyle()

## Değerler

### Belirtilmedi

When sending, IBM MQ classes for JMS does or does not generate and include an MQRFH2 header, depending on the value of WMQ\_TARGET\_CLIENT. Giriş sırasında JMSdeğeri olarak işlev görür.

### JMS

Gönderirken, IBM MQ classes for JMS otomatik olarak bir MQRFH2 üstbilgisi oluşturur ve bunu IBM MQ iletilerinde içerir.

When receiving, IBM MQ classes for JMS set the JMS message properties according to values in the MQRFH2 (if present); it does not present the MQRFH2 as part of the JMS message body.

### MQ

When sending, IBM MQ classes for JMS does not generate an MQRFH2.

receivingalınırken, IBM MQ classes for JMS , JMS ileti gövdesinin bir parçası olarak MQRFH2 ' yi gösterir.

## MSGRECALOG

Bağlantı tüketicisinin giriş kuyruğunda teslim edilmemiş iletiler alıp alıkoymadığı.

### Uygulanabilir Nesnelere

ConnectionFactory, QueueConnectionFactory, XAConnectionFactory, XAQueueConnectionFactory,

JMS yönetim aracı uzun adı: MSGRECADATION

JMS yönetim aracı kısa adı: MRET

### Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setMessageRetention ()
- MQConnectionFactory.getMessageRetention ()

## Değerler

### Evet

Teslim edilmeyen iletiler giriş kuyruğunda kalır. Bu varsayılan değerdir.

### Hayır

Teslim edilmeyen iletiler, yok etme seçeneklerine göre ele alınmaz.

## MSGSEÇEK

İleti seçmesinin IBM MQ classes for JMS ya da aracı tarafından yapılıp yapılmayacağını belirler. TRANSPORT değeri INDIRECT değerine sahipse, ileti seçimi her zaman aracı tarafından yapılır ve MSGSELECTION değeri yoksayılır. BROKERVER V1değerine sahip olduğunda, aracıya göre ileti seçimi desteklenmez.

### Uygulanabilir Nesnelere

ConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: MSGSELECTION

JMS yönetim aracı kısa adı: MSEL

### Programlı erişim

Ayarlayıcılar/alıcılar



- MQConnectionFactory.setMessageSelection ()
- MQConnectionFactory.getMessageSelection ()

## Değerler

### CLIENT

İleti seçimi IBM MQ classes for JMS tarafından yapılır. Bu varsayılan değerdir.

### Aracı

İleti seçimi aracı tarafından yapılır.

## ÇOK YA

Bir aracıya gerçek zamanlı bağlantıda çoklu yayını etkinleştirmek ve etkinleştirilmişse, aracıdan ileti tüketicisi için iletileri göndermek üzere çok hedefli olan tam yolu belirtmek için kullanılır. Özelliğin bir ileti üreticinin bir aracıya ileti göndermesine ilişkin bir etkisi yoktur.

## Uygulanabilir Nesnelere

ConnectionFactory, TopicConnectionFactory, Konu

JMS yönetim aracı uzun adı: MULTICAST

JMS yönetim aracı kısa adı: MCAST

## Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setMulticast()
- MQConnectionFactory.getMulticast()

## Değerler

### DEVRE DIŞI

İletiler, çoklu yayın iletimi kullanan bir ileti tüketicisine teslim edilmeyor. ConnectionFactory ve TopicConnectionFactory nesnelere için varsayılan değer budur.

### ASCF

İletiler, ileti tüketicisi ile ilişkili bağlantı üreticisine ilişkin çoklu yayın ayarına göre ileti tüketicisi olarak teslim edilir. Bağlantı üreticisine ilişkin çoklu yayın ayarı, ileti tüketicisinin oluşturulduğu sırada belirtilir. Bu değer yalnızca Konu nesnelere için geçerlidir ve Konu nesnelere için varsayılan değerdir.

### Etkinleştirildi

Konu, aracıda çoklu yayın için yapılandırıldıysa, iletiler çoklu yayın iletimi kullanan bir ileti tüketicisine teslim edilir. Konu güvenilir çoklu yayın için yapılandırıldıysa, güvenilir bir hizmet kalitesi kullanılır.

### GÜVENİLİR

Konu, aracıda güvenilir çoklu yayın için yapılandırıldıysa, iletiler güvenilir bir hizmet kalitesi ile çoklu yayın iletimi kullanılarak ileti tüketicisine teslim edilir. Konu güvenilir çoklu yayın için yapılandırılmamışsa, konu için bir ileti tüketicisi yaratamazsınız.

### NOTLAR

Konu, aracıda çoklu yayın için yapılandırıldıysa, iletiler çoklu yayın iletimi kullanılarak ileti tüketicisine teslim edilir. Bu konu güvenilir çoklu yayın için yapılandırılmış olsa da, güvenilir bir hizmet kalitesi kullanılmamaktadır.

## EN İYİ BİLDİRİM

Bu özellik, IBM MQ classes for JMS ' in bir iletiyi yayınlayan bir yayıncıya hemen denetimi geri verip vermeyeceğini ya da yalnızca çağrıyla ilişkili tüm işlemleri tamamladıktan sonra denetimi döndürüp döndürmeyeceğini ve sonucu yayıncıya bildirip bildiremeyeceğini belirler.

## Uygulanabilir Nesnelere

ConnectionFactory, TopicConnectionFactory

JMS yönetim aracı uzun adı: OPTIMIZE YAYIN

JMS yönetim aracı kısa adı: OPTPUB

## Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setOptimisticPublication ()
- MQConnectionFactory.getOptimisticPublication ()

## Değerler

### HAYIR

Bir yayıncı bir iletiyi yayınladığında, IBM MQ classes for JMS çağrıyla ilişkili tüm işlemleri tamamlayıp yayıncıya rapor verinceye kadar denetimi yayıncıya geri döndürmez. Bu varsayılan değerdir.

### EYET

Bir yayıncı bir iletiyi yayınladığında, IBM MQ classes for JMS , çağrıyla ilişkili tüm işlemleri tamamlamadan önce yayıncıya hemen denetimi döndürür ve sonucu yayıncıya bildirebilir. IBM MQ classes for JMS , sonucu yalnızca yayıncı iletiyi bildirdiğinde bildirir.

## KALDIRIM

Bu özellik, IBM MQ classes for JMS ' in yalnızca bir iletiyi kabul etmiş ya da kesinleştirmiş olan bir aboneye hemen geri dönmesini mi, yoksa yalnızca çağrıyla ilişkili tüm işlemleri tamamladıktan sonra denetimi mi döndüreceğini ve sonucu aboneye bildirip bildiremeyeceğini belirler.

## Uygulanabilir Nesnelere

ConnectionFactory, TopicConnectionFactory

JMS yönetim aracı uzun adı: OUTCOMPENOTIFICATION

JMS yönetim aracı kısa adı: NOTIFY

## Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setOutcomeNotification ()
- MQConnectionFactory.getOutcomeNotification ()

## Değerler

### EYET

When a subscriber acknowledges or commits a message, IBM MQ classes for JMS do not return control to the subscriber until it has completed all the processing associated with the call and can report the outcome to the subscriber. Bu varsayılan değerdir.

### HAYIR

When a subscriber acknowledges or commits a message, IBM MQ classes for JMS returns control to the subscriber immediately, before it has completed all the processing associated with the call and can report the outcome to the subscriber.

## Kalıcılık

Bir hedefe gönderilen iletilerin kalıcılığı.

## Uygulanabilir Nesnelere

Kuyruk, Konu

JMS yönetim aracı uzun adı: PERSISTENCE

JMS yönetim aracı kısa adı: PER

## Programlı erişim

Ayarlayıcılar/alıcılar

- MQDestination.setPersistence()
- MQDestination.getPersistence()

## Değerler

### Uygulama

Kalıcılık, JMS uygulaması tarafından tanımlanır. Bu varsayılan değerdir.

### QDEF

Kalıcılık, kuyruğun varsayılan değerini alır.

### PERS

İletiler kalıcı.

### OLMAYAN

İletiler kalıcı değil.

### YÜKSEK

Bu değerın kullanımına ilişkin ek bilgi için [JMS kalıcı iletileri başlıklı konuya](#) bakın.

## GİRİŞ TARİHİ

Bir oturumdaki her ileti dinleyicisinin kuyruğunda uygun bir ileti yoksa, her ileti dinleyici kuyruğundan ileti almak için yeniden denemeden önce geçen süre üst sınırı (milisaniye cinsinden) olur. Bir oturumda ileti dinleyicilerinin herhangi biri için uygun bir ileti bulunmuyorsa, bu özelliğın değerini artırmayı düşünün. Bu özellik yalnızca, TRANSPORT değeri BIND ya da CLIENT değerine sahip olduğunda ilişkilidir.

## Uygulanabilir Nesnelere

ConnectionFactory, QueueConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XAQueueConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: POLLINGINT

JMS yönetim aracı kısa adı: PINT

## Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setPollingInterval ()
- MQConnectionFactory.getPollingInterval ()

## Değerler

### 5000

Bu varsayılan değerdir.

### Pozitif bir tamsayı

## PORT

Kuyruk yöneticisine yönelik bir bağlantı için, kuyruk yöneticisinin dinlediği kapı numarası ya da bir aracıya gerçek zamanlı bağlantı için, aracının gerçek zamanlı bağlantılar için dinlediği kapının numarası.

### Uygulanabilir Nesnelere

ConnectionFactory, QueueConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XAQueueConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: PORT

JMS yönetim aracı kısa adı: PORT

### Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setPort()
- MQConnectionFactory.getPort()

### Değerler

#### 1414

TRANSPORT, CLIENT olarak ayarlandıysa, varsayılan değer budur.

#### 1506

TRANSPORT, DIRECT ya da DIRECTHTTP olarak ayarlandıysa, bu varsayılan değerdir.

### Pozitif bir tamsayı

## Öncelik

Bir hedefe gönderilen iletilere ilişkin öncelik.

### Uygulanabilir Nesnelere

Kuyruk, Konu

JMS yönetim aracı uzun adı: PRIORITY

JMS yönetim aracı kısa adı: PRI

### Programlı erişim

Ayarlayıcılar/alıcılar

- MQDestination.setPriority()
- MQDestination.getPriority()

### Değerler

#### Uygulama

Öncelik, JMS uygulaması tarafından tanımlanır. Bu varsayılan değerdir.

#### QDEF

Öncelik, kuyruk varsayılanının değerini alır.

#### 0-9 aralığında herhangi bir tamsayı

En yüksekten en yüksek

## SüREç

Bu özellik, denetimi IBM MQ classes for JMS' e geri vermeden önce, bir abonenin aldığı iletiyi hızlı bir şekilde işlemek için garanti edip etmeyeceğini belirler.

### Uygulanabilir Nesnelere

ConnectionFactory, TopicConnectionFactory

JMS yönetim aracı uzun adı: PROCESSDURATION

JMS yönetim aracı kısa adı: PROCUR

### Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setProcessSüre ()
- MQConnectionFactory.getProcessSüre ()

### Değerler

#### BİLİNMIYOR

Bir abone, aldığı herhangi bir iletiyi ne kadar hızlı işleyebileceğiyle ilgili herhangi bir garanti vermez. Bu varsayılan değerdir.

#### short

Bir abone, denetimi IBM MQ classes for JMS' e geri vermeden önce aldığı her iletiyi hızlı bir şekilde işlemeyi garanti eder.

## SAĞLAMA Sü

This property differentiates between the three IBM MQ messaging modes of operation: IBM MQ messaging provider normal mode, IBM MQ messaging provider normal mode with restrictions, and IBM MQ messaging provider migration mode.

The IBM MQ messaging provider normal mode uses all the features of an IBM MQ queue manager to implement JMS. Bu kip, JMS 2.0 API 'sini ve işlevlerini kullanmak için eniylenir. Kısıtlamalarla IBM MQ ileti alışverişi sağlayıcısı normal kipi, JMS 2.0 API 'sını kullanır, ancak paylaşılan abonelikler, gecikmeli teslim ya da zamanuyumsuz gönderme gibi yeni özellikleri kullanmaz.

### Uygulanabilir Nesnelere

ConnectionFactory, QueueConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XAQueueConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: PROVIDERVERSION

JMS yönetim aracı kısa adı: PVER

### Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setProviderVersion ()
- MQConnectionFactory.getProviderVersion ()

### Değerler

**PROVIDERVERSION** özelliğini 8 (normal kip), 7 (kısıtlamalarla normal kip), 6 (geçiş kipi) ya da belirlenmemiş (varsayılan değer) değerlerinden herhangi birine ayarlayabilirsiniz. **PROVIDERVERSION**

özelliđi için belirttiđiniz deđer bir dizgi olmalıdır. 8, 7 ya da 6 seeneklerinden birini belirtiyorsanız, bunu ařađıdaki biimlerden herhangi birinde yapabilirsiniz:

- V.R.M.F
- V.R.M
- V.R
- V

Burada V, R, M ve F sıfırdan büyük ya da sıfıra eřit tamsayı deđerleridir. Ek R, M ve F deđerleri isteđe bađlıdır ve ayrıntılı denetim gerektiđinde kullanmanız! (isteđe bađlı) için kullanılabilir. Örneđin, **PROVIDERVERSION** 7düzeyini kullanmak istiyorsanız, **PROVIDERVERSION=7, 7.0, 7.0.0** ya da **7.0.0.0**deđerini ayarlayabilirsiniz.

### 8 -Normal kip

JMS uygulaması, IBM MQ ileti alışveriři sađlayıcısının normal kipini kullanır. Normal kip, JMS' i gerekleřtirmek için IBM MQ kuyruk yöneticisinin tüm özelliklerini kullanır. Bu kip, JMS 2.0 API ve işlevselliđini kullanmak için eniyilenmiřtir.

800 komut düzeyiyle bir kuyruk yöneticisine bađlanıyorsanız, zamanuyumsuz gönderme, gecikmeli teslim ya da paylařılan abonelik gibi tüm JMS 2.0 API ve özellikleri kullanılabilir.

Bađlantı üreticisi ayarlarında belirtilen kuyruk yöneticisi bir IBM MQ 8.0.0 kuyruk yöneticisi deđilse, `createConnection` yöntemi bir kural dıřı durumla başarısız olur `JMSFMQ0003`.

IBM MQ ileti alışveriři sađlayıcısı normal kipi, paylařım etkileřimleri özelliđini kullanır ve paylařılabilen etkileřimlerin sayısı, sunucu bađlantısı kanalındaki **SHARECNV()** özelliđi tarafından denetlenir. Bu özellik 0olarak ayarlanırsa, IBM MQ ileti alışveriři sađlayıcısı normal kipini kullanamazsınız ve `createConnection` yöntemi bir kural dıřı durumla başarısız olur `JMSCC5007`.

### 7 -Kısıtlamalarla olađan kip

JMS uygulaması, IBM MQ ileti alışveriři sađlayıcısının normal kipini kısıtlamalarla kullanır. Bu kip, paylařılan abonelikler, gecikmeli teslim ya da zamanuyumsuz gönderme gibi yeni özellikleri deđil, JMS 2.0 API 'sini kullanır.

**PROVIDERVERSION** ayarını 7 olarak ayarlarsanız, yalnızca IBM MQ ileti alışveriři sađlayıcısı olađan ve kısıtlamalar kipiyle kurun " bu seenek kullanılabilir. Bađlantı üreticisi ayarlarında belirtilen kuyruk yöneticisi bir IBM WebSphere MQ 7.0.1ya da daha sonraki bir kuyruk yöneticisi deđilse, `createConnection` yöntemi `JMSFCC5008`kural dıřı durumuyla başarısız olur.

Olađan dođumdiđer dođumsan ... daha [ ı le a Bir farkındayım lar]. Normal kipi kullanarak kısıtlamalarla bađlanıyorsanız, 700 ile 800 arasında bir komut düzeyine sahip bir kuyruk yöneticisine bađlanırken, JMS 2.0 API 'sini kullanabilirsiniz, ancak zamanuyumsuz gönderme, gecikmeli teslim ya da paylařılan abonelik özelliklerini kullanamazsınız.

IBM MQ ileti alışveriři sađlayıcısı normal kipi, kısıtlamalar ile paylařım etkileřimleri özelliđini kullanır ve paylařılabilen etkileřimlerin sayısı, sunucu bađlantısı kanalındaki **SHARECNV()** özelliđi tarafından denetlenir. Bu özellik 0olarak ayarlanırsa, kısıtlamalar ile IBM MQ ileti alışveriři sađlayıcısı normal kipini kullanamazsınız ve `createConnection` yöntemi bir özel durumla başarısız olur `JMSCC5007`.

### 6 -Geiş kipi

JMS uygulaması IBM MQ ileti alışveriři sađlayıcısı geiş kipini kullanır.

IBM MQ classes for JMS , IBM WebSphere MQ 6.0ile birlikte verilen özellikleri ve algoritmaları kullanır. WebSphere Message Broker 6.0 ya da 6.1 ' e IBM WebSphere MQ Enterprise Transport 6.0kullanarak bađlanmak istiyorsanız, bu kipi kullanmanız gerekir. Bu kipi kullanarak bir IBM MQ 8.0 kuyruk yöneticisine bađlanabilirsiniz, ancak IBM MQ classes for JMS kuyruk yöneticisinin yeni özelliklerinin hibiri kullanılmaz; örneđin, önden okuma ya da akıř.

IBM MQ 8.0 ya da sonraki bir kuyruk yöneticisine bađlanan IBM MQ 8.0 ya da sonraki bir istemciniz varsa, ileti seđimi istemci sisteminde deđil, kuyruk yöneticisi tarafından yapılır.

IBM MQ ileti alışveriři sađlayıcısı geiş kipi belirtilirse ve herhangi-gereken sa: geerli seni JMS 2.0 API, API yöntemi çağırısı `JMSCC5007`kural dıřı durumuyla başarısız olur.

## belirlenmedi (varsayılan)

**PROVIDERVERSION** özelliği varsayılan olarak *belirlenmemiş* olarak ayarlanır.

JNDI içinde önceki bir IBM MQ classes for JMS sürümüyle oluşturulan bir bağlantı üreticisi, bağlantı üreticisi IBM MQ classes for JMS' nin yeni sürümüyle kullanıldığında bu değeri alır. Hangi işlem kipinin kullanıldığını saptamak için aşağıdaki algoritma kullanılır. Bu algoritma, `createConnection` yöntemi çağrıldığında kullanılır ve IBM MQ ileti alışverişi sağlayıcısı normal kipinin, kısıtlamalarla normal kipinin ya da IBM MQ ileti alışverişi sağlayıcısı geçiş kipinin gerekli olup olmadığını saptamak için bağlantı üreticisinin diğer yönlerini kullanır.

1. Önce, IBM MQ ileti alışverişi sağlayıcısının normal kipini kullanma girişiminde bulunuldu.
2. Bağlı kuyruk yöneticisi IBM MQ 8.0 ya da üstü değilse, IBM MQ ileti alışverişi sağlayıcısını normal kipte kullanma girişiminde bulunma kısıtlamaları vardır.
3. Bağlı kuyruk yöneticisi IBM WebSphere MQ 7.0.1ya da üstü değilse, bağlantı kapanır ve bunun yerine IBM MQ ileti alışverişi sağlayıcısı geçiş kipi kullanılır.
4. Sunucu bağlantı kanalındaki **SHARECNV** özelliği 0 olarak ayarlanırsa, bağlantı kapanır ve bunun yerine IBM MQ ileti alışverişi sağlayıcısı geçiş kipi kullanılır.
5. **BROKERVER** V1 olarak ya da varsayılan *belirlenmemiş* değere ayarlanırsa, IBM MQ ileti alışverişi sağlayıcısı normal kipi kullanılmaya devam eder ve bu nedenle, yayınlama/abone olma işlemleri yeni IBM WebSphere MQ 7.0.1ya da daha sonraki özellikleri kullanır.

Uyumluluğu ilgili ek bilgi için ALTER QMGR komutunun PSMODE parametresine ilişkin bilgi için [ALTER QMGR](#) başlıklı konuya bakın.

6. **BROKERVER** V2 olarak ayarlanırsa, yapılan işlem **BROKERQMGR** değerine bağlıdır:

- **BROKERQMGR** boşluk ise:

**BROKERCONQ** özelliği tarafından belirtilen kuyruk çıkış için açılabilir (çıkış başarılı olduğunda MQOPEN ) ve kuyruk yöneticisindeki **PSMODE** COMPAT ya da DISABLED olarak ayarlandıysa, IBM MQ ileti alışverişi sağlayıcısı geçiş kipi kullanılır.

- **BROKERCONQ** özelliği tarafından belirtilen kuyruk çıkış için açılmıyorsa ya da **PSMODE** özniteliği ENABLED : olarak ayarlandıysa:

IBM MQ ileti alışverişi sağlayıcısı normal kipi kullanılır.

- **BROKERQMGR** boşluk değilse :

IBM MQ ileti alışverişi sağlayıcısı geçiş kipi kullanılır.

Kullandığınız bağlantı üreticisini değiştiremezseniz, bağlantı üreticisinin herhangi bir ayarını geçersiz kılmak için `com.ibm.msg.client.wmq.overrideProviderVersion` özelliğini kullanabilirsiniz. Bu geçersiz kılma, JVM ' deki tüm bağlantı üreticileri için geçerlidir, ancak gerçek bağlantı üreticisi nesnelere değiştirilmez.

## İlgili görevler

JMS **PROVIDERVERSION** özelliğini yapılandırma

## PROXYHOSTNAME

Yetkili sunucu aracılığıyla bir aracıya gerçek zamanlı bağlantı kullanılırken yetkili sunucunun çalıştığı sistemin ana makine adı ya da IP adresi.

## Uygulanabilir Nesnelere

ConnectionFactory, TopicConnectionÜreticisi

JMS yönetim aracı uzun adı: PROXYHOSTNAME

JMS yönetim aracı kısa adı: PHHOST

## Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setProxyHostName()
- MQConnectionFactory.getProxyHostName()

## Değerler

### boş değerli

Yetkili sunucunun anasistem adı. Bu varsayılan değerdir.

## TANITIM

Yetkili sunucu aracılığıyla bir aracıya gerçek zamanlı bir bağlantı kullanılırken yetkili sunucunun dinlediği kapının numarası.

## Uygulanabilir Nesnelere

ConnectionFactory, TopicConnectionFactory

JMS yönetim aracı uzun adı: PROXYPET

JMS yönetim aracı kısa adı: PPORT

## Programlı erişim

Ayarlayıcılar/alıcılar

MQConnectionFactory.setProxyPort ()

MQConnectionFactory.getProxyKapısı ()

## Değerler

### 443

Yetkili sunucunun kapı numarası. Bu varsayılan değerdir.

## PUBKACT

IBM MQ classes for JMS öncesinde bir yayıncı tarafından yayınlanan ileti sayısı, aracıdan bir alındı bildirimi isteğinde bulunmadan önce.

Bu özelliğin değerini alçaltığınızda, IBM MQ classes for JMS daha sık kabul eder, bu nedenle yayıncının performansı azalır. Değeri yükselttiğinizde, aracı başarısız olursa, IBM MQ classes for JMS daha uzun bir süre için kural dışı durum yayınlamayı kabul eder. Bu özellik yalnızca, TRANSPORT değeri BIND ya da CLIENT değerine sahip olduğunda ilişkilidir.

## Uygulanabilir Nesnelere

ConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: PROXYPET

JMS yönetim aracı kısa adı: PPORT

## Programlı erişim

Ayarlayıcılar/alıcılar

MQConnectionFactory.setPubAckInterval()

MQConnectionFactory.getPubAckInterval()



## Değerler

25

Herhangi bir pozitif tamsayı, varsayılan değer olabilir.

## PUTASINCALLI

Bu özellik, ileti üreticilerinin bu hedefe ileti göndermek için zamanuyumsuz yerleştirmeleri kullanmalarına izin verilip verilmeyeceğini belirler.

### Uygulanabilir Nesnelere

Kuyruk, Konu

JMS yönetim aracı uzun adı: PUTASYNCALLIVE

JMS yönetim aracı kısa adı: PAALD

### Programlı erişim

Ayarlayıcılar/alıcılar

MQDestination.setPutAsyncAllowed()

MQDestination.getPutAsyncAllowed()

## Değerler

### AS\_DEST

Kuyruğa ya da konu tanımlamasına gönderme yaparak zamanuyumsuz yerleştirmeye izin verilip verilmediğini saptayın. Bu varsayılan değerdir.

### AS\_Q\_DEF

Kuyruk tanımlamasına gönderme yaparak zamanuyumsuz yerleştirmeye izin verilip verilmediğini saptayın.

### AS\_TOPIC\_DEF

Konu tanımlamasına gönderme yaparak zamanuyumsuz yerleştirmeye izin verilip verilmediğini saptayın.

### HAYIR

Zamanuyumsuz yerleştirmeye izin verilmez.

### EVET

Zamanuyumsuz yerleştirmeye izin verilir.

## YÖNETİM

Bağlanılacak kuyruk yöneticisinin adı.

Ancak, uygulamanız bir kuyruk yöneticisine bağlanmak için bir istemci kanal tanımlama çizelgesi kullanıyorsa, [IBM MQ classes for JMS ile istemci kanalı tanımlama çizelgesi kullanılması](#) başlıklı konuya bakın.

### Uygulanabilir Nesnelere

ConnectionFactory, QueueConnectionFactory, TopicConnectionFactory, Queue, XAConnectionFactory, XAQueueConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: QASYER

JMS yönetim aracı kısa adı: QMGR

## Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setQueueManager ()
- MQConnectionFactory.getQueueManager ()

## Değerler

### "" (boş dizgi)

Herhangi bir dizgi varsayılan değer olabilir.

## kuyruk

JMS kuyruk hedefinin adı. Bu, kuyruk yöneticisi tarafından kullanılan kuyruğun adıyla eşleşir.

## Uygulanabilir Nesnelere

Kuyruk

JMS yönetim aracı uzun adı: QUEUE

JMS yönetim aracı kısa adı: QU

## Değerler

### Herhangi bir dizgi

Geçerli bir IBM MQ kuyruğu adı.

### İlgili kavramlar

[IBM MQ nesnelere adlandırma kuralları >](#)

## OKUNABİLENİZİN

Bu özellik, ileti tüketicilerinin ve kuyruk tarayıcılarının, bu hedeften gelen kalıcı olmayan iletileri almadan önce, bu hedeften gelen kalıcı iletileri iç arabelleğe almak için önden okuma kullanmasına izin verilip verilmeyeceğini belirler.

## Uygulanabilir Nesnelere

Kuyruk, Konu

JMS yönetim aracı uzun adı: READAHEADINE

JMS yönetim aracı kısa adı: RAALD

## Programlı erişim

Ayarlayıcılar/alıcılar

- MQDestination.setReadAheadAllowed()
- MQDestination.getReadAheadAllowed()

## Değerler

### AS\_DEST

İleriye okuma işlemine, kuyruk ya da konu tanımından gönderme yaparak izin verilip verilmediğini belirleyin. Bu, denetim araçlarındaki varsayılan değerdir.

Programlardaki WMQConstants.WMQ\_READ\_AHEAD\_ALLOWED\_AS\_DEST seçeneğini kullanın.

### **AS\_Q\_DEF**

İleriye okuma işlemine, kuyruk tanımlamasına başvuruda bulunarak izin verilip verilmediğini belirleyin.

Programlarda WMQConstants.WMQ\_READ\_AHEAD\_ALLOWED\_AS\_Q\_DEF kullanın.

### **AS\_TOPIC\_DEF**

İleriye okumanın, konu tanımlamasına gönderme yaparak izin verilip verilmediğini belirleyin.

Programlarda WMQConstants.WMQ\_READ\_AHEAD\_ALLOWED\_AS\_TOPIC\_DEF kullanın.

### **HAYIR**

İlerlemenize izin verilmez.

Programlarda WMQConstants.WMQ\_READ\_AHEAD\_ALLOWED\_DISABLED kullanın.

### **EVET**

İlerlemenize izin verilir.

Programlardaki WMQConstants.WMQ\_READ\_AHEAD\_ALLOWED\_ENABLED seçeneğini kullanın.

## **READAHEADCLOSEPOLICY**

Zamanuyumsuz bir ileti dinleyicisine gönderilmekte olan iletiler için, ileti tüketicisi kapatıldığında iç okuma arabelleğindeki iletilere ne olur.

### **Uygulanabilir Nesnelere**

Kuyruk, Konu

JMS yönetim aracı uzun adı: READAHEADCLOSEPOLICY

JMS yönetim aracı kısa adı: RACP

### **Programlı erişim**

Ayarlayıcılar/alıcılar

- MQDestination.setReadAheadClosePolicy()
- MQDestination.getReadAheadClosePolicy()

### **Değerler**

#### **Tümü TESLİM**

İç okuma yazma arabelleğindeki tüm iletiler, uygulamanın döndürülmeden önce ileti dinleyicisine teslim edilir. Bu, denetim araçlarındaki varsayılan değerdir.

Programlardaki WMQConstants.WMQ\_READ\_AHEAD\_DELIVERALL ögesini kullanın.

#### **GELİŞ**

Geri dönmeye önce yalnızca yürürlükteki ileti dinleyicisi çağrısı tamamlanır, iç okuma arabelleğindeki iletiler bırakılır ve bu iletiler atılır.

Programlardaki WMQConstants.WMQ\_READ\_AHEAD\_DELIVERCURRENT ögesini kullanın.

## **RECEIVECCSTnt**

Kuyruk yöneticisi ileti dönüştürmesi için hedef CCSID ' yi belirleyen hedef özellik. RECEIVECONVERSION WMQ\_RECEIVE\_CONVERSION\_QMGRolarak ayarlanmadıkça, değer yoksayılr.

### **Uygulanabilir Nesnelere**

Kuyruk, Konu

JMS yönetim aracı uzun adı: RECEIVECCSID

JMS yönetim aracı kısa adı: RCCS

## Programlı erişim

### Ayarlayıcılar/Getter

- `MQDestination.setReceiveCCSID`
- `MQDestination.getReceiveCCSID`

## Değerler

### **WMQConstants.WMQ\_RECEIVE\_CC\_SID\_JVM\_DEFAULT**

0 -JVM ' yi kullan `Charset.defaultCharset`

### **1208**

UTF-8

### **CCSID**

Desteklenen kodlanmış karakter takımı tanıtıcısı.

## ALINMASI

Veri dönüştürme işleminin kuyruk yöneticisi tarafından gerçekleştirilip gerçekleştirileceğini belirleyen hedef özellik.

## Uygulanabilir Nesnelere

Kuyruk, Konu

JMS yönetim aracı uzun adı: `RECEIVECONVERSION`

JMS yönetim aracı kısa adı: `RCNV`

## Programlı erişim

### Ayarlayıcılar/Getter

- `MQDestination.setReceiveConversion`
- `MQDestination.getReceiveConversion`

## Değerler

### **WMQConstants.WMQ\_RECEIVE\_CONVERSION\_CLIENT\_MSG**

1 -Yalnızca JMS istemcisinde veri dönüştürme işlemi gerçekleştirin. The default value from up to V7.0, and from, and including, 7.0.1.5.

### **WMQConstants.WMQ\_RECEIVE\_CONVERSION\_QMGR**

2 -İstemciye bir ileti göndermeden önce kuyruk yöneticisinden veri dönüştürme işlemi gerçekleştirin. APAR IC72897 uygulanırsa, V7.0 ile V7.0.1.4 arasındaki varsayılan (ve yalnızca) değer.

## RECEIVEI

Bu özellik, bir abonenin, abone kuyruğunda kesinleştirilmemiş iletileri alıp alamayacağını belirler.

## Uygulanabilir Nesnelere

`ConnectionFactory`, `TopicConnection` Üreticisi

JMS yönetim aracı uzun adı: `RECEIVEISOLATION`

JMS yönetim aracı kısa adı: `RCVISOL`

## Değerler

### Kesinleştirildi

Abonenin yalnızca, abone kuyruğunda işlendiği iletileri alır. Bu, denetim araçlarındaki varsayılan değerdir.

Programlarda WMQConstants.WMQ\_RCVISOL\_COMMITTED komutunu kullanın.

### KESINLEMEYEN

Abone, abone kuyruğunda kesinleştirilmemiş iletileri alabilir.

Programlarda WMQConstants.WMQ\_RCVISOL\_UNCOMMITTED komutunu kullanın.

## YENİDEN ÇIKIŞ

Bir kanal alma çıkışı ya da art arda çalıştırılacağı bir dizi alma çıkışı dizisini tanımlar.

Additional configuration might be required in order for the IBM MQ classes for JMS to locate receive exits. Daha fazla bilgi için [Kanal çıkışlarını kullanmak için IBM MQ sınıflarının JMS için yapılandırılması başlıklı konuya bakın.](#)

### Uygulanabilir Nesnelere

ConnectionFactory, QueueConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XAQueueConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: RECEXIT

JMS yönetim aracı kısa adı: RCX

### Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setReceiveExit ()
- MQConnectionFactory.getReceiveExit ()

## Değerler

- null (null) Bu varsayılan değerdir.
- Virgüllerle ayrılmış bir ya da daha çok öğeyi içeren bir dizgi; burada her bir öğe aşağıdakilerden biri olabilir:
  - The name of a class that implements the WMQReceiveExit interface (for a channel receive exit written in Java).
  - A string in the format *libraryName(entryPointAd)* (for a channel receive exit not written in Java).

## YENİDEN KALDIRMA

Kanala geçirilen kullanıcı verileri, çağrıldığında çıkış çıkışlarına çıkar.

### Uygulanabilir Nesnelere

ConnectionFactory, QueueConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XAQueueConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: RECEXITINIT

JMS yönetim aracı kısa adı: RCXI

### Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setReceiveExitInit()
- MQConnectionFactory.getReceiveExitInit()

## Değerler

### boş değerli

Kullanıcı verilerinin bir ya da daha çok ögesini virgüllerle ayrılmış bir dizgi. Bu varsayılan değerdir.

## REPLYTOSTYLE

Alınan bir iletteki JMSReplyTo alanının nasıl oluşturulacağını belirler.

### Uygulanabilir Nesnelere

ConnectionFactory, QueueConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XAQueueConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: REPLYTOSTYLE

JMS yönetim aracı kısa adı: RTOST

### Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setReplyToStyle()
- MQConnectionFactory.getReplyToStyle()

## Değerler

### VARSAYILAN

MQMDile eşdeğer.

### RFH2

RFH2 üstbilgisinde sağlanan değeri kullanın. Gönderme uygulamasında bir JMSReplyTo değeri belirlendiye, bu değeri kullanın.

### Mqmd

MQMD tarafından sağlanan değeri kullanın. Bu davranış, IBM WebSphere MQ 6.0.2.4 ve 6.0.2.5 varsayılan davranışıyla eşdeğerdir.

Gönderme uygulaması tarafından ayarlanan JMSReplyTo değeri kuyruk yöneticisi adı içermiyorsa, alma kuyruğu yöneticisi kendi adını MQMD ' ye ekler. Bu parametreyi MQMDolarak ayarlıyorsanız, kullandığınız yanıtlama kuyruğu alma kuyruğu yöneticisidir. Bu parametreyi RFH2olarak ayarlıyorsanız, kullandığınız yanıt kuyruğu, gönderilen iletinin RFH2 ' da belirtilen kuyruk yöneticisinde, gönderme uygulaması tarafından belirlenmiş olan kuyruk yöneticisinde yer alıyor.

Gönderme uygulaması tarafından ayarlanan JMSReplyTo değeri bir kuyruk yöneticisi adı içeriyorsa, hem MQMD hem de RFH2 aynı değeri içermediği için bu değiştirgenin değeri önem göstermez.

## SIFIRA

Noktadan noktaya iletişim etki alanındaki bir ileti tüketicisi, almak istediği iletileri seçmek için bir ileti seçicisini kullandığında, IBM MQ classes for JMS kuyruğun MsgDeliverySequence özneliği tarafından belirlenen sırayla uygun iletiler için IBM MQ kuyruğunu arar.

IBM MQ classes for JMS uygun bir ileti bulup tüketicieye teslim ettikten sonra, IBM MQ classes for JMS , kuyruқта geçerli konumundan bir sonraki uygun iletiyi aramaya devam eder. IBM MQ classes for JMS , kuyruğun sonuna ulaşınca kadar ya da bu özelliğın değerine göre belirlenen süre (milisaniye) kadar süre bitinceye kadar kuyruқта arama yapmaya devam eder. Her durumda, IBM MQ classes for JMS aramayı devam ettirmek için kuyruğun başına dönün ve yeni bir zaman aralığı kesinleştirme yapın.

## Uygulanabilir Nesneler

ConnectionFactory, QueueConnectionFactory, XAConnectionFactory, XAQueueConnectionFactory

JMS yönetim aracı uzun adı: RESANINT

JMS yönetim aracı kısa adı: RINT

## Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setRescanInterval ()
- MQConnectionFactory.getRescanInterval ()

## Değerler

**5000**

Herhangi bir pozitif tamsayı, varsayılan değer olabilir.

## SEÇİLEN

Kanal güvenliği çıkışını tanımlar.

Additional configuration might be required in order for the IBM MQ classes for JMS to locate security exits. Daha fazla bilgi için [Kanal çıkışlarını kullanmak için IBM MQ sınıflarının JMS için yapılandırılması](#) başlıklı konuya bakın.

## Uygulanabilir Nesneler

ConnectionFactory, QueueConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XAQueueConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: SECEXIT

JMS yönetim aracı kısa adı: SXC

## Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setSecurityExit ()
- MQConnectionFactory.getSecurityExit ()

## Değerler

- null(null) Bu varsayılan değerdir.
- Virgüllerle ayrılmış bir ya da daha çok öğeyi içeren bir dizgi; burada her bir öğe aşağıdakilerden biri olabilir:
  - WMQSecurityExit arabirimini gerçekleştiren sınıfın adı ( Java' ta yazılmış bir kanal güvenlik çıkışı için).
  - *libraryName(entryPointAd)* biçimindeki bir dizgi ( Java içinde yazılmamış bir kanal güvenlik çıkışı için).

## GÜNCELLEMİYİ

Çağrıldığında kanal güvenlik çıkışa geçirilen kullanıcı verileri.

## Uygulanabilir Nesneler

ConnectionFactory, QueueConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XAQueueConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: SECEXITINIT

JMS yönetim aracı kısa adı: SCXI

## Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setSecurityExitInit()
- MQConnectionFactory.getSecurityExitInit()

## Değerler

### boş değerli

Herhangi bir dizgi varsayılan değer olabilir.

## SENDCHECKCOUNT

Tek bir etkileşim dışı JMS oturumu içinde zamanuyumsuz koyma hataları olup olmadığını denetlemek için izin verilecek gönderme çağrılarının sayısı.

## Uygulanabilir Nesneler

ConnectionFactory, QueueConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XAQueueConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: SENDCHECKCOUNT

JMS yönetim aracı kısa adı: SCC

## Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setSendCheckCount()
- MQConnectionFactory.getSendCheckCount()

## Değerler

### boş değerli

Herhangi bir dizgi varsayılan değer olabilir.

## SENDEXIT

Bir kanal gönderme çıkışını ya da art arda çalıştırılacak bir gönderme çıkışları dizisini tanımlar.

Additional configuration might be required in order for the IBM MQ classes for JMS to locate send exits. Daha fazla bilgi için [Kanal çıkışlarını kullanmak için IBM MQ sınıflarının JMS için yapılandırılması](#) başlıklı konuya bakın.

## Uygulanabilir Nesneler

ConnectionFactory, QueueConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XAQueueConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: SENDEXIT



JMS yönetim aracı kısa adı: SDX

## Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setSendExit ()
- MQConnectionFactory.getSendExit ()

## Değerler

- null (null) Bu varsayılan değerdir.
- Virgüllerle ayrılmış bir ya da daha çok öğeyi içeren bir dizgi; burada her bir öğe aşağıdakilerden biri olabilir:
  - The name of a class that implements the WMQSendExit interface (for a channel send exit written in Java).
  - A string in the format *libraryName(entryPointAd)* (for a channel send exit not written in Java).

## SENDEXITINIT

Kanal gönderme çıkışlarına geçirilen kullanıcı verileri çağrıldığında bu verileri gönderir.

### Uygulanabilir Nesnelere

ConnectionFactory, QueueConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XAQueueConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: SENDEXITINIT

JMS yönetim aracı kısa adı: SDXI

## Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setSendExitInit()
- MQConnectionFactory.getSendExitInit()

## Değerler

### boş değerli

Virgülle ayrılmış kullanıcı verilerinin bir ya da daha fazla öğesini içeren dizimlerden biri varsayılan değer olabilir.

## FARKLILIKLI

IBM MQ ileti alışverişi sağlayıcısı normal kipini ya da sınırlamalarla olağan kipini kullanan uygulamalar için bu özellik, bağlantı üreticisinden yaratılan JMS bağlantıları, oturumları ve bağlamları için paylaşım etkileşimleri işlevinin kullanılıp kullanılmayacağını belirler.

### Uygulanabilir Nesnelere

ConnectionFactory, QueueConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XAQueueConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: SHARECONVAL,

JMS yönetim aracı kısa adı: SCALD

## Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setShareConvAllowed()
- MQConnectionFactory.getShareConvAllowed()

## Değerler

### EVET

Aynı JVM içindeki bağlantı üreticisinden yaratılan JMS bağlantıları, oturumları ve bağlamları, uygun olduğunda bir kanal eşgörünümünü (TCP/IP bağlantısıyla eşlenir) paylaşabilir.

Bu, yönetim araçları için varsayılan değerdir.

Programlar için WMQConstants.WMQ\_SHARE\_CONV\_ALLOWED\_YESkomutunu kullanın.

### HAYIR

Bağlantı üreticisinden oluşturulan her JMS bağlantısı ve bu JMSbağlantılarından yaratılan her JMS oturumu, bir kuyruk yöneticisine kendi kanal yönetim ortamına (TCP/IP bağlantısı) sahiptir.

JMS bağlamları için, bağlantı üreticisinden yaratılan ilk bağlam iki kanal yönetim ortamı (TCP/IP bağlantıları) yaratır. İlkinden yaratılan diğer JMS bağlamlarının kendi kanal yönetim ortamı (TCP/IP bağlantısı) vardır.

Programlar için WMQConstants.WMQ\_SHARE\_CONV\_ALLOWED\_NOdeğerini kullanın.

## İlgili kavramlar

[IBM MQ ileti alışverişi sağlayıcısı işletim kipleri](#)

[JMS için IBM MQ sınıflarında TCP/IP bağlantısının paylaşılması](#)

## SPARSESUBS

Bir TopicSubscriber nesnesine ilişkin ileti alma ilkesini denetler.

## Uygulanabilir Nesnelere

ConnectionFactory, TopicConnectionÜreticisi

JMS yönetim aracı uzun adı: SPARSESUBS

JMS yönetim aracı kısa adı: SSUBS

## Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setSparseAbonelikler ()
- MQConnectionFactory.getSparseAbonelikler ()

## Değerler

### HAYIR

Abonelikler sık sık eşleşen iletiler alır. Bu, denetim araçları için varsayılan değerdir.

Programlar için false değerini kullanın.

### EVET

Abonelikler sık sık eşleşen iletileri alır. Bu değer, göz atma için abonelik kuyruğunun açılabilmesini gerektirir.

Programlar için true (doğru) seçeneğini kullanın.

## SSLCIPHERSUITE

TLS bağlantısı için kullanmak üzere CipherSuite .

### Uygulanabilir Nesnelere

ConnectionFactory, QueueConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XAQueueConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: SSLCIPHERSUITE

JMS yönetim aracı kısa adı: SCPHS

### Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setSSLCipherSuite ()
- MQConnectionFactory.getSSLCipherSuite ()

### Değerler

#### boş değeri

Bu varsayılan değeri. Daha fazla bilgi için bakınız: [TLS properties of JMS objects](#).

## SLSCRL

TLS sertifikası iptal olup olmadığını denetlemek için cRL sunucuları.

### Uygulanabilir Nesnelere

ConnectionFactory, QueueConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XAQueueConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: SSLCRL

JMS yönetim aracı kısa adı: SCRL

### Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setSSLCertStores ()
- MQConnectionFactory.getSSLCertStores ()

### Değerler

#### boş değeri

LDAP URL ' lerinin boşlukla ayrılmış listesi. Bu varsayılan değeri. Daha fazla bilgi için bakınız: [TLS properties of JMS objects](#).

## SSLFIPSREQUIRED

Bu özellik, bir TLS bağlantısının IBM Java JSSE FIPS sağlayıcısı (IBMJSSEFIPS) tarafından desteklenen bir CipherSuite kullanması gerekip gerekmediğini belirler.

**Not:** AIX, Linux, and Windows işletim sistemlerinde IBM MQ , "IBM Crypto for C" şifreleme modülü aracılığıyla FIPS 140-2 uyumluluğu sağlar. Bu modüle ilişkin sertifika Geçmiş durumuna taşındı.

Müşteriler, IBM Crypto for C certificate belgesini görüntüleyip NIST tarafından sağlanan tüm tavsiyeleri dikkate almalıdır. Yeni bir FIPS 140-3 modülü şu anda devam ediyor ve durumu [İşlem listesindeki NIST CMVP modüllerinde](#) olarak görüntülenebilir.

## Uygulanabilir Nesneler

ConnectionFactory, QueueConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XAQueueConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: SSLFIPSREQUIRED

JMS yönetim aracı kısa adı: SFIPS

## Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setSSLFipsRequired ()
- MQConnectionFactory.getSSLFipsGerekli ()

## Değerler

### HAYIR

TLS bağlantısı, IBM Java JSSE FIPS sağlayıcısı (IBMJSSEFIPS) tarafından desteklenmeyen herhangi bir CipherSuite kullanılabilir.

Bu varsayılan değerdir. Programlarda false değerini kullanın.

### EVET

TLS bağlantısı, IBMJSSEFIPS tarafından desteklenen bir CipherSuite kullanılmalıdır.

Programlarda true değerini kullanın.

## SLAYICI ADI

TLS için, kuyruk yöneticisi tarafından sağlanan eşleşmesi gereken *ayırt edici ad* iskeleti.

## Uygulanabilir Nesneler

ConnectionFactory, QueueConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XAQueueConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: SSLPEERNAME

JMS yönetim aracı kısa adı: SPER

## Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setSSLPeerAd ()
- MQConnectionFactory.getSSLPeerAd ()

## Değerler

### boş değerli

Bu varsayılan değerdir. Daha fazla bilgi için bakınız: [TLS properties of JMS objects](#).

## SSLRESTCOUNT

TLS için, şifreleme için kullanılan gizli anahtardan önce bir bağlantı tarafından gönderilen ve alınan toplam bayt sayısı yeniden görüşülmektedir.

## Uygulanabilir Nesneler

ConnectionFactory, QueueConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XAQueueConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: SSLRESETCOUNT

JMS yönetim aracı kısa adı: SRC

### Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setSSLResetCount ()
- MQConnectionFactory.getSSLResetCount ()

### Değerler

0

Sıfır ya da 999, 999, 999 'dan küçük ya da bu değere eşit herhangi bir artı tamsayı. Bu varsayılan değerdir. Daha fazla bilgi için bakınız: [TLS properties of JMS objects](#).

## DEYİM YENİDEN

Bir abonenin kuyruk yöneticisiyle bağlantısını kaybettiğinde algılayan uzun süre çalışan işlemin yenilenmesi arasındaki aralık (milisaniye).

Bu özellik, yalnızca SUBSTORE değeri QUEUE değeri olduğunda ilişkilidir.

### Uygulanabilir Nesneler

ConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: STATREFRESHINT

JMS yönetim aracı kısa adı: SRI

### Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setStatusRefreshInterval()
- MQConnectionFactory.getStatusRefreshInterval()

### Değerler

60000

Herhangi bir pozitif tamsayı, varsayılan değer olabilir. Daha fazla bilgi için bakınız: [TLS properties of JMS objects](#).

## ALT DÜZEN

Burada IBM MQ classes for JMS , etkin aboneliklerle ilgili kalıcı verileri saklar.

### Uygulanabilir Nesneler

ConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: SUBSTORE

JMS yönetim aracı kısa adı: SS

### Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setSubscriptionStore ()

- MQConnectionFactory.getSubscriptionStore ()

## Değerler

### Aracı

Aboneliklerin ayrıntılarını tutmak için aracıya dayalı abonelik deposunu kullanın. Bu, denetim araçları için varsayılan değerdir.

Programlar için WMQConstants.WMQ\_SUBSTORE\_BROKERprogramını kullanın.

### MIGRATE

Abonelik bilgilerini, kuyruk tabanlı abonelik deposundan aracıya dayalı abonelik deposuna aktarın.

Programlar için WMQConstants.WMQ\_SUBSTORE\_MIGRATEöğesini kullanın.

### kuyruk

Aboneliklerin ayrıntılarını tutmak için kuyruk tabanlı abonelik deposunu kullanın.

Programlar için WMQConstants.WMQ\_SUBSTORE\_QUEUEeşçeneğini kullanın.

## SYNCPOINTALLGETS

Bu özellik, tüm alımların uyumluluk noktası altında gerçekleştirilip gerçekleştirilmeyeceğini belirler.

### Uygulanabilir Nesnelere

ConnectionFactory, QueueConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XAQueueConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: SYNCPOINTALLGETS

JMS yönetim aracı kısa adı: SPAG

### Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setSyncpointAllGets()
- MQConnectionFactory.getSyncpointAllGets()

## Değerler

### Hayır

Bu varsayılan değerdir.

### Evet

## HEDEF İSTEMCİ

Bu özellik, hedef uygulamalarla bilgi alışverişi yapmak için IBM MQ RFH2 biçiminin kullanılıp kullanılmayacağını belirler.

### Uygulanabilir Nesnelere

Kuyruk, Konu

JMS yönetim aracı uzun adı: TARGCLIENT

JMS yönetim aracı kısa adı: TC

### Programlı erişim

Ayarlayıcılar/alıcılar

- MQDestination.setTargetClient()

- MQDestination.getTargetClient()

## Değerler

### JMS

İletinin hedefi bir JMS uygulamasıdır. Bu, denetim araçları için varsayılan değerdir.  
Programlar için WMQConstants.WMQ\_CLIENT\_JMS\_COMPLIANT seçeneğini kullanın.

### MQ

İletinin hedefi, JMS dışı bir IBM MQ uygulamasıdır.  
Programlar için WMQConstants.WMQ\_CLIENT\_NONJMS\_MQ programını kullanın.

## TARGCLIENTEŞLEŞTİRME

Bu özellik, gelen bir iletinin JMSReplyTo üstbilgisi tarafından tanımlanan kuyruğa gönderilen bir yanıt iletisinin, yalnızca gelen iletinin bir MQRFH2 üstbilgisi varsa bir MQRFH2 üstbilgisine sahip olup olmadığını belirler.

### Uygulanabilir Nesneler

ConnectionFactory, QueueConnectionFactory, XAConnectionFactory, XAQueueConnectionFactory

JMS yönetim aracı uzun adı: TARGCLIENTMATCHILE

JMS yönetim aracı kısa adı: TCM

### Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setTargetClientMatching()
- MQConnectionFactory.getTargetClientMatching()

## Değerler

### EVET

Gelen iletinin bir MQRFH2 üstbilgisi yoksa, iletinin JMSReplyTo üstbilgi alanından türetilen kuyruk nesnesinin TARGCLIENT özelliği MQ'ya gönderilir. İletide bir MQRFH2 üstbilgisi varsa, TARGClient özelliği JMS olarak ayarlanır. Bu, denetim araçları için varsayılan değerdir.

Programlar için true (doğru) seçeneğini kullanın.

### HAYIR

Gelen bir iletinin JMSReplyTo üstbilgisinden türetilen kuyruk nesnesinin TARGCLIENT özelliği her zaman JMSolarak ayarlanır.

Programlar için false değerini kullanın.

## TEMPMODEL

JMS geçici kuyruklarının yaratıldığı model kuyruğunun adı.

### Uygulanabilir Nesneler

ConnectionFactory, QueueConnectionFactory, XAConnectionFactory, XAQueueConnectionFactory

JMS yönetim aracı uzun adı: TEMPMODEL

JMS yönetim aracı kısa adı: TM

### Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setTemporaryModel ()
- MQConnectionFactory.getTemporaryModel ()

## Değerler

### SYSTEM.DEFAULT.MODEL.QUEUE

Herhangi bir dizgi varsayılan değer olabilir.

## TEMPQPREFIX

IBM MQ dinamik kuyruğunun adını oluşturmak için kullanılan önek.

### Uygulanabilir Nesnelere

ConnectionFactory, QueueConnectionFactory, XAConnectionFactory, XAQueueConnectionFactory

JMS yönetim aracı uzun adı: TEMPQPREFIX

JMS yönetim aracı kısa adı: TQP

### Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setTempQPrefix ()
- MQConnectionFactory.getTempQPrefix ()

## Değerler

" " (boş dizgi)

Kullanılan önek, diğer tüm altyapılarda z/OS ve AMQ . \* üzerinde CSQ . \* önekidir. Bunlar varsayılan değerlerdir.

### Kuyruk öneki

Kuyruk öneki, bir IBM MQ nesne tanımlayıcısında (MQOD yapısı) *DynamicQName* alanının içeriği oluşturmak için kurallara uyan herhangi bir dizgidir, ancak son boş olmayan karakterin yıldız işareti olması gerekir.

## TEMPTOPICFIX

Geçici konular yaratırken, JMS " TEMP /TEMPTOPREPPREFIX/unique\_id " biçiminde bir konu dizgisi oluşturur ya da bu özellik varsayılan değer olarak bırakılırsa, yalnızca " TEMP /unique\_id " değeri olur. Boş olmayan TEMPTOPICPREFIX değerinin belirtilmesi, bu bağlantı altında yaratılan geçici konulara aboneler için yönetilen kuyruklar yaratmak üzere belirli model kuyruklarının tanımlanmasına olanak sağlar.

### Uygulanabilir Nesnelere

ConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: TEMPTOPICPREFIX

JMS yönetim aracı kısa adı: TTP

### Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setTempTopicPrefix()
- MQConnectionFactory.getTempTopicPrefix()



## Değerler

Bir IBM MQ konu dizgisi için yalnızca geçerli karakterlerden oluşan, boş değerli olmayan dizgi. Varsayılan değer " " (boş dizgi) değeridir.

## Konu

JMS konu hedefinin adı, bu değer kuyruk yöneticisi tarafından bir yayının ya da aboneliğin konu dizgisi olarak kullanılır.

## Uygulanabilir Nesneler

Konu

JMS yönetim aracı uzun adı: KONU

JMS yönetim aracı kısa adı: TOP

## Değerler

### Herhangi bir dizgi

Geçerli bir IBM MQ konu dizgisi oluşturan bir dize. When using IBM MQ as a messaging provider with WebSphere Application Server, specify a value that matches the name by which the topic is known for administrative purposes within WebSphere Application Server.

### İlgili kavramlar

[Konu dizgileri](#)

## Aktarım

Bir kuyruk yöneticisine ya da aracıya yönelik bağlantının niteliği.

## Uygulanabilir Nesneler

ConnectionFactory, QueueConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XAQueueConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: TRANSPORT

JMS yönetim aracı kısa adı: TRAN

## Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setTransportType ()
- MQConnectionFactory.getTransportType ()

## Değerler

### Bağ Tanımla

Bağ tanımları kipindeki bir kuyruk yöneticisine yönelik bağlantı için. Bu, denetim araçları için varsayılan değerdir.

Programlar için WMQConstants.WMQ\_CM\_BINDINGSöğesini kullanın.

### CLIENT

İstemci kipinde bir kuyruk yöneticisine yönelik bağlantı için.

Programlar için WMQConstants.WMQ\_CM\_CLIENTprogramını kullanın.

### Doğrudan

HTTP tünellemesi kullanmayan bir aracıya gerçek zamanlı bir bağlantı için.

Programlar için WMQConstants.WMQ\_CM\_DIRECT\_TCPIPseçeneğini kullanın.

## YÖN TUŞU

HTTP tünellemesi kullanan bir aracıya gerçek zamanlı bağlantı için. Yalnızca HTTP 1.0 desteklenir.  
Programlar için WMQConstants.WMQ\_CM\_DIRECT\_HTTPkullanın.

## İlgili kavramlar

[“IBM MQ classes for JMS nesnelere ilişkin özellikler arasındaki bağımlılıklar” sayfa 1869](#)  
Bazı özelliklerin geçerliliği, diğer özelliklerin belirli değerlerine bağlıdır.

## JOKER BIÇIMI

Bu özellik, kullanılacak genel arama karakteri sözdiziminin hangi sürümünü kullanılacağını belirler.

### Uygulanabilir Nesnelere

ConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: WILDCARDFORMAT

JMS yönetim aracı kısa adı: WCFMT

### Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setWildcardFormat()
- MQConnectionFactory.getWildcardFormat()

### Değerler

#### YALNIZ

Yalnızca, aracı sürüm 2 'de kullanıldığı gibi, konu düzeyi genel arama karakterlerini tanır. Bu, denetim araçları için varsayılan değerdir.

Programlar için WMQConstants.WMQ\_WILDCARD\_TOPIC\_ONLYögesini kullanın.

#### YALNIZCA

Yalnızca, aracı sürüm 1 'de kullanıldığı gibi, karakter genel arama karakterlerini tanır.

Programlar için WMQConstants.WMQ\_WILDCARD\_CHAR\_ONLYögesini kullanın.

## ENCODING özelliği

ENCODING özelliği, on iki olası birleşimde üç alt özelliğinden oluşur.

ENCODING özelliğinin alabileceği geçerli değerler üç alt özellikten oluşturulur:

#### tamsayı kodlaması

Olağan ya da ters çevrildi

#### ondalık kodlama

Olağan ya da ters çevrildi

#### kayan noktalı kodlama

IEEE normal, IEEE tersine çevrildi ya da z/OS

ENCODING özelliği, aşağıdaki sözdizimine sahip üç karakterlik bir dizgi olarak ifade edilir:

```
{N|R}{N|R}{N|R|3}
```

Bu dizgide:

- N olağan anlamına gelir
- R tersine çevrildi
- 3 , z/OS anlamına gelir

- İlk karakter *tamsayı kodlaması* 'ı temsil eder
- İkinci karakter *ondalık encoding* değerini gösterir
- Üçüncü karakter *kayan nokta kodlaması* temsil eder.

Bu, ENCODING özelliği için 12 olası değer kümesi sağlar.

Java platformu için uygun kodlama değerlerini belirleyen ek bir değer ( NATIVE dizgisi) vardır.

Aşağıdaki örneklerde, ENCODING için geçerli birleşimler gösterilmektedir:

```
ENCODING (NNR)
ENCODING (NATIVE)
ENCODING (RR3)
```

## TLS properties of JMS objects

SSLCIPHERSUITE özelliğini kullanarak Transport Layer Security (TLS) şifrelemesini etkinleştirin. Daha sonra başka birkaç özellik kullanarak TLS şifrelemesinin özelliklerini değiştirebilirsiniz.

TRANSPORT (CLIENT) belirttiğinizde, SSLCIPHERSUITE özelliğini kullanarak TLS ' nin şifrelenmiş iletişimini etkinleştirebilirsiniz. Set this property to a valid CipherSuite provided by your JSSE provider; it must match the CipherSpec named on the SVRCONN channel named by the CHANNEL property.

Ancak, CipherSpecs (SVRCONN kanalında belirtildiği gibi) ve CipherSuites ( ConnectionFactory nesnesinde belirtildiği gibi), aynı TLS şifreleme algoritmalarını göstermek için farklı adlandırma şemaları kullanın. SSLCIPHERSUITE özelliğinde tanınan bir CipherSpec adı belirtilirse, JMSAdmin bir uyarı yayınlar ve CipherSpec ile eşdeğer CipherSuite ile eşler. IBM MQ ve JMSAdmin tarafından tanınan CipherSpecs listesi için bkz. [TLS CipherSpecs ve CipherSuites in IBM MQ classes for JMS](#) .

IBM Java JSSE FIPS sağlayıcısı (IBMJSSEFIPS) tarafından desteklenen bir CipherSuite kullanmak için bağlantı gerekiyorsa, bağlantı üreticisinin SSLFIPSREQUIRE özelliğini YES değerine ayarlayın. Bu özelliğin varsayılan değeri NO (Hayır) değeridir. Bu, bir bağlantının desteklenen herhangi bir CipherSuite ' i kullanabileceğini gösterir. SSLCIPHERSUITE ayarlanmadıysa, özellik yoksayıdır.

SSLPEERNAME, kanal tanımlarında ayarlanabilen SSLPEER parametresinin biçimiyle eşleşir. Bu, virgülle ya da noktalı virgüllerle ayrılmış öznitelik ad-değer çiftlerinin listesidir. Örneğin:

```
SSLPEERNAME(CN=QMGR.*, OU=IBM, OU=WEBSPPHERE)
```

Ad ve değer kümesi, *ayırt edici* adoluşturur. Ayırt edici adlar ve bunların IBM MQ ile kullanımı hakkında daha fazla ayrıntı için bkz. [Securing IBM MQ](#).

Örnek verilen örnek, sunucu tarafından bağlanma sırasında gösterilen sertifikana sertifikayı denetler. Bağlantının başarılı olması için sertifikanda QMGR. başlangıcında bir Ortak Ad olmalıdır. ve en az iki Kuruluş Birimi adı olmalıdır, bunlardan ilki IBM ve ikinci WEBSPPHERE. Denetleme büyük ve küçük harfe duyarlı değildir.

SSLPEERNAME ayarlanmadıysa, böyle bir denetleme gerçekleştirilmez. SSLCIPHERSUITE ayarlanmadıysa SSLPEERNAME yoksayıdır.

SSLCRL özelliği, sıfır ya da daha fazla CRL (Sertifika İptal Listesi) sunucularını belirtir. Bu özelliğin kullanılması için Java 2 v1.4 üzerinde bir JVM olması gerekir. Bu, formdaki girdilerin boşlukla ayrılmış bir listesidir:

```
ldap:// hostname:[ port ]
```

isteğe bağlı olarak tek/. port atlanırsa, varsayılan LDAP kapısı 389 olarak kabul edilir. Bağlanma sırasında, sunucu tarafından sunulan TLS sertifikası belirtilen CRL sunucularına göre denetlenir. CRL güvenliği hakkında daha fazla bilgi için bkz. [Securing IBM MQ](#) .

SSLCRL ayarlanmadıysa, böyle bir denetleme gerçekleştirilmez. SSLCIPHERSUITE ayarlanmadıysa, SLCRL yoksayıdır.

SSLRESETCOUNT özelliđi, şifreleme için kullanılan gizli anahtardan önce bir bağlantı tarafından gönderilen ve alınan toplam bayt sayısını temsil eder. Gönderilen bayt sayısı, şifrelemeden önceki sayıdır ve alınan bayt sayısı, şifre çözme işleminden sonra gelen sayıdır. Bayt sayısı, IBM MQ classes for JMStarafından gönderilen ve alınan denetim bilgilerini de içerir.

Örneđin, 4 MB ' lik veri akıldıktan sonra yeniden anlaşma sağlanan bir TLS etkin MQI kanalı üzerinden bir bağlantı yaratmak üzere kullanılacak bir ConnectionFactory nesnesini yapılandırmak için, JMSAdmin komutunu aşağıdaki komutu verin:

```
ALTER CF(my.cf) SSLRESETCOUNT(4194304)
```

Varsayılan değeri olan SSLRESETCOUNT değeri sıfırsa, gizli anahtar hiçbir zaman yeniden anlaşılacaktır. SSLCIPHERSUITE ayarlanmadıysa, SSLRESETCOUNT özelliđi yoksayıdır.

## IBM MQ Message Service Client (XMS) for .NET başvuru

Bu başvuru kısmı, IBM MQ Message Service Client (XMS) for .NET (XMS .NET) sınıf arabirimleriyle ve XMS ile tanımlanan nesne özellikleri hakkında bilgi sağlar.

### .NET Arabirimler

Bu bölümde, .NET sınıfı arabirimleri ve bunların özellikleri ve yöntemleri açıklanmaktadır.

Aşağıdaki tablo, IBM .XMS ad alanı içinde tanımlanan arabirimleri özetlemektedir.

Çizelge 872. .NET sınıf arabirimlerinin özeti	
Arabirim	Tanım
<a href="#">"IBytesMessage" sayfa 1922</a>	Bir bayt iletisi, gövdesi byte akışından oluşan bir iletidir.
<a href="#">"IConnection" sayfa 1932</a>	Bağlantı nesnesi, uygulamanın ileti sunucusu için etkin bağlantısını gösterir.
<a href="#">"IConnectionFactory" sayfa 1934</a>	Bir uygulama, bağlantı oluşturmak için bir bağlantı üreticisi kullanır.
<a href="#">"IConnectionMetaVerileri" sayfa 1936</a>	Bir ConnectionMetaVeri nesnesi, bir bağlantıyla ilgili bilgi sağlar.
<a href="#">"Varış Noktası" sayfa 1936</a>	Bir hedef, bir uygulamanın iletileri göndereceđi ya da bir uygulamanın iletileri aldığı bir kaynaktır ya da her ikisi de olabilir.
<a href="#">"ExceptionHandler" sayfa 1937</a>	Bir uygulama, bağlantıyla ilgili bir sorunun zamanuyumsuz olarak bildirilmesi için bir kural dışı durum dinleyicisi kullanır.
<a href="#">"IllegalStateKural Dışı Durumu" sayfa 1938</a>	XMS throws this exception if an application calls a method at an incorrect or inappropriate time, or if XMS is not in an appropriate state for the requested operation.
<a href="#">"InitialContext" sayfa 1938</a>	Bir uygulama, denetlenen nesnelere havuzundan alınan nesne tanımlamalarından nesne yaratmak için InitialContext nesnesini kullanır.
<a href="#">"InvalidClientIDException" sayfa 1940</a>	XMS, bir uygulama bir bağlantı için istemci tanıtıcısı belirlemeye çalışırsa, ancak istemci tanıtıcısı geçerli değilse ya da önceden kullanımda olduđu için bu kural dışı durumu atar.

Çizelge 872. .NET sınıf arabirimlerinin özeti (devamı var)

Arabirim	Tanım
<a href="#">“InvalidDestinationKural Dışı Durumu” sayfa 1940</a>	Bir uygulama geçerli olmayan bir hedef belirtiyorsa,XMS bu kural dışı durumu atar.
<a href="#">“InvalidSelectorKural Dışı Durumu” sayfa 1941</a>	Bir uygulama, sözdizimi geçerli olmayan bir ileti seçici ifadesi sağlıyorsa,XMS bu kural dışı durumu verir.
<a href="#">“IMapMessage” sayfa 1941</a>	Eşlem iletisi, her değerin ilişkili bir veri tipine sahip olduğu bir ad-değer çiftleri kümesinden oluşan bir iletidir.
<a href="#">“İleti” sayfa 1950</a>	Bir ileti nesnesi, uygulamanın gönderdiği ya da aldığı bir iletiyi gösterir. IMessage is a superclass for the message classes such as IMapMessage.
<a href="#">“IMessageConsumer” sayfa 1955</a>	Bir uygulama, bir hedefe gönderilen iletileri almak için bir ileti tüketicisi kullanır.
<a href="#">“MessageEOFException” sayfa 1958</a>	XMS , bir uygulama byte iletilerinin gövdeini okurken bir byte ileti akışının sonuna rastlanırsa,XMS bu kural dışı durumu döndürür.
<a href="#">“MessageFormatKural Dışı Durumu” sayfa 1958</a>	XMS throws this exception if XMS encounters a message with a format that is not valid.
<a href="#">“IMessageListener (temsilci)” sayfa 1958</a>	Bir uygulama, iletileri zamanuyumsuz olarak almak için bir ileti dinleyicisini kullanır.
<a href="#">“MessageNotReadableException” sayfa 1959</a>	XMS , bir uygulama yalnızca yazma işlemi olan bir iletinin gövdenini okumayı denediğinde bu kural dışı durumu atar.
<a href="#">“MessageNotWritableException” sayfa 1959</a>	XMS , bir uygulama salt okunur bir iletinin gövdesine yazma girişiminde bulunursa bu kural dışı durumu atar.
<a href="#">“IMessageProducer” sayfa 1959</a>	Bir uygulama, bir hedefe ileti göndermek için ileti üreticiyi kullanır.
<a href="#">“IObjectMessage” sayfa 1965</a>	Bir nesne iletisi, gövdesi serileştirilmiş bir Java ya da .NET nesnesinden oluşan bir iletidir.
<a href="#">“IPropertyContext” sayfa 1965</a>	IPropertyContext , özellikleri alan ve ayarlayan yöntemler içeren, soyut bir üst sınıftır. Bu yöntemler diğer sınıflar tarafından devralınır.
<a href="#">“IQueueBrowser” sayfa 1974</a>	Bir uygulama, kuyruğu kaldırmadan kuyruklardaki iletilere göz atmak için bir kuyruk tarayıcısı kullanır.
<a href="#">“İstekte Bulunan” sayfa 1976</a>	Bir uygulama, istek iletisini göndermek için istekte bulunan bir programı kullanır ve ardından yanıtı bekler ve alır.
<a href="#">“ResourceAllocationÖzel Durumu” sayfa 1977</a>	XMS throws this exception if XMS cannot allocate the resources required by a method.

Çizelge 872. .NET sınıf arabirimlerinin özeti (devamı var)	
Arabirim	Tanım
<a href="#">"SecurityException" sayfa 1978</a>	XMS throws this exception if the user identifier and password provided to authenticate an application are rejected. XMS , bir yetki denetimi başarısızlıkla sonuçlanırsa ve bir yöntemin tamamlanmasını önlediğinde bu kural dışı durumu da atar.
<a href="#">"ISession" sayfa 1978</a>	Oturum, ileti göndermek ve almak için tek bir iş parçacıklı bağlamdır.
<a href="#">"IStreamMessage" sayfa 1988</a>	Akış iletisi, her değerin ilişkili bir veri tipine sahip olduğu bir değerler akışından oluşan bir iletidir.
<a href="#">"ITextMessage" sayfa 1997</a>	Metin iletisi, gövdesi bir dizeyi oluşturan bir iletidir.
<a href="#">"TransactionInProgressException" sayfa 1997</a>	XMS , bir işlem devam etmekte olduğu için geçerli olmayan bir işlem isteğinde bulunduysa, bu kural dışı durumu döndürür.
<a href="#">"TransactionRolledBackException" sayfa 1998</a>	XMS , bir uygulama yürürlükteki hareketi kesinleştirmek için Session.commit() uygulaması çağırıyorsa, ancak işlem geriye işlenirse bu kural dışı durumu atar.
XMSC	.NET için, XMS özellik adları ve değerleri, bu sınıfta genel sabitler olarak tanımlanır. Daha fazla ayrıntı için bkz. <a href="#">"XMS nesnelerinin özellikleri" sayfa 2001.</a>
<a href="#">"XMSException" sayfa 1998</a>	XMS bir .NET yöntemi çağırmasını işlerken bir hata saptarsa, XMS kural dışı durum yayınlar. Kural dışı durum, hatayla ilgili bilgileri sarmalayan bir nesnedir.  Farklı tiplerde XMS kural dışı durumu vardır ve bir XMSException nesnesi yalnızca bir kural dışı durum tipidir. Ancak, XMSException sınıfı, diğer XMS kural dışı durum sınıflarının üst sınıfıdır. XMS , diğer kural dışı durum tiplerinden hiçbirinin uygun olmadığı durumlarda bir XMSException nesnesi atar.
<a href="#">"XMSFactoryFactory" sayfa 1999</a>	Bir uygulama yönetilen nesnelere kullanmıyorsa, bağlantı üreticileri, kuyruklar ve konular yaratmak için bu sınıfı kullanın.

Her yöntemin tanımı, yöntem çağrı işlerken bir hata algılandığında XMS ' un döndürebileceği kural dışı durum kodlarını listeler. Her bir kural dışı durum kodu, karşılık gelen bir kural dışı durum içeren adlandırılmış değişmeziyle temsil edilir.

## IBytesMessage

Bir bayt iletisi, gövdesi byte akışından oluşan bir iletidir.

### Devralma sıradüzeni:

```

IBM.XMS.IPropertyContext
|
+---- IBM.XMS.IMessage
|
+---- IBM.XMS.IBytesMessage

```

## **.NET özellikler**

*BodyLength -Body Length (Gövde Uzunluğu)*

### **Arabirim:**

```
Int64 BodyLength
{
    get;
}
```

İletinin gövdesi salt okunurken, iletinin gövdesinin uzunluğunu bayt cinsinden alın.

Döndürülen değer, iletiyi okumak için imlecini bulunduğu yere bakmaksızın, tüm gövdenin uzunluğuna sahip olur.

### **Özel durumlar:**

- XMSEException
- MessageNotReadableException

## **Yöntemler**

*ReadBoolean -Boolean değerini okuyun*

### **Arabirim:**

```
Boolean ReadBoolean();
```

Bayt ileti akışından bir Boole değeri okuyun.

### **Parametreler:**

Yok

### **Döndürülen:**

Okunan boole değeri.

### **Özel durumlar:**

- XMSEException
- MessageNotReadableException
- MessageEOFException

*ReadSignedByte-Okuma Byte 'ı*

### **Arabirim:**

```
Int16 ReadSignedByte();
```

Bayt ileti akışından, imzalı 8 bitlik bir tamsayı olarak bir sonraki baytı okuyun.

### **Parametreler:**

Yok

### **Döndürülen:**

Okunan bayt.

### **Özel durumlar:**

- XMSEException
- MessageNotReadableException
- MessageEOFException

## ReadBytes -Okuma Baytları

### Arabirim:

```
Int32 ReadBytes(Byte[] array);  
Int32 ReadBytes(Byte[] array, Int32 length);
```

İmlecin yürürlükteki konumundan başlayarak, byte ileti akımından oluşan bayt dizisini okuyun.

### Parametreler:

#### dizi (çıkış)

Okunan bayt dizisine sahip arabellek. Çağrı, arabelleğin uzunluğundan büyük ya da bu uzunluğa eşit olmadan önce akıştan okunacaksa kalan bayt sayısı doldurulur. Ters durumda, arabellek kalan tüm baytlarla kısmen doldurulur.

Girişte boş değerli bir gösterge belirtirseniz, yöntem bunları okumadan baytların üzerine atlar. Çağrı, arabelleğin uzunluğundan büyük ya da bu uzunluğa eşit olmadan önce akıştan okunacaksa kalan bayt sayısı, atlanan bayt sayısı arabelleğin uzunluğuna eşit olur. Ters durumda, kalan tüm byte 'lar atlanır. İmleç, bayt ileti akımında okunmak üzere bir sonraki konuma kalır.

#### uzunluk (giriş)

Arabelleğin bayt cinsinden uzunluğu

### Döndürülen:

Arabelleğe okunan bayt sayısı. Arabellek kısmen doldurulduysa, okunacağı başka bayt kalmadığını belirten değer, arabellek uzunluğundan az olur. Aramadan önce akıştan okunmak üzere kalan herhangi bir byte yoksa, değer XMSC\_END\_OF\_STREAM olur.

Giriş sırasında boş değerli bir gösterge belirtirseniz, yöntem değer döndürmez.

### Özel durumlar:

- XMSEException
- MessageNotReadableException

## ReadChar -Okuma karakteri

### Arabirim:

```
Char ReadChar();
```

Bayt ileti akışından sonraki 2 baytı bir karakter olarak okuyun.

### Parametreler:

Yok

### Döndürülen:

Okunan karakter.

### Özel durumlar:

- XMSEException
- MessageNotReadableException
- MessageEOFException

## ReadDouble -Çift Duyarlıklı Yüzer Nokta Numarasını Oku

### Arabirim:

```
Double ReadDouble();
```

Çift duyarlıklı kayan noktalı sayı olarak bayt ileti akışından sonraki 8 baytı okuyun.



**Parametreler:**

Yok

**Döndürülen:**

Okunan çift duyarlılık kayan noktalı sayı.

**Özel durumlar:**

- XMSEException
- MessageNotReadableException
- MessageEOFException

*ReadFloat -Kayan Nokta Numarasını okuyun*

**Arabirim:**

```
Single ReadFloat();
```

Bir kayar noktalı sayı olarak, bayt ileti akışından sonraki 4 bayta ilişkin bilgileri okuyun.

**Parametreler:**

Yok

**Döndürülen:**

Okunan kayan noktalı sayı.

**Özel durumlar:**

- XMSEException
- MessageNotReadableException
- MessageEOFException

*ReadInt -Tamsayı Oku*

**Arabirim:**

```
Int32 ReadInt();
```

Bir sonraki 4 baytı, byte ileti akışından imzalı 32 bitlik bir tamsayı olarak okuyun.

**Parametreler:**

Yok

**Döndürülen:**

Okunan tamsayı.

**Özel durumlar:**

- XMSEException
- MessageNotReadableException
- MessageEOFException

*ReadLong -Uzun Tamsayı Oku*

**Arabirim:**

```
Int64 ReadLong();
```

Bir sonraki 8 baytı, byte ileti akışından imzalı 64 bitlik bir tamsayı olarak okuyun.

**Parametreler:**

Yok

**Döndürülen:**

Okunan uzun tamsayı.

**Özel durumlar:**

- XMSEException
- MessageNotReadableException
- MessageEOFException

*ReadShort -Kısa Tamsayı Oku*

**Arabirim:**

```
Int16 ReadShort();
```

Bayt ileti akışından sonraki 2 baytı, imzalı 16 bitlik bir tamsayı olarak okuyun.

**Parametreler:**

Yok

**Döndürülen:**

Okunan kısa tamsayı.

**Özel durumlar:**

- XMSEException
- MessageNotReadableException
- MessageEOFException

*ReadByte -İmzalanmamış Byte 'ı Oku*

**Arabirim:**

```
Byte ReadByte();
```

Bayt ileti akışından imzalanmamış 8 bitlik bir tamsayı olarak bir sonraki baytı okuyun.

**Parametreler:**

Yok

**Döndürülen:**

Okunan bayt.

**Özel durumlar:**

- XMSEException
- MessageNotReadableException
- MessageEOFException

*ReadUnsignedKısa-İmzalanmamış Kısa Tamsayı*

**Arabirim:**

```
Int32 ReadUnsignedShort();
```

İmzalanmamış 16 bitlik bir tamsayı olarak bayt ileti akışından sonraki 2 baytı okuyun.

**Parametreler:**

Yok

**Döndürülen:**

Okunmamış işaretli kısa tamsayı.

**Özel durumlar:**

- XMSEException
- MessageNotReadableException
- MessageEOFException

*ReadUTF -UTF Dizesini Oku*

**Arabirim:**

```
String ReadUTF();
```

Bayt ileti akışından UTF-8içinde kodlanan bir dizgi okuyun.

**Not:** ReadUTF() çağrılmadan önce, arabelleğin imlecinin byte ileti akışının başlangıcına işaret ettiğini doğrulayın.

**Parametreler:**

Yok

**Döndürülen:**

Okunan dizeyi sarmalayan bir String nesnesi.

**Özel durumlar:**

- XMSEException
- MessageNotReadableException
- MessageEOFException

*Sıfırla-İlk Duruma Getir*

**Arabirim:**

```
void Reset();
```

İletinin gövdeini salt okunur kipine getirin ve imleci, byte ileti akışının başlangıcındaki yerini yeniden konumlayın.

**Parametreler:**

Yok

**Döndürülen:**

Hükümsüz

**Özel durumlar:**

- XMSEException
- MessageNotReadableException

*WriteBoolean -Yazma Boole Değeri*

**Arabirim:**

```
void WriteBoolean(Boolean value);
```

Bayt ileti akışına bir boole değeri yazın.

**Parametreler:****değer (giriş)**

Yazılacak boole değeri.

**Döndürülen:**

Hükümsüz

**Özel durumlar:**

- XMSEException
- MessageNotWritableException

*WriteByte -Yazma Byte 'ı*

**Arabirim:**

```
void WriteByte(Byte value);  
void WriteSignedByte(Int16 value);
```

Bayt ileti akışına bir bayt yazın.

**Parametreler:****değer (giriş)**

Yazılacak bayt.

**Döndürülen:**

Hükümsüz

**Özel durumlar:**

- XMSEException
- MessageNotWritableException

*WriteBytes -Yazma Baytları*

**Arabirim:**

```
void WriteBytes(Byte[] value);
```

Bayt ileti akışlarına bir bayt dizisi yazın.

**Parametreler:****değer (giriş)**

Yazılacak bayt dizisi.

**Döndürülen:**

Hükümsüz

**Özel durumlar:**

- XMSEException
- MessageNotWritableException

*WriteBytes -Yazma Kısmi Byte Dizisi*

**Arabirim:**

```
void WriteBytes(Byte[] value, int offset, int length);
```

Belirtilen uzunluğun tanımladığı şekilde, byte ileti akışına kısmi byte dizisi yazın.

**Parametreler:****değer (giriş)**

Yazılacak bayt dizisi.

**görelî konum (giriş)**

Yazılacak bayt dizisine ilişkin başlangıç noktası.

**uzunluk (giriş)**

Yazılacak bayt sayısı.

**Döndürülen:**

Hükümsüz

**Özel durumlar:**

- XMSEException
- MessageNotWritableException

*WriteChar -Yazma Karakteri*

**Arabirim:**

```
void WriteChar(Char value);
```

Önce, byte ileti akışına 2 byte, önce yüksek sıralı bayt olarak bir karakter yazın.

**Parametreler:****değer (giriş)**

Yazılacak karakter.

**Döndürülen:**

Hükümsüz

**Özel durumlar:**

- XMSEException
- MessageNotWritableException

*WriteDouble -Çift Duyarlıklı Kayan Nokta Numarası Yaz*

**Arabirim:**

```
void WriteDouble(Double value);
```

Çift duyarlıklı kayan noktalı sayıyı uzun bir tamsayıya dönüştür ve uzun tamsayıyı, önce yüksek sıralı bayt olan 8 bayt olarak byte ileti akımuna yazın.

**Parametreler:****değer (giriş)**

Yazılacak çift duyarlıklı kayan noktalı sayı.

**Döndürülen:**

Hükümsüz

**Özel durumlar:**

- XMSEException
- MessageNotWritableException

*WriteFloat -Kayan Noktalı Sayı Yazma*

**Arabirim:**

```
void WriteFloat(Single value);
```

Bir kayan noktalı sayıyı bir tamsayıya dönüştürüp tamsayıyı, önce yüksek sıralı bayt olan 4 bayt olarak byte ileti akışına yazın.

**Parametreler:****değer (giriş)**

Yazılacak kayan noktalı sayı.

**Döndürülen:**

Hükümsüz

**Özel durumlar:**

- XMSEException
- MessageNotWritableException

*WriteInt -Yazma Tamsayı*

**Arabirim:**

```
void WriteInt(Int32 value);
```

Bayt ileti akışına 4 byte, önce yüksek sıralı bayt olarak bir tamsayı yazın.

**Parametreler:****değer (giriş)**

Yazılacak tamsayı.

**Döndürülen:**

Hükümsüz

**Özel durumlar:**

- XMSEException
- MessageNotWritableException

*WriteLong -Yazma Uzun Tamsayı*

**Arabirim:**

```
void WriteLong(Int64 value);
```

Bayt ileti akışına önce 8 bayt, yüksek sıralı bayt olarak uzun bir tamsayı yazın.

**Parametreler:****değer (giriş)**

Yazılacak uzun tamsayı.

**Döndürülen:**

Hükümsüz

**Özel durumlar:**

- XMSEException
- MessageNotWritableException

*WriteObject -Yazma Nesnesi*

**Arabirim:**

```
void WriteObject(Object value);
```

Belirtilen nesneyi byte ileti akışına yazın.

**Parametreler:****değer (giriş)**

Yazılacak nesne; bir temel tipe gönderme olması gerekir.

**Döndürülen:**

Hükümsüz

**Özel durumlar:**

- XMSEException
- MessageNotWritableException

*WriteShort -Yazma Kısa Tamsayı*

**Arabirim:**

```
void WriteShort(Int16 value);
```

Bayt ileti akışına kısa bir tamsayı yazın; 2 byte olarak, önce yüksek sıralı byte 'ı yazın.

**Parametreler:****değer (giriş)**

Yazılacak kısa tamsayı.

**Döndürülen:**

Hükümsüz

**Özel durumlar:**

- XMSEException
- MessageNotWritableException

*WriteUTF -UTF dizgisi yazın*

**Arabirim:**

```
void WriteUTF(String value);
```

Write a string, encoded in UTF-8, to the bytes message stream.

**Parametreler:****değer (giriş)**

Yazılacak dizeyi sarsalayan bir String nesnesi.

**Döndürülen:**

Hükümsüz

**Özel durumlar:**

- XMSEException
- MessageNotWritableException

***Edinilmiş özellikler ve yöntemler***

Aşağıdaki özellikler IMessage arabiriminden edinilir:

JMSCorrelationID, JMSDeliveryMode, JMSDestination, JMSExpiration, JMSMessageID, JMSPriority, JMSReteslim, JMSReplyTo, JMSTimestamp, JMSType, Özellikler

Aşağıdaki yöntemler IMessage arabiriminden edinilir:

clearBody, clearProperties, PropertyExists

Aşağıdaki yöntemler IPropertyContext arabiriminden edinilir:

GetBooleanProperty, GetByteProperty, GetBytesProperty, GetCharProperty, GetDoubleProperty, GetFloatProperty, GetIntProperty, GetLongProperty, GetObjectProperty, GetShortProperty, GetStringProperty, SetBooleanProperty, SetByteProperty, SetBytesProperty, SetCharProperty,

[SetDoubleProperty](#), [SetFloatProperty](#), [SetIntProperty](#), [SetLongProperty](#), [SetObjectProperty](#), [SetShortProperty](#), [SetStringProperty](#)

## IConnection

Bağlantı nesnesi, uygulamanın ileti sunucusu için etkin bağlantısını gösterir.

### Devralma sıradüzeni:

```
IBM.XMS.IPropertyContext
|
+----IBM.XMS.IConnection
```

Connection nesnesine ilişkin XMS tanımlı özelliklerinin bir listesi için bkz. [“Bağlantı Özellikleri” sayfa 2001](#).

### .NET özellikler

*ClientID* -İstemci tanıtıcısını al ve ayarla

#### Arabirim:

```
String ClientID
{
    get;
    set;
}
```

Bağlantıya ilişkin istemci tanıtıcısını alın ve ayarlayın.

İstemci tanıtıcısı, ConnectionFactorydosyası içinde yönetici tarafından önceden yapılandırılabilir ya da ClientIDayarına göre atanmış olabilir.

İstemci tanıtıcısı yalnızca, yayınlama/abone olma etki alanındaki sürekli abonelikleri desteklemek için kullanılır ve noktadan noktaya iletişim alanında yoksayılr.

Bir uygulama, bir bağlantı için istemci tanıtıcısı ayarlarsa, bağlantı yaratıldıktan hemen sonra ve bağlantıda başka bir işlem gerçekleştirilmeden önce uygulamanın yapması gerekir. Uygulama bu noktadan sonra bir istemci tanıtıcısı ayarlamaya çalışırsa, çağrı kural dışı durum IllegalStatekural dışı durumunu atar.

Bu özellik, bir aracıya gerçek zamanlı bağlantı için geçerli değildir.

#### Özel durumlar:

- XMSException
- IllegalStateKural Dışı Durumu
- InvalidClientIDException

*ExceptionListener* -Kural Dışı Durum Dinleyicisi Al ve Ayarla

#### Arabirim:

```
ExceptionListener ExceptionListener
{
    get;
    set;
}
```

Bağlantıda kayıtlı olan kural dışı durum dinleyicisini alın ve bağlantıyla bir kural dışı durum dinleyicisi kaydettirin.

Bağlantıda bir kural dışı durum dinleyicisi kaydedilmediyse, yöntem boş değer döndürür. Bağlantıda bir kural dışı durum dinleyicisi zaten kayıtlı ise, kural dışı durum dinleyicisi yerine boş değer belirterek kayıt işlemini iptal edebilirsiniz.



Kural dışı durum dinleyicilerinin kullanılmasına ilişkin ek bilgi edinmek için [Using message and exception listeners in .NET](#)' e bakın.

#### Özel durumlar:

- XMSEException

*Meta Veri-Meta Verileri Al*

#### Arabirim:

```
ICollectionMetadata Metadata
{
    get;
}
```

Bağlantıya ilişkin meta verileri alın.

#### Özel durumlar:

- XMSEException

#### Yöntemler

*Bağlantıyı Kapat-Bağlantıyı Kapat*

#### Arabirim:

```
void Close();
```

Bağlantıyı kapatın.

Bir uygulama önceden kapatılmış bir bağlantıyı kapatmaya çalışırsa, arama yoksayılır.

#### Parametreler:

Yok

#### Döndürülen:

Hükümsüz

#### Özel durumlar:

- XMSEException

*CreateSession -Oturum Yarat*

#### Arabirim:

```
ISession CreateSession(Boolean transacted,
                        AcknowledgeMode acknowledgeMode);
```

Bir oturum oluşturun.

#### Parametreler:

##### **translated (giriş)**

True değeri, oturumun aktarıldığı anlamına gelir. False değeri, oturumun aktarılamadığı anlamına gelir.

Bir aracıya gerçek zamanlı bağlantı için değer False olmalıdır.

##### **acknowledgeMode (giriş)**

Bir uygulama tarafından alınan iletilerin nasıl kabul edildiğine ilişkin bilgi içerir. Değer, AcknowledgeMode sıralı değer listeleyicisinden biri olmalıdır:

AcknowledgeMode.AutoAcknowledge

AcknowledgeMode.ClientAcknowledge  
AcknowledgeMode.DupsOkAcknowledge

Bir aracıya gerçek zamanlı bağlantı için değer AcknowledgeMode.AutoAcknowledge ya da AcknowledgeMode.DupsOkAcknowledge olmalıdır.

Oturum hareket ettiyse bu değıştirge yoksayılr. Alındı bildirim i kipleri hakkında daha fazla bilgi için bkz. [İleti alındı bildirim i](#).

**Döndürülen:**

Oturum nesnesi

**Özel durumlar:**

- XMSException

*Başlatma-Bağlantı Başlat*

**Arabirim:**

```
void Start();
```

Bağlantıya ilişkin gelen iletilerin sağlananını başlatın ya da yeniden başlatın. Bağlantı önceden başlatıldıysa, arama yoksayılr.

**Parametreler:**

Yok

**Döndürülen:**

Hükümsüz

**Özel durumlar:**

- XMSException

*Durdur-Bağlantıyı Durdur*

**Arabirim:**

```
void Stop();
```

Bağlantı için gelen iletilerin sağlananını durdurun. Bağlantı önceden durdurulduysa, arama yoksayılr.

**Parametreler:**

Yok

**Döndürülen:**

Hükümsüz

**Özel durumlar:**

- XMSException

***Edinilmiş özellikler ve yöntemler***

Aşğıdaki yöntemler [IPropertyContext](#) arabiriminden edinilir:

[GetBooleanProperty](#), [GetByteProperty](#), [GetBytesProperty](#), [GetCharProperty](#), [GetDoubleProperty](#), [GetFloatProperty](#), [GetIntProperty](#), [GetLongProperty](#), [GetObjectProperty](#), [GetShortProperty](#), [GetStringProperty](#), [SetBooleanProperty](#), [SetByteProperty](#), [SetBytesProperty](#), [SetCharProperty](#), [SetDoubleProperty](#), [SetFloatProperty](#), [SetIntProperty](#), [SetLongProperty](#), [SetObjectProperty](#), [SetShortProperty](#), [SetStringProperty](#)

**IConnectionFactory**

Bir uygulama, bağlantı oluşturmak için bir bağlantı üreticisi kullanır.

## Devralma sıradüzeni:

```
IBM.XMS.IPropertyContext
|
+----IBM.XMS.IConnectionFactory
```

Bir ConnectionFactory nesnesine ilişkin XMS tanımlı özelliklerinin bir listesi için bkz. [“ConnectionFactoryözellikleri” sayfa 2002.](#)

## Yöntemler

*CreateConnection -Create Connection Factory (Bağlantı Üreticisi Yarat) (varsayılan kullanıcı kimliğini kullanarak)*

### Arabirim:

```
IConnectionFactory CreateConnection();
```

Varsayılan özelliklerle bir bağlantı üreticisi yaratın.

IBM MQ ve XMLC\_USERID ' ye bağlanıyorsanız, kuyruk yöneticisi varsayılan olarak oturum açmış kullanıcının userID değerini kullanır. Tek tek kullanıcıların daha fazla bağlantı düzeyinde kimlik doğrulamasına gereksinim duyarsanız, IBM MQ’ünde yapılandırılmış bir istemci kimlik doğrulaması çıkışı yazabilirsiniz.

### Parametreler:

Yok

### Özel durumlar:

- XMSEException

*CreateConnection -Bağlantı Yarat (belirli bir kullanıcı kimliğini kullanarak)*

### Arabirim:

```
IConnectionFactory CreateConnection(String userId, String password);
```

Belirtilen kullanıcı kimliğini kullanarak bir bağlantı yaratın.

IBM MQ ve XMLC\_USERID ' ye bağlanıyorsanız, kuyruk yöneticisi varsayılan olarak oturum açmış kullanıcının userID değerini kullanır. Tek tek kullanıcıların daha fazla bağlantı düzeyinde kimlik doğrulamasına gereksinim duyarsanız, IBM MQ’ünde yapılandırılmış bir istemci kimlik doğrulaması çıkışı yazabilirsiniz.

Bağlantı durdurulmuş kipte yaratılır. Uygulama **ConnectionFactory.start()** çağrılıncaya kadar hiçbir ileti teslim edilmez.

### Parametreler:

#### userId (giriş)

Bir dizgi nesnesi, uygulamanın kimliğini doğrulamak için kullanılacak kullanıcı kimliğini sarsalıyor. Boş değer sağlıyorsa, kimlik doğrulamadan bağlantıyı yaratma girişiminde bulunmanız gerekir.

#### parola (giriş)

Uygulamanın kimliğini doğrulamak için kullanılacak parolayı sarmalayan bir String nesnesi. Boş değer sağlıyorsa, kimlik doğrulamadan bağlantıyı yaratma girişiminde bulunmanız gerekir.

### Döndürülen:

Bağlantı nesnesi.

### Özel durumlar:

- XMSEException

- XMS\_X\_SECURITY\_EXCEPTION

## **Edinilmiş özellikler ve yöntemler**

Aşağıdaki yöntemler [IPropertyContext](#) arabiriminden edinilir:

[GetBooleanProperty](#), [GetByteProperty](#), [GetBytesProperty](#), [GetCharProperty](#), [GetDoubleProperty](#), [GetFloatProperty](#), [GetIntProperty](#), [GetLongProperty](#), [GetObjectProperty](#), [GetShortProperty](#), [GetStringProperty](#), [SetBooleanProperty](#), [SetByteProperty](#), [SetBytesProperty](#), [SetCharProperty](#), [SetDoubleProperty](#), [SetFloatProperty](#), [SetIntProperty](#), [SetLongProperty](#), [SetObjectProperty](#), [SetShortProperty](#), [SetStringProperty](#)

## **IConnectionMetaVerileri**

Bir ConnectionMetaVeri nesnesi, bir bağlantıyla ilgili bilgi sağlar.

### **Devralma sıradüzeni:**

```
IBM.XMS.IPropertyContext
|
+---- IBM.XMS.IConnectionMetaData
```

Bir ConnectionMetaVeri nesnesine ilişkin XMS tanımlı özelliklerinin bir listesi için bkz. [“ConnectionMetaVerileri Özellikleri” sayfa 2007](#).

## **.NET özellikler**

*JMSXPropertyNames -JMS Tanımlı İleti Özelliklerini Al*

### **Arabirim:**

```
System.Collections.IEnumerator JMSXPropertyNames
{
    get;
}
```

Bağlantı tarafından desteklenen, JMS tanımlı ileti özelliklerinin adlarına ilişkin sıralı değer listesi döndürür.

JMS tanımlı ileti özellikleri, aracıya gerçek zamanlı bir bağlantı tarafından desteklenmez.

### **Özel durumlar:**

- XMSException

## **Edinilmiş özellikler ve yöntemler**

Aşağıdaki yöntemler [IPropertyContext](#) arabiriminden edinilir:

[GetBooleanProperty](#), [GetByteProperty](#), [GetBytesProperty](#), [GetCharProperty](#), [GetDoubleProperty](#), [GetFloatProperty](#), [GetIntProperty](#), [GetLongProperty](#), [GetObjectProperty](#), [GetShortProperty](#), [GetStringProperty](#), [SetBooleanProperty](#), [SetByteProperty](#), [SetBytesProperty](#), [SetCharProperty](#), [SetDoubleProperty](#), [SetFloatProperty](#), [SetIntProperty](#), [SetLongProperty](#), [SetObjectProperty](#), [SetShortProperty](#), [SetStringProperty](#)

## **Variş Noktası**

Bir hedef, bir uygulamanın iletileri göndereceği ya da bir uygulamanın iletileri aldığı bir kaynaktır ya da her ikisi de olabilir.

### **Devralma sıradüzeni:**

```
IBM.XMS.IPropertyContext
```

```
|  
+----IBM.XMS.IDestination
```

Hedef nesneye ilişkin XMS tanımlı özelliklerinin bir listesi için bkz. "[Hedef Özellikleri](#)" sayfa 2008.

## **.NET özellikler**

*Ad-Hedef Adı Al*

### **Arabirim:**

```
String Name  
{  
  get;  
}
```

Hedefin adını alın. Ad, bir kuyruğun adını ya da bir konunun adını sarmalayan bir dizilimdir.

### **Özel durumlar:**

- XMSEException

*TypeId -Hedef Tipi Al*

### **Arabirim:**

```
DestinationType TypeId  
{  
  get;  
}
```

Hedefin tipini alın. Hedefin tipi aşağıdaki değerlerden biridir:

```
DestinationType.Queue  
DestinationType.Topic
```

### **Özel durumlar:**

- XMSEException

## **Edinilmiş özellikler ve yöntemler**

Aşağıdaki yöntemler [IPropertyContext](#) arabiriminden edinilir:

[GetBooleanProperty](#), [GetByteProperty](#), [GetBytesProperty](#), [GetCharProperty](#), [GetDoubleProperty](#), [GetFloatProperty](#), [GetIntProperty](#), [GetLongProperty](#), [GetObjectProperty](#), [GetShortProperty](#), [GetStringProperty](#), [SetBooleanProperty](#), [SetByteProperty](#), [SetBytesProperty](#), [SetCharProperty](#), [SetDoubleProperty](#), [SetFloatProperty](#), [SetIntProperty](#), [SetLongProperty](#), [SetObjectProperty](#), [SetShortProperty](#), [SetStringProperty](#)

## **ExceptionHandler**

Bir uygulama, bağlantıyla ilgili bir sorunun zamanuysuz olarak bildirilmesi için bir kural dışı durum dinleyicisi kullanır.

### **Devralma sıradüzeni:**

Yok

Bir uygulama yalnızca, iletileri zamanuysuz olarak tüketebilmek için ve başka hiçbir amaçla bir bağlantı kullanıyorsa, uygulamanın bir sorunla ilgili bilgi edinebilmenin tek yolu bir kural dışı durum dinleyicisi kullanılmasıdır. Başka durumlarda, bir kural dışı durum dinleyicisi, XMS' a bir sonraki zamanuysuz çağrıya kadar beklemekten daha kısa bir şekilde bir sorunla ilgili bilgi edinmek için daha acil bir yol sağlayabilir.

## Ata

*ExceptionListener -Kural Dışı Durum Dinleyici*

### Arabirim:

```
public delegate void ExceptionListener(Exception ex)
```

Bir sorunla ilgili uygulamayı bir bağlantıyla bilgilendir.

Bu yetkilendirmeyi uygulayan yöntemler, bağlantıyla birlikte kaydedilebilir.

Kural dışı durum dinleyicilerinin kullanılmasına ilişkin ek bilgi edinmek için [Using message and exception listeners in .NET](#)' e bakın.

### Parametreler:

#### kural dışı durum (giriş)

XMStarafından oluşturulan bir kural dışı duruma işaret eden gösterge.

### Döndürülen:

Hükümsüz

## IllegalStateKural Dışı Durumu

XMS throws this exception if an application calls a method at an incorrect or inappropriate time, or if XMS is not in an appropriate state for the requested operation.

### Devralma sıradüzeni:

```
IBM.XMS.XMSException
|
+----IBM.XMS.Exception
|
+----IBM.XMS.IllegalStateException
```

## Edinilmiş özellikler ve yöntemler

Aşağıdaki yöntemler [XMSException](#) arabiriminden edinilir:

[GetErrorCode](#), [GetLinkedException](#)

## InitialContext

Bir uygulama, denetlenen nesnelere havuzundan alınan nesne tanımlamalarından nesne yaratmak için InitialContext nesnesini kullanır.

### Devralma sıradüzeni:

Yok

## .NET özellikler

*Ortam-Ortamı alın*

### Arabirim:

```
Hashtable Environment
{
    get;
}
```

Ortamı alın.

### Özel durumlar:

- Kural dışı durumlar, kullanılmakta olan dizin hizmetine özgüdür.

## ***Oluşturucular***

*InitialContext -Başlangıç Bağlamı Yarat*

### Arabirim:

```
InitialContext(Hashtable env);
```

Bir InitialContext nesnesi yaratın.

### Parametreler:

Denetlenen nesnelerin havuzlarıyla bağlantı kurmak için gereken bilgiler, oluşturucuda bir ortam hashtable 'ında sağlanır.

### Özel durumlar:

- XMSEException

## ***Yöntemler***

*AddToOrtamı-Ortama Yeni Bir Özellik Ekle*

### Arabirim:

```
Object AddToEnvironment(String propName, Object propVal);
```

Ortama yeni bir özellik ekleyin.

### Parametreler:

#### **propName (giriş)**

Eklenecek özelliğin adını sarmalayan bir String nesnesi.

#### **propVal (giriş)**

Eklenecek özelliğin değeri.

### Döndürülen:

Özelliğin eski değeri.

### Özel durumlar:

- Kural dışı durumlar, kullanılmakta olan dizin hizmetine özgüdür.

*Kapat-Bu bağlamı kapat*

### Arabirim:

```
void Close()
```

Bu bağlamı kapatın.

### Parametreler:

Yok

### Döndürülen:

Yok

### Özel durumlar:

- Kural dışı durumlar, kullanılmakta olan dizin hizmetine özgüdür.

## Arama-Başlangıç Bağlamında Nesne Arama

### Arabirim:

```
Object Lookup(String name);
```

Denetlenen nesnelere havuzundan alınan bir nesne tanımlamasından nesne yaratın.

### Parametreler:

#### ad (giriş)

Alınacak denetimli nesnenin adını sarsalayan bir String nesnesi. Ad yalın bir ad ya da karmaşık bir ad olabilir. Ek ayrıntılar için [Denetlenen nesnelere alınması başlıklı konuya](#) bakın.

### Döndürülen:

Alınmakta olan nesnenin tipine bağlı olarak bir IConnectionFactory ya da bir IDestination değeri. İşlev dizine erişebiliyorsa, ancak gerekli nesneyi bulamazsa, boş değer döndürülür.

### Özel durumlar:

- Kural dışı durumlar, kullanılmakta olan dizin hizmetine özgüdür.

## RemoveFromEnvironment-Ortamdan Bir Özelliği Kaldırma

### Arabirim:

```
Object RemoveFromEnvironment(String propName);
```

Bir özelliği ortamdaki bir özelliği kaldırın.

### Parametreler:

#### propName (giriş)

Kaldırılacak özelliğin adını sarmalayan bir String nesnesi.

### Döndürülen:

Kaldırılan nesne.

### Özel durumlar:

- Kural dışı durumlar, kullanılmakta olan dizin hizmetine özgüdür.

## InvalidClientIDException

XMS , bir uygulama bir bağlantı için istemci tanıtıcısı belirlemeye çalışırsa, ancak istemci tanıtıcısı geçerli değilse ya da önceden kullanımda olduğu için bu kural dışı durumu atar.

### Devralma sıradüzeni:

```
IBM.XMS.XMSEException
|
+----IBM.XMS.XMSEException
|
+----IBM.XMS.InvalidClientIDException
```

### Edinilmiş özellikler ve yöntemler

Aşağıdaki yöntemler [XMSEException](#) arabiriminden edinilir:

[GetErrorCode](#), [GetLinkedException](#)

## InvalidDestinationKural Dışı Durumu

Bir uygulama geçerli olmayan bir hedef belirtiyorsa,XMS bu kural dışı durumu atar.



### Devralma sıradüzeni:

```
IBM.XMS.XMSEException
|
+----IBM.XMS.XMSEException
|
+----IBM.XMS.InvalidDestinationException
```

### Edinilmiş özellikler ve yöntemler

Aşağıdaki yöntemler [XMSEException](#) arabiriminden edinilir:

[GetErrorCode](#), [GetLinkedException](#)

### InvalidSelectorKural Dışı Durumu

Bir uygulama, sözdizimi geçerli olmayan bir ileti seçici ifadesi sağlıyorsa,XMS bu kural dışı durumu verir.

### Devralma sıradüzeni:

```
IBM.XMS.XMSEException
|
+----IBM.XMS.XMSEException
|
+----IBM.XMS.InvalidSelectorException
```

### Edinilmiş özellikler ve yöntemler

Aşağıdaki yöntemler [XMSEException](#) arabiriminden edinilir:

[GetErrorCode](#), [GetLinkedException](#)

### IMapMessage

Eşlem iletisi, her değer ilişkili bir veri tipine sahip olduğu bir ad-değer çiftleri kümesinden oluşan bir iletidir.

### Devralma sıradüzeni:

```
IBM.XMS.IPropertyContext
|
+----IBM.XMS.IMessage
|
+----IBM.XMS.IMapMessage
```

Bir uygulama ad-değer çiftinin değerini aldığı anda, değer XMS tarafından başka bir veri tipine dönüştürülebilir. Bu örtük dönüştürme biçimiyle ilgili daha fazla bilgi için, [XMS iletisinin gövdesi](#) içindeki harita iletilerine ilişkin bilgilere bakın.

### .NET özellikler

*MapNames -Eşlem Adlarını Al*

### Arabirim:

```
System.Collections.IEnumerator MapNames
{
    get;
}
```

Eşlem iletisinin gövdesindeki adlara ilişkin bir sıralı değer listesi alın.

## Özel durumlar:

- XMSEException

## Yöntemler

*GetBoolean -Boole Değeri Al*

### Arabirim:

```
Boolean GetBoolean(String name);
```

Eşlem iletisinin gövdesinden ada göre tanımlanan Boole değerini alın.

### Parametreler:

#### ad (giriş)

Bir String nesnesi, boole değerini tanımlayan adı sarmalayan bir nesne.

### Döndürülen:

Eşlem iletisinin gövdesinden alınan Boole değeri.

## Özel durumlar:

- XMSEException

*GetByte -Byte Al*

### Arabirim:

```
Byte GetByte(String name);  
Int16 GetSignedByte(String name);
```

Eşlemin gövdesinden ada göre tanımlanan baytı alın.

### Parametreler:

#### ad (giriş)

Byte 'ı tanımlayan adı sarmalayan bir String nesnesi.

### Döndürülen:

Harita iletisinin gövdesinden alınan bayt. Byte üzerinde veri dönüştürme işlemi gerçekleştirilmez.

## Özel durumlar:

- XMSEException

*GetBytes -Alma Baytları*

### Arabirim:

```
Byte[] GetBytes(String name);
```

Eşlem iletisinin gövdesinden ada göre tanımlanan bayt dizisini alın.

### Parametreler:

#### ad (giriş)

Bir dize nesnesi, bayt dizisini tanımlayan adı sarmalayan bir nesnedir.

### Döndürülen:

Dizideki bayt sayısı.

## Özel durumlar:

- XMSEException

## *GetChar -Karakter Al*

### **Arabirim:**

```
Char GetChar(String name);
```

Eşlemin gövdesinden ada göre tanımlanan karakteri alın.

### **Parametreler:**

#### **ad (giriş)**

Bir String nesnesi, karakteri tanımlayan adı sarmalayan bir nesnedir.

### **Döndürülen:**

Eşlem iletisinin gövdesinden alınan karakter.

### **Özel durumlar:**

- XMSEException

## *GetDouble -Çift Duyarlıklı Kayan Nokta Numarası Al*

### **Arabirim:**

```
Double GetDouble(String name);
```

Eşlem iletisinin gövdesinden ada göre tanımlanan çift duyarlıklı kayan nokta numarasını alın.

### **Parametreler:**

#### **ad (giriş)**

Çift duyarlıklı kayan nokta numarasını tanımlayan adı sarmalayan bir String nesnesi.

### **Döndürülen:**

İleti eşleme iletisinin gövdesinden alınan çift duyarlıklı kayan noktalı sayı.

### **Özel durumlar:**

- XMSEException

## *GetFloat -Kayan Noktalı Sayı Al*

### **Arabirim:**

```
Single GetFloat(String name);
```

Eşlem iletisinin gövdesinden ada göre tanımlanan kayan noktalı sayıyı alın.

### **Parametreler:**

#### **ad (giriş)**

Bir dize nesnesi, kayan nokta numarasını tanımlayan adı sarmalayan bir nesnedir.

### **Döndürülen:**

Eşleme iletisinin gövdesinden alınan kayan noktalı sayı.

### **Özel durumlar:**

- XMSEException

## *GetInt -Tamsayı Al*

### **Arabirim:**

```
Int32 GetInt(String name);
```

Eşlem iletisinin gövdesinden ada göre tanımlanan tamsayıyı alın.

**Parametreler:****ad (giriş)**

Bir dizgi nesnesi, tamsayıyı tanımlayan adı sarmalayan bir nesne.

**Döndürülen:**

Eşlem iletisinin gövdesinden alınan tamsayı.

**Özel durumlar:**

- XMSEException

*GetLong -Uzun Tamsayı Al*

**Arabirim:**

```
Int64 GetLong(String name);
```

Eşlem iletisinin gövdesinden ada göre tanımlanan uzun tamsayıyı alın.

**Parametreler:****ad (giriş)**

Uzun tamsayıyı tanımlayan adı sarmalayan bir String nesnesi.

**Döndürülen:**

Eşlem iletisinin gövdesinden alınan uzun tamsayı.

**Özel durumlar:**

- XMSEException

*GetObject -Nesne Al*

**Arabirim:**

```
Object GetObject(String name);
```

Eşlem iletisinin gövdesinden bir ad-değer çiftinin değerine bir başvuru alın. Ad-değer çifti adıyla tanımlanır.

**Parametreler:****ad (giriş)**

Ad-değer çiftinin adını sarsalayan bir String nesnesi.

**Döndürülen:**

Aşağıdaki nesne tiplerinden biri olan değer:

Boolean  
Byte  
Byte[]  
Char  
Double  
Single  
Int32  
Int64  
Int16  
String

**Özel durumlar:**

XMSEException

*GetShort -Kısa Tamsayı Al*

**Arabirim:**

```
Int16 GetShort(String name);
```

Eşlem iletisinin gövdesinden ada göre tanımlanan kısa tamsayıyı alın.

**Parametreler:**

**ad (giriş)**

Bir dizgi nesnesi, kısa tamsayıyı tanımlayan adı sarkıyor.

**Döndürülen:**

Eşlem iletisinin gövdesinden alınan kısa tamsayı.

**Özel durumlar:**

- XMSEException

*GetString -Get String*

**Arabirim:**

```
String GetString(String name);
```

Eşlemin gövdesinden ada göre tanımlanan dizgiyi alın.

**Parametreler:**

**ad (giriş)**

Bir String nesnesi, eşlem iletisinin gövdesindeki dizeyi tanımlayan adı sarkıyor.

**Döndürülen:**

Eşleme iletisinin gövdesinden alınan dizeyi sarmalayan bir String nesnesi. Veri dönüştürme gerekiyorsa, bu değer dönüştürmenin ardından dizgidir.

**Özel durumlar:**

- XMSEException

*ItemExists -Denetim Adı-Değer çifti Var*

**Arabirim:**

```
Boolean ItemExists(String name);
```

Eşlem iletisinin gövdesinin, belirtilen ada sahip bir ad-değer çifti içerip içermediğini denetleyin.

**Parametreler:**

**ad (giriş)**

Ad-değer çiftinin adını sarsalayan bir String nesnesi.

**Döndürülen:**

- True, if the body of the map message contains a name-value pair with the specified name.
- False, if the body of the map message does not contain a name-value pair with the specified name.

**Özel durumlar:**

- XMSEException

*SetBoolean -Boole Değerini Ayarla*

**Arabirim:**

```
void SetBoolean(String name, Boolean value);
```

Eşlem iletisinin gövdesinde bir boole değeri ayarlayın.

**Parametreler:****ad (giriş)**

Bir dizgi nesnesi, eşlem iletisinin gövdesindeki boole değerini saptamak için adı sarkıyor.

**değer (giriş)**

Ayarlanacak boole değeri.

**Döndürülen:**

Hükümsüz

**Özel durumlar:**

- XMSEException

*SetByte -Set Byte*

**Arabirim:**

```
void SetByte(String name, Byte value);  
void SetSignedByte(String name, Int16 value);
```

Eşlem iletisinin gövdesinde bir bayt belirleyin.

**Parametreler:****ad (giriş)**

Bir dizgi nesnesi, eşlem iletisinin gövdesindeki byte 'ı tanıtmak için adı sarkıyor.

**değer (giriş)**

Ayarlanacak bayt.

**Döndürülen:**

Hükümsüz

**Özel durumlar:**

- XMSEException

*SetBytes -Set Bytes*

**Arabirim:**

```
void SetBytes(String name, Byte[] value);
```

Eşlem iletisinin gövdesinde bir bayt dizisi ayarlayın.

**Parametreler:****ad (giriş)**

Bir String nesnesi, eşlem iletisinin gövdesindeki byte dizilerinin tanınabilmek için adını sarkıyor.

**değer (giriş)**

Ayarlanacak bayt dizisi.

**Döndürülen:**

Hükümsüz

**Özel durumlar:**

- XMSEException

## *SetChar -Karakter kümesi*

### **Arabirim:**

```
void SetChar(String name, Char value);
```

Eşlem iletisinin gövdesinde 2 byte 'lık bir karakter belirleyin.

### **Parametreler:**

#### **ad (giriş)**

Bir String nesnesi, eşlem iletisinin gövdesindeki karakteri tanımlamak için adı sarkıyor.

#### **değer (giriş)**

Ayarlanacak karakter.

### **Döndürülen:**

Hükümsüz

### **Özel durumlar:**

- XMSEException

## *SetDouble -Çift Duyarlıklı Kayan Nokta Numarası Ayarla*

### **Arabirim:**

```
void SetDouble(String name, Double value);
```

Eşlem iletisinin gövdesinde bir çift duyarlıklı kayan noktalı sayı belirleyin.

### **Parametreler:**

#### **ad (giriş)**

Eşlem iletisinin gövdesindeki çift duyarlıklı kayan noktalı sayıyı tanımlamak için adı sarmalayan bir String nesnesi.

#### **değer (giriş)**

Ayarlanacak çift duyarlıklı kayan noktalı sayı.

### **Döndürülen:**

Hükümsüz

### **Özel durumlar:**

- XMSEException

## *SetFloat -Değişken Nokta Numarasını Ayarla*

### **Arabirim:**

```
void SetFloat(String name, Single value);
```

Eşlem iletisinin gövdesinde bir kayan noktalı sayı belirleyin.

### **Parametreler:**

#### **ad (giriş)**

Bir dizgi nesnesi, eşlem iletisinin gövdesindeki kayan noktalı sayıyı tanımlamak için adı sarkıyor.

#### **değer (giriş)**

Ayarlanacak kayan noktalı sayı.

### **Döndürülen:**

Hükümsüz

### **Özel durumlar:**

- XMSEException

## *SetInt -Tamsayı Ayarla*

### **Arabirim:**

```
void SetInt(String name, Int32 value);
```

Eşlem iletisinin gövdesinde bir tamsayı belirtin.

### **Parametreler:**

#### **ad (giriş)**

Bir dizgi nesnesi, eşlem iletisinin gövdesindeki tamsayıyı tanımlamak için adı sarkıyor.

#### **değer (giriş)**

Ayarlanacak tamsayı.

### **Döndürülen:**

Hükümsüz

### **Özel durumlar:**

- XMSEException

## *SetLong -Uzun Tamsayı Belirle*

### **Arabirim:**

```
void SetLong(String name, Int64 value);
```

Eşlem iletisinin gövdesinde uzun bir tamsayı belirtin.

### **Parametreler:**

#### **ad (giriş)**

Bir String nesnesi, eşlem iletisinin gövdesindeki uzun tamsayıyı tanımlamak için adı sarkıyor.

#### **değer (giriş)**

Ayarlanacak uzun tamsayı.

### **Döndürülen:**

Hükümsüz

### **Özel durumlar:**

- XMSEException

## *SetObject -Nesneyi Ayarla*

### **Arabirim:**

```
void SetObject(String name, Object value);
```

Eşlem iletisinin gövdesinde bir XMS temel tipi olması gereken bir değer ayarlayın.

### **Parametreler:**

#### **ad (giriş)**

Bir dizgi nesnesi, eşlem iletisinin gövdesindeki değeri tanımlamak için adı sarkıyor.

#### **değer (giriş)**

Ayarlanacak değeri içeren bayt dizisi.

### **Döndürülen:**

Hükümsüz

### **Özel durumlar:**

- XMSEException



*SetShort -Kısa Tamsayı Ayarla*

**Arabirim:**

```
void SetShort(String name, Int16 value);
```

Eşlem iletisinin gövdesinde kısa bir tamsayı belirtin.

**Parametreler:**

**ad (giriş)**

Bir dizgi nesnesi, eşlem iletisinin gövdesindeki kısa tamsayıyı tanımlamak için adı sarkıyor.

**değer (giriş)**

Ayarlanacak kısa tamsayı.

**Döndürülen:**

Hükümsüz

**Özel durumlar:**

- XMSEException

*SetString -Dizgi Ayarla*

**Arabirim:**

```
void SetString(String name, String value);
```

Eşlem iletisinin gövdesinde bir dizgi belirleyin.

**Parametreler:**

**ad (giriş)**

Bir String nesnesi, eşlem iletisinin gövdesindeki dizeyi tanımlamak için adı sarsalıyor.

**değer (giriş)**

Ayarlanacak dizeyi sarmalayan bir String nesnesi.

**Döndürülen:**

Hükümsüz

**Özel durumlar:**

- XMSEException

***Edinilmiş özellikler ve yöntemler***

Aşağıdaki özellikler IMessage arabiriminden edinilir:

JMSCorrelationID, JMSDeliveryMode, JMSDestination, JMSExpiration, JMSMessageID, JMSPriority, JMSReteslim, JMSReplyTo, JMSTimestamp, JMSType, Özellikler

Aşağıdaki yöntemler IMessage arabiriminden edinilir:

clearBody, clearProperties, PropertyExists

Aşağıdaki yöntemler IPropertyContext arabiriminden edinilir:

GetBooleanProperty, GetByteProperty, GetBytesProperty, GetCharProperty, GetDoubleProperty, GetFloatProperty, GetIntProperty, GetLongProperty, GetObjectProperty, GetShortProperty, GetStringProperty, SetBooleanProperty, SetByteProperty, SetBytesProperty, SetCharProperty, SetDoubleProperty, SetFloatProperty, SetIntProperty, SetLongProperty, SetObjectProperty, SetShortProperty, SetStringProperty

## İleti

Bir ileti nesnesi, uygulamanın gönderdiği ya da aldığı bir iletiyi gösterir. IMessage is a superclass for the message classes such as IMapMessage.

### Devralma sıradüzeni:

```
IBM.XMS.IPropertyContext
|
+----IBM.XMS.IMessage
```

İleti nesnesindeki JMS ileti üstbilgisi alanlarının bir listesi için bkz. [XMS iletisine ilişkin üstbilgi alanları](#). Bir ileti nesnesinin JMS tanımlı özelliklerinin bir listesi için [İletinin JMS tanımlı özellikleribaşlıklı konuya](#) bakın. Bir ileti nesnesinin IBM tanımlı özelliklerinin bir listesi için bkz. [IBM-bir iletinin tanımlı özellikleri](#). İleti nesnesine ilişkin JMS\_IBM\_MQMD\* özelliklerinin bir listesi için bkz. [“JMS\\_IBM\\_MQMD\\* özellikleri” sayfa 2011](#)

İletiler, çöp toplayıcı tarafından silinir. Bir ileti silindiğinde, bu ileti kullanmakta olduğu kaynakları boşaltır.

## .NET özellikler

*GetJMSCorrelationKimlik-Alma ve Ayarlama JMSCorrelationID*

### Arabirim:

```
String JMSCorrelationID
{
    get;
    set;
}
```

İletinin ilinti tanıtıcısını String nesnesi olarak ayarlayın ve ayarlayın.

### Özel durumlar:

- XMSException

*JMSDeliveryMode -Get and Set JMSDeliveryMode*

### Arabirim:

```
DeliveryMode JMSDeliveryMode
{
    get;
    set;
}
```

İletiyeye ilişkin teslim kipini alın ve ayarlayın.

İletinin teslim kipi aşağıdaki değerlerden biridir:

```
DeliveryMode.Persistent
DeliveryMode.NonPersistent
```

Gönderilmeyen yeni oluşturulan bir ileti için teslim kipi DeliveryMode' dir. Kalıcı, teslim kipinin DeliveryMode.NonPersistent olduğu bir aracıya gerçek zamanlı bağlantı dışında. Alınan bir ileti için yöntem, receiving.send () çağrısı tarafından ayarlanan uygulama teslim kipini döndürür; ancak, alma kipi JMSDeliveryMode ayarlanarak teslim kipini değiştirmedeği sürece, ileti, IMessageProducer.send () çağrısıyla ayarlanan teslim kipini döndürür.

### Özel durumlar:

- XMSException

**Arabirim:**

```
IDestination JMSDestination
{
    get;
    set;
}
```

İletinin hedefini alın ve ayarlayın.

Hedef, ileti gönderildiğinde IMessageProducer.send () çağrısıyla ayarlanır. JMSDestination değeri yok sayıldı. Ancak, alınan bir iletinin hedefini değiştirmek için JMSDestination 'ı kullanabilirsiniz.

Gönderilmeyen yeni oluşturulan bir ileti için, gönderme uygulaması JMSDestination ayarlanarak bir hedef belirlemezse, yöntem boş değerli bir hedef nesne döndürür. For a message that was received, the method returns a Destination object for the destination that was set by the IMessageProducer.send() call when the message was sent unless the receiving application changes the destination by setting JMSDestination.

**Özel durumlar:**

- XMSEException

**Arabirim:**

```
Int64 JMSExpiration
{
    get;
    set;
}
```

İletinin son kullanma tarihini alın ve ayarlayın.

Süre bitim süresi, ileti gönderildiğinde IMessageProducer.send () çağrısına göre ayarlanır. Değeri, gönderen uygulama tarafından belirtildiği gibi, iletinin gönderildiği zamana kadar, iletinin gönderildiği saat eklenerek hesaplanır. Süre bitimi, 1 Ocak 1970 'te 00:00:00 GMT ' den bu yana milisaniye olarak ifade edilir.

Gönderilmeyen yeni oluşturulan bir ileti için, gönderme uygulaması JMSExpiration ayarını ayarlayarak farklı bir süre bitim süresi ayarlamadıkça, süre bitim saati 0 olur. For a message that was received, the method returns the expiration time that was set by the IMessageProducer.send() call when the message was sent unless the receiving application changes the expiration time by setting JMSExpiration.

If the time to live is 0, the IMessageProducer.send() call sets the expiration time to 0 to indicate that the message does not expire.

XMS , süresi dolmuş iletileri atar ve bunları uygulamalara teslim etmez.

**Özel durumlar:**

- XMSEException

**Arabirim:**

```
String JMSMessageID
{
    get;
    set;
}
```

İletinin ileti tanıtıcısını alın ve ileti tanıtıcısını sarmalayan bir dizgi nesnesi olarak ayarlayın.

İleti tanıtıcısı, ileti gönderildiğinde IMessageProducer.send () çağrısıyla belirlenir. For a message that was received, the method returns the message identifier that was set by the IMessageProducer.send() call when the message was sent unless the receiving application changes the message identifier by setting JMSPriority.

İletin ileti tanıtıcısı yoksa, yöntem boş değer döndürür.

#### Özel durumlar:

- XMSEException

*JMSPriority- JMSPriority Al*

#### Arabirim:

```
Int32 JMSPriority
{
    get;
    set;
}
```

İletin önceliğini alın ve ayarlayın.

Öncelik, ileti gönderildiğinde IMessageProducer.send () çağrısına göre ayarlanır. The value is an integer in the range 0, the lowest priority, to 9, the highest priority.

Gönderilmeyen yeni oluşturulan bir ileti için, gönderme uygulaması JMSPriority ayarını tanımlayarak farklı bir öncelik ayarlamadıkça, öncelik 4 olur. Alınan bir ileti için, yöntem, JMSPriority değerini ayarlayarak, alma uygulamasının önceliği değiştirmedeği sürece, ileti gönderildiğinde, yöntem IMessageProducer.send () çağrısıyla ayarlanan önceliği döndürür.

#### Özel durumlar:

- XMSEException

*JMSReverled-Get ve Set JMSRever*

#### Arabirim:

```
Boolean JMSRedelivered
{
    get;
    set;
}
```

İletin yeniden teslim edilip edilmediğine ilişkin bir gösterge elde edin ve iletinin yeniden teslim edilip edilmediğini belirtin. Bu gösterge, ileti alındığında IMessageConsumer.alncall () çağrısına göre ayarlanır.

Bu özellik aşağıdaki değerlere sahiptir:

- True, if the message is being redelivered.
- False, if the message is not being redelivered.

Bir aracıya gerçek zamanlı bağlantı için değer her zaman False' dir.

İleti gönderilmeden önce, JMSResaled tarafından ayarlanan yeniden teslim edilmenin bir göstergesi, ileti gönderildiğinde IMessageProducer.send () çağrısı dikkate alınmaz ve ileti alındığında IMessageConsumer.receive () çağrısıyla yoksayılır ve değiştirilir. Ancak, alınan bir iletinin göstergesini değiştirmek için JMSReverled olanağını kullanabilirsiniz.

#### Özel durumlar:

- XMSEException

*JMSReplyTo -GetJMSReplyToGet' u alın ve ayarlayın.*

#### **Arabirim:**

```
IDestination JMSReplyTo
{
    get;
    set;
}
```

İletinin yanıtının gönderileceği hedefi alın ve ayarlayın.

Bu özelliğin değeri, iletiye verilen yanıtın gönderileceği hedef nesne nesnesidir. Boş değerli bir hedef nesne, yanıt beklenmediği anlamına gelir.

#### **Özel durumlar:**

- XMSEException

*JMSTimestamp-JMSTimestamp 'ı Al ve Ayarla*

#### **Arabirim:**

```
Int64 JMSTimestamp
{
    get;
    set;
}
```

İletinin gönderildiği saati alın ve ayarlayın.

Zaman damgası, ileti gönderildiğinde IMessageProducer.send () çağrısına göre ayarlanır ve 1 Ocak 1970 'te 00:00:00 GMT ' den bu yana milisaniye olarak ifade edilir.

Gönderilmeyen yeni oluşturulan bir ileti için, gönderme uygulaması JMSTimestamp ayarlanarak farklı bir zaman damgası ayarlamadıkça, zaman damgası 0 olur. For a message that was received, the method returns the time stamp that was set by the IMessageProducer.send() call when the message was sent unless the receiving application changes the time stamp by setting JMSTimestamp.

#### **Özel durumlar:**

- XMSEException

#### **Notlar:**

1. Zaman damgası tanımsızsa, yöntem 0 değerini döndürür, ancak kural dışı durum yayınlanmaz.

*JMSType-JMSType 'ı Al ve Ayarla*

#### **Arabirim:**

```
String JMSType
{
    get;
    set;
}
```

İletinin tipini alın ve ayarlayın.

JMSType değeri, iletinin tipini sarmalayan bir dizilimdir. Veri dönüştürme gerekliyse, bu değer dönüştürmenin ardından tiptir.

#### **Özel durumlar:**

- XMSEException

## PropertyNames -Özellikleri Al

### Arabirim:

```
System.Collections.IEnumerator PropertyNames
{
    get;
}
```

İletinin ad özelliklerine ilişkin sıralı değer listesi alın.

### Özel durumlar:

- XMSException

## Yöntemler

### Alındı Bilgisi-Akbilgi

### Arabirim:

```
void Acknowledge();
```

Bu iletiyi ve oturum tarafından alınan önceden kabul edilmemiş tüm iletileri kabul edin.

Bir uygulama, oturumun alındı bildirimini kipi AcknowledgeMode.ClientAcknowledgeise bu yöntemi çağırabilir. Oturum başka bir alındı bildirimini kipine sahipse ya da işlem hareketliyse, yonteme yapılan çağrılar yoksayılr.

Alınan ancak kabul edilmeyen iletiler yeniden teslim edilebilir.

İletilerin kabul edilmesi hakkında daha fazla bilgi için bkz. [../com.ibm.mq.dev.doc/xms\\_cmesack.dita#xms\\_cmesack](http://com.ibm.mq.dev.doc/xms_cmesack.dita#xms_cmesack).

### Parametreler:

Yok

### Döndürülen:

Hükümsüz

### Özel durumlar:

- XMSException
- IllegalStateKural Dışı Durumu

### ClearBody -Gövde Temizle

### Arabirim:

```
void ClearBody();
```

İletinin gövdeini temizleyin. Üstbilgi alanları ve ileti özellikleri temizlenmez.

Bir uygulama bir ileti gövdeyi temizlerse, gövde yeni oluşturulan bir iletide boş bir gövle aynı durumda kalır. Yeni oluşturulan bir iletide boş bir gövdenin durumu, ileti gövdesinin tipine bağlıdır. Ek bilgi için [XMS iletisinin gövdesinebakın](#).

Bir uygulama, vücudun hangi durumda olursa olsun, bir ileti gövdinden herhangi bir zamanda temizlenemez. Bir ileti gövdesi salt okunursa, uygulamanın gövdeye yazabileceği tek yol, uygulamanın önce göveyi temizleyebilmesini sağlar.

### Parametreler:

Yok

**Döndürülen:**

Hükümsüz

**Özel durumlar:**

- XMSEException

*ClearProperties -Özellikleri Temizle*

**Arabirim:**

```
void ClearProperties();
```

İletinin özelliklerini temizleyin. Üstbilgi alanları ve ileti gövdesi temizlenmez.

Bir uygulama bir iletinin özelliklerini temizlerse, özellikler okunabilir ve yazılabilir hale gelir.

Bir uygulama, özelliklerin hangi durumda olursa olsun, her an bir iletinin özelliklerini temizleyebilir. Bir iletinin özellikleri salt okunur ise, uygulamanın yazılabilir hale gelebileceği tek yol, uygulamanın önce özellikleri temizleyebilmesini sağlar.

**Parametreler:**

Yok

**Döndürülen:**

Hükümsüz

**Özel durumlar:**

- XMSEException

*PropertyExists -Denetleme Özelliği Var*

**Arabirim:**

```
Boolean PropertyExists(String propertyName);
```

İletinin belirtilen ada sahip bir özelliği olup olmadığını denetleyin.

**Parametreler:****propertyName (giriş)**

Özelliğin adını sarmalayan bir String nesnesi.

**Döndürülen:**

- True, if the message has a property with the specified name.
- False, if the message does not have a property with the specified name.

**Özel durumlar:**

- XMSEException

**Edinilmiş özellikler ve yöntemler**

Aşağıdaki yöntemler IPropertyContext arabiriminden edinilir:

GetBooleanProperty, GetByteProperty, GetBytesProperty, GetCharProperty, GetDoubleProperty, GetFloatProperty, GetIntProperty, GetLongProperty, GetObjectProperty, GetShortProperty, GetStringProperty, SetBooleanProperty, SetByteProperty, SetBytesProperty, SetCharProperty, SetDoubleProperty, SetFloatProperty, SetIntProperty, SetLongProperty, SetObjectProperty, SetShortProperty, SetStringProperty

**IMessageConsumer**

Bir uygulama, bir hedefe gönderilen iletleri almak için bir ileti tüketicisi kullanır.

## Devralma sıradüzeni:

```
IBM.XMS.IPropertyContext
|
+---- IBM.XMS.IMessageConsumer
```

Bir MessageConsumer nesnesine ilişkin XMS tanımlı özelliklerinin bir listesi için bkz. [“MessageConsumerözellikleri” sayfa 2014.](#)

## .NET özellikler

*MessageListener -İleti Dinleyicisini Al ve Ayarla*

### Arabirim:

```
MessageListener MessageListener
{
    get;
    set;
}
```

İleti tüketicisi ile kaydedilen ileti dinleyicisini edinin ve ileti dinleyicisine ileti dinleyicisini kaydettirin.

İleti tüketicisi ile kayıtlı bir ileti dinleyicisi yoksa, MessageListener boş değerli olur. İleti dinleyicisi zaten ileti tüketicisi ile kayıtlıysa, bunun yerine boş değer belirterek kayıt işlemini iptal edebilirsiniz.

İleti dinleyicileri kullanma hakkında daha fazla bilgi için bkz. [.NET ' te ileti ve kural dışı durum dinleyicilerinin kullanılması.](#)

### Özel durumlar:

- XMSEception

*MessageSelector -İleti Seçiciyi Al*

### Arabirim:

```
String MessageSelector
{
    get;
}
```

İleti tüketicisi için ileti seçiciyi alın. Dönüş değeri, ileti seçici ifadesini sarmalayan bir String nesnesidir. Veri dönüştürme gerekiyorsa, bu değer dönüştürmeden sonraki ileti seçici ifadesidir. İleti tüketicisi bir ileti seçiciye sahip değilse, MessageSelector değeri bir boş Dizgi nesnesidir.

### Özel durumlar:

- XMSEception

## Yöntemler

*Kapat-İleti Tüketicisi Kapat*

### Arabirim:

```
void Close();
```

İleti tüketicisini kapatın.

Bir uygulama önceden kapatılmış bir ileti tüketicisi kapatmaya çalışırsa, arama yoksayılr.

### Parametreler:

Yok



**Döndürülen:**

Hükümsüz

**Özel durumlar:**

- XMSEException

*Alma-Alma*

**Arabirim:**

```
IMessage Receive();
```

İleti tüketicisi için bir sonraki iletiyi alın. Çağrı, bir ileti için süresiz olarak ya da ileti tüketicisi kapatılıncaya kadar bekler.

**Parametreler:**

Yok

**Döndürülen:**

İleti nesnesine bir gösterge. Çağrı bir iletiyi beklerken ileti tüketicisi kapatılırsa, yöntem boş değerli bir ileti nesnesine ilişkin bir gösterge döndürür.

**Özel durumlar:**

- XMSEException

*Alma-Alma (bekleme aralığıyla birlikte)*

**Arabirim:**

```
IMessage Receive(Int64 delay);
```

İleti tüketicisi için bir sonraki iletiyi alın. Çağrı, ileti için yalnızca belirli bir dönemi bekler ya da ileti tüketicisi kapatılıncaya kadar bekler.

**Parametreler:****gecikme (giriş)**

Çağrıların bir ileti için bekleyeceği süre (milisaniye). Bir bekleme aralığı 0 belirlerseniz, arama ileti için süresiz olarak bekler.

**Döndürülen:**

İleti nesnesine bir gösterge. Bekleme süresi boyunca hiçbir ileti gelmezse ya da çağrı bir iletiyi beklerken ileti tüketicisi kapatılırsa, yöntem boş değerli bir Message nesnesine bir gösterge döndürür, ancak kural dışı durum yayınlanmaz.

**Özel durumlar:**

- XMSEException

*ReceiveNoBekleme-Bekleme ile Alma*

**Arabirim:**

```
IMessage ReceiveNoWait();
```

İleti tüketicisi için bir sonraki iletiyi alırsanız, hemen bir sonraki iletiyi alın.

**Parametreler:**

Yok

**Döndürülen:**

Bir ileti nesnesine ilişkin gösterge. Hemen kullanılabilir bir ileti yoksa, yöntem boş değerli bir ileti nesnesine işaretçi döndürür.

## Özel durumlar:

- [XMSEException](#)

## Edinilmiş özellikler ve yöntemler

Aşağıdaki yöntemler [IPropertyContext](#) arabiriminden edinilir:

[GetBooleanProperty](#), [GetByteProperty](#), [GetBytesProperty](#), [GetCharProperty](#), [GetDoubleProperty](#), [GetFloatProperty](#), [GetIntProperty](#), [GetLongProperty](#), [GetObjectProperty](#), [GetShortProperty](#), [GetStringProperty](#), [SetBooleanProperty](#), [SetByteProperty](#), [SetBytesProperty](#), [SetCharProperty](#), [SetDoubleProperty](#), [SetFloatProperty](#), [SetIntProperty](#), [SetLongProperty](#), [SetObjectProperty](#), [SetShortProperty](#), [SetStringProperty](#)

## MessageEOFException

XMS , bir uygulama byte iletilerinin gövdeini okurken bir byte ileti akışının sonuna rastlanırsa,XMS bu kural dışı durumu döndürür.

### Devralma sıradüzeni:

```
IBM.XMS.XMSEException
|
+----IBM.XMS.XMSEException
|
+----IBM.XMS.MessageEOFException
```

## Edinilmiş özellikler ve yöntemler

Aşağıdaki yöntemler [XMSEException](#) arabiriminden edinilir:

[GetErrorCode](#), [GetLinkedException](#)

## MessageFormatKural Dışı Durumu

XMS throws this exception if XMS encounters a message with a format that is not valid.

### Devralma sıradüzeni:

```
IBM.XMS.XMSEException
|
+----IBM.XMS.XMSEException
|
+----IBM.XMS.MessageFormatException
```

## Edinilmiş özellikler ve yöntemler

Aşağıdaki yöntemler [XMSEException](#) arabiriminden edinilir:

[GetErrorCode](#), [GetLinkedException](#)

## IMessageListener (temsilci)

Bir uygulama, iletileri zamanuyumsuz olarak almak için bir ileti dinleyicisini kullanır.

### Devralma sıradüzeni:

Yok

## Ata

## MessageListener - İleti Dinleyicisi

### Arabirim:

```
public delegate void MessageListener(IMessage msg);
```

İleti tüketicisine zamanuyumsuz bir ileti sunun.

Bu yetkilendirmeyi uygulayan yöntemler, bağlantıyla birlikte kaydedilebilir.

İleti dinleyicileri kullanma hakkında daha fazla bilgi için bkz. [Using message and exception listeners in .NET](#).

### Parametreler:

#### mesg (giriş)

İleti nesnesi.

### Döndürülen:

Hükümsüz

## MessageNotReadableException

XMS , bir uygulama yalnızca yazma işlemi olan bir iletinin gövdenini okumayı denediğinde bu kural dışı durumu atar.

### Devralma sıradüzeni:

```
IBM.XMS.XMSException
|
+----IBM.XMS.XMSException
|
+----IBM.XMS.MessageNotReadableException
```

### Edinilmiş özellikler ve yöntemler

Aşağıdaki yöntemler [XMSException](#) arabiriminden edinilir:

[GetErrorCode](#), [GetLinkedException](#)

## MessageNotWritableException

XMS , bir uygulama salt okunur bir iletinin gövdesine yazma girişiminde bulunursa bu kural dışı durumu atar.

### Devralma sıradüzeni:

```
IBM.XMS.XMSException
|
+----IBM.XMS.XMSException
|
+----IBM.XMS.MessageNotWritableException
```

### Edinilmiş özellikler ve yöntemler

Aşağıdaki yöntemler [XMSException](#) arabiriminden edinilir:

[GetErrorCode](#), [GetLinkedException](#)

## IMessageProducer

Bir uygulama, bir hedefe ileti göndermek için ileti üreticiyi kullanır.

## Devralma sıradüzeni:

```
IBM.XMS.IPropertyContext
|
+----IBM.XMS.IMessageProducer
```

Bir MessageProducer nesnesine ilişkin XMS tanımlı özelliklerinin bir listesi için bkz. [“MessageProducerözellikleri” sayfa 2015.](#)

## .NET özellikler

*DeliveryMode -Varsayılan Teslim Kipini Al ve Ayarla*

### Arabirim:

```
DeliveryMode DeliveryMode
{
    get;
    set;
}
```

İleti üreticisi tarafından gönderilen iletiler için varsayılan teslim kipini alın ve ayarlayın.

Varsayılan teslim kipi aşağıdaki değerlerden birine sahiptir:

```
DeliveryMode.Persistent
DeliveryMode.NonPersistent
```

Bir aracıya gerçek zamanlı bağlantı için değer `DeliveryMode.NonPersistent` olmalıdır.

Varsayılan değer, varsayılan değer `DeliveryMode.NonPersistent` olduğu bir aracıya gerçek zamanlı bağlantı dışında `DeliveryMode.Persistent` değeridir.

### Özel durumlar:

- `XMSException`

*Hedef-Hedef Al*

### Arabirim:

```
IDestination Destination
{
    get;
}
```

İleti üreticisine ilişkin hedefi alın.

### Parametreler:

Yok

### Döndürülen:

Hedef nesne. İleti üreticinin bir hedefi yoksa, yöntem boş değerli bir hedef nesne döndürür.

### Özel durumlar:

- `XMSException`

*DisableMsgKimliği-İleti Tanıtıcısı İşaretini Al ve Ayarla İşaretini Ayarla*

### Arabirim:

```
Boolean DisableMessageID
{
    get;
```

```
    set;
}
```

Alma uygulamasının ileti üreticisi tarafından gönderilen iletilere eklenmesi için ileti tanıtıcıları gerektirip gerektirmediğine ilişkin bir bilgi edinin ve alıcı uygulamanın ileti üreticisi tarafından gönderilen iletilere eklenmesi için ileti tanıtıcıları gerektirip gerektirmediğini belirtin.

Kuyruk yöneticisine ya da bir aracıya gerçek zamanlı bir bağlantı üzerinde yapılan bir bağlantıda bu işaret yoksayılr. Bir hizmet bütünleştirme veriyoluna yapılan bağlantıda, işaret onurlandırılır.

DisabledMsgTanıtıcısı şu değerlere sahiptir:

- `True`, bir alma uygulaması ileti üreticinin gönderdiği iletilerde ileti tanıtıcıları gerektirmiyorsa.
- `False`, alma uygulaması, ileti üreticisi tarafından gönderilen iletilere eklenecek ileti tanıtıcıları gerektiriyorsa.

#### Özel durumlar:

- `XMSEException`

*DisableMsgTS-Zaman Damgası İşaretini Al ve Ayarla İşaretini Ayarla*

#### Arabirim:

```
Boolean DisableMessageTimestamp
{
    get;
    set;
}
```

Alma uygulamasının, ileti üreticisi tarafından gönderilen iletilere eklenecek zaman damgalarını gerektirip gerektirmediğine ilişkin bir bilgi edinin ve bir alma uygulamasının, ileti üreticisi tarafından gönderilen iletilere dahil edilecek zaman damgalarını gerektirip gerektirmediğini belirtin.

Bir aracıya gerçek zamanlı bir bağlantı üzerinde bu işaret yoksayılr. Kuyruk yöneticisine ya da hizmet bütünleştirme veriyoluna yönelik bir bağlantıda, işaret onurlandırılır.

TSDisableMsgDisableMsgTS aşağıdaki değerlere sahiptir:

- `True`, bir alma uygulamasının ileti üreticisi tarafından gönderilen iletilere dahil edilmesi için zaman damgalarını gerektirmemesi durumunda.
- `False`, alma uygulaması, ileti üreticisi tarafından gönderilen iletilere dahil edilecek zaman damgalarını gerektiriyorsa.

#### Döndürülen:

#### Özel durumlar:

- `XMSEException`

*Öncelik-Varsayılan Önceliği Al ve Ayarla*

#### Arabirim:

```
Int32 Priority
{
    get;
    set;
}
```

İleti üreticisi tarafından gönderilen iletiler için varsayılan öncelik değerini alın ve ayarlayın.

Varsayılan ileti önceliğinin değeri, en düşük öncelik olan 0 aralığındaki bir tamsayıdır; en yüksek öncelik 9'a, en yüksek önceliğe sahip olur.

Bir aracıya gerçek zamanlı bir bağlantıda, bir iletinin önceliği yok sayılır.

### Özel durumlar:

- XMSEException

*TimeToLive-Varsayılan Zamanı Canlı Olarak Al ve Etkin Olarak Ayarla*

### Arabirim:

```
Int64 TimeToLive
{
    get;
    set;
}
```

Bir iletinin süresi dolmadan önce var olan varsayılan süreyi elde edin ve ayarlayın.

Zaman, ileti üreticinin iletiyi gönderdiğinde ve milisaniye olarak yaşamanın varsayılan zamanı olduğu zamandan itibaren ölçülür. 0 değeri, bir iletinin hiçbir zaman süre bitimine geçmediği anlamına gelir.

Bir aracıya gerçek zamanlı bağlantı için bu değer her zaman 0 'tır.

### Özel durumlar:

- XMSEException

### Yöntemler

*İleti Üreticisini Kapat-Kapat*

### Arabirim:

```
void Close();
```

İleti üreticisini kapatın.

Bir uygulama önceden kapatılmış bir ileti üreticisini kapatmaya çalışırsa, arama yoksayılr.

### Parametreler:

Yok

### Döndürülen:

Hükümsüz

### Özel durumlar:

- XMSEException

*Gönder-Gönder*

### Arabirim:

```
void Send(IMessage msg) ;
```

İleti üreticisi yaratıldığında belirlenen hedefe bir ileti gönderir. İleti üreticisi varsayılan teslim kipini, önceliğini ve yaşama süresini kullanarak iletiyi gönderin.

### Parametreler:

**msg (giriş)**

İleti nesnesi.

### Döndürülen:

Hükümsüz

### Özel durumlar:

- XMSEException

- MessageFormatKural Dışı Durumu
- InvalidDestinationKural Dışı Durumu

*Gönder-Gönder (teslim kipi belirtme, öncelik ve yaşama süresi)*

**Arabirim:**

```
void Send(IMessage msg,  
         DeliveryMode deliveryMode,  
         Int32 priority,  
         Int64 timeToLive);
```

İleti üreticisi yaratıldığında belirlenen hedefe bir ileti gönderir. Belirtilen teslim kipini, önceliğini ve yaşama süresini kullanarak iletiyi gönderin.

**Parametreler:**

**msg (giriş)**

İleti nesnesi.

**deliveryMode (giriş)**

İletiyeye ilişkin teslim kipi; aşağıdaki değerlerden biri olmalıdır:

DeliveryMode.Persistent  
DeliveryMode.NonPersistent

Bir aracıya gerçek zamanlı bağlantı için değer DeliveryMode.NonPersistent olmalıdır.

**öncelik (giriş)**

İletinin önceliği. Değer, en düşük öncelik için 0 aralığında, en düşük öncelik için 9 değerine sahip bir tamsayı olabilir. Bir aracıya gerçek zamanlı bir bağlantı üzerinde değer yok sayılır.

**timeToLive (giriş)**

İletinin milisaniye cinsinden yaşaması için kullanılan süre. 0 değeri, iletinin hiçbir zaman süre bitimine geçmediği anlamına gelir. Bir aracıya gerçek zamanlı bağlantı için değer 0 olmalıdır.

**Döndürülen:**

Hükümsüz

**Özel durumlar:**

- XMSException
- MessageFormatKural Dışı Durumu
- InvalidDestinationKural Dışı Durumu
- IllegalStateKural Dışı Durumu

*Gönder-Gönder (belirtilen hedefe)*

**Arabirim:**

```
void Send(IDestination dest, IMessage msg) ;
```

İleti üreticisi yaratıldığında hedef belirtilmediyse, belirtilen hedefe bir ileti gönderir. İleti üreticisi varsayılan teslim kipini, önceliğini ve yaşama süresini kullanarak iletiyi gönderin.

Genellikle, bir ileti üreticisi yarattığınızda bir hedef belirtiyorsunuz, ancak yapmazsanız, ileti gönderdiğinizde hedef belirtmeniz gerekir.

**Parametreler:**

**en dest (giriş)**

Hedef nesne.

**msg (giriş)**

İleti nesnesi.

**Döndürülen:**

Hükümsüz

**Özel durumlar:**

- XMSException
- MessageFormatKural Dışı Durumu
- InvalidDestinationKural Dışı Durumu

*Gönder-Gönder (belirli bir hedefe bir teslim kipi, öncelik ve yaşama süresi belirtme)*

**Arabirim:**

```
void Send(IDestination dest,
          IMessage msg,
          DeliveryMode deliveryMode,
          Int32 priority,
          Int64 timeToLive) ;
```

İleti üreticisi yaratıldığında hedef belirtilmediyse, belirtilen hedefe bir ileti gönderir. Belirtilen teslim kipini, önceliğini ve yaşama süresini kullanarak iletiyi gönderin.

Genellikle, bir ileti üreticisi yarattığınızda bir hedef belirtiyorsunuz, ancak yapmazsanız, ileti gönderdiğinizde hedef belirtmeniz gerekir.

**Parametreler:****en dest (giriş)**

Hedef nesne.

**msg (giriş)**

İleti nesnesi.

**deliveryMode (giriş)**

İletiye ilişkin teslim kipi; aşağıdaki değerlerden biri olmalıdır:

DeliveryMode.Persistent

DeliveryMode.NonPersistent

Bir aracıya gerçek zamanlı bağlantı için değer DeliveryMode.NonPersistent olmalıdır.

**öncelik (giriş)**

İletinin önceliği. Değer, en düşük öncelik için 0 aralığında, en düşük öncelik için 9 değerine sahip bir tamsayı olabilir. Bir aracıya gerçek zamanlı bir bağlantı üzerinde değer yok sayılır.

**timeToLive (giriş)**

İletinin milisaniye cinsinden yaşaması için kullanılan süre. 0 değeri, iletinin hiçbir zaman süre bitimine geçmediği anlamına gelir. Bir aracıya gerçek zamanlı bağlantı için değer 0 olmalıdır.

**Döndürülen:**

Hükümsüz

**Özel durumlar:**

- XMSException
- MessageFormatKural Dışı Durumu
- InvalidDestinationKural Dışı Durumu
- IllegalStateKural Dışı Durumu

**Edinilmiş özellikler ve yöntemler**

Aşağıdaki yöntemler [IPropertyContext](#) arabiriminden edinilir:

[GetBooleanProperty](#), [GetByteProperty](#), [GetBytesProperty](#), [GetCharProperty](#), [GetDoubleProperty](#), [GetFloatProperty](#), [GetIntProperty](#), [GetLongProperty](#), [GetObjectProperty](#), [GetShortProperty](#), [GetStringProperty](#), [SetBooleanProperty](#), [SetByteProperty](#), [SetBytesProperty](#), [SetCharProperty](#),



[SetDoubleProperty](#), [SetFloatProperty](#), [SetIntProperty](#), [SetLongProperty](#), [SetObjectProperty](#), [SetShortProperty](#), [SetStringProperty](#)

## IObjectMessage

Bir nesne iletisi, gövdesi serileştirilmiş bir Java ya da .NET nesnesinden oluşan bir iletidir.

### Devralma sıradüzeni:

```
IBM.XMS.IPropertyContext
|
+----IBM.XMS.IMessage
|
+----IBM.XMS.IObjectMessage
```

## .NET özellikler

*Nesne-Nesneyi Bayt Olarak Al ve Ayarla*

### Arabirim:

```
System.Object Object
{
    get;
    set;
}

Byte[] GetObject();
```

Nesne iletisinin gövdelerini oluşturan nesneyi alın ve ayarlayın.

### Özel durumlar:

- XMSException
- MessageNotReadableException
- MessageEOFException
- MessageNotWritableException

## Edinilmiş özellikler ve yöntemler

Aşağıdaki özellikler [IMessage](#) arabiriminden edinilir:

[JMSCorrelationID](#), [JMSDeliveryMode](#), [JMSDestination](#), [JMSExpiration](#), [JMSPriority](#), [JMSMessageID](#), [JMSRetslim](#), [JMSReplyTo](#), [JMSTimestamp](#), [JMSType](#), [Özellikler](#)

Aşağıdaki yöntemler [IMessage](#) arabiriminden edinilir:

[clearBody](#), [clearProperties](#), [PropertyExists](#)

Aşağıdaki yöntemler [IPropertyContext](#) arabiriminden edinilir:

[GetBooleanProperty](#), [GetByteProperty](#), [GetBytesProperty](#), [GetCharProperty](#), [GetDoubleProperty](#), [GetFloatProperty](#), [GetIntProperty](#), [GetLongProperty](#), [GetObjectProperty](#), [GetShortProperty](#), [GetStringProperty](#), [SetBooleanProperty](#), [SetByteProperty](#), [SetBytesProperty](#), [SetCharProperty](#), [SetDoubleProperty](#), [SetFloatProperty](#), [SetIntProperty](#), [SetLongProperty](#), [SetObjectProperty](#), [SetShortProperty](#), [SetStringProperty](#)

## IPropertyContext

IPropertyContext , özellikleri alan ve ayarlayan yöntemler içeren, soyut bir üst sınıftır. Bu yöntemler diğer sınıflar tarafından devralınır.

### Devralma sıradüzeni:

Yok

## Yöntemler

*GetBoolean* özelliği-*Boolean* Özelliği Al

### Arabirim:

```
Boolean GetBooleanProperty(String property_name);
```

Belirtilen ada sahip Boole özelliğinin değerini alın.

### Parametreler:

#### özellik\_adi (giriş)

Özelliğın adını sarmalayan bir String nesnesi.

### Döndürülen:

Özelliğın deęeri.

### İş parçacığı bağlamı:

Alt sınıf tarafından belirlenir

### Özel durumlar:

- XMSException

*GetByte* özelliği-*Byte* Özelliği Al

### Arabirim:

```
Byte GetByteProperty(String property_name) ;  
Int16 GetSignedByteProperty(String property_name) ;
```

Ad ile tanımlanan bayt özelliğinin değerini alın.

### Parametreler:

#### özellik\_adi (giriş)

Özelliğın adını sarmalayan bir String nesnesi.

### Döndürülen:

Özelliğın deęeri.

### İş parçacığı bağlamı:

Alt sınıf tarafından belirlenir

### Özel durumlar:

- XMSException

*GetBytesProperty*-*Get Byte Array* özelliği

### Arabirim:

```
Byte[] GetBytesProperty(String property_name) ;
```

Ad ile tanımlanan bayt dizisi özelliğinin değerini alın.

### Parametreler:

#### özellik\_adi (giriş)

Özelliğın adını sarmalayan bir String nesnesi.

### Döndürülen:

Dizideki bayt sayısı.

### İş parçacığı bağlamı:

Alt sınıf tarafından belirlenir

**Özel durumlar:**

- XMSEException

*GetCharözelliği-Karakter Özelliği Al*

**Arabirim:**

```
Char GetCharProperty(String property_name) ;
```

Ad ile tanımlanan 2 baytlık karakter özelliğinin değerini alın.

**Parametreler:****özellik\_adi (giriş)**

Özelliğın adını sarmalayan bir String nesnesi.

**Döndürülen:**

Özelliğın deęeri.

**İş parçacığı bağlamı:**

Alt sınıf tarafından belirlenir

**Özel durumlar:**

- XMSEException

*GetDoubleözelliği-Double Precision Floating Point Özelliği Al*

**Arabirim:**

```
Double GetDoubleProperty(String property_name) ;
```

Ad ile tanımlanan çift duyarlıklı kayan nokta özelliğinin değerini alın.

**Parametreler:****özellik\_adi (giriş)**

Özelliğın adını sarmalayan bir String nesnesi.

**Döndürülen:**

Özelliğın deęeri.

**İş parçacığı bağlamı:**

Alt sınıf tarafından belirlenir

**Özel durumlar:**

- XMSEException

*GetFloatÖzelliği-Kayar Noktalı özelliği Al*

**Arabirim:**

```
Single GetFloatProperty(String property_name) ;
```

Ad ile tanımlanan kayan nokta özelliğinin değerini alın.

**Parametreler:****özellik\_adi (giriş)**

Özelliğın adını sarmalayan bir String nesnesi.

**Döndürülen:**

Özelliğın deęeri.

**İş parçacığı bağlamı:**

Alt sınıf tarafından belirlenir

### Özel durumlar:

- XMSEException

*GetIntÖzelliği- GetIntÖzelliği*

### Arabirim:

```
Int32 GetIntProperty(String property_name) ;
```

Ad ile tanımlanan tamsayı özelliğinin değerini alın.

### Parametreler:

#### **özellik\_adi (giriş)**

Özelliğın adını sarmalayan bir String nesnesi.

### Döndürülen:

Özelliğın deęeri.

### İş parçacığı bağlamı:

Alt sınıf tarafından belirlenir

### Özel durumlar:

- XMSEException

*GetLongözelliği-Uzun Tamsayı Özelliği Al*

### Arabirim:

```
Int64 GetLongProperty(String property_name) ;
```

Ad ile tanımlanan uzun tamsayı özelliğinin değerini alın.

### Parametreler:

#### **özellik\_adi (giriş)**

Özelliğın adını sarmalayan bir String nesnesi.

### Döndürülen:

Özelliğın deęeri.

### İş parçacığı bağlamı:

Alt sınıf tarafından belirlenir

### Özel durumlar:

- XMSEException

*GetObjectözelliği-Nesne özelliğini al*

### Arabirim:

```
Object GetObjectProperty( String property_name) ;
```

Ad ile tanımlanan özelliğın deęerini ve veri tipini alın.

### Parametreler:

#### **özellik\_adi (giriş)**

Özelliğın adını sarmalayan bir String nesnesi.

### Döndürülen:

Aşağıdaki nesne tiplerinden biri olan özelliğın deęeri:

Boolean

Byte

Byte[]  
Char  
Double  
Single  
Int32  
Int64  
Int16  
String

**İş parçacığı bağlamı:**

Alt sınıf tarafından belirlenir

**Özel durumlar:**

- XMSEException

*GetShortÖzelliği-Kısa Tamsayı Özelliği Al*

**Arabirim:**

```
Int16 GetShortProperty(String property_name) ;
```

Ad ile tanımlanan kısa tamsayı özelliğinin değerini alın.

**Parametreler:**

**özellik\_adi (giriş)**

Özelliğın adını sarmalayan bir String nesnesi.

**Döndürülen:**

Özelliğın deęeri.

**İş parçacığı bağlamı:**

Alt sınıf tarafından belirlenir

**Özel durumlar:**

- XMSEException

*GetStringÖzelliği- GetStringÖzelliği*

**Arabirim:**

```
String GetStringProperty(String property_name) ;
```

Ad ile tanımlanan dizgi özelliğinin deęerini alın.

**Parametreler:**

**özellik\_adi (giriş)**

Özelliğın adını sarmalayan bir String nesnesi.

**Döndürülen:**

Özelliğın deęeri olan dizeyi sarmalayan bir String nesnesi. Veri dönüştürme gerekiyorsa, bu deęer dönüştürmenin ardından dizgidir.

**İş parçacığı bağlamı:**

Alt sınıf tarafından belirlenir

**Özel durumlar:**

- XMSEException

## *SetBooleanözelliği-Boole Özelliği Ayarla*

### **Arabirim:**

```
void SetBooleanProperty( String property_name, Boolean value) ;
```

Ad ile tanımlanan Boole özelliğinin değerini ayarlayın.

### **Parametreler:**

#### **özellik\_adi (giriş)**

Özelliğın adını sarmalayan bir String nesnesi.

#### **değer (giriş)**

Özelliğın deęeri.

### **Döndürülen:**

Hükümsüz

### **İş parçacığı baęlamı:**

Alt sınıf tarafından belirlenir

### **Özel durumlar:**

- XMSException
- MessageNotWritableException

## *SetByteözelliği-Byte Özelliği Ayarla*

### **Arabirim:**

```
void SetByteProperty( String property_name, Byte value) ;  
void SetSignedByteProperty( String property_name, Int16 value) ;
```

Ad tarafından tanımlanan byte özelliğinin değerini ayarlayın.

### **Parametreler:**

#### **özellik\_adi (giriş)**

Özelliğın adını sarmalayan bir String nesnesi.

#### **değer (giriş)**

Özelliğın deęeri.

### **Döndürülen:**

Hükümsüz

### **İş parçacığı baęlamı:**

Alt sınıf tarafından belirlenir

### **Özel durumlar:**

- XMSException
- MessageNotWritableException

## *SetBytesProperty-Set Byte Array özelliği*

### **Arabirim:**

```
void SetBytesProperty( String property_name, Byte[] value ) ;
```

Ad tarafından tanımlanan bayt dizisi özelliğinin değerini ayarlayın.

### **Parametreler:**

#### **özellik\_adi (giriş)**

Özelliğın adını sarmalayan bir String nesnesi.

**değer (giriş)**

Bir bayt dizisi olan özelliğin değeri.

**Döndürülen:**

Hükümsüz

**İş parçacığı bağlamı:**

Alt sınıf tarafından belirlenir

**Özel durumlar:**

- XMSEException
- MessageNotWritableException

*SetCharÖzelliği-Karakter Takımı Özelliği*

**Arabirim:**

```
void SetCharProperty( String property_name, Char value) ;
```

Ad tarafından tanımlanan 2 baytlık karakter özelliğinin değerini ayarlayın.

**Parametreler:****özellik\_adi (giriş)**

Özelliğin adını sarmalayan bir String nesnesi.

**değer (giriş)**

Özelliğin değeri.

**Döndürülen:**

Hükümsüz

**İş parçacığı bağlamı:**

Alt sınıf tarafından belirlenir

**Özel durumlar:**

- XMSEException
- MessageNotWritableException

*SetDoubleözelliği-Double Precision Floating Point özelliği ayarla*

**Arabirim:**

```
void SetDoubleProperty( String property_name, Double value) ;
```

Ad tarafından tanımlanan çift duyarlıklı kayan nokta özelliğinin değerini ayarlayın.

**Parametreler:****özellik\_adi (giriş)**

Özelliğin adını sarmalayan bir String nesnesi.

**değer (giriş)**

Özelliğin değeri.

**Döndürülen:**

Hükümsüz

**İş parçacığı bağlamı:**

Alt sınıf tarafından belirlenir

**Özel durumlar:**

- XMSEException
- MessageNotWritableException

## *SetFloatÖzelliđi-Deđiřken Nokta özelliđini ayarla*

### **Arabirim:**

```
void SetFloatProperty( String property_name, Single value) ;
```

Ad tarafından tanımlanan kayar noktalı özellik deđerini ayarlayın.

### **Parametreler:**

#### **özelliđ\_adi (giriř)**

Özelliđin adını sarmalayan bir String nesnesi.

#### **deđer (giriř)**

Özelliđin deđerini.

### **Döndürülen:**

Hükümsüz

### **İř parçacığı bađlamı:**

Alt sınıf tarafından belirlenir

### **Özel durumlar:**

- XMSEException
- MessageNotWritableException

## *SetIntÖzelliđi-Tamsayı Özelliđi Ayarla*

### **Arabirim:**

```
void SetIntProperty( String property_name, Int32 value) ;
```

Ad ile tanımlanan tamsayı özelliđinin deđerini ayarlayın.

### **Parametreler:**

#### **özelliđ\_adi (giriř)**

Özelliđin adını sarmalayan bir String nesnesi.

#### **deđer (giriř)**

Özelliđin deđerini.

### **Döndürülen:**

Hükümsüz

### **İř parçacığı bađlamı:**

Alt sınıf tarafından belirlenir

### **Özel durumlar:**

- XMSEException
- MessageNotWritableException

## *SetLongÖzellik-Uzun Tamsayı Özelliđi Ayarla*

### **Arabirim:**

```
void SetLongProperty( String property_name, Int64 value) ;
```

Ad tarafından tanımlanan uzun tamsayı özelliđinin deđerini ayarlayın.

### **Parametreler:**

#### **özelliđ\_adi (giriř)**

Özelliđin adını sarmalayan bir String nesnesi.



**değer (giriş)**

Özelliğın deęeri.

**Döndürülen:**

Hükümsüz

**İş parçacığı baęlamı:**

Alt sınıf tarafından belirlenir

**Özel durumlar:**

- XMSException
- MessageNotWritableException

*SetObjectÖzellięi-Nesne özellięini ayarla*

**Arabirim:**

```
void SetObjectProperty( String property_name, Object value) ;
```

Ad tarafından tanımlanan bir özelliğın deęerini ve veri tipini ayarlayın.

**Parametreler:****özellię\_adi (giriş)**

Özelliğın adını sarmalayan bir String nesnesi.

**objectType (giriş)**

Aşğıdaki nesne tiplerinden biri olması gereken özelliğın deęeri:

Boolean  
Byte  
Byte[]  
Char  
Double  
Single  
Int32  
Int64  
Int16  
String

**değer (giriş)**

Özelliğın bayt dizisi olarak deęeri.

**uzunluk (giriş)**

Dizideki bayt sayısı.

**Döndürülen:**

Hükümsüz

**İş parçacığı baęlamı:**

Alt sınıf tarafından belirlenir

**Özel durumlar:**

- XMSException
- MessageNotWritableException

*SetShortÖzellię-Kısa Tamsayı Özellięi Ayarla*

**Arabirim:**

```
void SetShortProperty( String property_name, Int16 value) ;
```

Ad tarafından tanımlanan kısa tamsayı özelliğının deęerini ayarlayın.

**Parametreler:****özellik\_adi (giriş)**

Özelliğin adını sarmalayan bir String nesnesi.

**değer (giriş)**

Özelliğin değeri.

**Döndürülen:**

Hükümsüz

**İş parçacığı bağlamı:**

Alt sınıf tarafından belirlenir

**Özel durumlar:**

- XMSEException
- MessageNotWritableException

*SetStringÖzelliği-Dizgi Özelliğini Ayarla*

**Arabirim:**

```
void SetStringProperty( String property_name, String value);
```

Ad ile tanımlanan dizgi özelliğinin değerini ayarlayın.

**Parametreler:****özellik\_adi (giriş)**

Özelliğin adını sarmalayan bir String nesnesi.

**değer (giriş)**

Özelliğin değeri olan dizeyi sarmalayan bir String nesnesi.

**Döndürülen:**

Hükümsüz

**İş parçacığı bağlamı:**

Alt sınıf tarafından belirlenir

**Özel durumlar:**

- XMSEException
- MessageNotWritableException

**IQueueBrowser**

Bir uygulama, kuyruğu kaldırmadan kuyruklardaki iletilere göz atmak için bir kuyruk tarayıcısı kullanır.

**Devralma sıradüzeni:**

```
IBM.XMS.IPropertyContext
System.Collections.IEnumerable
|
+---- IBM.XMS.IQueueBrowser
```

**.NET özellikler**

*MessageSelector -İleti Seçiciyi Al*

**Arabirim:**

```
String MessageSelector
{
```

```
    get;  
}
```

Kuyruk tarayıcısı için ileti seçiciyi alın.

İleti seçici, ileti seçici ifadesini sarmalayan bir String nesnesidir. Veri dönüştürme gerekiyorsa, bu değer dönüştürmeden sonraki ileti seçici ifadesidir. Kuyruk tarayıcısının bir ileti seçicisi yoksa, yöntem boş bir String nesnesi döndürür.

#### Özel durumlar:

- XMSEException

*Kuyruk-Kuyruk Alma*

#### Arabirim:

```
IDestination Queue  
{  
    get;  
}
```

Kuyruk tarayıcısıyla ilişkili kuyruğu, kuyruğu gösteren hedef nesne olarak alın.

#### Özel durumlar:

- XMSEException

#### Yöntemler

*Kapat-Kuyruk Tarayıcısını Kapat*

#### Arabirim:

```
void Close();
```

Kuyruk tarayıcısını kapatın.

Bir uygulama önceden kapatılmış bir kuyruk tarayıcısını kapatmaya çalışırsa, arama yoksayılır.

#### Parametreler:

Yok

#### Döndürülen:

Hükümsüz

#### Özel durumlar:

- XMSEException

*GetEnumerator -İletileri Al*

#### Arabirim:

```
IEnumerator GetEnumerator();
```

Kuyruklardaki iletilerin bir listesini alın.

Yöntem, İleti nesnelерinin listesini sarmalayan bir sıralı değer listeleyci döndürür. İleti nesnelерinin sırası, iletilerin kuyruktan alınacağı sıralama düzeniyle aynıdır. Daha sonra, uygulama sırayla her iletiye göz atmak için sıralı değer numaracığı kullanabilir.

İletiler kuyruğa yerleştirildiği ve kuyruktan kaldırıldığı için, sıralı değer listeleyci dinamik olarak güncellenir. Uygulamanın kuyruktan sonraki iletiye göz atmak için `IEnumerator.MoveNext()` çağrısının her çağrısında, ileti kuyruğun yürürlükteki içeriğini yansıtır.

Bir uygulama bu yöntemi bir kuyruk tarayıcısı için bir kereden fazla çağırırsa, her çağrı yeni bir sıralı değer listeleme döndürür. Bu nedenle, uygulama bir kuyruktaki iletilere göz atmak ve kuyruğun içinde birden çok konum korumak için birden çok sıralı değer listeleme kullanabilir.

**Parametreler:**

Yok

**Döndürülen:**

Yineleyici nesnesi.

**Özel durumlar:**

- XMSEException

**Edinilmiş özellikler ve yöntemler**

Aşağıdaki yöntemler [IPropertyContext](#) arabiriminden edinilir:

[GetBooleanProperty](#), [GetByteProperty](#), [GetBytesProperty](#), [GetCharProperty](#), [GetDoubleProperty](#), [GetFloatProperty](#), [GetIntProperty](#), [GetLongProperty](#), [GetObjectProperty](#), [GetShortProperty](#), [GetStringProperty](#), [SetBooleanProperty](#), [SetByteProperty](#), [SetBytesProperty](#), [SetCharProperty](#), [SetDoubleProperty](#), [SetFloatProperty](#), [SetIntProperty](#), [SetLongProperty](#), [SetObjectProperty](#), [SetShortProperty](#), [SetStringProperty](#)

**İstekte Bulunan**

Bir uygulama, istek iletisini göndermek için istekte bulunan bir programı kullanır ve ardından yanıtı bekler ve alır.

**Devralma sıradüzeni:**

Yok

**Oluşturucular**

*İstekte Bulunanın-İstekte Bulunanın Yaratılması*

**Arabirim:**

```
Requestor(ISession sess, IDestination dest);
```

İstekte bulunanın oluşturulması.

**Parametreler:****takıntı (giriş)**

Bir Oturum nesnesi. Oturum hareket edilmemelidir ve aşağıdaki alındı bildirim kiplerinden birine sahip olmalıdır:

`AcknowledgeMode.AutoAcknowledge`

`AcknowledgeMode.DupsOkAcknowledge`

**en dest (giriş)**

Uygulamanın istek iletileri gönderebileceği hedefi gösteren hedef nesne.

**İş parçacığı bağlamı:**

İstekçiyle ilişkili oturum

**Özel durumlar:**

- XMSEException

**Yöntemler**

## Yakın-Kapanış İstek Sahibi

### Arabirim:

```
void Close();
```

İstekte bulunanın kapatılması.

Bir uygulama önceden kapatılmış bir istekte bulunmayı kapatmaya çalışırsa, arama yoksayılır.

**Not:** Bir uygulama bir istek isteğini kapattığında, ilişkili oturum da kapanmaz. Bu bakımdan XMS , JMS ' ye kıyasla farklı davranır.

### Parametreler:

Yok

### Döndürülen:

Hükümsüz

### İş parçacığı bağlamı:

Herhangi

### Özel durumlar:

- XMSEException

## İstek-İstek Yanıtı

### Arabirim:

```
IMessage Request(IMessage requestMessage);
```

Bir istek iletisi gönderin ve istek iletisini alan uygulamadan bir yanıt almak ve almak için bekleyin.

Bu yöntem öbeklerine bir yanıt alınincaya kadar ya da oturum sona erinceye kadar bir çağrı (hangisi daha erken olursa olsun).

### Parametreler:

#### requestMessage (giriş)

İleti nesnesi, istek iletisini sarsalıyor.

### Döndürülen:

İleti nesnesine ilişkin bir işaretçi, yanıt iletisini sarsalıyor.

### İş parçacığı bağlamı:

İstekçiyle ilişkili oturum

### Özel durumlar:

- XMSEException

## ResourceAllocationÖzel Durumu

XMS throws this exception if XMS cannot allocate the resources required by a method.

### Devralma sıradüzeni:

```
IBM.XMS.XMSEException
|
+----IBM.XMS.XMSEException
|
+----IBM.XMS.ResourceAllocationException
```

## Edinilmiş özellikler ve yöntemler

Aşağıdaki yöntemler [XMSEException](#) arabiriminden edinilir:

[GetErrorCode](#), [GetLinkedException](#)

## SecurityException

XMS throws this exception if the user identifier and password provided to authenticate an application are rejected. XMS , bir yetki denetimi başarısızlıkla sonuçlanırsa ve bir yöntemin tamamlanmasını önlediğinde bu kural dışı durumu da atar.

### Devralma sıradüzeni:

```
IBM.XMS.XMSEException
|
+----IBM.XMS.XMSEException
|
+----IBM.XMS.SecurityException
```

## Edinilmiş özellikler ve yöntemler

Aşağıdaki yöntemler [XMSEException](#) arabiriminden edinilir:

[GetErrorCode](#), [GetLinkedException](#)

## ISession

Oturum, ileti göndermek ve almak için tek bir iş parçacıklı bağlamdır.

### Devralma sıradüzeni:

```
IBM.XMS.IPropertyContext
|
+----IBM.XMS.ISession
```

Bir Oturum nesnesinin XMS tanımlı özelliklerinin bir listesi için bkz. "[Oturum Özellikleri](#)" sayfa 2015.

## .NET özellikler

*AcknowledgeMode -Onay Kipini Al*

### Arabirim:

```
AcknowledgeMode AcknowledgeMode
{
    get;
}
```

Oturum için alındı bildirimini modunu alın.

Onay kipi, oturum oluşturulduğunda belirtilir.

Oturum hareket edilmediği sürece, alındı bildirimini kipi aşağıdaki değerlerden biridir:

```
AcknowledgeMode.AutoAcknowledge
AcknowledgeMode.ClientAcknowledge
AcknowledgeMode.DupsOkAcknowledge
```

Alındı bildirimini kipleri hakkında daha fazla bilgi için bkz. [İleti alındı bildirimini](#).

Hareket eden bir oturumda alındı bildirimini kipi yoktur. Oturum çevrilebiliyorsa, yöntem `AcknowledgeMode.SessionTransacted` değerini döndürür.

### Özel durumlar:

- XMSEException

## Hareket Eden-Hareket Edip Etmeyeceğini Belirleyin

### Arabirim:

```
Boolean Transacted
{
    get;
}
```

Oturumun hareket edip etmeyeceğini belirleyin.

Transacted (derleme sonucu):

- Doğru, eğer oturum hareket ettiyse.
- Oturum hareket edilmezse, False (Yanlış) değerini kullanın.

Bir aracıya gerçek zamanlı bağlantı için, yöntem her zaman False değerini döndürür.

### Özel durumlar:

- XMSEException

### Yöntemler

#### Oturumu Kapat-Oturumu Kapat

### Arabirim:

```
void Close();
```

Oturumu kapatın. Oturum hareketliyse, devam etmekte olan herhangi bir hareket geriye işlenir.

Bir uygulama önceden kapatılmış bir oturumu kapatmaya çalışırsa, arama yoksayılr.

### Parametreler:

Yok

### Döndürülen:

Hükümsüz

### İş parçacığı bağlamı:

Herhangi

### Özel durumlar:

- XMSEException

#### Kesinleştir-Kesinleştir

### Arabirim:

```
void Commit();
```

Geçerli harekette işlenen tüm iletileri kesinleştir.

Oturum, hareket eden bir oturum olmalıdır.

### Parametreler:

Yok

### Döndürülen:

Hükümsüz

### Özel durumlar:

- XMSEException
- IllegalStateKural Dışı Durumu

- TransactionRolledBackException

*CreateBrowser -Kuyruk Tarayıcısı Yarat*

**Arabirim:**

```
IQueueBrowser CreateBrowser(IDestination queue) ;
```

Belirlenen kuyruk için bir kuyruk tarayıcısı yaratın.

**Parametreler:**

**kuyruk (giriş)**

Kuyruğu gösteren bir hedef nesne.

**Döndürülen:**

QueueBrowser nesnesi.

**Özel durumlar:**

- XMSException
- InvalidDestinationKural Dışı Durumu

*CreateBrowser -Kuyruk Tarayıcısı Yarat (ileti seçici ile)*

**Arabirim:**

```
IQueueBrowser CreateBrowser(IDestination queue, String selector) ;
```

İleti seçiciyi kullanarak, belirlenen kuyruk için bir kuyruk tarayıcısı yaratın.

**Parametreler:**

**kuyruk (giriş)**

Kuyruğu gösteren bir hedef nesne.

**seçici (giriş)**

Bir ileti seçici ifadesini sarmalayan bir String nesnesi. Kuyruk tarayıcıya, yalnızca, ileti seçici ifadesiyle eşleşen özelliklere sahip iletiler teslim edilir.

Boş Dizi nesnesi, kuyruk tarayıcısı için herhangi bir ileti seçiciye sahip olmadığı anlamına gelir.

**Döndürülen:**

QueueBrowser nesnesi.

**Özel durumlar:**

- XMSException
- InvalidDestinationKural Dışı Durumu
- InvalidSelectorKural Dışı Durumu

*CreateBytesMessage-Bytes İletisi Oluştur*

**Arabirim:**

```
IBytesMessage CreateBytesMessage();
```

Bir bayt iletisi oluşturun.

**Parametreler:**

Yok

**Döndürülen:**

BytesMessage nesnesi.



### Özel durumlar:

- XMSException
- IllegalStateKural Dışı Durumu (Oturum kapatılmış)

*CreateConsumer -Tüketici yarat*

### Arabirim:

```
IMessageConsumer CreateConsumer(IDestination dest) ;
```

Belirtilen hedef için bir ileti tüketicisi oluşturun.

### Parametreler:

#### en dest (giriş)

Hedef nesne.

### Döndürülen:

MessageConsumer nesnesi.

### Özel durumlar:

- XMSException
- InvalidDestinationKural Dışı Durumu

*CreateConsumer -Tüketici yarat (ileti seçici ile)*

### Arabirim:

```
IMessageConsumer CreateConsumer(IDestination dest,  
String selector) ;
```

İleti seçiciyi kullanarak, belirtilen hedef için ileti tüketicisi yaratın.

### Parametreler:

#### en dest (giriş)

Hedef nesne.

#### seçici (giriş)

Bir ileti seçici ifadesini sarmalayan bir String nesnesi. İleti tüketicisi, yalnızca, ileti seçici ifadesiyle eşleşen özelliklere sahip iletiler iletilmeye teslim edilir.

Boş Dizgi nesnesi, ileti tüketicisi için ileti seçicisinin olmadığı anlamına gelir.

### Döndürülen:

MessageConsumer nesnesi.

### Özel durumlar:

- XMSException
- InvalidDestinationKural Dışı Durumu
- InvalidSelectorKural Dışı Durumu

*CreateConsumer -Tüketici yarat (ileti seçici ve yerel ileti işareti ile)*

### Arabirim:

```
IMessageConsumer CreateConsumer(IDestination dest,  
String selector,  
Boolean noLocal) ;
```

İleti seçiciyi kullanarak belirtilen hedef için bir ileti tüketicisi yaratın ve hedef, ileti tüketicisinin kendi bağlantısı tarafından yayınlanan iletileri alıp almadığını belirten bir konudur.

**Parametreler:****en dest (giriş)**

Hedef nesne.

**seçici (giriş)**

Bir ileti seçici ifadesini sarmalayan bir String nesnesi. İleti tüketicisi, yalnızca, ileti seçici ifadesiyle eşleşen özelliklere sahip iletiler iletiye teslim edilir.

Boş Dizgi nesnesi, ileti tüketicisi için ileti seçicisinin olmadığı anlamına gelir.

**noLocal (giriş)**

True değeri, ileti tüketicisinin kendi bağlantısı tarafından yayınlanan iletileri almadığı anlamına gelir. False değeri, ileti tüketicisi tarafından yayınlanan iletileri kendi bağlantısı tarafından alacak anlamına gelir. Varsayılan değer False değeridir.

**Döndürülen:**

MessageConsumer nesnesi.

**Özel durumlar:**

- XMSEException
- InvalidDestinationKural Dışı Durumu
- InvalidSelectorKural Dışı Durumu

*CreateDurableAbonesi-Durable Abonesi Oluştur*

**Arabirim:**

```
IMessageConsumer CreateDurableSubscriber(IDestination dest,  
                                         String subscription) ;
```

Belirtilen konu için dayanıklı bir abone yaratın.

Bu yöntem, bir aracıya gerçek zamanlı bağlantı için geçerli değildir.

Kalıcı aboneler hakkında daha fazla bilgi için bkz. [Durable aboneleri](#).

**Parametreler:****en dest (giriş)**

Konuyu simgeleyen bir hedef nesne. Konu geçici bir konu olmamalıdır.

**abonelik (giriş)**

Bir dizgi nesnesi, kalıcı aboneliği tanımlayan bir adı sarkıyor. Adın, bağlantı için istemci tanıtıcısı içinde benzersiz olması gerekir.

**Döndürülen:**

Dayanıklı aboneyi gösteren MessageConsumer nesnesi.

**Özel durumlar:**

- XMSEException
- InvalidDestinationKural Dışı Durumu

*CreateDurableAbonesi-Durable Abonesi Oluştur (ileti seçici ve yerel ileti işareti ile)*

**Arabirim:**

```
IMessageConsumer CreateDurableSubscriber(IDestination dest,  
                                         String subscription,  
                                         String selector,  
                                         Boolean noLocal) ;
```

İleti seçiciyi kullanarak ve kalıcı abonenin kendi bağlantısı tarafından yayınlanan iletileri alıp almadığını belirterek, belirtilen konu için dayanıklı bir abone yaratın.

Bu yöntem, bir aracıya gerçek zamanlı bağlantı için geçerli değildir.

Kalıcı aboneler hakkında daha fazla bilgi için bkz. [Durable aboneleri](#).

#### **Parametreler:**

##### **en dest (giriş)**

Konuyu simgeleyen bir hedef nesne. Konu geçici bir konu olmamalıdır.

##### **abonelik (giriş)**

Bir dizgi nesnesi, kalıcı aboneliği tanımlayan bir adı sarkıyor. Adın, bağlantı için istemci tanıtıcısı içinde benzersiz olması gerekir.

##### **seçici (giriş)**

Bir ileti seçici ifadesini sarmalayan bir String nesnesi. Yalnızca, ileti seçici ifadesiyle eşleşen özelliklere sahip iletiler dayanıklı aboneye teslim edilir.

Boş Dizgi nesnesi, dayanıklı abone için ileti seçiciye sahip olmadığı anlamına gelir.

##### **noLocal (giriş)**

True değeri, kalıcı abonenin kendi bağlantısı tarafından yayınlanan iletileri almadığı anlamına gelir. False değeri, kalıcı abonenin kendi bağlantısı tarafından yayınlanan iletileri alacağı anlamına gelir. Varsayılan değer False değeridir.

#### **Döndürülen:**

Dayanıklı aboneyi gösteren MessageConsumer nesnesi.

#### **Özel durumlar:**

- XMSException
- InvalidDestinationKural Dışı Durumu
- InvalidSelectorKural Dışı Durumu

*CreateMapIletisi-Eşlem Iletisi Yarat*

#### **Arabirim:**

```
IMapMessage CreateMapMessage();
```

Bir eşlem iletisi yaratın.

#### **Parametreler:**

Yok

#### **Döndürülen:**

MapMessage nesnesi.

#### **Özel durumlar:**

- XMSException
- IllegalStateKural Dışı Durumu (Oturum kapatılmış)

*CreateMessage -İleti yarat*

#### **Arabirim:**

```
IMessage CreateMessage();
```

Vücuduna sahip olmayan bir ileti oluşturun.

#### **Parametreler:**

Yok

#### **Döndürülen:**

İleti nesnesi.

**Özel durumlar:**

- XMSEException
- IllegalStateKural Dışı Durumu (Oturum kapatılmış)

*CreateObjectİletisi-Nesne İletisi Oluştur*

**Arabirim:**

```
IObjectMessage CreateObjectMessage();
```

Bir nesne iletisi oluşturun.

**Parametreler:**

Yok

**Döndürülen:**

ObjectMessage nesnesi.

**Özel durumlar:**

- XMSEException
- IllegalStateKural Dışı Durumu (Oturum kapatılmış)

*CreateProducer -Üretici yarat*

**Arabirim:**

```
IMessageProducer CreateProducer(IDestination dest) ;
```

İletileri belirtilen hedefe göndermek için bir ileti üreticisi yaratın.

**Parametreler:****en dest (giriş)**

Hedef nesne.

Boş değerli bir hedef nesne belirtirseniz, ileti üreticisi hedef nesne olmadan yaratılır. Bu durumda, bir ileti göndermek için, uygulamanın ileti üreticisini her kullanımında bir hedef belirtmesi gerekir.

**Döndürülen:**

MessageProducer nesnesi.

**Özel durumlar:**

- XMSEException
- InvalidDestinationKural Dışı Durumu

*CreateQueue -Kuyruk Yarat*

**Arabirim:**

```
IDestination CreateQueue(String queue) ;
```

İleti alışverişi sunucusunda bir kuyruğu temsil etmek için bir hedef nesne yaratın.

Bu yöntem, ileti sistemi sunucusunda kuyruğu yaratmaz. Bir uygulamanın bu yöntemi çağırabilmesi için önce kuyruğu oluşturmanız gerekir.

**Parametreler:****kuyruk (giriş)**

Bir String nesnesi, kuyruğun adını sarsalıyor ya da kuyruğu tanımlayan bir URI (uniform resource identifier; bir tip kaynak tanıtıcısını (URI) sarsalıyor).

**Döndürülen:**

Kuyruğu gösteren hedef nesne.

**Özel durumlar:**

- XMSEException

*CreateStreamIletisi-Akış İletisi Oluştur*

**Arabirim:**

```
IStreamMessage CreateStreamMessage();
```

Bir akış iletisi oluşturun.

**Parametreler:**

Yok

**Döndürülen:**

StreamMessage nesnesi.

**Özel durumlar:**

- XMSEException
- XMS\_ILLEGAL\_STATE\_EXCEPTION

*CreateTemporaryKuyruğu-Geçici Kuyruk Yarat*

**Arabirim:**

```
IDestination CreateTemporaryQueue() ;
```

Geçici bir kuyruk yaratın.

Geçici kuyruğun kapsamı bağlantıdır. Yalnızca, bağlantı tarafından yaratılan oturumlar geçici kuyruğu kullanabilir.

Geçici kuyruk açık bir şekilde silininceye ya da bağlantı sona erinceye kadar kalır; hangisi daha erken olursa olsun.

Geçici kuyruklara ilişkin ek bilgi için [Geçici hedefler](#) konusuna bakın.

**Parametreler:**

Yok

**Döndürülen:**

Geçici kuyruğu gösteren Hedef nesnesi.

**Özel durumlar:**

- XMSEException

*CreateTemporaryKonusu-Geçici Konu Yarat*

**Arabirim:**

```
IDestination CreateTemporaryTopic() ;
```

Geçici bir konu oluşturun.

Geçici konunun kapsamı, bağlantıdır. Yalnızca, bağlantı tarafından yaratılan oturumlar geçici konuyu kullanabilir.

Geçici konu, açık bir şekilde silininceye ya da bağlantı sona erinceye kadar devam eder.

Geçici konular hakkında daha fazla bilgi için bkz. [Geçici hedefler](#).

**Parametreler:**

Yok

**Döndürülen:**

Geçici konuyu simgeleyen Hedef nesnesi.

**Özel durumlar:**

- XMSEException

*CreateTextİletisi-Metin İletisi Oluştur*

**Arabirim:**

```
ITextMessage CreateTextMessage();
```

Boş gövdesi içeren bir metin iletisi oluşturun.

**Parametreler:**

Yok

**Döndürülen:**

TextMessage nesnesi.

**Özel durumlar:**

- XMSEException

*CreateTextİletisi-Metin İletisi Oluştur (kullanıma hazırlandı)*

**Arabirim:**

```
ITextMessage CreateTextMessage(String initialValue);
```

Cesedi belirtilen metinle ilk kullanıma hazırlanmış bir metin iletisi oluşturun.

**Parametreler:****initialValue (giriş)**

Metin iletisinin gövesini kullanıma hazırlamak için metni sarmalayan bir String nesnesi.

Yok

**Döndürülen:**

TextMessage nesnesi.

**Özel durumlar:**

- XMSEException

*CreateTopic -Konu Oluştur*

**Arabirim:**

```
IDestination CreateTopic(String topic) ;
```

Bir konuyu temsil etmek için bir hedef nesne yaratın.

**Parametreler:****konu (giriş)**

Bir dizgi nesnesi, konunun adını sarsalıyor ya da konuyu tanımlayan bir URI (uniform resource identifier; bir tip kaynak tanıtıcısını (URI) sarsalıyor.

**Döndürülen:**

Konuyu simgeleyen Hedef nesnesi.

**Özel durumlar:**

- XMSEException

*Kurtar-Yeniden Kurtar*

**Arabirim:**

```
void Recover();
```

Oturumu kurtarın. İleti teslimi durdurulur ve tanınmayan en eski iletiyle yeniden başlatılır.

Oturum, hareket eden bir oturum olmamalıdır.

Bir oturumu kurtarma hakkında daha fazla bilgi için bkz. [İleti onayı](#).

**Parametreler:**

Yok

**Döndürülen:**

Hükümsüz

**Özel durumlar:**

- XMSEException
- IllegalStateKural Dışı Durumu

*Geridönüş-Geri Alma*

**Arabirim:**

```
void Rollback();
```

Geçerli harekette işlenen tüm iletileri geri alma.

Oturum, hareket eden bir oturum olmalıdır.

**Parametreler:**

Yok

**Döndürülen:**

Hükümsüz

**Özel durumlar:**

- XMSEException
- IllegalStateKural Dışı Durumu

*Aboneliği Kaldır-Aboneliği Kaldır*

**Arabirim:**

```
void Unsubscribe(String subscription);
```

Dayanıklı bir aboneliği silin. İleti alışverişi sunucusu, sürdürmekte olduğu dayanıklı aboneliğin kaydını siler ve kalıcı aboneye başka ileti göndermez.

Bir uygulama, aşağıdaki koşullardan herhangi birinde kalıcı bir aboneliği silemez:

- Kalıcı abonelik için etkin bir ileti tüketicisi varken
- Tüketilen bir ileti, bekleyen bir hareketin bir parçası olmakla birlikte
- Tüketilen bir ileti onaylanmazken

Bu yöntem, bir aracıya gerçek zamanlı bağlantı için geçerli değildir.

**Parametreler:****abonelik (giriş)**

Kalıcı aboneliği tanımlayan adı sarmalayan bir String nesnesi.

**Döndürülen:**

Hükümsüz

**Özel durumlar:**

- XMSException
- InvalidDestinationKural Dışı Durumu
- IllegalStateKural Dışı Durumu

**Edinilmiş özellikler ve yöntemler**

Aşağıdaki yöntemler [IPropertyContext](#) arabiriminden edinilir:

[GetBooleanProperty](#), [GetByteProperty](#), [GetBytesProperty](#), [GetCharProperty](#), [GetDoubleProperty](#), [GetFloatProperty](#), [GetIntProperty](#), [GetLongProperty](#), [GetObjectProperty](#), [GetShortProperty](#), [GetStringProperty](#), [SetBooleanProperty](#), [SetByteProperty](#), [SetBytesProperty](#), [SetCharProperty](#), [SetDoubleProperty](#), [SetFloatProperty](#), [SetIntProperty](#), [SetLongProperty](#), [SetObjectProperty](#), [SetShortProperty](#), [SetStringProperty](#)

**IStreamMessage**

Akış iletisi, her değerin ilişkili bir veri tipine sahip olduğu bir değerler akışından oluşan bir iletidir. Gövdenin içeriği sırayla yazılır ve okunur.

**Devralma sıradüzeni:**

```
IBM.XMS.IPropertyContext
|
+----IBM.XMS.IMessage
|
+----IBM.XMS.IStreamMessage
```

Bir uygulama iletisi akışından bir değer okuduğunda, değer XMS tarafından başka bir veri tipine dönüştürülebilir. Bu örtük dönüştürme biçimiyle ilgili daha fazla bilgi için bkz. [XMS iletisinin gövdesi](#).

**Yöntemler**

*ReadBoolean* - Boolean değerini okuyun

**Arabirim:**

```
Boolean ReadBoolean();
```

İleti akışından bir Boole değeri okuyun.

**Parametreler:**

Yok

**Döndürülen:**

Okunan boole değeri.

**Özel durumlar:**

- XMSException
- MessageNotReadableException
- MessageEOFException



## *ReadByte -Byte Okuma Deęeri*

### **Arabirim:**

```
Int16 ReadSignedByte();  
Byte ReadByte();
```

İleti akışından imzalı 8 bitlik bir tamsayı okuyun.

### **Parametreler:**

Yok

### **Döndürülen:**

Okunan bayt.

### **Özel durumlar:**

- XMSEException
- MessageNotReadableException
- MessageEOFException

## *ReadBytes -Okuma Baytları*

### **Arabirim:**

```
Int32 ReadBytes(Byte[] array);
```

İleti akışından bir bayt dizisi okuyun.

### **Parametreler:**

#### **dizi (giriş)**

Okunan bayt dizisine ve byte cinsinden arabelleğin uzunluęuna sahip arabellek.

Dizideki bayt sayısı arabelleğin uzunluęundan az ya da bu uzunluęa eşitse, tüm dizi arabelleęe okunuyor. Dizideki bayt sayısı arabelleğin uzunluęundan büyükse, arabellek dizinin bir bölüşüyle doldurulur ve bir iç imleç sonraki baytın okunabilmesinin konumunu gösterir. A subsequent call to readBytes() reads bytes from the array starting from the current position of the cursor.

Girişte boş deęerli bir gösterge belirtirseniz, arama, okumadan byte dizilerinin üzerine atlar.

### **Döndürülen:**

Arabelleęe okunan bayt sayısı. Arabellek kısmen doldurulsa, deęer, okunmak üzere kalan dizide başka byte kalmadığını gösteren arabellek uzunluęundan az olur. Aramadan önce diziden okunmak üzere kalan herhangi bir byte yoksa, deęer XMSE\_END\_OF\_BYTEARRAY olur.

Giriş sırasında boş deęerli bir gösterge belirtirseniz, yöntem deęer döndürmez.

### **Özel durumlar:**

- XMSEException
- MessageNotReadableException
- MessageEOFException

## *ReadChar -Okuma karakteri*

### **Arabirim:**

```
Char ReadChar();
```

İleti akışından 2 byte 'lık bir karakter okuyun.

### **Parametreler:**

Yok

**Döndürülen:**

Okunan karakter.

**Özel durumlar:**

- XMSEException
- MessageNotReadableException
- MessageEOFException

*ReadDouble -Çift Duyarlıklı Yüzer Nokta Numarasını Oku*

**Arabirim:**

```
Double ReadDouble();
```

İleti akışından, 8 baytlık çift duyarlıklı kayan noktalı sayı okuyun.

**Parametreler:**

Yok

**Döndürülen:**

Okunan çift duyarlıklı kayan noktalı sayı.

**Özel durumlar:**

- XMSEException
- MessageNotReadableException
- MessageEOFException

*ReadFloat -Kayan Nokta Numarasını okuyun*

**Arabirim:**

```
Single ReadFloat();
```

İleti akışından 4 baytlık bir kayan noktalı sayı okuyun.

**Parametreler:**

Yok

**Döndürülen:**

Okunan kayan noktalı sayı.

**Özel durumlar:**

- XMSEException
- MessageNotReadableException
- MessageEOFException

*ReadInt -Tamsayı Oku*

**Arabirim:**

```
Int32 ReadInt();
```

İleti akışından, imzalı 32 bitlik bir tamsayı okuyun.

**Parametreler:**

Yok

**Döndürülen:**

Okunan tamsayı.

**Özel durumlar:**

- XMSEException
- MessageNotReadableException
- MessageEOFException

*ReadLong -Uzun Tamsayı Oku*

**Arabirim:**

```
Int64 ReadLong();
```

İleti akışından imzalı 64 bitlik bir tamsayı okuyun.

**Parametreler:**

Yok

**Döndürülen:**

Okunan uzun tamsayı.

**Özel durumlar:**

- XMSEException
- MessageNotReadableException
- MessageEOFException

*ReadObject -Nesneyi Oku*

**Arabirim:**

```
Object ReadObject();
```

İleti akışından bir değer okuyun ve veri tipini döndürün.

**Parametreler:**

Yok

**Döndürülen:**

Aşağıdaki nesne tiplerinden biri olan değer:

Boolean  
Byte  
Byte[]  
Char  
Double  
Single  
Int32  
Int64  
Int16  
String

**Özel durumlar:**

XMSEException

*ReadShort -Kısa Tamsayı Oku*

**Arabirim:**

```
Int16 ReadShort();
```

İleti akışından imzalı 16 bitlik bir tamsayı okuyun.

**Parametreler:**

Yok

**Döndürülen:**

Okunan kısa tamsayı.

**Özel durumlar:**

- XMSEException
- MessageNotReadableException
- MessageEOFException

*ReadString -Dize Oku*

**Arabirim:**

```
String ReadString();
```

İleti akışından bir dizgi okuyun. Gerekliyse, XMS dizgideki karakterleri yerel kod sayfasına dönüştürür.

**Parametreler:**

Yok

**Döndürülen:**

Okunan dizeyi sarmalayan bir String nesnesi. Veri dönüştürme gerekiyorsa, bu dönüştürme işleminden sonra dizgidir.

**Özel durumlar:**

- XMSEException
- MessageNotReadableException
- MessageEOFException

*Sıfırla-İlk Duruma Getir*

**Arabirim:**

```
void Reset();
```

İletinin gövdeini salt okunur kipine getirin ve imleci ileti akışının başlangıcındaki yerini yeniden konumlayın.

**Parametreler:**

Yok

**Döndürülen:**

Hükümsüz

**Özel durumlar:**

- XMSEException
- MessageNotReadableException
- MessageEOFException

*WriteBoolean -Yazma Boole Değeri*

**Arabirim:**

```
void WriteBoolean(Boolean value);
```

İleti akışına bir boole değeri yazın.

**Parametreler:****değer (giriş)**

Yazılacak boole değeri.

**Döndürülen:**

Hükümsüz

**Özel durumlar:**

- XMSEException
- MessageNotWritableException

*WriteByte -Yazma Byte 'ı*

**Arabirim:**

```
void WriteByte(Byte value);  
void WriteSignedByte(Int16 value);
```

İleti akışa bir bayt yazın.

**Parametreler:****değer (giriş)**

Yazılacak bayt.

**Döndürülen:**

Hükümsüz

**Özel durumlar:**

- XMSEException
- MessageNotWritableException

*WriteBytes -Yazma Baytları*

**Arabirim:**

```
void WriteBytes(Byte[] value);
```

İleti akışa bir bayt dizisi yazın.

**Parametreler:****değer (giriş)**

Yazılacak bayt dizisi.

**uzunluk (giriş)**

Dizideki bayt sayısı.

**Döndürülen:**

Hükümsüz

**Özel durumlar:**

- XMSEException
- MessageNotWritableException

*WriteChar -Yazma Karakteri*

**Arabirim:**

```
void WriteChar(Char value);
```

İleti akışına 2 byte, önce yüksek sıralı bayt olarak bir karakter yazın.

**Parametreler:****değer (giriş)**

Yazılacak karakter.

**Döndürülen:**

Hükümsüz

**Özel durumlar:**

- XMSEException
- MessageNotWritableException

*WriteDouble -Çift Duyarlıklı Kayan Nokta Numarası Yaz*

**Arabirim:**

```
void WriteDouble(Double value);
```

Çift duyarlıklı kayan noktalı sayıyı uzun bir tamsayıya dönüştür ve uzun tamsayıyı ileti akışına 8 bayt, yüksek sıralı bayt olarak yazın.

**Parametreler:****değer (giriş)**

Yazılacak çift duyarlıklı kayan noktalı sayı.

**Döndürülen:**

Hükümsüz

**Özel durumlar:**

- XMSEException
- MessageNotWritableException

*WriteFloat -Kayan Noktalı Sayı Yazma*

**Arabirim:**

```
void WriteFloat(Single value);
```

Bir kayan noktalı sayıyı bir tamsayıya dönüştürüp tamsayıyı 4 bayt, yüksek sıralı bayt olarak ileti akışına yazın.

**Parametreler:****değer (giriş)**

Yazılacak kayan noktalı sayı.

**Döndürülen:**

Hükümsüz

**Özel durumlar:**

- XMSEException
- MessageNotWritableException

*WriteInt -Yazma Tamsayı*

**Arabirim:**

```
void WriteInt(Int32 value);
```

İleti akışına ilk olarak 4 bayt, yüksek sıralı bayt olarak bir tamsayı yazın.

**Parametreler:****değer (giriş)**

Yazılacak tamsayı.

**Döndürülen:**

Hükümsüz

**Özel durumlar:**

- XMSEException
- MessageNotWritableException

*WriteLong -Yazma Uzun Tamsayı*

**Arabirim:**

```
void WriteLong(Int64 value);
```

İleti akışına önce 8 bayt, yüksek sıralı bayt olarak uzun bir tamsayı yazın.

**Parametreler:****değer (giriş)**

Yazılacak uzun tamsayı.

**Döndürülen:**

Hükümsüz

**Özel durumlar:**

- XMSEException
- MessageNotWritableException

*WriteObject -Yazma Nesnesi*

**Arabirim:**

```
void WriteObject(Object value);
```

Belirtilen veri tipi ile ileti akışına bir değer yazın.

**Parametreler:****objectType (giriş)**

Aşağıdaki nesne tiplerinden biri olması gereken değer:

Boolean  
Byte  
Byte[]  
Char  
Double  
Single  
Int32  
Int64  
Int16  
String

**değer (giriş)**

Yazılacak değeri içeren bir bayt dizisi.

**uzunluk (giriş)**

Dizideki bayt sayısı.

**Döndürülen:**

Hükümsüz

**Özel durumlar:**

- XMSEException

*WriteShort -Yazma Kısa Tamsayı*

**Arabirim:**

```
void WriteShort(Int16 value);
```

İleti akışına 2 byte, önce yüksek sıralı bayt olarak kısa bir tamsayı yazın.

**Parametreler:****değer (giriş)**

Yazılacak kısa tamsayı.

**Döndürülen:**

Hükümsüz

**Özel durumlar:**

- XMSEException
- MessageNotWritableException

*WriteString -Yazma Dizgisi*

**Arabirim:**

```
void WriteString(String value);
```

İleti akışına bir dizgi yazın.

**Parametreler:****değer (giriş)**

Yazılacak dizeyi sarsalayan bir String nesnesi.

**Döndürülen:**

Hükümsüz

**Özel durumlar:**

- XMSEException
- MessageNotWritableException

***Edinilmiş özellikler ve yöntemler***

Aşağıdaki özellikler [IMessage](#) arabiriminden edinilir:

[JMSCorrelationID](#), [JMSDeliveryMode](#), [JMSDestination](#), [JMSExpiration](#), [JMSMessageID](#), [JMSPriority](#), [JMSReteslim](#), [JMSReplyTo](#), [JMSTimestamp](#), [JMSType](#), [Özellikler](#)

Aşağıdaki yöntemler [IMessage](#) arabiriminden edinilir:

[clearBody](#), [clearProperties](#), [PropertyExists](#)

Aşağıdaki yöntemler [IPROPERTYContext](#) arabiriminden edinilir:

[GetBooleanProperty](#), [GetByteProperty](#), [GetBytesProperty](#), [GetCharProperty](#), [GetDoubleProperty](#), [GetFloatProperty](#), [GetIntProperty](#), [GetLongProperty](#), [GetObjectProperty](#), [GetShortProperty](#), [GetStringProperty](#), [SetBooleanProperty](#), [SetByteProperty](#), [SetBytesProperty](#), [SetCharProperty](#), [SetDoubleProperty](#), [SetFloatProperty](#), [SetIntProperty](#), [SetLongProperty](#), [SetObjectProperty](#), [SetShortProperty](#), [SetStringProperty](#)



## ITextMessage

Metin iletisi, gövdesi bir dizeyi oluşturan bir iletidir.

### Devralma sıradüzeni:

```
IBM.XMS.IPropertyContext
|
+----IBM.XMS.IMessage
|
+----IBM.XMS.ITextMessage
```

## .NET özellikler

*Metin-Metni Al ve Ayarla*

### Arabirim:

```
String Text
{
    get;
    set;
}
```

Metin iletisinin gövdeini oluşturan dizgiyi alın ve ayarlayın.

Gerekliyse, XMS dizgiadaki karakterleri yerel kod sayfasına dönüştürür.

### Özel durumlar:

- XMSException
- MessageNotReadableException
- MessageNotWritableException
- MessageEOFException

## Edinilmiş özellikler ve yöntemler

Aşağıdaki özellikler [IMessage](#) arabiriminden edinilir:

[JMSCorrelationID](#), [JMSDeliveryMode](#), [JMSDestination](#), [JMSExpiration](#), [JMSMessageID](#), [JMSPriority](#), [JMSRetslim](#), [JMSReplyTo](#), [JMSTimestamp](#), [JMSType](#), [Özellikler](#)

Aşağıdaki yöntemler [IMessage](#) arabiriminden edinilir:

[clearBody](#), [clearProperties](#), [PropertyExists](#)

Aşağıdaki yöntemler [IPropertyContext](#) arabiriminden edinilir:

[GetBooleanProperty](#), [GetByteProperty](#), [GetBytesProperty](#), [GetCharProperty](#), [GetDoubleProperty](#), [GetFloatProperty](#), [GetIntProperty](#), [GetLongProperty](#), [GetObjectProperty](#), [GetShortProperty](#), [GetStringProperty](#), [SetBooleanProperty](#), [SetByteProperty](#), [SetBytesProperty](#), [SetCharProperty](#), [SetDoubleProperty](#), [SetFloatProperty](#), [SetIntProperty](#), [SetLongProperty](#), [SetObjectProperty](#), [SetShortProperty](#), [SetStringProperty](#)

## TransactionInProgressException

XMS , bir işlem devam etmekte olduğu için geçerli olmayan bir işlem isteğinde bulunduysa, bu kural dışı durumu döndürür.

### Devralma sıradüzeni:

```
IBM.XMS.XMSException
|
+----IBM.XMS.XMSException
```

```
|
+----IBM.XMS.TransactionInProgressException
```

### **Edinilmiş özellikler ve yöntemler**

Aşağıdaki yöntemler `XMSEException` arabiriminden edinilir:

[GetErrorCode](#), [GetLinkedException](#)

### **TransactionRolledBackException**

XMS , bir uygulama yürürlükteki hareketi kesinleştirmek için `Session.commit()` uygulaması çağırıyorsa, ancak işlem geriye işlenirse bu kural dışı durumu atar.

**Devralma sıradüzeni:**

```
IBM.XMS.XMSEException
|
+----IBM.XMS.XMSEException
|
+----IBM.XMS.TransactionRolledBackException
```

### **Edinilmiş özellikler ve yöntemler**

Aşağıdaki yöntemler `XMSEException` arabiriminden edinilir:

[GetErrorCode](#), [GetLinkedException](#)

### **XMSEException**

XMS bir .NET yöntemi çağırmasını işlerken bir hata saptarsa, XMS kural dışı durum yayınlar. Kural dışı durum, hatayla ilgili bilgileri sarmalayan bir nesnedir.

**Devralma sıradüzeni:**

```
System.Exception
|
+----IBM.XMS.XMSEException
```

Farklı tiplerde XMS kural dışı durumu vardır ve bir `XMSEException` nesnesi yalnızca bir kural dışı durum tipidir. Ancak, `XMSEException` sınıfı, diğer XMS kural dışı durum sınıflarının üst sınıfıdır. XMS , diğer kural dışı durum tiplerinden hiçbirinin uygun olmadığı durumlarda bir `XMSEException` nesnesi atar.

### **.NET özellikler**

*ErrorCode -Hata Kodu Al*

**Arabirim:**

```
public String ErrorCode
{
    get {return errorCode_;}
}
```

Hata kodunu alın.

**Özel durumlar:**

- `XMSEException`

*LinkedException -Bağlı Kural Dışı Durumu Al*

**Arabirim:**

```
public Exception LinkedException
{
    get { return linkedException_;}
    set { linkedException_ = value;}
}
```

Kural dışı durumlar zincirinde bir sonraki kural dışı durumu alın.

Zincirde başka kural dışı durumlar yoksa, yöntem boş değer döndürür.

**Özel durumlar:**

- XMSEException

## **XMSFactoryFactory**

Bir uygulama yönetilen nesnelere kullanılmıyorsa, bağlantı üreticileri, kuyruklar ve konular yaratmak için bu sınıfı kullanın.

**Devralma sıradüzeni:**

Yok

### **.NET özellikler**

*Meta Veri-Meta verileri al*

**Arabirim:**

```
IConnectionMetaData MetaData
```

XMSFactoryFactory nesnesinin bağlantı tipine uygun meta verileri alın.

**Özel durumlar:**

Yok

### **Yöntemler**

*CreateConnectionÜreticisi-Bağlantı Üreticisi Yarat*

**Arabirim:**

```
IConnectionFactory CreateConnectionFactory();
```

Bildirilmiş tip için bir ConnectionFactory nesnesi yaratın.

**Parametreler:**

Yok

**Döndürülen:**

ConnectionFactory nesnesi.

**Özel durumlar:**

- XMSEException

## CreateQueue -Kuyruk Yarat

### Arabirim:

```
IDestination CreateQueue(String name);
```

İleti alışverişi sunucusunda bir kuyruğu temsil etmek için bir hedef nesne yaratın.

Bu yöntem, ileti sistemi sunucusunda kuyruğu yaratmaz. Bir uygulamanın bu yöntemi çağırabilmesi için önce kuyruğu oluşturmanız gerekir.

### Parametreler:

#### ad (giriş)

Bir String nesnesi, kuyruğun adını sarsalıyor ya da kuyruğu tanımlayan bir URI (uniform resource identifier; bir tip kaynak tanıtıcısını (URI) sarsalıyor).

### Döndürülen:

Kuyruğu gösteren hedef nesne.

### Özel durumlar:

- XMSEException

## CreateTopic -Konu Oluştur

### Arabirim:

```
IDestination CreateTopic(String name);
```

Bir konuyu temsil etmek için bir hedef nesne yaratın.

### Parametreler:

#### ad (giriş)

Bir dizgi nesnesi, konunun adını sarsalıyor ya da konuyu tanımlayan bir URI (uniform resource identifier; bir tip kaynak tanıtıcısını (URI) sarsalıyor).

### Döndürülen:

Konuyu simgeleyen Hedef nesnesi.

### Özel durumlar:

- XMSEException

## GetInstance (GetInstance)- XMSFactoryFactoryyönetim ortamını alın

### Arabirim:

```
static XMSFactoryFactory GetInstance(int connectionType);
```

Bir XMSFactoryFactoryyönetim ortamı yaratın. XMS uygulaması, gerekli protokol tipine uygun bir ConnectionFactory nesnesine başvuru almak için bir XMSFactoryFactory nesnesi kullanır. Bu ConnectionFactory nesnesi, yalnızca bu protokol tipi için bağlantı üretebilir.

### Parametreler:

#### connectionType (giriş)

ConnectionFactory nesnesinin bağlantı ürettiği bağlantı tipi:

- XMSC.CT\_WPM
- XMSC.CT\_RTT
- XMSC.CT\_WMQ

### Döndürülen:

Bildirilmiş bağlantı tipine adanmış XMSFactoryFactory nesnesi.

## Özel durumlar:

- NotSupported()

## XMS nesnelere özellikleri

Bu kısım, XMS ile tanımlanan nesne özelliklerini içerir.

Bu kısım, aşağıdaki nesne tipleriyle ilgili bilgileri içerir:

- [“Bağlantı Özellikleri” sayfa 2001](#)
- [“ConnectionFactory özellikleri” sayfa 2002](#)
- [“ConnectionMetaVerileri Özellikleri” sayfa 2007](#)
- [“Hedef Özellikleri” sayfa 2008](#)
- [“InitialContext özellikleri” sayfa 2009](#)
- [“İleti Özellikleri” sayfa 2010](#)
- [“MessageConsumer özellikleri” sayfa 2014](#)
- [“MessageProducer özellikleri” sayfa 2015](#)
- [“Oturum Özellikleri” sayfa 2015](#)

Her nesne tipinin tanımı, belirtilen tipteki bir nesnenin özelliklerini listeler ve her özelliğin kısa bir açıklamasını sağlar.

Bu bölüm ayrıca her özelliğin bir tanımını da sağlar (bkz. [“Özellik Tanımları” sayfa 2015](#)).

Bir uygulama bu bölümde açıklanan nesnelere kendi özelliklerini tanımlıyorsa, bu, hataya neden olmaz, ancak beklenmedik sonuçlara neden olabilir.

**Not:** Bu bölümdeki özellik adları ve değerleri, C ve C++ için kullanılan form `XMSC.NAME` biçiminde gösterilir. Ancak, .NET’te, özellik adının biçimi, nasıl kullanmanıza bağlı olarak `XMSC.NAME` ya da `XMSC_NAME` olabilir:

- Bir özellik belirtiyorsanız, özellik adı aşağıdaki örnekte gösterildiği gibi `XMSC.NAME` biçiminde olmalıdır:

```
cf.SetStringProperty(XMSC.WMQ_CHANNEL, "DOTNET.SVRCONN");
```

- Bir dizgi belirtiyorsanız, özellik adı aşağıdaki örnekte gösterildiği gibi `XMSC_NAME` biçiminde olmalıdır:

```
cf.SetStringProperty("XMSC_WMQ_CHANNEL", "DOTNET.SVRCONN");
```

.NET’te, özellik adları ve değerleri, XMSC sınıfında sabit değerler olarak sağlanır. Bu sabitler dizgileri tanımlar ve herhangi bir XMS.NET uygulaması tarafından kullanılır. Önceden tanımlanmış bu değişmezleri kullanıyorsanız, özellik adları ve değerleri `XMSC.NAME` biçimlerinde bulunur; örneğin, `XMSC.USERID`, `XMLC_USERID` yerine.

Veri tipleri aynı zamanda C/C++ için de kullanılır. .NET içinde ilgili değerleri [.NET için veri tipleri](#) içinde bulabilirsiniz.

## Bağlantı Özellikleri

Daha ayrıntılı başvuru bilgilerine ilişkin bağlantılarla Connection nesnesinin özelliklerine genel bakış.

Çizelge 873. Bağlantı Özellikleri	
Özelliğin adı	Tanım
<a href="#">“XMLC_WMQ_RESOLVED_QUEUE_YÖNETİCELERİ” sayfa 2048</a>	Bu özellik, bağlı olduğu kuyruk yöneticisinin adını elde etmek için kullanılır.
<a href="#">“XMLC_WMQ_RESOLVED_QUEUE_MANAGER_ID” sayfa 2049</a>	Bu özellik, bağlantı tamamlandıktan sonra kuyruk yöneticisinin tanıtıcısıyla doldurulur.

Çizelge 873. Bağlantı Özellikleri (devamı var)	
Özelliğin adı	Tanım
<u>XMSSC_WPM_CONNECTION_PROTOCOL</u>	İleti alışverişi altyapısına bağlantı için kullanılan iletişim protokolü. Bu özellik salt okunur.
<u>XMLC_WPM_HOST_NAME</u>	Uygulamanın bağlı olduğu ileti alışverişi altyapısını içeren sistemin anasistem adı ya da IP adresi. Bu özellik salt okunur.
<u>XMLC_WPM_ME_NAME</u>	Uygulamanın bağlı olduğu ileti alışverişi altyapısının adı. Bu özellik salt okunur.
<u>XMLC_WPM_PORT</u>	Uygulamanın bağlı olduğu ileti alışverişi altyapısı tarafından dinlenen kapının sayısı. Bu özellik salt okunur.

Connection nesnesi, bağlantıyı yaratmak için kullanılan bağlantı üreticisinin özelliklerinden türetilen salt okunur özelliklere de sahiptir. Bu özellikler yalnızca, bağlantı yaratıldığı sırada ayarlanan bağlantı üreticisi özelliklerinden değil, ayarlanmamış özelliklerin varsayılan değerlerinden türetilir. Özellikler arasında, yalnızca uygulamanın bağlı olduğu ileti alışverişi sunucusu tipi için ilgili olanlar da bulunur. Özelliklerin adları, bağlantı üreticisi özelliklerinin adlarıyla aynıdır.

## ConnectionFactoryözellikleri

Daha ayrıntılı başvuru bilgilerine bağlantılar içeren ConnectionFactory nesnesinin özelliklerine genel bakış.

Çizelge 874. ConnectionFactoryözellikleri	
Özelliğin adı	Tanım
<u>"XMSC_ASYNC_EXCAIMS" sayfa 2024</u>	Bu özellik, XMS 'nin bir ExceptionListener ' i yalnızca bir bağlantı kesildiğinde mi, yoksa bir XMS API çağrısına zamanuyumsuz olarak bir kural dışı durum oluştuğunda mı bilgilendirdiğini belirler. Bu özellik, ExceptionListener kaydı olan bu ConnectionFactory ' den oluşturulan tüm Connections için geçerlidir.
<b>V 9.2.4</b> <u>"XMSC_WMQ_BALANCING_APPLICATION_TYPE" sayfa 2032</u>	Dengeleme seçeneğinin tipi
<b>V 9.2.4</b> <u>"XMLC_WMQ_BALANCING_SEÇENEKLER" sayfa 2033</u>	Verme uygulaması tarafından ayarlanan dengeleme seçenekleri
<b>V 9.2.4</b> <u>"XMSC_WMQ_BALANCING_TIMEOUT" sayfa 2033</u>	Yeniden dengeleme işlemi uygulama etkinliğini kesintiye uğratabileceği için zamanaşımı oluştu.
<u>XMLC_CLIENT_ID</u>	Bir bağlantıya ilişkin istemci tanıtıcısı.
<u>XMLC_CONNECTION_TYPE</u>	Bir uygulamanın bağlandığı ileti sistemi sunucusunun tipi.
<u>XMLC_PAROLASI</u>	Bir ileti sistemi sunucusuna bağlanmaya çalışıldığında uygulamanın kimliğini doğrulamak için kullanılacak parola.
<u>"XMLC_RTT_BROKER_PING_INTERVAL" sayfa 2029</u>	XMS .NET ' in milisaniye cinsinden zaman aralığı, herhangi bir etkinliği saptamak için Gerçek Zamanlı ileti sistemi sunucusuna olan bağlantıyı denetler.

Çizelge 874. ConnectionFactoryözellikleri (devamı var)	
Özellğin adı	Tanım
<u>XMSC_RTT_CONNECTION_PROTOCOL</u>	Bir aracıya gerçek zamanlı bağlantı için kullanılan iletişim protokolü.
<u>XMLC_RTT_HOST_NAME</u>	Aracının çalıştığı sistemin anasistem adı ya da IP adresi.
<u>XMLC_RTT_LOCAL_ADRESİ</u>	Bir aracıya gerçek zamanlı bağlantı için kullanılacak yerel ağ arabiriminin anasistem adı ya da IP adresi.
<u>XMLC_RTT_MULTICAST</u>	Bir bağlantı üreticisi ya da hedefi için çoklu yayın ayarı.
<u>XMLC_RTT_PORT</u>	Bir aracının gelen istekleri dinlediği kapının numarası.
<u>XMLC_USERID</u>	Bir ileti sistemi sunucusuna bağlanmayı denediğinde uygulamanın kimliğini doğrulamak için kullanılacak kullanıcı kimliği.
<u>XMLC_WMQ_BROKER_CONTROLO</u>	Bir aracı tarafından kullanılan denetim kuyruğunun adı. <b>Not:</b> This property can be used with version 2.0 of IBM Message Service Client for .NET but has no effect for an application connected to an IBM WebSphere MQ 7.0 queue manager unless the XMSC_WMQ_PROVIDER_VERSION property of the connection factory is set to a version number less than 7.
<u>XMLC_WMQ_BROKER_PUBQ</u>	Uygulamaların yayınladıkları iletileri gönderdikleri bir aracı tarafından izlenen kuyruğun adı. <b>Not:</b> This property can be used with version 2.0 of IBM Message Service Client for .NET but has no effect for an application connected to an IBM WebSphere MQ 7.0 queue manager unless the XMSC_WMQ_PROVIDER_VERSION property of the connection factory is set to a version number less than 7.
<u>XMLC_WMQ_BROKER_QMGR</u>	Aracının bağlı olduğu kuyruk yöneticisinin adı. <b>Not:</b> This property can be used with version 2.0 of IBM Message Service Client for .NET but has no effect for an application connected to an IBM WebSphere MQ 7.0 queue manager unless the XMSC_WMQ_PROVIDER_VERSION property of the connection factory is set to a version number less than 7.
<u>XMLC_WMQ_BROKER_SUBQ</u>	Sürekli olmayan bir ileti tüketicisine ilişkin abone kuyruğunun adı. <b>Not:</b> This property can be used with version 2.0 of IBM Message Service Client for .NET but has no effect for an application connected to an IBM WebSphere MQ 7.0 queue manager unless the XMSC_WMQ_PROVIDER_VERSION property of the connection factory is set to a version number less than 7.

Çizelge 874. ConnectionFactoryözellikleri (devamı var)	
Özelliğın adı	Tanım
<a href="#">XMLC_WMQ_BROKER_VERSION</a>	Uygulama tarafından bir bağlantı ya da hedef için kullanılan aracının tipi. <b>Not:</b> This property can be used with version 2.0 of IBM Message Service Client for .NET but has no effect for an application connected to an IBM WebSphere MQ 7.0 queue manager unless the XMSC_WMQ_PROVIDER_VERSION property of the connection factory is set to a version number less than 7.
<a href="#">"XMLC_WMQ_CCDTURL" sayfa 2035</a>	İstemci kanal tanımlama çizelgesini içeren dosyanın adını ve yerini tanıtan ve dosyaya nasıl erişilebileceğini belirten URL.
<a href="#">XMLC_WMQ_CHANNEL</a>	Bağlantı için kullanılacak kanalın adı.
<a href="#">"XMLC_WMQ_CLIENT_RECONNECT_OPTIONS" sayfa 2036</a>	Bu özellik, bu üretici tarafından yaratılan yeni bağlantılara ilişkin istemci yeniden bağlanma seçeneklerini belirler.
<a href="#">"XMLC_WMQ_CLIENT_RECONNECT_TIMEOUT" sayfa 2037</a>	Bu özellik, bir istemci bağlantısının yeniden bağlanmayı denediği süreyi (saniye olarak) belirtir.
<a href="#">XMLC_WMQ_CONNECTION_MODE</a>	Bir uygulamanın kuyruk yöneticisine bağlanma kipi.
<a href="#">"XMLC_WMQ_CONNECTION_NAME_LIST" sayfa 2037</a>	Bu özellik, istemcinin bağlantısı kesildikten sonra istemcinin yeniden bağlanmayı denediği anasistemleri belirler.
<a href="#">XMSSC_WMQ_FAIL_IF_QUIESCE</a>	Uygulamanın bağlı olduğu kuyruk yöneticisi susturma durumundaysa, belirli yöntemlere yapılan çağrılarının başarısız olup olmadığını belirler.
<a href="#">XMLC_WMQ_HOST_NAME</a>	Kuyruk yöneticisinin çalıştığı sistemin anasistem adı ya da IP adresi.
<a href="#">XMLC_WMQ_LOCAL_ADRESİ</a>	Kuyruk yöneticisine yönelik bir bağlantı için bu özellik, kullanılacak yerel ağ arabirimini ya da kullanılacak yerel kapı ya da yerel kapı aralığını ya da her ikisini de belirtir.
<a href="#">XMSSC_WMQ_MESSAGE_SELECTION</a>	İleti seçiminin XMS istemcisi tarafından mı, yoksa aracı tarafından mı yapılacağını belirler. <b>Not:</b> This property can be used with version 2.0 of IBM Message Service Client for .NET but has no effect for an application connected to an IBM WebSphere MQ 7.0 queue manager unless the XMSC_WMQ_PROVIDER_VERSION property of the connection factory is set to a version number less than 7.
<a href="#">XMLC_WMQ_MSG_BATCH_SIZE</a>	Zamanuyumsuz ileti teslimi kullanılırken bir toplu işte kuyruktan alınacak ileti sayısı üst sınırı. <b>Not:</b> This property can be used with version 2.0 of IBM Message Service Client for .NET but has no effect for an application connected to an IBM WebSphere MQ 7.0 queue manager unless the XMSC_WMQ_PROVIDER_VERSION property of the connection factory is set to a version number less than 7.



Çizelge 874. ConnectionFactoryözellikleri (devamı var)	
Özellğin adı	Tanım
<u>XMLC_WMQ_POLLING_INTERVAL</u>	Bir oturum içindeki her ileti dinleyicinin kuyruğunda uygun bir ileti yoksa, bu değer, her ileti dinleyicisinin kuyruğundan ileti almaya yeniden çalışmadan önce geçmesi gereken milisaniye cinsinden aralık üst sınırıdır.  <b>Not:</b> This property can be used with version 2.0 of IBM Message Service Client for .NET but has no effect for an application connected to an IBM WebSphere MQ 7.0 queue manager unless the XMSC_WMQ_PROVIDER_VERSION property of the connection factory is set to a version number less than 7.
<u>"XMSC_WMQ_PROVIDER_VERSION" sayfa 2046</u>	Uygulamanın bağlanmayı planladığı kuyruk yöneticisinin sürümü, yayını, değişiklik düzeyi ve düzeltme paketi.
<u>XMLC_WMQ_PORT</u>	Kuyruk yöneticisinin gelen istekleri dinlediği kapının numarası.
<u>XMLC_WMQ_PUB_ACK_INTERVAL</u>	XMS istemcisi aracından bir alındı bildirim istemeden önce bir yayıncı tarafından yayınlanan iletilerin sayısı.  <b>Not:</b> This property can be used with version 2.0 of IBM Message Service Client for .NET but has no effect for an application connected to an IBM WebSphere MQ 7.0 queue manager unless the XMSC_WMQ_PROVIDER_VERSION property of the connection factory is set to a version number less than 7.
<u>"XMSC_WMQ_PUT_ASYNC_ALLOWES" sayfa 2041</u>	Bu özellik, ileti üreticilerinin bu hedefe ileti göndermek için zamanuyumsuz girişler kullanmalarına izin verilip verilmeyeceğini belirler.
<u>XMLC_WMQ_QMGR_CCSD</u>	İleti Kuyruğu Arabirimi 'nde (MQI) tanımlanan karakter verileri alanlarının XMS istemcisi ile IBM MQ istemcisi arasında değiş tokuş edildiği, kodlanmış karakter takımının ya da kod sayfasının tanıtıcısı (CCSID).
<u>XMLC_WMQ_QUEUE_YÖNETİCİSİ</u>	Bağlanılacak kuyruk yöneticisinin adı.
<u>XMLC_WMQ_RECEIVE_EXIT</u>	çalıştırılacak kanal alma çıkışı tanımlar.
<u>XMLC_WMQ_RECEIVE_EXIT_INIT</u>	Bir kanala geçirilen kullanıcı verileri, çağrıldığında çıkış alır.
<u>XMLC_WMQ_SECURITY_EXIT</u>	Bir kanal güvenlik çıkışı tanımlar.
<u>XMLC_WMQ_SECURITY_EXIT_INIT</u>	Bir kanal güvenlik çıkışına çağrıldığında geçirilen kullanıcı verileri.
<u>"XMLC_WMQ_SEND_CHECK_COUNT" sayfa 2050</u>	Tek bir hareket etmeyen XMS oturumunda, zamanuyumsuz koyma hatalarının denetlenmesi arasında izin verilecek gönderme çağrılarının sayısı.
<u>XMLC_WMQ_SEND_EXIT</u>	Bir kanal gönderme çıkışı tanımlar.
<u>XMLC_WMQ_SEND_EXIT_INIT</u>	Çağrıldığında kanal gönderme çıkışlarına geçirilen kullanıcı verileri.

Çizelge 874. ConnectionFactoryözellikleri (devamı var)	
Özellğin adı	Tanım
<a href="#">“XMLC_WMQ_SHARE_CONV_ALLOWED” sayfa 2050</a>	Bir istemci bağlantısının, kanal tanımlamaları eşleşiyorsa, aynı işlemde aynı kuyruk yöneticisine yapılan diğer üst düzey XMS bağlantılarla yuvasını paylaşım paylaşılamayabileceğini belirler. Bu özellik, uygulama geliştirme, bakım ya da işletim nedenleriyle gerektiğinde Connections 'ın ayrı yuvalarda tam olarak yalıtılmasına olanak sağlamak için sağlanır.
<a href="#">XMLC_WMQ_SSL_CERT_STORES</a>	Kuyruk yöneticisine SSL bağlantısında kullanılacak sertifika iptal listelerini (CRL) bulunduran sunucuların yerleri.
<a href="#">XMLC_WMQ_SSL_CIPHER_SPEC</a>	Bir kuyruk yöneticisine güvenli bir bağlantıda kullanılacak CipherSpec ' in adı.
<a href="#">XMSC_WMQ_SSL_CIPHER_SUITE</a>	Bir kuyruk yöneticisine TLS bağlantısında kullanılacak CipherSuite ' in adı. Güvenli bağlantının kararlaştırılmasında kullanılan protokol, belirtilen CipherSuite' e bağlıdır.
<a href="#">XMLC_WMQ_SSL_CRYPT_HW</a>	İstemci sistemine bağlı şifreleme donanımına ilişkin yapılandırma ayrıntıları.
<a href="#">XMLC_WMQ_SSL_FIPS_REQUIRED</a>	Bu özelliğin değeri, bir uygulamanın FIPS uyumlu olmayan şifreleme takımlarını kullanıp kullanamayacağını belirler. Bu özellik true olarak ayarlanırsa, istemci-sunucu bağlantısı için yalnızca FIPS algoritmaları kullanılır.
<a href="#">XMLC_WMQ_SSL_KEY_REPOSITORY</a>	Anahtarların ve sertifikaların saklandığı anahtar veritabanı dosyasının konumu.
<a href="#">XMLC_WMQ_SSL_KEY_RESETCOUNT</a>	KeyResetSayısı, gizli anahtar yeniden anlaşılmadan önce SSL etkileşimi içinde gönderilen ve alınan toplam şifrelenmemiş bayt sayısını temsil eder.
<a href="#">XMLC_WMQ_SSL_PEER_NAME</a>	Kuyruk yöneticisine SSL bağlantısında kullanılacak eşdüzey ad.
<a href="#">XMSSC_WMQ_SYNCPOINT_ALL_GETS</a>	Eşitleme noktası denetimi içindeki kuyruklardan tüm iletilerin alınıp alınmayacağı.
<a href="#">“XMLC_WMQ_TARGET_İSTEMCENİ” sayfa 2057</a>	
<a href="#">XMLC_WMQ_TEMP_Q_PREFIX</a>	Uygulama bir XMS geçici kuyruğu oluşturduğunda oluşturulan IBM MQ dinamik kuyruğunun adını oluşturmak için kullanılan önek.
<a href="#">XMSC_WMQ_TEMP_TOPIC_PREFIX</a>	Geçici konular oluştururken XMS , "TEMP/TEMPTOPICPREFIX/unique_id" biçiminde bir konu dizgisi oluşturur ya da bu özellik varsayılan değeri içeriyorsa, bu dizgi ("TEMP/unique_id") oluşturulur. Boş olmayan bir değer belirtilmesi, abonelerin bu bağlantı altında yaratılan geçici konulara ilişkin yönetilen kuyruklarının yaratılması için belirli model kuyruklarının tanımlanmasına olanak sağlar.
<a href="#">XMLC_WMQ_TEMPORARY_MODEL</a>	Uygulama bir XMS geçici kuyruğu yarattığında dinamik kuyruğun yaratıldığı IBM MQ model kuyruğunun adı.

<i>Çizelge 874. ConnectionFactoryözellikleri (devamı var)</i>	
<b>Özelliğın adı</b>	<b>Tanım</b>
<u>XMLC_WPM_BUS_ADı</u>	Bir bağlantı üreticisi için, uygulamanın bağlandığı hizmet tümleştirme veriyolunun adı ya da hedef için, hedefin bulunduğı hizmet tümleştirme veriyolunun adı.
<u>XMSSC_WPM_CONNECTION_PROXIMITY</u>	Bağlantı için bağlantı yakınlığı ayarı.
<u>XMLC_WPM_DUR_SUB_HOME</u>	Bir bağlantıya ya da hedefe ilişkin tüm sürekli aboneliklerin yönetildiğı ileti alışverişi altyapısının adı.
<u>XMLC_WPM_LOCAL_ADRESİ</u>	Bir hizmet bütünleştirme veriyolu bağlantısı için bu özellik, kullanılacak yerel ağ arabirimini ya da kullanılacak yerel kapı ya da yerel kapı aralığını ya da her ikisini de belirtir.
<u>XMLC_WPM_NON_PERSISTENT_MAP</u>	Bağlantı kullanılarak gönderilen kalıcı olmayan iletilerin güvenilirlik düzeyi.
<u>XMLC_WPM_PERSISTENT_MAP</u>	Bağlantı kullanılarak gönderilen kalıcı iletilerin güvenilirlik düzeyi.
<u>XMSC_WPM_PROVIDER_ENDERS</u>	Önyükleme sunucularının bir ya da daha çok uç noktası adresi sırası.
<u>XMLC_WPM_TARGET_GROUP</u>	İleti alışverişi altyapılarının hedef grubunun adı.
<u>XMSSC_WPM_TARGET_MAITLE</u>	İleti alışverişi altyapılarının hedef grubunun önemi.
<u>XMSC_WPM_TARGET_TRANSPORT_CHAIN</u>	Uygulamanın bir ileti alışverişi altyapısına bağlanmak için kullanması gereken gelen iletim zincirinin adı.
<u>XMLC_WPM_TARGET_TYPE</u>	İleti alışverişi altyapılarının hedef grubunun tipi.
<u>XMSC_WPM_TEMP_Q_PREFIX</u>	Uygulama bir XMS geçici kuyruğı yarattığında hizmet tümleştirme veriyolunda yaratılan geçici kuyruğun adını oluşturmak için kullanılan önek.
<u>XMSC_WPM_TEMP_TOPIC_PREFIX</u>	Uygulama tarafından oluşturulan geçici bir konunun adını oluşturmak için kullanılan önek.

## **ConnectionMetaVerileri Özellikleri**

Daha ayrıntılı başvuru bilgilerine bağlantılar içeren ConnectionMetaVeri nesnesinin özelliklerine genel bakış.

<i>Çizelge 875. ConnectionMetaVerileri Özellikleri</i>	
<b>Özelliğın adı</b>	<b>Tanım</b>
<u>XMLC_JMS_MAJOR_VERSION</u>	XMS belirtiminin temel aldığı JMS belirtiminin ana sürüm numarası. Bu özellik salt okunur.
<u>XMLC_JMS_MINOR_VERSION</u>	XMS belirtiminin temel aldığı JMS belirtiminin ikincil sürüm numarası. Bu özellik salt okunur.
<u>XMLC_JMS_VERSION</u>	XMS belirtiminin temel aldığı JMS belirtiminin sürüm tanıtıcısı. Bu özellik salt okunur.
<u>XMLC_MAJOR_VERSION</u>	XMS istemcisinin sürüm numarası. Bu özellik salt okunur.
<u>XMLC_MINOR_VERSION</u>	XMS istemcisinin yayın numarası. Bu özellik salt okunur.
<u>XMLC_PROVIDER_NAME</u>	XMS istemcisinin sağlayıcısı. Bu özellik salt okunur.

Çizelge 875. ConnectionMetaVerileri Özellikleri (devamı var)

Özelliğın adı	Tanım
<u>XMSC_SÜRÜM</u>	cliXMS' nin sürüm tanıtıcısı. Bu özellik salt okunur.

## Hedef Özellikleri

Daha ayrıntılı başvuru bilgilerine ilişkin bağlantılarla, Hedef nesnenin özelliklerine genel bakış.

Çizelge 876. Hedef Özellikleri

Özelliğın adı	Tanım
<u>XMLC_DELIVERY_MODE</u>	Hedefe gönderilen iletilerin teslim kipi.
<u>XMSC_PRIORITY</u>	Hedefe gönderilen iletilerin önceliğı.
<u>XMLC_RTT_MULTICAST</u>	Bir bağlantı üreticisi ya da hedefi için çoklu yayın ayarı.
<u>XMLC_TIME_TO_LIVE</u>	Hedefe gönderilen iletiler için yaşam süresi.
<u>XMLC_WMQ_BROKER_VERSION</u>	Uygulama tarafından bir bağlantı ya da hedef için kullanılan aracının tipi.
<u>XMLC_WMQ_CCSD</u>	XMS istemcisi iletiyi hedefe ilettiğinde, iletinin gövdesindeki karakter verileri dizgilerinin içinde olduğı kodlanmış karakter takımının ya da kod sayfasının tanıtıcısı (CCSID).
<u>XMLC_WMQ_DUR_SUBQ</u>	Hedeften ileti alan sürekli bir aboneye ilişkin abone kuyruğunun adı. <b>Not:</b> This property can be used with version 2.0 of IBM Message Service Client for .NET but has no effect for an application connected to an IBM WebSphere MQ 7.0 queue manager unless the XMSC_WMQ_PROVIDER_VERSION property of the connection factory is set to a version number less than 7.
<u>XMLC_WMQ_ENCODING</u>	XMS istemcisi iletiyi hedefe ilettiğinde, iletinin gövdesindeki sayısal verilerin nasıl gösterileceğı.
<u>XMSSC_WMQ_FAIL_IF_QUIESCE</u>	Uygulamanın bağı olduğı kuyruk yöneticisi susturma durumundaysa, belirli yöntemlere yapılan çağrıların başarısız olup olmadığını belirler.
<u>"XMLC_WMQ_MESSAGE_BÖLGESİ" sayfa 2039</u>	Bu özellik, bir XMS uygulamasının bir IBM MQ iletinin MQRFH2 ' sini ileti bilgi yükünün (ileti gövdesinin bir parçası olarak) bir parçası olarak işleyip işlemediğini belirler.
<u>"XMSC_WMQ_MQMD_MESSAGE_CONTEXT" sayfa 2040</u>	XMS uygulaması tarafından ayarlanacak ileti bağı düzeyini belirler. Bu özelliğın yürürlüğe girmesi için uygulamanın uygun bağı yetkisiyle çalışması gerekir.
<u>"XMLC_WMQ_MQMD_READENABLED" sayfa 2041</u>	Bu özellik, bir XMS uygulamasının MQMD alanlarının değerlerini alıp almayacağını belirler.
<u>"XMSSC_WMQ_MQMD_WRITE_ENABLED" sayfa 2041</u>	Bu özellik, bir XMS uygulamasının MQMD alanlarının değerlerini ayarlayıp ayarlayamayacağını belirler.

<i>Çizelge 876. Hedef Özellikleri (devamı var)</i>	
<b>Özellik adı</b>	<b>Tanım</b>
<a href="#">“XMSC_WMQ_READ_AHEAD_ALLOWD” sayfa 2042</a>	Bu özellik, ileti tüketicilerinin ve kuyruk tarayıcılarının, bu hedeften kalıcı olmayan, işlemsel olmayan iletileri almadan önce iç arabelleğe almak için önden okuma özelliğini kullanıp kullanamayacağını belirler.
<a href="#">“XMSC_WMQ_READ_AHEAD_CLOSE_POLICY” sayfa 2042</a>	Bu özellik, zamanuyumsuz bir ileti dinleyicisine teslim edilen iletiler için, ileti tüketicisi kapatıldığında iç önden okuma arabelleğindeki iletilere ne olacağını belirler.
<a href="#">“XMLC_WMQ_RECEIVE_CCSD” sayfa 2047</a>	Kuyruk yöneticisi ileti dönüştürmesi için hedef CCSID 'yi belirleyen hedef özellik. XMLC_WMQ_RECEIVE_CONVERSION WMQ_RECEIVE_CONVERSION_QMGR değerine ayarlanmıyorsa, değer yoksayılr.
<a href="#">“XMLC_WMQ_RECEIVE_CONVERSION” sayfa 2048</a>	Veri dönüştürme işleminin kuyruk yöneticisi tarafından gerçekleştirilip gerçekleştirilmeyeceğini belirleyen hedef özellik.
<a href="#">XMLC_WMQ_TARGET_İSTEMCI</a>	Hedefe gönderilen iletilerin bir MQRFH2 üstbilgisi içerip içermediğini belirler.
<a href="#">XMSC_WMQ_TEMP_TOPIC_PREFIX</a>	Geçici konular oluştururken XMS, "TEMP/TEMPTOPICPREFIX/unique_id" biçiminde bir konu dizgisi oluşturur ya da bu özellik varsayılan değeri içeriyorsa, bu dizgi ("TEMP/unique_id") oluşturulur. Boş olmayan bir değer belirtilmesi, abonelerin bu bağlantı altında yaratılan geçici konulara ilişkin yönetilen kuyruklarının yaratılması için belirli model kuyruklarının tanımlanmasına olanak sağlar.
<a href="#">XMLC_WPM_BUS_ADı</a>	Bir bağlantı üreticisi için, uygulamanın bağlandığı hizmet tümleştirme veriyolunun adı ya da hedef için, hedefin bulunduğu hizmet tümleştirme veriyolunun adı.
<a href="#">XMSC_WPM_TOPIC_SPACE</a>	Konuyu içeren konu alanının adı.

## **InitialContextözellikleri**

Daha ayrıntılı başvuru bilgilerine bağlantılar içeren, InitialContext nesnesinin özelliklerine genel bakış.

<i>Çizelge 877. InitialContextözellikleri</i>	
<b>Özellik adı</b>	<b>Tanım</b>
<a href="#">XMLC_IC_PROVIDER_URL</a>	COS adlandırma hizmetinin web hizmetiyle aynı sunucuda olması gerekmeyecek şekilde JNDI adlandırma dizinini bulmak için kullanılır.
<a href="#">XMSC_IC_SECURITY_AUTHENTICATION</a>	Java Bağlam arabirimi SECURITY_AUTHENTICATION ögesine dayalı olarak. Bu özellik yalnızca COS adlandırma bağlamı için geçerlidir.
<a href="#">XMSC_IC_SECURITY_CREDENTIAL</a>	Java Bağlam arabirimi SECURITY_CREDENTIALS. Bu özellik yalnızca COS adlandırma bağlamı için geçerlidir.
<a href="#">XMSC_IC_SECURITY_PRINCIPAL</a>	Java Bağlam arabirimi SECURITY_PRINCIPAL değerine dayalı olarak. Bu özellik yalnızca COS adlandırma bağlamı için geçerlidir.

<i>Çizelge 877. InitialContextözellikleri (devamı var)</i>	
<b>Özelliğın adı</b>	<b>Tanım</b>
<u>XMSC_IC_SECURITY_PROTOCOL</u>	Java Bağlam arabirimi SECURITY_PROTOCOL değerine dayalı Bu özellik yalnızca COS adlandırma bağlamı için geçerlidir.
<u>XMLC_IC_URL</u>	LDAP ve FileSystem bağlamları için, yönetilen nesnelere içeren havuzun adresi. COS adlandırma bağlamları için, dizindeki nesnelere tarayan web hizmetinin adresi.

## İleti Özellikleri

Daha ayrıntılı başvuru bilgilerine bağlantılar içeren, İleti nesnesinin özelliklerine genel bakış.

<i>Çizelge 878. İleti Özellikleri</i>	
<b>Özelliğın adı</b>	<b>Tanım</b>
<u>JMS_IBM_CHARACTER_SET</u>	XMS istemcisi iletiyi istenen hedefe ilettiğinde, iletinin gövdesindeki karakter verileri dizgilerinin içinde olduğu kodlanmış karakter takımının ya da kod sayfasının tanıtıcısı (CCSID). XMS içinde bu özellik sayısal bir değere sahiptir ve CCSID ile eşlenir. Ancak, bu özellik bir JMS özelliğine dayalıdır; dizgi tipi değeri vardır ve bu sayısal CCSID 'yi gösteren Java karakter takımıyla eşlenir.
<u>JMS_IBM_ENCODING</u>	XMS istemcisi iletiyi hedeflenen hedefine ilettiğinde, iletinin gövdesindeki sayısal verilerin nasıl temsil edileceği.
<u>JMS_IBM_EXCEPTIONMESSAGE</u>	İletinin kural dışı durum hedefine neden gönderildiğini açıklayan metin. Bu özellik salt okunur.
<u>JMS_IBM_EXCEPTIONPROBLEMINHEDEF</u>	İletinin kural dışı durum hedefine gönderilmeden önce olduğu hedefin adı.
<u>JMS_IBM_EXCEPTIONREASON</u>	İletinin kural dışı durum hedefine gönderilme nedenini gösteren neden kodu.
<u>JMS_IBM_EXCETTIONTIMESTAMP</u>	İletinin kural dışı durum hedefine gönderildiği zaman.
<u>JMS_IBM_FEEDBACK</u>	Rapor iletisinin niteliğini gösteren kod.
<u>JMS_IBM_FORMA</u>	İletideki uygulama verilerinin türü.
<u>JMS_IBM_LAST_MSG_IN_GROUP</u>	İletinin bir ileti grubundaki son ileti olup olmadığını belirtin.
<u>JMS_IBM_MSGTYPE</u>	İletinin tipi.
<u>JMS_IBM_PUTAPPLTYPE</u>	İletiyi gönderen uygulamanın tipi.
<u>JMS_IBM_PUTDATE</u>	İletinin gönderildiği tarih.
<u>JMS_IBM_PUTTIME</u>	İletinin gönderildiği saat.
<u>JMS_IBM_REPORT_COA</u>	Özgün iletideki uygulama verilerinin bir rapor iletisine ne kadar eklenmesi gerektiğini belirten 'varış sırasında onayla' rapor iletilerini isteyin.
<u>JMS_IBM_REPORT_COD</u>	Özgün iletideki uygulama verilerinin bir rapor iletisine ne kadar eklenmesi gerektiğini belirten 'teslimde onayla' rapor iletilerini isteyin.

Çizelge 878. İleti Özellikleri (devamı var)	
Özelliğin adı	Tanım
<u>JMS_IBM_REPORT_DISCARD_MSG</u>	İletinin hedeflenen hedefine teslim edilememesi durumunda atılmasını isteyin.
<u>JMS_IBM_REPORT_EXCETION</u>	Özgün iletideki uygulama verilerinin bir rapor iletisine ne kadar eklenmesi gerektiğini belirten kural dışı durum raporu iletileri isteyin.
<u>JMS_IBM_REPORT_EXPIRATION</u>	Özgün iletideki uygulama verilerinin bir rapor iletisine ne kadar dahil edilmesi gerektiğini belirten, süre bitim raporu iletileri isteyin.
<u>JMS_IBM_REPORT_NAN</u>	Olumsuz eylem bildirim raporu iletileri isteyin.
<u>JMS_IBM_REPORT_PAN</u>	Pozitif eylem bildirim raporu iletileri isteyin.
<u>JMS_IBM_REPORT_PASS_CORREL_ID</u>	Herhangi bir raporun ya da yanıt iletisinin ilinti tanıtıcısının, özgün iletinin ilinti tanıtıcısıyla aynı olması isteği.
<u>JMS_IBM_REPORT_PASS_MSG_ID</u>	Herhangi bir raporun ya da yanıt iletisinin ileti tanıtıcısının, özgün iletinin ileti tanıtıcısıyla aynı olduğunu isteyin.
<u>JMS_IBM_RETAIN</u>	Bu özelliğin ayarlanması, kuyruk yöneticisine bir iletinin Ulaşılan Yayın olarak kabul edileceğini gösterir.
<u>JMS_IBM_SYSTEM_MESSAGEID</u>	İletiyi hizmet bütünleştirme veriyolu içinde benzersiz olarak tanımlayan bir tanıtıcı. Bu özellik salt okunur.
<u>JMSX_APPID</u>	İletiyi gönderen uygulamanın adı.
<u>JMSX_DELIVERY_COUNT</u>	İletiyi teslim etme girişimi sayısı.
<u>JMSX_GROUPID</u>	İletinin ait olduğu ileti grubunun tanıtıcısı.
<u>JMSX_GROUPSEQ</u>	İleti grubundaki iletinin sıra numarası.
<u>JMSX_USERID</u>	İletiyi gönderen uygulamayla ilişkili kullanıcı kimliği.

### JMS\_IBM\_MQMD\* özellikleri

IBM Message Service Client for .NET , istemci uygulamalarının API ' leri kullanarak MQMD alanlarını okumalarını/yazmalarını sağlar. Ayrıca, MQ ileti verilerine erişim de sağlar. MQMD ' ye varsayılan olarak erişim geçersiz kılınır ve XMSC\_WMQ\_MQMD\_WRITE\_ENABLED ve XMLC\_WMQ\_MQMD\_READ\_ENAB\_READENABLED hedef özellikleri kullanılarak uygulama tarafından belirtik olarak etkinleştirilmelidir. Bu iki özellik birbirlerinden bağımsızdır.

StrucId ve Version dışındaki tüm MQMD alanları ek İleti nesnesi özellikleri olarak ortaya çıkar ve JMS\_IBM\_MQMD önekli olur.

JMS\_IBM\_MQMD\* özellikleri, önceki çizelgede açıklanan JMS\_IBM\* gibi diğer özelliklerden daha öncelikli olarak uygulanır.

### İletilerin gönderilmesi

StrucId ve Version dışındaki tüm MQMD alanları temsil edilir. Bu özellikler yalnızca MQMD alanlarına gönderme yapar; hem MQMD ' de hem de MQRFH2 üstbilgisinde bir özellik oluşur, MQRFH2 içindeki sürüm belirlenmez ya da çıkarılmaz. JMS\_IBM\_MQMD\_BackoutCount dışında, bu özelliklerden herhangi biri ayarlanabilir. JMS\_IBM\_MQMD\_BackoutCount için ayarlanan herhangi bir değer yok sayılır.

Bir özelliğin uzunluk üst sınırı varsa ve siz çok uzun bir değer sağlıyorsa, değer kesilir.

Bazı özellikler için, Hedef nesnede XMSC\_WMQ\_MQMD\_MESSAGE\_CONTEXT özelliğini de ayarlamanız gerekir. Uygulamanın, bu özelliğin geçerli olması için uygun bağlam yetkisi ile çalışıyor olması gerekir. XMSSC\_WMQ\_MQMD\_MESSAGE\_CONTEXT değerini uygun bir değere ayarlamadıysanız, özellik değeri yoksayılır. XMSSC\_WMQ\_MQMD\_MESSAGE\_CONTEXT değerini uygun bir değere ayarladığınızda, ancak kuyruk yöneticisi için yeterli bağlam yetkiniz yoksa, kural dışı durum yayınlanır. XMSC\_WMQ\_MQMD\_MESSAGE\_CONTEXT için belirli değerleri gerektiren özellikler aşağıdaki gibidir.

Aşağıdaki özellikler XMSSC\_WMQ\_MQMD\_MESSAGE\_CONTEXT için XMLC\_WMQ\_MDCTX\_SET\_IDENTITY\_CONTEXT ya da XMLC\_WMQ\_MDCTX\_SET\_ALL\_CONTEXT olarak ayarlanmasını gerektirir:

- JMS\_IBM\_MQMD\_UserIdentifier
- JMS\_IBM\_MQMD\_AccountingToken
- JMS\_IBM\_MQMD\_ApplIdentityVerileri

Aşağıdaki özellikler XMSSC\_WMQ\_MQMD\_MESSAGE\_CONTEXT için XMSC\_WMQ\_MDCTX\_SET\_ALL\_CONTEXT olarak ayarlanmasını gerektirir:

- JMS\_IBM\_MQMD\_PutApplTipi
- JMS\_IBM\_MQMD\_PutApplAdı
- JMS\_IBM\_MQMD\_PutDate
- JMS\_IBM\_MQMD\_PutTime
- JMS\_IBM\_MQMD\_ApplOriginVerileri

## İletileri alma

XMLC\_WMQ\_MQMD\_READ\_READ\_ENABLED özelliği true olarak ayarlandıysa, üreten uygulama kümesinin ürettiği gerçek özelliklerden bağımsız olarak tüm bu özellikler alınmış bir iletiyle kullanılabilir. Bir uygulama, ilk olarak JMS belirtimine göre tüm özellikler temizlenmedikçe, alınan iletinin özelliklerini değiştiremez. Alınan ileti, özellikler değiştirilmeden iletilebilir.

**Not:** Uygulamanız, XMLC\_WMQ\_MQMD\_READ\_READ\_READ\_READ\_ENABLED özelliği true olarak ayarlanmış bir hedeften bir ileti alırsa ve bunu XMSC\_WMQ\_MQMD\_WRITE\_ENABLED değerine ayarlanmış bir hedefe iletirse, alınan iletinin tüm MQMD alanı değerleri, iletilen iletiye kopyalanıyor. Özellikler tablosu

<i>Çizelge 879. MQMD alanlarını temsil eden ileti nesnesinin özellikleri</i>		
Özellik	Tanım	Tip
JMS_IBM_MQMD_REPORT	Rapor iletileri için seçenekler	System.Int32
JMS_IBM_MQMD_MSGTYPE	İleti tipi	System.Int32
JMS_IBM_MQMD_EXPIRY	ileti kullanım süresi	System.Int32
JMS_IBM_MQMD_FEEDBACK	Geribildirim ya da neden kodu	System.Int32
JMS_IBM_MQMD_ENCODING	İleti verilerinin sayısal kodlaması	System.Int32
JMS_IBM_MQMD_CODEDCHARSETID	İleti verilerinin karakter kümesi tanıtıcısı	System.Int32
JMS_IBM_MQMD_FORMA	İleti verilerinin adını biçimle	System.String
JMS_IBM_MQMD_PRIORITY	İleti önceliği	System.Int32
<b>Not:</b> JMS_IBM_MQMD_PRIORITY için 0-9 aralığında olmayan bir değer atarsanız, bu değer JMS belirtimini ihlal eder.		
JMS_IBM_MQMD_PERSISTENCE	İleti kalıcılığı	System.Int32



Çizelge 879. MQMD alanlarını temsil eden ileti nesnesinin özellikleri (devamı var)

Özellik	Tanım	Tip
JMS_IBM_MQMD_MSGID <b>Not:</b> JMS belirtimi, ileti tanıtıcısının JMS sağlayıcısı tarafından ayarlanması gerektiğini ve bunun benzersiz ya da boş değer olması gerektiğini belirtiyor. JMS_IBM_MQMD_MSGID değerine bir değer atarsanız, bu değer JMSMessageID'ye kopyalanır. Bu nedenle JMS sağlayıcısı tarafından ayarlanmamış ve benzersiz olmayabilir: bu değer JMS belirtimini ihlal eder.	İleti Tanıtıcısı	Bayt Dizisi <b>Not:</b> Bir iletideki bayt dizisi özelliklerinin kullanımı, JMS belirtimini ihlal eder.
JMS_IBM_MQMD_CORRELID <b>Not:</b> JMS_IBM_MQMD_CORRELID değerine, 'ID:' dizisiyle başlayan bir değer atarsanız, bu değer JMS belirtimini ihlal eder.	İlinti tanıtıcısı	Bayt Dizisi <b>Not:</b> Bir iletideki bayt dizisi özelliklerinin kullanımı, JMS belirtimini ihlal eder.
JMS_IBM_MQMD_BACKUTCOUNT	Geriletme sayacı	System.Int32
JMS_IBM_MQMD_REPLYTOQ	Yanıt kuyruğunun adı	System.String
JMS_IBM_MQMD_REPLYTOQMGR	Yanıt kuyruğu yöneticisinin adı	System.String
JMS_IBM_MQMD_USERIDENTIFIER	Kullanıcı kimliği	System.String
JMS_IBM_MQMD_ACCOUNTINGTOKEN	Hesap simgesi	Bayt Dizisi <b>Not:</b> Bir iletideki bayt dizisi özelliklerinin kullanımı, JMS belirtimini ihlal eder.
JMS_IBM_MQMD_APPLIDENTITDATA	Kimlikle ilgili uygulama verileri	System.String
JMS_IBM_MQMD_PUTAPPLTYPE	İletiyi koyan uygulamanın tipi	System.Int32
JMS_IBM_MQMD_PUTAPPLNAME	İletiyi koyan uygulamanın adı	System.String
JMS_IBM_MQMD_PUTDATE	İletinin konulduğu tarih	System.String
JMS_IBM_MQMD_PUTTIME	İletinin konulduğu saat	System.String
JMS_IBM_MQMD_APPLORIGINDATA	Köken ile ilgili uygulama verileri	System.String
JMS_IBM_MQMD_GROUPID	Grup tanıtıcısı	Bayt Dizisi <b>Not:</b> Bir iletideki bayt dizisi özelliklerinin kullanımı, JMS belirtimini ihlal eder.
JMS_IBM_MQMD_MSGSEQSAYI	Grup içindeki yerel iletinin sıra numarası	System.Int32
JMS_IBM_MQMD_OFFSET	Mantıksal iletinin başlangıcındaki fiziksel iletelerde verilerin görelî konumu	System.Int32
JMS_IBM_MQMD_MSGFLAGS	İleti İşaretleri	System.Int32
JMS_IBM_MQMD_ORIGINALLENGTH	Özgün iletinin uzunluğu	System.Int32

Ek ayrıntılar için [MQMD](#) başlıklı konuya bakın.

## Örnekler

Bu örnek, MQMD.UserIdentifier , "JoeBloggs" olarak ayarlanır.

```
// Create a ConnectionFactory, connection, session, producer, message
// ...

// Create a destination
// ...

// Enable MQMD write
dest.setBooleanProperty(XMSC_WMQ_MQMD_WRITE_ENABLED,
    XMSC_WMQ_MQMD_WRITE_ENABLED_YES);

// Optionally, set a message context if applicable for this MD field
dest.setIntProperty(XMSC_WMQ_MQMD_MESSAGE_CONTEXT,
    XMSC_WMQ_MDCTX_SET_IDENTITY_CONTEXT);

// On the message, set property to provide custom UserId
msg.setStringProperty(JMS_IBM_MQMD_USERIDENTIFIER, "JoeBloggs");

// Send the message
// ...
```

JMS\_IBM\_MQMD\_USERIDENTIFIER ayarlanmadan önce XMSC\_WMQ\_MQMD\_MESSAGE\_CONTEXT ayarını ayarlamanız gerekir. XMSC\_WMQ\_MQMD\_MESSAGE\_CONTEXT kullanımı hakkında daha fazla bilgi için, İleti nesnesi özelliklerine bakın.

Benzer şekilde, bir ileti almadan önce ve daha sonra getStringözelligi gibi iletinin alma yöntemlerini kullanarak MQMD alanlarının içeriğini XMSC\_WMQ\_MQMD\_READ\_ENABLED değerine true değerine ayarlayarak çıkartabilirsiniz. Alınan özellikler salt okunurdur.

Bu örnek, MQMD.ApplIdentityData , bir iletinin kuyruğundan ya da bir konudan alındı.

```
// Create a ConnectionFactory, connection, session, consumer
// ...

// Create a destination
// ...

// Enable MQMD read
dest.setBooleanProperty(XMSC_WMQ_MQMD_READ_ENABLED, XMSC_WMQ_MQMD_READ_ENABLED_YES);

// Receive a message
// ...

// Get required MQMD field value using a property
System.String value = rcvMsg.getStringProperty(JMS_IBM_MQMD_APPLIDENTITYDATA);
```

## MessageConsumerözellikleri

Daha ayrıntılı başvuru bilgilerine bağlantılar içeren MessageConsumer nesnesinin özelliklerine genel bakış.

Çizelge 880. MessageConsumerözellikleri	
Özellğin adı	Tanım
<a href="#">XMSC_IS_SUBSCRIPTION_MULTICAST</a>	İletilerin WebSphere MQ Multicast Transportkullanılarak ileti tüketicisine teslim edilip edilmediğini gösterir. Bu özellik salt okunur.
<a href="#">XMSC_IS_SUBSCRIPTION_RELIABLE_MULTICAST</a>	İletilerin güvenilir bir hizmet kalitesiyle WebSphere MQ Multicast Transport kullanılarak ileti tüketicisine teslim edilip edilmediğini gösterir. Bu özellik salt okunur.

Ayrıntılar için [IMessageConsumer](#)' in .NET özellikleri dosyasına bakın.

## MessageProducer özellikleri

Daha ayrıntılı başvuru bilgilerine bağlantılar içeren, MessageProducer nesnesinin özelliklerine genel bakış.

Bkz. [.Daha fazla ayrıntı için IMessageProducer](#) ' in NET özellikleri.

## Oturum Özellikleri

Daha ayrıntılı başvuru bilgilerine bağlantılar içeren, Oturum nesnesinin özelliklerine genel bakış.

Bkz. [.Daha fazla ayrıntı için NET UZUUse](#) .

## Özellik Tanımları

Bu kısım, her nesne özelliği için bir tanımlama sağlar.

Her özellik tanımlaması aşağıdaki bilgileri içerir:

- Özelliğin veri tipi
- Özelliğin sahip olduğu nesne tipleri
- Hedef özelliği için, tek tip bir kaynak tanıtıcısında kullanılacak ad (URI)
- Özelliğe ilişkin daha ayrıntılı bir açıklama
- Özelliğin geçerli değerleri
- Özelliğin varsayılan değeri

Adları aşağıdaki örneklerden biriyle başlayan özellikler yalnızca belirtilen bağlantı tipi için anlamlıdır:

### **XMLC\_RTT**

Özellikler yalnızca bir aracıya gerçek zamanlı bağlantı için anlamlıdır. Özelliklerin adları, `xmsc_rtt`. hüstbilgi dosyasında adlandırılmış sabitler olarak tanımlanır.

### **XMLC\_WMQ**

Özellikler yalnızca, bir uygulama bir IBM MQ kuyruk yöneticisine bağlandığında anlamlıdır. Özelliklerin adları, `xmsc_wmq`. hüstbilgi dosyasında adlandırılmış sabitler olarak tanımlanır.

### **XMLC\_WPM**

Özellikler yalnızca, bir uygulama bir WebSphere hizmet tümleştirme veriyoluna bağlandığında anlamlıdır. Özelliklerin adları, `xmsc_wpm`. hüstbilgi dosyasında adlandırılmış sabitler olarak tanımlanır.

Tanımlamalarında aksi belirtilmedikçe, geri kalan özellikler tüm bağlantı tipleri için anlamlıdır. Özelliklerin adları, `xmsc`. hüstbilgi dosyasında adlandırılmış sabitler olarak tanımlanır. Adları JMSX örneğiyle başlayan özellikler, bir iletinin JMS tanımlı özellikleridir ve adları JMS\_IBM örneğiyle başlayan özellikler, bir iletinin IBM tanımlı özellikleridir. İletilerin özellikleri hakkında daha fazla bilgi için bkz. [XMS iletisinin özellikleri](#).

Aksi belirtilmediği sürece, her özellik, hem noktadan noktaya iletişim, hem de yayınlama abone olma etki alanlarında ilişkilidir.

Bir uygulama, özellik salt okunur olarak atanmadıkça, herhangi bir özelliğin değerini alabilir ve ayarlayabilirler.

## **JMS\_IBM\_XX\_ENCODE\_CASE\_ONE charter\_set**

### **Veri tipi:**

System.Int32

### **Özellik:**

İleti

XMS istemcisi iletiyi istenen hedefe ilettiğinde, iletinin gövdesindeki karakter verileri dizgilerinin içinde olduğu kodlanmış karakter takımının ya da kod sayfasının tanıtıcısı (CCSID). XMS içinde bu özellik sayısal bir değere sahiptir ve CCSID ile eşlenir. Ancak, bu özellik bir JMS özelliğine dayalıdır; dizgi tipi değeri

vardır ve bu sayısal CCSID ' yi gösteren Java karakter takımıyla eşlenir. This property overrides any CCSID specified for the destination by the XMLC\_WMQ\_CCSID property.

Varsayılan olarak özellik ayarlanmaz.

Bir uygulama bir hizmet bütünleştirme veriyoluna bağlandığında bu özellik ilgili değildir.

## **JMS\_IBM\_ENCODING**

### **Veri tipi:**

System.Int32

### **Özellik:**

İleti

XMS istemcisi iletiyi hedeflenen hedefine ilettiğinde, iletinin gövdesindeki sayısal verilerin nasıl temsil edileceği. This property overrides any encoding specified for the destination by the XMLC\_WMQ\_ENCODING property. Bu özellik, ikili tamsayıların, paketlenmiş onlu tamsayıların ve kayan noktalı sayıların gösterilmesini belirtir.

Özelliğin geçerli değerleri, bir ileti tanımlayıcısının **Encoding** alanında belirtilebilecek değerlerle aynıdır.

Bir uygulama, özelliği ayarlamak için aşağıdaki adı belirtilen değişmezleri kullanabilir:

### **Adlandırılmış sabit**

### **Anlamı**

MQENC\_INTEGER\_NORMAL

Normal tamsayı kodlaması

MQENC\_INTEGER\_TERS

Tersine çevrilmiş tamsayı kodlaması

MQENC\_DECIMAL\_NORMAL

Olağan paketlenmiş onlu kodlama

MQENC\_DECIMAL\_TERDI

Tersine çevrilmiş paketlenmiş ondalık kodlama

MQENC\_FLOAT\_IEEE\_NORMAL

Olağan IEEE kayar noktalı kodlama

MQENC\_FLOAT\_IEEEE\_TERS

Ters IEEE kayar noktalı kodlama

MQENC\_FLOAT\_S390

z/OS mimarisi kayan nokta kodlaması

MQENC\_NATIVE

Yerli makine kodlaması

Özelliğe ilişkin bir değer oluşturmak için, uygulama bu değişmezlerden üçünü aşağıdaki gibi ekleyebilir:

- İkili tamsayıların gösterimini belirtmek için, adı MQENC\_XX\_ENCODE\_CASE\_ONE integer ile başlayan bir değişmez.
- Paketlenmiş onlu tamsayıların gösterimini belirtmek için, adı MQENC\_DECIMAL ile başlayan bir değişmez.
- Yüzer nokta sayılarının gösterimini belirtmek için, adı MQENC\_FLOAT olan bir değişmez.

Diğer bir seçenek olarak, uygulama özelliği, değeri ortama bağlı olan MQENC\_NATIVE değerine ayarlayabilir.

Varsayılan olarak özellik ayarlanmaz.

Bir uygulama bir hizmet bütünleştirme veriyoluna bağlandığında bu özellik ilgili değildir.

## **JMS\_IBM\_EXCEPTIONMESSAGE**

### **Veri tipi:**

Dizgi

### **Özellik:**

İleti

İletinin kural dışı durum hedefine neden gönderildiğini açıklayan metin. Bu özellik salt okunur.

Bu özellik, yalnızca bir uygulama bir hizmet bütünleştirme veriyoluna bağlandığında ve bir kural dışı durum hedefinden bir ileti aldığı anda anlamlıdır.

## ***JMS\_IBM\_EXCEPTIONPROBLEMINVARIŞ***

**Veri tipi:**

Dizgi

**Özellik:**

İleti

İletinin kural dışı durum hedefine gönderilmeden önce olduğu hedefin adı.

Bu özellik, yalnızca bir uygulama bir hizmet bütünleştirme veriyoluna bağlandığında ve bir kural dışı durum hedefinden bir ileti alındığında anlamlıdır.

## ***JMS\_IBM\_EXCEPTIONREASON***

**Veri tipi:**

System.Int32

**Özellik:**

İleti

İletinin kural dışı durum hedefine gönderilme nedenini gösteren neden kodu.

Bu özellik, yalnızca bir uygulama bir hizmet bütünleştirme veriyoluna bağlandığında ve bir kural dışı durum hedefinden bir ileti alındığında anlamlıdır.

## ***JMS\_IBM\_EXCEPTIMESTAMP***

**Veri tipi:**

System.Int64

**Özellik:**

İleti

İletinin kural dışı durum hedefine gönderildiği zaman.

Saat, 1 Ocak 1970 'te 00:00:00 GMT ' den bu yana milisaniye olarak ifade edilir.

Bu özellik, yalnızca bir uygulama bir hizmet bütünleştirme veriyoluna bağlandığında ve bir kural dışı durum hedefinden bir ileti alındığında anlamlıdır.

## ***JMS\_IBM\_FEEDBACK***

**Veri tipi:**

System.Int32

**Özellik:**

İleti

Rapor iletisinin niteliğini gösteren kod.

Özelliğin geçerli değerleri, bir ileti tanımlayıcısının **Feedback** alanında belirtilebilecek geribildirim kodlarıdır ve neden kodlarıdır.

Varsayılan olarak özellik ayarlanmaz.

## ***JMS\_\_IBM\_***

**Veri tipi:**

Dizgi

**Özellik:**

İleti

İletideki uygulama verilerinin türü.

Özelliğin geçerli değerleri, bir ileti tanımlayıcısının **Format** alanında belirtilebilecek değerlerle aynıdır.

Varsayılan olarak özellik ayarlanmaz.

Bir uygulama bir hizmet bütünleştirme veriyoluna bağlandığında bu özellik ilgili değildir.

### **JMS\_IBM\_LAST\_MSG\_IN\_GROUP**

**Veri tipi:**

System.Boolean

**Özellik:**

İleti

İletinin bir ileti grubundaki son ileti olup olmadığını belirtin.

İleti, bir ileti grubundaki son iletiyse, özelliği true olarak ayarlayın. Ters durumda, özelliği false olarak ayarlayın ya da özelliği ayarlamayın. Varsayılan olarak özellik ayarlanmaz.

True değeri, ileti tanımlayıcısının **MsgFlags** alanında belirtilebilen MQMF\_LAST\_MSG\_IN\_GROUP durum işaretine karşılık gelir.

Yayınla/abone olma etki alanında bu özellik yok sayılır ve bir uygulama bir hizmet tümleştirme veriyoluna bağlandığında, bu özellik uygun değildir.

### **JMS\_IBM\_MSGTYPE**

**Veri tipi:**

System.Int32

**Özellik:**

İleti

İletinin tipi.

Özelliğin geçerli değerleri şunlardır:

**Geçerli değer**

MQMT\_DATAGRAM

MQMT\_REQUEST

MQMT\_REPLY

MQMT\_REPORT

**Anlamı**

İleti, yanıt gerektirmeyen bir iletidir.

İleti, yanıt gerektiren bir iletidir.

İleti bir yanıt iletidir.

İleti, bir rapor iletidir.

Bu değerler, bir ileti tanımlayıcısının **MsgType** alanında belirtilebilecek ileti tipleriyle karşılık gelir.

Varsayılan olarak özellik ayarlanmaz.

Bir uygulama bir hizmet bütünleştirme veriyoluna bağlandığında bu özellik ilgili değildir.

### **JMS\_IBM\_PUTAPPLTYPE**

**Veri tipi:**

System.Int32

**Özellik:**

İleti

İletiyi gönderen uygulamanın tipi.

Özelliğin geçerli değerleri, bir ileti tanımlayıcısının **PutApp1Type** alanında belirtilebilecek uygulama tipleridir.

Varsayılan olarak özellik ayarlanmaz.

Bir uygulama bir hizmet bütünleştirme veriyoluna bağlandığında bu özellik ilgili değildir.

### **JMS\_IBM\_PUTDATE**

**Veri tipi:**

Dizgi

**Özellik:**

İleti

İletinin gönderildiği tarih.

Özelliğin geçerli değerleri, bir ileti tanımlayıcısının **PutDate** alanında belirtilebilecek değerlerle aynıdır.

Varsayılan olarak özellik ayarlanmaz.

Bir uygulama bir hizmet bütünleştirme veriyoluna bağlandığında bu özellik ilgili değildir.

**JMS\_IBM\_PUTTIME****Veri tipi:**

Dizgi

**Özellik:**

İleti

İletinin gönderildiği saat.

Özelliğin geçerli değerleri, bir ileti tanımlayıcısının **PutTime** alanında belirtilebilecek değerlerle aynıdır.

Varsayılan olarak özellik ayarlanmaz.

Bir uygulama bir hizmet bütünleştirme veriyoluna bağlandığında bu özellik ilgili değildir.

**JMS\_IBM\_REPORT\_COA****Veri tipi:**

System.Int32

**Özellik:**

İleti

Özgün iletideki uygulama verilerinin bir rapor iletisine ne kadar eklenmesi gerektiğini belirten 'varış sırasında onayla' rapor iletilerini isteyin.

Özelliğin geçerli değerleri şunlardır:

<b>Geçerli değer</b>	<b>Anlamı</b>
MQRO_COA	Bir rapor iletisinde yer alan özgün iletiden uygulama verileri olmadan, 'varışta onayla' raporu iletilerine istek iletin.
MQRO_CO_WITH_DATA	Bir rapor iletisinde yer alan özgün iletiden ilk 100 bayt olan uygulama verilerinin ilk 100 baytı ile birlikte 'varışta doğrulama' isteği iletilerini isteyin.
MQRO_COA_WITHL_FULL_DATA	Bir rapor iletisinde yer alan özgün iletiden tüm uygulama verileriyle birlikte 'varışta onayla' iletisi iletilerini isteyin.

Bu değerler, bir ileti tanımlayıcısının **Report** alanında belirtilebilecek rapor seçeneklerine karşılık gelir. Bu seçenekler hakkında daha fazla bilgi için bkz. [Rapor \(MQUZE\)](#).

Varsayılan olarak özellik ayarlanmaz.

**JMS\_IBM\_REPORT\_COD****Veri tipi:**

System.Int32

**Özellik:**

İleti

Özgün iletideki uygulama verilerinin bir rapor iletisine ne kadar eklenmesi gerektiğini belirten 'teslimde onayla' rapor iletilerini isteyin.

Özelliğin geçerli değerleri şunlardır:

**Geçerli değer**

MQRO\_COD

MQRO\_COD\_WITH\_DATA

MQRO\_COD\_WITH\_FULL\_DATA

**Anlamı**

Bir rapor iletisinde yer alan özgün iletiden herhangi bir uygulama verisi olmadan, 'teslimde teslim edin' isteği iletilsin.

Bir rapor iletisinde yer alan özgün iletiden ilk 100 bayt olan uygulama verilerinin ilk 100 baytı ile, 'teslimde teslim edilme' isteğini isteyin.

Bir rapor iletisinde yer alan özgün iletiden tüm uygulama verileriyle birlikte 'teslimde teslim edin' isteği iletilsin.

Bu değerler, bir ileti tanımlayıcısının **Report** alanında belirtilebilecek rapor seçeneklerine karşılık gelir. Varsayılan olarak özellik ayarlanmaz.

**JMS\_IBM\_REPORT\_DISCARD\_MSG****Veri tipi:**

System.Int32

**Özellik:**

İleti

İletinin hedeflenen hedefine teslim edilememesi durumunda atılmasını isteyin.

İletiyi MQRO\_DISCARD\_MSG olarak ayarlamak için, iletinin amaçlanan hedefine teslim edilemezse atıldığını isteyin. İletinin, bunun yerine bir ölü mektup kuyruğuna konmasını ya da bir kural dışı durum hedefine gönderilmesini istiyorsanız, bu özelliği ayarlamayın. Varsayılan olarak özellik ayarlanmaz.

MQRO\_DISCARD\_MSG değeri, bir ileti tanımlayıcısının **Report** alanında belirtilebilecek bir rapor seçeneğine karşılık gelir.

**JMS\_IBM\_REPORT\_EXCETION****Veri tipi:**

System.Int32

**Özellik:**

İleti

Özgün iletideki uygulama verilerinin bir rapor iletisine ne kadar eklenmesi gerektiğini belirten kural dışı durum raporu iletileri isteyin.

Özelliğin geçerli değerleri şunlardır:

**Geçerli değer**

MQRO\_EXCEPTION

MQRO\_EXCEPTION\_WITH\_DATA

MQRO\_EXCEPTION\_WITH\_FULL\_DATA

**Anlamı**

Bir rapor iletisinde yer alan özgün iletiden uygulama verileri olmadan, istek kural dışı durumu rapor iletileri.

Bir rapor iletisinde yer alan özgün iletiden ilk 100 bayt uygulama verisi içeren, istek kural dışı durumu rapor iletileri.

Bir rapor iletisinde yer alan özgün iletiden tüm uygulama verilerine sahip istek kural dışı durum raporu iletileri.

Bu değerler, bir ileti tanımlayıcısının **Report** alanında belirtilebilecek rapor seçeneklerine karşılık gelir. Varsayılan olarak özellik ayarlanmaz.



## **JMS\_IBM\_REPORT\_EXPIRATION**

**Veri tipi:**

System.Int32

**Özellik:**

İleti

Özgün iletideki uygulama verilerinin bir rapor iletisine ne kadar dahil edilmesi gerektiğini belirten, süre bitim raporu iletileri isteyin.

Özelliğin geçerli değerleri şunlardır:

**Geçerli değer**

MQRO\_EXPIRATION

**Anlamı**

Bir rapor iletisinde yer alan özgün iletiden herhangi bir uygulama verisi olmadan, süre bitimi rapor iletilerini talep edin.

MQRO\_EXPIRATION\_WITH\_DATA

Bir rapor iletisinde yer alan özgün iletiden ilk 100 bayt olan uygulama verilerinin ilk 100 baytı ile son kullanma tarihi raporu isteğinde bulunmaya devam edin.

MQRO\_EXPIRATION\_WITH\_FULL\_DATA

Bir rapor iletisinde yer alan özgün iletiden tüm uygulama verileriyle birlikte, süre bitimi rapor iletilerini talep edin.

Bu değerler, bir ileti tanımlayıcısının **Report** alanında belirtilebilecek rapor seçeneklerine karşılık gelir.

Varsayılan olarak özellik ayarlanmaz.

## **JMS\_IBM\_REPORT\_NAN**

**Veri tipi:**

System.Int32

**Özellik:**

İleti

Olumsuz eylem bildirim raporu iletileri isteyin.

Negatif işlem bildirim raporu iletileri istemek için özelliği MQRO\_NAN olarak ayarlayın. Olumsuz işlem bildirim raporu iletileri gerekmiyorsa, özelliği ayarlamayın. Varsayılan olarak özellik ayarlanmaz.

MQRO\_NAN değeri, bir ileti tanımlayıcısının **Report** alanında belirtilebilecek bir rapor seçeneğine karşılık gelir.

## **JMS\_IBM\_REPORT\_PAN**

**Veri tipi:**

System.Int32

**Özellik:**

İleti

Pozitif eylem bildirim raporu iletileri isteyin.

Pozitif işlem bildirim raporu iletileri istemek için özelliği MQRO\_PAN olarak ayarlayın. Pozitif işlem bildirim raporu iletileri gerektirmiyorsa, özelliği ayarlamayın. Varsayılan olarak özellik ayarlanmaz.

MQRO\_PAN değeri, bir ileti tanımlayıcısının **Report** alanında belirtilebilecek bir rapor seçeneğine karşılık gelir.

## **JMS\_IBM\_REPORT\_PASS\_CORREL\_ID**

### **Veri tipi:**

System.Int32

### **Özellik:**

İleti

Herhangi bir raporun ya da yanıt iletisinin ilinti tanıtıcısının, özgün iletinin ilinti tanıtıcısıyla aynı olması isteği.

Özelliğin geçerli değerleri şunlardır:

#### **Geçerli değer**

MQRO\_PASS\_COREL\_ID

#### **Anlamı**

Herhangi bir rapor ya da yanıt iletisinin ilinti tanımlayıcısının, özgün iletinin ilinti tanıtıcısıyla aynı olduğunu isteyin.

MQRO\_COPY\_MSG\_ID\_TO\_CORREL\_ID

Herhangi bir rapor ya da yanıt iletisinin ilinti tanımlayıcısının, özgün iletinin ileti tanıtıcısıyla aynı olduğunu isteyin.

Bu değerler, bir ileti tanımlayıcısının **Report** alanında belirtilebilecek rapor seçeneklerine karşılık gelir.

Özelliğin varsayılan değeri MQRO\_COPY\_MSG\_ID\_TO\_CORREL\_ID 'dir.

## **JMS\_IBM\_REPORT\_PASS\_MSG\_ID**

### **Veri tipi:**

System.Int32

### **Özellik:**

İleti

Herhangi bir raporun ya da yanıt iletisinin ileti tanıtıcısının, özgün iletinin ileti tanıtıcısıyla aynı olduğunu isteyin.

Özelliğin geçerli değerleri şunlardır:

#### **Geçerli değer**

MQRO\_PASS\_MSG\_ID

#### **Anlamı**

Herhangi bir rapor ya da yanıt iletisinin ileti tanıtıcısının, özgün iletinin ileti tanıtıcısıyla aynı olduğunu kabul edin.

MQRO\_NEW\_MSG\_ID

Her bir rapor ya da yanıt iletisi için yeni bir ileti tanıtıcısı oluşturulduğunu isteyin.

Bu değerler, bir ileti tanımlayıcısının **Rapor** alanında belirtilebilecek rapor seçeneklerine karşılık gelir.

Özelliğin varsayılan değeri MQRO\_NEW\_MSG\_ID 'dir.

## **JMS\_IBM\_RETAIN**

### **Veri tipi:**

System.Int32

### **Özellik:**

İleti

Bu özelliğin ayarlanması, kuyruk yöneticisine bir iletinin Ulaşılan Yayın olarak kabul edileceğini gösterir. Bir abone konulardan ileti aldığı anda, önceki yayınlarda alınan iletilerin ötesinde, abone olduktan hemen sonra ek iletiler alabilirler. Bu iletiler, abone olunan konular için isteğe bağlı tutulan yayınlardır. Abonelikte eşleşen her konu için, alıkonan bir yayın varsa, yayını abone olan ileti tüketicisine teslim etmek için kullanılır.

RETAIN\_YAYIN, bu özellik için tek geçerli değer. Varsayılan olarak bu özellik belirlenmez.

**Not:** Bu özellik yalnızca yayınlama/abone olma etki alanında anlamlıdır

### ***JMS\_IBM\_SYSTEM\_MESSAGEID***

**Veri tipi:**

Dizgi

**Özellik:**

İleti

İletiyi hizmet bütünleştirme veriyolu içinde benzersiz olarak tanımlayan bir tanıtıcı. Bu özellik salt okunur.

Bu özellik, yalnızca bir uygulama bir hizmet tümleştirme veriyoluna bağlandığında anlamlıdır.

### ***JMS\_APPID***

**Veri tipi:**

Dizgi

**Özellik:**

İleti

İletiyi gönderen uygulamanın adı.

Bu özellik, JMS adı JMSXAppIDolan JMS tanımlı özellidir. Özelliğe ilişkin ek bilgi için *Java Message Service Specification, Sürüm 1.1* başlıklı konuya bakın.

Varsayılan olarak özellik ayarlanmaz.

Bu özellik, bir aracıya gerçek zamanlı bağlantı için geçerli değildir.

### ***JMSX\_DELIVERY\_COUNT***

**Veri tipi:**

System.Int32

**Özellik:**

İleti

İletiyi teslim etme girişimi sayısı.

Bu özellik, JMS adı JMSXDeliveryCountolan JMS özelliğine sahip özeldir. Özelliğe ilişkin ek bilgi için *Java Message Service Specification, Sürüm 1.1* başlıklı konuya bakın.

Varsayılan olarak özellik ayarlanmaz.

Bu özellik, bir aracıya gerçek zamanlı bağlantı için geçerli değildir.

### ***JMSX\_GROUPID***

**Veri tipi:**

Dizgi

**Özellik:**

İleti

İletin ait olduğu ileti grubunun tanıtıcısı.

Bu özellik, JMS adı JMSXGroupIDolan JMS tanımlı özellidir. Özelliğe ilişkin ek bilgi için *Java Message Service Specification, Sürüm 1.1* başlıklı konuya bakın.

Varsayılan olarak özellik ayarlanmaz.

Bu özellik, bir aracıya gerçek zamanlı bağlantı için geçerli değildir.

### ***JMSX\_GROUPSEQ***

**Veri tipi:**

System.Int32

**Özellik:**

İleti

İleti grubundaki iletinin sıra numarası.

Bu özellik, JMS adı JMSXGroupSeqolan JMS tanımlı özellidir. Özelliğe ilişkin ek bilgi için *Java Message Service Specification, Sürüm 1.1* başlıklı konuya bakın.

Varsayılan olarak özellik ayarlanmaz.

Bu özellik, bir aracıya gerçek zamanlı bağlantı için geçerli değildir.

**JMSX\_USERID****Veri tipi:**

Dizgi

**Özellik:**

İleti

İletiyi gönderen uygulamayla ilişkili kullanıcı kimliği.

Bu özellik, JMS adı JMSXUserIDolan JMS tanımlı özellidir. Özelliğe ilişkin ek bilgi için *Java Message Service Specification, Sürüm 1.1* başlıklı konuya bakın.

Varsayılan olarak özellik ayarlanmaz.

Bu özellik, bir aracıya gerçek zamanlı bağlantı için geçerli değildir.

**XMSC\_ASYNC\_EXCAIMS****Veri tipi:**

System.Int32

**Özellik:**

ConnectionFactory

**Uygulanabilir nesnelere:**

JMS denetim aracı uzun adı: ASYNCEXCEPTION

JMS yönetim aracı kısa adı: AEX

Bu özellik, XMS 'nin bir ExceptionListener ' i yalnızca bir bağlantı kesildiğinde mi, yoksa bir XMS API çağırısına zamanuyumsuz olarak bir kural dışı durum oluştuğunda mı bildiğini belirler. Bu özellik, ExceptionListener kaydı olan bu ConnectionFactory ' den oluşturulan tüm Connections için geçerlidir.

Bu özelliğe ilişkin geçerli değerler şunlardır:

**XMLC\_ASYNC\_EXCEPTIONS\_ALL**

Zamanuyumlu bir API çağırısının kapsamı dışında, zamanuyumsuz olarak algılanan herhangi bir kural dışı durum oluştu ve tüm bağlantı bozuk kural dışı durumları ExceptionListener' e gönderildi.

**XMSC\_ASYNC\_EXCEPTIONS\_CONNECTIONBROKEN**

Yalnızca, bozuk bir bağlantının ExceptionListener' e gönderildiğini gösteren kural dışı durumlar vardır. Zamanuyumsuz işleme sırasında oluşan diğer kural dışı durumlar ExceptionListener(ExceptionListener) olarak bildirilmez ve bu nedenle, uygulama bu kural dışı durumlarla ilgili bilgilendirilmez.

Varsayılan olarak bu özellik XMC\_ASYNC\_EXCEPTIONS\_ALL olarak ayarlıdır.

**XMLC\_CLIENT\_ID****Veri tipi:**

Dizgi

**Özellik:**

ConnectionFactory

**Uygulanabilir nesnelere:**

JMS denetim aracı uzun adı: CLIENTID

JMS yönetim aracı kısa adı: CID

Bir bağlantıya ilişkin istemci tanıtıcısı.

İstemci tanıtıcısı yalnızca, yayınlama/abone olma etki alanındaki sürekli abonelikleri desteklemek için kullanılır ve noktadan noktaya iletişim alanında yoksayılr. İstemci tanıtıcılarını ayarlama hakkında daha fazla bilgi için bkz. [ConnectionFactories and Connection objects](#).

Bu özellik, bir aracıyla gerçek zamanlı bağlantı için uygun değildir.

## ***XMLC\_CONNECTION\_TYPE***

### **Veri tipi:**

System.Int32

### **Özellik:**

ConnectionFactory

Bir uygulamanın bağlandığı ileti sistemi sunucusunun tipi.

Özelliğin geçerli değerleri şunlardır:

#### **Geçerli değer**

XMLC\_CT\_RTT

XMLC\_CT\_WMQ

XMLC\_CT\_WPM

#### **Anlamı**

Bir aracıya gerçek zamanlı bağlantı.

Bir IBM MQ kuyruk yöneticisiyle bağlantı.

WebSphere Application Server service integration busile bağlantı.

Varsayılan olarak özellik ayarlanmaz.

## ***XMLC\_DELIVERY\_MODE***

### **Veri tipi:**

System.Int32

### **Özellik:**

Hedef

### **Bir URI ' de kullanılan ad:**

persistence (bir IBM MQ hedefi için)

deliveryMode (bir WebSphere varsayılan ileti sistemi sağlayıcısı hedefi için)

### **Uygulanabilir nesnelere:**

JMS yönetim aracı uzun adı: PERSISTENCE

JMS denetim aracı kısa adı: PER

Hedefe gönderilen iletilerin teslim kipi.

Özelliğin geçerli değerleri şunlardır:

#### **Geçerli değer**

XMSC\_DELIVERY\_NOT\_PERSISTENT

#### **Anlamı**

Hedefe gönderilen bir ileti kalıcı değil. İleti üreticisinin varsayılan teslim kipi ya da gönderme çağrısında belirtilen herhangi bir teslim kipi yoksayılr. If the destination is an IBM MQ queue, the value of the queue attribute *DefPersistence* is also ignored.

## Geçerli değer

XMSC\_DELIVERY\_PERSISTENT

XMLC\_DELIVERY\_AS\_APP

XMLC\_DELIVERY\_AS\_DEST

## Anlamı

Hedefe gönderilen bir ileti kalıcıdır. İleti üreticisinin varsayılan teslim kipi ya da gönderme çağrısında belirtilen herhangi bir teslim kipi yoksayılr. If the destination is an IBM MQ queue, the value of the queue attribute *DefPersistence* is also ignored.

Hedefe gönderilen bir ileti, Gönderme çağrısında belirlenen teslim kipine sahip olur. Gönderme çağrısı herhangi bir teslim kipi belirtmezse, bunun yerine ileti üreticisinin varsayılan teslim kipi kullanılır. Hedef bir IBM MQ kuyruğuysa, *DefPersistence* kuyruk özniteliğinin değeri yoksayılr.

Hedef bir IBM MQ kuyruğıyse, kuyruğa konan bir ileti, *DefPersistence* kuyruk özniteliğinin değeri tarafından belirtilen teslim kipine sahiptir. İleti üreticisinin varsayılan teslim kipi ya da gönderme çağrısında belirtilen herhangi bir teslim kipi yoksayılr.

Hedef bir IBM MQ kuyruğı değilse, anlam XMC\_DELIVERY\_AS\_APP ile aynı olur.

Varsayılan değer XMSC\_DELIVERY\_AS\_APP ' dir.

## ***XMLC\_IC\_PROVIDER\_URL***

### **Veri tipi:**

Dizgi

### **Özellik:**

InitialContext

COS adlandırma hizmetinin web hizmetiyle aynı sunucuda olması gerekmeyecek şekilde JNDI adlandırma dizinini bulmak için kullanılır.

## ***XMSC\_IC\_SECURITY\_AUTHENTICATION***

### **Veri tipi:**

Dizgi

### **Özellik:**

InitialContext

Java Bağlam arabirimi SECURITY\_AUTHENTICATION ögesine dayalı olarak. Bu özellik yalnızca COS adlandırma bağlamı için geçerlidir.

## ***XMSC\_IC\_SECURITY\_CREDENTIAL***

### **Veri tipi:**

Dizgi

### **Özellik:**

InitialContext

Java Bağlam arabirimi SECURITY\_CREDENTIALS. Bu özellik yalnızca COS adlandırma bağlamı için geçerlidir.

## ***XMSC\_IC\_SECURITY\_PRINCIPAL***

**Veri tipi:**

Dizgi

**Özellik:**

InitialContext

Java Bağlam arabirimi SECURITY\_PRINCIPAL değerine dayalı olarak. Bu özellik yalnızca COS adlandırma bağlamı için geçerlidir.

## ***XMSC\_IC\_SECURITY\_PROTOCOL***

**Veri tipi:**

Dizgi

**Özellik:**

InitialContext

Java Bağlam arabirimi SECURITY\_PROTOCOL değerine dayalı Bu özellik yalnızca COS adlandırma bağlamı için geçerlidir.

## ***XMLC\_IC\_URL***

**Veri tipi:**

Dizgi

**Özellik:**

InitialContext

LDAP ve FileSystem bağlamları için, denetlenen nesnelere içeren havuzun adresi.

LDAP ve FileSystem bağlamları için, yönetilen nesnelere içeren havuzun adresi.

## ***XMSC\_IS\_SUBSCRIPTION\_MULTICAST***

**Veri tipi:**

System.Boolean

**Özellik:**

MessageConsumer

İletilerin WebSphere MQ Multicast Transport kullanılarak ileti tüketicisine teslim edilip edilmediğini gösterir. Bu özellik salt okunur.

İletiler WebSphere MQ Multicast Transport kullanılarak ileti tüketicisine teslim ediliyorsa özelliğin değeri true (doğru) değeridir. Ters durumda, değeri false (yanlış) olur.

Bu özellik, yalnızca bir aracıya gerçek zamanlı bağlantı için geçerlidir.

## ***XMSC\_IS\_SUBSCRIPTION\_RELIABLE\_MULTICAST***

**Veri tipi:**

System.Boolean

**Özellik:**

MessageConsumer

İletilerin güvenilir bir hizmet kalitesiyle WebSphere MQ Multicast Transport kullanılarak ileti tüketicisine teslim edilip edilmediğini gösterir. Bu özellik salt okunur.

İletiler güvenilir bir hizmet kalitesi ile WebSphere MQ Multicast Transport kullanılarak ileti tüketicisine teslim ediliyorsa, özelliğin değeri true olur. Ters durumda, değeri false (yanlış) olur.

Bu özellik, yalnızca bir aracıya gerçek zamanlı bağlantı için geçerlidir.

## ***XMLC\_JMS\_MAJOR\_VERSION***

**Veri tipi:**

System.Int32

**Özellik:**

ConnectionMetaVerileri

XMS belirtiminin temel aldığı JMS belirtiminin ana sürüm numarası. Bu özellik salt okunur.

## ***XMLC\_JMS\_MINOR\_VERSION***

**Veri tipi:**

System.Int32

**Özellik:**

ConnectionMetaVerileri

XMS belirtiminin temel aldığı JMS belirtiminin ikincil sürüm numarası. Bu özellik salt okunur.

## ***XMLC\_JMS\_VERSION***

**Veri tipi:**

Dizgi

**Özellik:**

ConnectionMetaVerileri

XMS belirtiminin temel aldığı JMS belirtiminin sürüm tanıtıcısı. Bu özellik salt okunur.

## ***XMLC\_MAJOR\_SÜRÜMÜ***

**Veri tipi:**

System.Int32

**Özellik:**

ConnectionMetaVerileri

XMS istemcisinin sürüm numarası. Bu özellik salt okunur.

## ***XMLC\_MINOR\_SÜRÜMÜ***

**Veri tipi:**

System.Int32

**Özellik:**

ConnectionMetaVerileri

XMS istemcisinin yayın numarası. Bu özellik salt okunur.

## ***XML\_PAROLA***

**Veri tipi:**


Bayt dizisi

**Özellik:**

ConnectionFactory

Bir ileti sistemi sunucusuna bağlanmaya çalışıldığında uygulamanın kimliğini doğrulamak için kullanılacak parola. Parola, [XMLC\\_USERID](#) özelliği ile kullanılır.

Varsayılan olarak özellik ayarlanmaz.

 Multiplatforms'ta IBM MQ 'a bağlanıyorsanız ve bağlantı üreticisinin XMLC\_USERID özelliğini ayarlıyorsanız, bu, oturum açmış kullanıcının **userid** ile eşleşmesi gerekir. Bu özellikleri ayarlamadıysanız, kuyruk yöneticisi varsayılan olarak oturum açmış olan kullanıcının **userid** değerini kullanır. Tek tek kullanıcıların daha fazla bağlantı düzeyinde kimlik doğrulamasına gereksinim duyarsanız, IBM MQ'ünde yapılandırılmış bir istemci kimlik doğrulaması çıkışı yazabilirsiniz. İstemci



kimlik doğrulaması çıkışı yaratılmasına ilişkin ek bilgi için [İstemci uygulaması için planlama kimlik doğrulaması](#) başlıklı konuya bakın.

**z/OS** IBM MQ for z/OS ' a bağlanırken kullanıcının kimliğini doğrulamak için bir güvenlik çıkışı kullanmanız gerekir.

## ***XMLC\_PRIORITY***

### **Veri tipi:**

System.Int32

### **Özellik:**

Hedef

### **Bir URI ' de kullanılan ad:**

öncelik

Hedefe gönderilen iletilerin önceliği.

Özelliğin geçerli değerleri şunlardır:

### **Geçerli değer**

0 aralığında bir tamsayı, en düşük öncelik 9 'a, en yüksek önceliğe

### **Anlamı**

Hedefe gönderilen bir ileti, belirtilen önceliğe sahip. İleti üreticisinin varsayılan önceliği ya da Gönderme çağrısında belirlenen öncelik sayısı dikkate alınmaz. If the destination is an IBM MQ queue, the value of the queue attribute **DefPriority** is also ignored.

XMLC\_PRIORITY\_AS\_APP

Hedefe gönderilen bir ileti, Gönderme çağrısında belirtilen önceliğe sahiptir. Gönderme çağrısı herhangi bir öncelik belirtmezse, bunun yerine ileti üreticisinin varsayılan önceliği kullanılır. Hedef bir IBM MQ kuyruğuysa, **DefPriority** kuyruk özniteliğinin değeri yoksayılır.

XMLC\_PRIORITY\_AS\_DEST

Hedef bir IBM MQ kuyruğıysa, kuyruğa konan bir ileti, **DefPriority** kuyruk özniteliğinin değeri tarafından belirtilen önceliğe sahip olur. İleti üreticisinin varsayılan önceliği ya da Gönderme çağrısında belirlenen öncelik sayısı dikkate alınmaz.

Hedef bir IBM MQ kuyruğu değilse, anlam, XMLC\_PRIORITY\_AS\_APP ile aynı olur.

Varsayılan değer XMSC\_PRIORITY\_AS\_APP ' dir.

WebSphere MQ Real-Time Transport ve WebSphere MQ Multicast Transport , bir iletinin önceliğine dayalı olarak hiçbir işlem kabul etmimiz.

## ***XMLC\_PROVIDER\_NAME***

### **Veri tipi:**

Dizgi

### **Özellik:**

ConnectionMetaVerileri

XMS istemcisinin sağlayıcısı. Bu özellik salt okunur.

## ***XMLC\_RTT\_BROKER\_PING\_INTERVAL***

### **Veri tipi:**

System.Int32

### **Özellik:**

ConnectionFactory

XMS.NET ' in milisaniye cinsinden zaman aralığı, herhangi bir etkinliği saptamak için Gerçek Zamanlı ileti sistemi sunucusuna olan bağlantıyı denetler. Bir etkinlik algılanmıyorsa, istemci ping işlemi başlatır; ping işlemi için herhangi bir yanıt algılanmıyorsa bağlantı kapatılır.

Özelliğin varsayılan değeri 30000 'dir.

### ***XMSC\_RTT\_CONNECTION\_PROTOCOL***

**Veri tipi:**

System.Int32

**Özellik:**

ConnectionFactory

Bir aracıya gerçek zamanlı bağlantı için kullanılan iletişim protokolü.

Özelliğin değeri XMLC\_RTT\_CP\_TCP olmalıdır; bu, TCP/IP üzerinden bir aracıya gerçek zamanlı bağlantı anlamına gelir. Varsayılan değer XMSC\_RTT\_CP\_TCP ' dir.

### ***XMLC\_RTT\_HOST\_NAME***

**Veri tipi:**

Dizgi

**Özellik:**

ConnectionFactory

Aracının çalıştığı sistemin anasistem adı ya da IP adresi.

Bu özellik, aracıyı tanımlamak için XMLC\_RTT\_PORT özelliği ile kullanılır.

Varsayılan olarak özellik ayarlanmaz.

### ***XMLC\_RTT\_LOCAL\_ADRESI***

**Veri tipi:**

Dizgi

**Özellik:**

ConnectionFactory

Bir aracıya gerçek zamanlı bağlantı için kullanılacak yerel ağ arabiriminin anasistem adı ya da IP adresi.

Bu özellik yalnızca, uygulamanın çalıştığı sistemde iki ya da daha çok ağ arabirimi varsa ve gerçek zamanlı bağlantı için hangi arabirimin kullanılması gerektiğini belirtmenize gerek duyarsa yararlıdır. Sistemde yalnızca bir ağ arabirimi varsa, yalnızca o arabirim kullanılabilir. Sistemde iki ya da daha çok ağ arabirimi varsa ve özellik ayarlanmadıysa, arabirim rasgele seçilir.

Varsayılan olarak özellik ayarlanmaz.

### ***XMLC\_RTT\_MULTICAST***

**Veri tipi:**

System.Int32

**Özellik:**

ConnectionFactory ve Hedef

**Bir URI ' de kullanılan ad:**

multicast

Bir bağlantı üreticisi ya da hedefi için çoklu yayın ayarı. Yalnızca konu olan bir hedef bu özelliğe sahip olabilir.

Bir uygulama bu özelliği, bir aracıyla gerçek zamanlı bir bağlantıyla ilişki içinde çoklu yayınlamayı etkinleştirmek için kullanır ve çoklu yayın etkinse, aracından ileti tüketicisi için ileti göndermek için çok hedefli bir şekilde kullanılan yolu belirtmek için kullanılır. Özelliğin, bir ileti üreticinin aracıya ileti göndermesine ilişkin bir etkisi yoktur.

Özelliğın geçerli deęerleri řunlardır:

**Geçerli deęer**

XMSC\_RTT\_MULTICAST\_DISABLEAL

**Anlamı**

İletiler, WebSphere MQ Multicast Transportkullanılarak ileti tüketicisine teslim edilmiyor. Bu deęer, bir ConnectionFactory nesnesi için varsayılan deęerdir.

XMLC\_RTT\_MULTICAST\_ASCF

İletiler, ileti tüketicisi ile ilişkili bağlantı üreticisine ilişkin çoklu yayın ayarına göre ileti tüketicisi olarak teslim edilir. Bağlantı oluşturucuna ilişkin çoklu yayın ayarı, bağlantının oluşturulduğu sırada belirtilir. Bu deęer yalnızca bir Hedef nesne için geçerlidir ve Hedef nesne için varsayılan deęerdir.

XMSC\_RTT\_MULTICAST\_ENABLED

Konu, aracıda çoklu yayın için yapılandırıldıysa, iletiler WebSphere MQ Multicast Transportkullanılarak bir ileti tüketicisine teslim edilir. Konu güvenilir çoklu yayın için yapılandırıldıysa, güvenilir bir hizmet kalitesi kullanılır.

XMSC\_RTT\_MULTICAST\_GÜVENİLİR

Konu, aracıda güvenilir çoklu yayın için yapılandırıldıysa, iletiler güvenilir bir hizmet kalitesi ile WebSphere MQ Multicast Transport kullanılarak bir ileti tüketicisine teslim edilir. Konu güvenilir çoklu yayın için yapılandırılmamışsa, konu için bir ileti tüketicisi yaratamazsınız.

XMSC\_RTT\_MULTICAST\_NOT\_GÜVENİLİR

Konu, aracıda çoklu yayın için yapılandırıldıysa, iletiler WebSphere MQ Multicast Transportkullanılarak bir ileti tüketicisine teslim edilir. Bu konu güvenilir çoklu yayın için yapılandırılmış olsa da, güvenilir bir hizmet kalitesi kullanılmamaktadır.

***XMLC\_RTT\_KAPISI***

**Veri tipi:**

System.Int32

**Özellik:**

ConnectionFactory

Bir aracının gelen istekleri dinlediğı kapının numarası. Aracıyla, bu bağlantı noktasını dinlemek için Real-timeInput ya da Real-timeOptimizedAkış iletisi işleme düğümünü yapılandırmanızdır.

Bu özellik, aracıyı tanıtmak için XMLC\_RTT\_HOST\_NAME özelliğı ile kullanılır.

Özelliğın varsayılan deęeri XMLC\_RTT\_DEFAULT\_PORT ya da 1506 'tur.

***XMLC\_TIME\_TO\_LIVE***

**Veri tipi:**

System.Int32

**Özellik:**

Hedef

**Bir URI ' de kullanılan ad:**

süre bitimi ( IBM MQ hedefi için)

timeToLive (bir WebSphere varsayılan ileti sistemi sağlayıcısı hedefi için)

Hedefe gönderilen iletiler için yaşam süresi.

Özelliğin geçerli değerleri şunlardır:

**Geçerli değer**

**Anlamı**

0

Hedefe gönderilen bir iletinin süresi hiçbir zaman sona ermez.

Pozitif bir tamsayı

Hedefe gönderilen bir iletinin milisaniye olarak (milisaniye) yaşaması için belirtilen süre var. İleti üreticisinin varsayılan süresi ya da gönderme çağrısında belirlenen herhangi bir süre için varsayılan süre yoksayılır.

XMLC\_TIME\_TO\_LIVE\_AS\_APP

Hedefe gönderilen bir iletinin, Gönderme çağrısında belirlenen süre içinde yaşama süresi vardır. Gönderme çağrısı, yaşamak için zaman belirtmezse, bunun yerine ileti üreticisinin yaşaması için varsayılan süre kullanılır.

Varsayılan değer XMLC\_TIME\_TO\_LIVE\_AS\_APP ' dir.

## **XMLC\_USERID**

**Veri tipi:**

Dizgi

**Özellik:**

ConnectionFactory

Bir ileti sistemi sunucusuna bağlanmayı denediğinde uygulamanın kimliğini doğrulamak için kullanılabilir kullanıcı kimliği. Kullanıcı kimliği, [XMLC\\_PARPPARE](#) özelliğiyle kullanılır.

Varsayılan olarak özellik ayarlanmaz.

**Multi** IBM MQ for Multiplatforms' a bağlanıyorsanız ve bağlantı üreticisinin XMLC\_USERID özelliğini ayarlıyorsanız, bu, oturum açmış kullanıcının **userid** ile eşleşmesi gerekir. Bu özellikleri ayarlamadıysanız, kuyruk yöneticisi varsayılan olarak oturum açmış olan kullanıcının **userid** değerini kullanır. Tek tek kullanıcılar için daha fazla bağlantı düzeyinde kimlik doğrulaması gerekiyorsa, IBM MQ içinde yapılandırılmış bir istemci kimlik doğrulama çıkışı yazabilirsiniz. İstemci kimlik doğrulaması çıkışı yaratılmasına ilişkin ek bilgi için [İstemci uygulaması için planlama kimlik doğrulaması](#) başlıklı konuya bakın.

**z/OS** IBM MQ for z/OS ' a bağlanırken kullanıcının kimliğini doğrulamak için bir güvenlik çıkışı kullanmanız gerekir.

## **XMLSC\_FORT**

**Veri tipi:**

Dizgi

**Özellik:**

ConnectionMetaVerileri

cliXMS' nin sürüm tanıtıcısı. Bu özellik salt okunur.

## **V 9.2.4 XMLSC\_WMQ\_BALANCING\_APPLICATION\_TYPE**

**Veri tipi:**

System.Int32

**Özellik:**

ConnectionFactory

Özelliğin geçerli değerleri şunlardır:

### Geçerli değer

XMSC\_WMQ\_BALANCING\_APPLICATION\_TYPE\_SIMPLE

XMSC\_WMQ\_BALANCING\_APPLICATION\_TYPE\_REQUEST\_REPLY

### Anlamı

Basit dengeleme; Tek tip kümelerde uygulama yeniden dengelemeinde açıklananlara ek olarak belirli kurallar uygulanmaz. Bu varsayılan değerdir.

İstek-Yanıt dengeleme; her bir MQPUT çağrısından sonra, bir yanıt iletisi için eşleşen bir MQGET çağrısının beklenmesi beklenir. Böyle bir ileti alınana kadar dengeleme gecikmeli ya da istek iletisi EXPIRY aşımıştır

Ayrıca, bu özellik `client.ini` dosyasında da ayarlanabilir. Tercihin sırası şöyledir:

1. Uygulamadaki özellikler ayarlandı
2. `mqclient.ini` dosyasında Application stanza (Uygulama desteği) ile eşleşen değer.
3. `mqclient.ini` dosyasındaki Application defaults stanza (Uygulama varsayılan ayarları).

## V 9.2.4 XMSC\_WMQ\_BALANCING\_SEÇENEKLER

### Veri tipi:

System.Int32

### Özellik:

ConnectionFactory

Özelliğin geçerli değerleri şunlardır:

### Geçerli değer

XMSC\_WMQ\_BALANCING\_OPTIONS\_NONE

XMSC\_WMQ\_BALANCING\_OPTIONS\_IGNORE\_TRANSACTIONS

### İlgili değer

Hiçbir seçenek ayarlanmadı. Bu varsayılan değerdir

Bu seçeneğin ayarlanması, bir hareketin ortasında bile uygulamaların yeniden dengelenmesine olanak sağlar.

Ayrıca, bu özellik `client.ini` dosyasında da ayarlanabilir. Tercihin sırası şöyledir:

1. Uygulamadaki özellikler ayarlandı
2. `mqclient.ini` dosyasında Application stanza (Uygulama desteği) ile eşleşen değer.
3. `mqclient.ini` dosyasındaki Application defaults stanza (Uygulama varsayılan ayarları).

## V 9.2.4 XMSC\_WMQ\_BALANCING\_TIMEOUT

### Veri tipi:

System.Int32

### Özellik:

ConnectionFactory

Özelliğin geçerli değerleri şunlardır:

### Geçerli değer

XMSC\_WMQ\_BALANCING\_TIMEOUT\_IMMEDIATE

XMSC\_WMQ\_BALANCING\_TIMEOUT\_AS\_DEFAULT

XMSC\_WMQ\_BALANCING\_TIMEOUT\_NEVER

### Anlamı

Anında zamanaşımı oluşur

Set default timeout değeri. Bu varsayılan değerdir

Zamanaşımı oluşmaz

**Not:** Yalnızca tanımlı değerlerden bir değer ya da 0-999999999 saniye arasında bir değer sağlamanız gerekir.

Ayrıca, bu özellik `client.ini` dosyasında da ayarlanabilir. Tercihin sırası şöyledir:

1. Uygulamadaki özellikler ayarlandı
2. `mqclient.ini` dosyasında Application stanza (Uygulama desteği) ile eşleşen değer.
3. `mqclient.ini` dosyasındaki Application defaults stanza (Uygulama varsayılan ayarları).

### ***XMLC\_WMQ\_BROKER\_CONTROLQ***

**Veri tipi:**

Dizgi

**Özellik:**

ConnectionFactory

Bir aracı tarafından kullanılan denetim kuyruğunun adı.

Özelliğin varsayılan değeri `SYSTEM.BROKER.CONTROL.QUEUE`.

Bu özellik, yalnızca yayınlama/abone olma etki alanında anlamlıdır.

### ***XMLC\_WMQ\_BROKER\_PUBQ***

**Veri tipi:**

Dizgi

**Özellik:**

ConnectionFactory

Uygulamaların yayınladıkları iletileri gönderdikleri bir aracı tarafından izlenen kuyruğun adı.

Özelliğin varsayılan değeri `SYSTEM.BROKER.DEFAULT.STREAM`.

Bu özellik, yalnızca yayınlama/abone olma etki alanında anlamlıdır.

### ***XMLC\_WMQ\_BROKER\_QMGR***

**Veri tipi:**

Dizgi

**Özellik:**

ConnectionFactory

Aracının bağlı olduğu kuyruk yöneticisinin adı.

Varsayılan olarak özellik ayarlanmaz.

Bu özellik, yalnızca yayınlama/abone olma etki alanında anlamlıdır.

### ***XMLC\_WMQ\_BROKER\_SUBQ***

**Veri tipi:**

Dizgi

**Özellik:**

ConnectionFactory

Sürekli olmayan bir ileti tüketicisine ilişkin abone kuyruğunun adı.

Abone kuyruğunun adı aşağıdaki karakterlerle başlamalıdır:

`SYSTEM.JMS.ND`.

Tüm kalıcı olmayan ileti okuyucularının bir abone kuyruğunu paylaşmasını istiyorsanız, paylaşılan kuyruğun tam adını belirtin. Uygulamanın kalıcı olmayan bir ileti tüketicisi oluşturabilmesi için, belirtilen adı taşıyan bir kuyruk var olmalıdır.

Her kalıcı olmayan ileti tüketicisinin iletileri kendi özel abone kuyruğundan almasını istiyorsanız, bir yıldız işareti (\*) ile biten bir kuyruk adı belirtin. Daha sonra, bir uygulama dayanıklı olmayan bir ileti tüketicisi yarattığında, XMS istemcisi ileti tüketicisi tarafından özel kullanım için dinamik bir kuyruk yaratır. XMS istemcisi, dinamik kuyruğu yaratmak için kullanılan nesne tanımlayıcısında yer alan **DynamicQName** alanının içeriğini ayarlamak için özelliğin değerini kullanır.

Özelliğin varsayılan değeri SYSTEM.JMS.ND.SUBSCRIBER.QUEUE; bu, XMS ' un varsayılan olarak paylaşılan kuyruk yaklaşımını kullandığı anlamına gelir.

Bu özellik, yalnızca yayınlama/abone olma etki alanında anlamlıdır.

## ***XMLC\_WMQ\_BROKER\_VERSION***

### **Veri tipi:**

System.Int32

### **Özellik:**

ConnectionFactory ve Hedef

### **Bir URI ' de kullanılan ad:**

brokerVersion

Uygulama tarafından bir bağlantı ya da hedef için kullanılan aracının tipi. Yalnızca konu olan bir hedef bu özelliğe sahip olabilir.

Özelliğin geçerli değerleri şunlardır:

### **Geçerli değer**

XMSC\_WMQ\_BROKER\_V1

### **Anlamı**

Uygulama bir IBM MQ yayınlama/abone olma aracısını kullanıyor.

The application can also use this value if you migrate from IBM MQ publish/subscribe to WebSphere Message Broker but did not change the application.

XMSC\_WMQ\_BROKER\_V2

Uygulama, IBM Integration Bus aracısını kullanıyor.

XMSC\_WMQ\_BROKER\_BELIRTIOMEMIŞ

Aracı yeni düzeye geçirildikten sonra, RFH2 üstbilgilerinin artık kullanılmaması için bu özelliği ayarlayın. Geçişten sonra, bu özellik artık ilgili değildir.

Bir connectionfactory için varsayılan değer XMSSC\_WMQ\_BROKER\_BELIRTIOMEMIŞ, ancak varsayılan olarak, özellik bir hedef için ayarlanmamış demektir. Bir hedef için özelliğin ayarlanması, bağlantı üreticisi özelliği tarafından belirtilen herhangi bir değeri geçersiz kılar.

## ***XMLC\_WMQ\_CCDTURL***

### **Veri tipi:**

System.String

### **Özellik:**

ConnectionFactory

### **Uygulanabilir nesnelere:**

JMS yönetim aracı uzun adı: CCDTURL

JMS yönetim aracı kısa adı: CCDT

İstemci kanal tanımlama çizelgesini içeren dosyanın adını ve yerini tanıtan ve dosyaya nasıl erişilebileceğini belirten URL.

Varsayılan olarak bu özellik belirlenmez.

## ***XMLC\_WMQ\_CCSID***

### **Veri tipi:**

System.Int32

**Özellik:**

Hedef

**Bir URI ' de kullanılan ad:**

CCSID

XMS istemcisi iletiyi hedefe ilettiğinde, iletinin gövdesindeki karakter verileri dizgilerinin içinde olduğu kodlanmış karakter takımının ya da kod sayfasının tanıtıcısı (CCSID). Tek bir ileti için ayarlandıysa, JMS\_IBM\_CHARACTER\_SET özelliği, bu özellik tarafından hedef için belirtilen CCSID ' yi geçersiz kılar.

Özelliğin varsayılan değeri 1208 'dir.

Bu özellik, hedeften alınan iletilere değil, yalnızca hedefe gönderilen iletilerdir.

***XMLC\_WMQ\_CHANNEL*****Veri tipi:**

Dizgi

**Özellik:**

ConnectionFactory

**Uygulanabilir nesnelere:**

JMS yönetim aracı uzun adı: KANAL

JMS yönetim aracı kısa adı: CHAN

Bağlantı için kullanılacak kanalın adı.

Varsayılan olarak özellik ayarlanmaz.

Bu özellik, yalnızca bir uygulama istemci kipinde bir kuyruk yöneticisine bağlandığında anlamlıdır.

***XMLC\_WMQ\_CLIENT\_RECONNECT\_OPTIONS*****Veri tipi:**

Dizgi

**Özellik:**

ConnectionFactory

**Uygulanabilir nesnelere:**

JMS yönetim aracı uzun adı: CLIENTRECONNECTORTIONS

JMS denetim aracı kısa adı: CROPT

Bu özellik, bu üretici tarafından yaratılan yeni bağlantılara ilişkin istemci yeniden bağlanma seçeneklerini belirler. XMSC içinde bulunur ve aşağıdakilerden biridir:

- WMQ\_CLIENT\_RECONNECT\_AS\_DEF (varsayılan). `mqclient.ini` dosyasında belirtilen değeri kullanın. Channels stanza içindeki **DefRecon** özelliğini kullanarak değeri ayarlayın. Aşağıdakilerden birine ayarlanabilir:
  1. EVET. WMQ\_CLIENT\_RECONNECT seçeneği olarak işlev görür
  2. No. Varsayılan. Herhangi bir yeniden bağlanma seçeneği belirtmiyor
  3. QMGR. WMQ\_CLIENT\_RECONNECT\_Q\_MGR seçeneği olarak işlev görür
  4. Geçersiz kılındı. WMQ\_CLIENT\_RECONNECT\_DISABLED seçeneği olarak işlev görür
- WMQ\_CLIENT\_RECONNECT. Bağlantı adı listesinde belirtilen kuyruk yöneticilerinden herhangi birine yeniden bağlanın.
- WMQ\_CLIENT\_RECONNECT\_Q\_MGR. Başlangıçta bağlı olduğu kuyruk yöneticisine yeniden bağlanır. Bağlanma girişiminde bulunursa (bağlantı adı listesinde belirtilen) kuyruk yöneticisinin başlangıçta bağlı kuyruk yöneticisine farklı bir QMID değeri varsa, bu değer MQRC\_RECONNECT\_QMID\_MISMATCH değerini döndürür.
- WMQ\_CLIENT\_RECONNECT\_DEVRE dışı. Yeniden bağlantı devre dışı bırakıldı.



## ***XMLC\_WMQ\_CLIENT\_RECONNECT\_TIMEOUT***

**Veri tipi:**

Dizgi

**Özellik:**

ConnectionFactory

**Uygulanabilir nesnelere:**

JMS yönetim aracı uzun adı: CLIENTRECONNECTTIMEOUT

JMS yönetim aracı kısa adı: CRT

XMLC\_WMQ\_CLIENT\_RECONNECT\_TIMEOUT özelliği yalnızca Yönetilen XMS .NET istemcisi için geçerlidir.

Bu özellik, bir istemci bağlantısının yeniden bağlanmayı denediği süreyi (saniye olarak) belirtir.

Bu süre için yeniden bağlanma girişiminde bulunduktan sonra, istemci MQRC\_RECONNECT\_FAILED ile başarısız olur. Bu özelliğe ilişkin varsayılan ayar XMSC.WMQ\_CLIENT\_RECONNECT\_TIMEOUT\_DEFAULT.

Bu özelliğin varsayılan değeri 1800 'dür.

## ***XMLC\_WMQ\_CONNECTION\_MODE***

**Veri tipi:**

System.Int32

**Özellik:**

ConnectionFactory

Bir uygulamanın kuyruk yöneticisine bağlanma kipi.

Özelliğin geçerli değerleri şunlardır:

<b>Geçerli değer</b>	<b>Anlamı</b>
XMLC_WMQ_CM_BINDINGS	En iyi performans için, bağ tanımları kipinde bir kuyruk yöneticisine yönelik bağlantı. Bu değer, C/C + + için varsayılan değerdir.
XMLC_WMQ_CM_İSTEMCENİ	Tam olarak yönetilen bir yığını sağlamak için istemci kipinde bir kuyruk yöneticisine yönelik bağlantı. Bu değer, .NET için varsayılan değerdir.
XMLC_WMQ_CM_CLIENT_UNCEOED (yalnızca .NET için)	Yönetilmeyen bir istemci yığınına zorlayan bir kuyruk yöneticisine yönelik bağlantı.

## ***XMLC\_WMQ\_CONNECTION\_NAME\_LIST***

**Veri tipi:**

Dizgi

**Özellik:**

ConnectionFactory

**Uygulanabilir nesnelere:**

JMS denetim aracı uzun adı: CONNECTIONNAMELIST

JMS yönetim aracı kısa adı: CNLIST

Bu özellik, istemcinin bağlantısı kesildikten sonra istemcinin yeniden bağlanmayı denediği anasistemleri belirler.

Bağlantı adı listesi, anasistem/ip kapı çiftlerinin virgülle ayrılmış listesidir. Bu özelliğe ilişkin varsayılan ayar WMQ\_CONNECTION\_NAME\_LIST\_VARSAYILIDIR.

Örneğin,127.0.0.1 (1414) ,host2.example.com(1400)

Bu özelliğin varsayılan ayarı localhost (1414)' dir.

## ***XMLC\_WMQ\_DUR\_SUBQ***

### **Veri tipi:**

Dizgi

### **Özellik:**

Hedef

Hedeften ileti alan sürekli bir aboneye ilişkin abone kuyruğunun adı. Yalnızca konu olan bir hedef bu özelliğe sahip olabilir.

Abone kuyruğunun adı aşağıdaki karakterlerle başlamalıdır:

SYSTEM.JMS.D.

Tüm dayanıklı abonelerin bir abone kuyruğunu paylaşmasını istiyorsanız, paylaşılan kuyruğun tam adını belirtin. Uygulamanın kalıcı bir abone oluşturabilmesi için, belirtilen adı taşıyan bir kuyruk var olmalıdır.

Her kalıcı abonenin, iletileri kendi özel abone kuyruğundan almasını istiyorsanız, bir yıldız (\*) ile biten bir kuyruk adı belirtin. Daha sonra, bir uygulama dayanıklı bir abone oluşturduğunda, XMS istemcisi, kalıcı abone tarafından özel kullanım için dinamik bir kuyruk yaratır. XMS istemcisi, dinamik kuyruğu yaratmak için kullanılan nesne tanımlayıcısında yer alan **DynamicQName** alanının içeriğini ayarlamak için özelliğin değerini kullanır.

Özelliğin varsayılan değeri SYSTEM.JMS.D.SUBSCRIBER.QUEUE; bu, XMS ' un varsayılan olarak paylaşılan kuyruk yaklaşımını kullandığı anlamına gelir.

Bu özellik, yalnızca yayınlama/abone olma etki alanında anlamlıdır.

## ***XMLC\_WMQ\_ENCODING***

### **Veri tipi:**

System.Int32

### **Özellik:**

Hedef

XMS istemcisi iletiyi hedefe ilettiğinde, iletinin gövdesindeki sayısal verilerin nasıl gösterileceği. Tek bir ileti için ayarlandıysa, JMS\_IBM\_ENCODING özelliği, bu özellik tarafından hedef için belirtilen kodlamayı geçersiz kılar. Bu özellik, ikili tamsayıların, paketlenmiş onlu tamsayıların ve kayan noktalı sayıların gösterilmesini belirtir.

Özelliğin geçerli değerleri, bir ileti tanımlayıcısının **Encoding** alanında belirtilebilecek değerlerle aynıdır.

Bir uygulama, özelliği ayarlamak için aşağıdaki adı belirtilen değişmezleri kullanabilir:

### **Adlandırılmış sabit**

### **Anlamı**

MQENC\_INTEGER\_NORMAL

Normal tamsayı kodlaması

MQENC\_INTEGER\_TERS

Tersine çevrilmiş tamsayı kodlaması

MQENC\_DECIMAL\_NORMAL

Olağan paketlenmiş onlu kodlama

MQENC\_DECIMAL\_TERDI

Tersine çevrilmiş paketlenmiş ondalık kodlama

MQENC\_FLOAT\_IEEE\_NORMAL

Olağan IEEE kayar noktalı kodlama

MQENC\_FLOAT\_IEEEE\_TERS

Ters IEEE kayar noktalı kodlama

MQENC\_FLOAT\_S390

z/OS mimarisi kayan nokta kodlaması

MQENC\_NATIVE

Yerli makine kodlaması

Özelliğe ilişkin bir değer oluşturmak için, uygulama bu değişmezlerden üçünü aşağıdaki gibi ekleyebilir:

- İkili tamsayıların gösterimini belirtmek için, adı MQENC\_XX\_ENCODE\_CASE\_ONE integer ile başlayan bir değişmez.

- Paketlenmiş onlu tamsayıların gösterimini belirtmek için, adı MQENC\_DECIMAL ile başlayan bir değişmez.
- Yüzer nokta sayılarının gösterimini belirtmek için, adı MQENC\_FLOAT olan bir değişmez.

Diğer bir seçenek olarak, uygulama özelliği, değeri ortama bağlı olan MQENC\_NATIVE değerine ayarlayabilir.

Özelliğin varsayılan değeri MQENC\_NATIVE değeridir.

Bu özellik, hedeften alınan iletilere değil, yalnızca hedefe gönderilen iletilerdir.

## ***XMSC\_WMQ\_FAIL\_IF\_QUIESCE***

### **Veri tipi:**

System.Int32

### **Özellik:**

ConnectionFactory ve Hedef

### **Bir URI ' de kullanılan ad:**

failIfQuiesce

### **Uygulanabilir nesnelere:**

JMS yönetim aracı uzun adı: FAILIFQUIESCE

JMS yönetim aracı kısa adı: FIQ

Uygulamanın bağlı olduğu kuyruk yöneticisi susturma durumundaysa, belirli yöntemlere yapılan çağrılar başarısız olup olmadığını belirler.

Özelliğin geçerli değerleri şunlardır:

<b>Geçerli değer</b>	<b>Anlamı</b>
XMLC_WMQ_FIQ_YES	Kuyruk yöneticisi susturulmuş durumdaysa, bazı yöntemlere yönelik çağrılar başarısız olur. Uygulama kuyruk yöneticisinin susturulmuş olduğunu saptarsa, uygulama hemen görevini tamamlayabilir ve kuyruk yöneticisinin durmasını sağlayarak bağlantıyı kapatabilir.
XMLC_WMQ_FIQ_NO	Kuyruk yöneticisi susturulmuş durumda olduğu için yöntem çağrıları başarısız oldu. Bu değeri belirtirseniz, uygulama kuyruk yöneticisinin susturulmuş olduğunu algılayamaz. Uygulama kuyruk yöneticisine yönelik işlemleri gerçekleştirmeye devam edebilir ve bu nedenle kuyruk yöneticisinin durmasını engelleyebilir.

Bir bağlantı üreticisine ilişkin varsayılan değer XMLC\_WMQ\_FIQ\_YES ise, ancak varsayılan olarak, özellik bir hedef için ayarlanmamış demektir. Bir hedef için özelliğin ayarlanması, bağlantı üreticisi özelliği tarafından belirtilen herhangi bir değeri geçersiz kılar.

## ***XMLC\_WMQ\_MESSAGE\_BÖLGESİ***

### **Veri tipi:**

System.Int32

### **Özellik:**

Hedef

Bu özellik, bir XMS uygulamasının bir IBM MQ iletisinin MQRFH2 ' sini ileti bilgi yükünün (ileti gövdesinin bir parçası olarak) bir parçası olarak işleyip işlemediğini belirler.

**Not:** Bir hedefe ileti gönderilirken XMLC\_WMQ\_MESSAGE\_BOLD özelliği, varolan XMS Hedef özelliği XMLC\_WMQ\_TARGET\_CLIENT ögesinin yerini alır.

Bu özelliğe ilişkin geçerli değerler şunlardır:

### **XMLC\_WMQ\_MESSAGE\_BODY\_JMS**

**Alma:** Gelen XMS ileti tipi ve gövdesi, alınan IBM MQ iletisinde MQRFH2 (varsa) ya da MQMD (MQRFH2 yoksa) içeriğiyle belirlenir.

**Gönder:** Giden XMS ileti gövdesi, XMS İleti özellikleri ve üstbilgi alanlarına dayalı olarak önlenmiş ve otomatik oluşturulan bir MQRFH2 üstbilgisi içerir.

### **XMLC\_WMQ\_MESSAGE\_BODY\_MQ**

**Alma:** Gelen XMS ileti tipi her zaman ByteMessage'dır, alınan IBM MQ iletisinin içeriğinden bağımsız olarak ya da alınan MQMD' nin biçim alanı. XMS ileti gövdesi, temeldeki ileti sistemi sağlayıcısı API çağırısı tarafından döndürülen değiştirilmemiş ileti verileridir. İleti gövdesindeki verilerin karakter takımı ve kodlaması, MQMD ' nin CodedCharSetId ve Encoding alanları tarafından belirlenir. İleti gövdesindeki verilerin biçimi, MQMD ' nin Biçim alanı tarafından belirlenir.

**Gönder:** Giden XMS ileti gövdesi, uygulama bilgi yükünü is; olarak içerir; gövdeye otomatik olarak oluşturulan IBM MQ üstbilgisi eklenmez.

### **XMSSC\_WMQ\_MESSAGE\_BODY\_BELIRTILMEMİŞ**

**Alma:** XMS istemcisi bu özellik için uygun bir değer belirler. Alma yolunda, bu değer WMQ\_MESSAGE\_BODY\_JMS özellik değeridir.

**Gönder:** XMS istemcisi bu özellik için uygun bir değer belirler. Gönderme yolunda bu değer XMLC\_WMQ\_TARGET\_CLIENT özellik değeri olur.

Varsayılan olarak bu özellik XMSSC\_WMQ\_MESSAGE\_BODY\_BELIRTILMEMİŞ olarak ayarlanır.

### **XMSC\_WMQ\_MQMD\_MESSAGE\_CONTEXT**

#### **Veri tipi:**

System.Int32

#### **Özellik:**

Hedef

XMS uygulaması tarafından ayarlanacak ileti bağlamı düzeyini belirler. Bu özelliğin yürürlüğe girmesi için uygulamanın uygun bağlam yetkisiyle çalışması gerekir.

Bu özelliğe ilişkin geçerli değerler şunlardır:

#### **XMLC\_WMQ\_MDCTX\_VARSAYILANI**

Giden iletiler için, MQOPER API çağırısı ve MQPMO yapısı belirttik ileti bağlamı seçeneklerini belirtmez.

#### **XMLC\_WMQ\_MDCTX\_SET\_IDENTITY\_CONTEXT**

MQOPER API çağırısı, MQOO\_SET\_IDENTITY\_CONTEXT ileti bağlamı seçeneğini belirtir ve MQPMO yapısı MQPMO\_SET\_IDENTITY\_CONTEXT belirtisini belirtir.

#### **XMLC\_WMQ\_MDCTX\_SET\_ALL\_CONTEXT**

MQOPEP API çağırısı, MQOO\_SET\_ALL\_CONTEXT ileti bağlamı seçeneğini belirtir ve MQPMO yapısı MQPMO\_SET\_ALL\_CONTEXT ' yi belirtir.

Varsayılan olarak bu özellik XMSC\_WMQ\_MDCTX\_default değerine ayarlıdır.

**Not:** Bir uygulama WebSphere Application Server service integration bus' e bağlandığında bu özellik ilgili değildir.

Aşağıdaki özellikler, aranan etkinin olması için ileti gönderirken XMLC\_WMQ\_MDCTX\_SET\_IDENTITY\_CONTEXT ya da XMLC\_WMQ\_MDCTX\_SET\_ALL\_CONTEXT özellik değerine ayarlanacak XMSC\_WMQ\_MQMD\_MESSAGE\_CONTEXT özelliğini gerektirir:

- JMS\_IBM\_MQMD\_USERIDENTIFIER
- JMS\_IBM\_MQMD\_ACCOUNTINGTOKEN
- JMS\_IBM\_MQMD\_APPLIDENTITDATA

Aşağıdaki özellikler, aranan etkinin olması için ileti gönderirken XMLC\_WMQ\_MDCTX\_SET\_ALL\_CONTEXT özellik değerine ayarlanacak XMSSC\_WMQ\_MQMD\_MESSAGE\_CONTEXT özelliğinin değerini gerektirir:

- JMS\_IBM\_MQMD\_PUTAPPLTYPE
- JMS\_IBM\_MQMD\_PUTAPPLNAME
- JMS\_IBM\_MQMD\_PUTDATE
- JMS\_IBM\_MQMD\_PUTTIME
- JMS\_IBM\_MQMD\_APPLORIGINDATA

### ***XMLC\_WMQ\_MQMD\_READENABLED***

**Veri tipi:**

System.Int32

**Özellik:**

Hedef

Bu özellik, bir XMS uygulamasının MQMD alanlarının değerlerini alıp almayacağını belirler.

Bu özelliğe ilişkin geçerli değerler şunlardır:

#### **XMLC\_WMQ\_READENABLED\_NO**

İletileri gönderirken, gönderilen bir iletteki JMS\_IBM\_MQMD\* özellikleri, MQMD 'deki güncellenen alan değerlerini yansıtacak şekilde güncellenmez.

İleti alınırken, bazı ya da tümü gönderen tarafından ayarlanmış olsa da, alınan bir iletide JMS\_IBM\_MQMD\* özelliklerinin hiçbiri kullanılabilir değil.

#### **XMLC\_WMQ\_READ\_ENABLED\_YES**

İletileri gönderirken, gönderilen bir iletteki tüm JMS\_IBM\_MQMD\* özellikleri, gönderenin belirttik olarak ayarlamadığı özellikler de içinde olmak üzere, MQMD 'deki güncellenmiş alan değerlerini yansıtacak şekilde güncellenir.

İleti alınırken, tüm JMS\_IBM\_MQMD\* özellikleri, gönderenin belirttik olarak ayarlamadığı özellikler de içinde olmak üzere, alınan bir iletelerde kullanılabilir.

Varsayılan olarak bu özellik XMSC\_WMQ\_READ\_ENABLED\_NO olarak ayarlıdır.

### ***XMSSC\_WMQ\_MQMD\_WRITE\_ENABLED***

**Veri tipi:**

System.Int32

**Özellik:**

Hedef

Bu özellik, bir XMS uygulamasının MQMD alanlarının değerlerini ayarlayıp ayarlayamayacağını belirler.

Bu özelliğe ilişkin geçerli değerler şunlardır:

#### **XMLC\_WMQ\_WRITE\_ENABLED\_NO**

Tüm JMS\_IBM\_MQMD\* özellikleri yoksayılr ve değerleri, temeldeki MQMD yapısıyla kopyalanmaz.

#### **XMLC\_WMQ\_WRITE\_ENABLED\_YES**

JMS\_IBM\_MQMD\* özellikleri işlendi. Değerleri, temeldeki MQMD yapısına kopyalanır.

Varsayılan olarak bu özellik XMLC\_WMQ\_WRITE\_ENABLED\_NO olarak ayarlıdır.

### ***XMSC\_WMQ\_PUT\_ASYNC\_ALLOWES***

**Veri tipi:**

System.Int32

**Özellik:**

Hedef

Bu özellik, ileti üreticilerinin bu hedefe ileti göndermek için zamanuyumsuz girişler kullanmalarına izin verilip verilmeyeceğini belirler.

Bu özelliğe ilişkin geçerli değerler şunlardır:

**XMLC\_WMQ\_PUT\_ASYNC\_ALLOWED\_AS\_DEST**

Kuyruğa ya da konu tanımlamasına gönderme yaparak zamanuyumsuz yerleştirmeye izin verilip verilmediğini saptayın.

**XMLC\_WMQ\_PUT\_ASYNC\_ALLOWED\_AS\_Q\_DEF**

Kuyruk tanımlamasına gönderme yaparak zamanuyumsuz yerleştirmeye izin verilip verilmediğini saptayın.

**XMLC\_WMQ\_PUT\_ASYNC\_ALLOWED\_AS\_TOPIC\_DEF**

Konu tanımlamasına gönderme yaparak zamanuyumsuz yerleştirmeye izin verilip verilmediğini saptayın.

**XMLC\_WMQ\_PUT\_ASYNC\_ALLOWED\_DISABET**

Zamanuyumsuz yerleştirmeye izin verilmez.

**XMLC\_WMQ\_PUT\_ASYNC\_ALLOWED\_ENABLED**

Zamanuyumsuz yerleştirmeye izin verilir.

Varsayılan olarak bu özellik XMSC\_WMQ\_PUT\_ASYNC\_ALLOWED\_AS\_DEST olarak ayarlıdır.

**Not:** Bir uygulama WebSphere Application Server service integration bus' a bağlanırken bu özellik ilgili değildir.

**XMSC\_WMQ\_READ\_AHEAD\_ALLOWED**

**Veri tipi:**

System.Int32

**Özellik:**

Hedef

Bu özellik, ileti tüketicilerinin ve kuyruk tarayıcılarının, bu hedeften kalıcı olmayan, işlemsel olmayan iletileri almadan önce iç arabelleğe almak için önden okuma özelliğini kullanıp kullanamayacağını belirler.

Bu özelliğe ilişkin geçerli değerler şunlardır:

**XMSC\_WMQ\_READ\_AHEAD\_ALLOWED\_AS\_Q\_DEF**

İleriye okuma işlemine, kuyruk tanımlamasına başvuruda bulunarak izin verilip verilmediğini belirleyin.

**XMSC\_WMQ\_READ\_AHEAD\_ALLOWED\_AS\_KONU\_DEF**

İleriye okumanın, konu tanımlamasına gönderme yaparak izin verilip verilmediğini belirleyin.

**XMSC\_WMQ\_READ\_AHEAD\_ALLOWED\_AS\_DEST**

İleriye okuma işlemine, kuyruk ya da konu tanımından gönderme yaparak izin verilip verilmediğini belirleyin.

**XMSC\_WMQ\_READ\_AHEAD\_ALLOWED\_DISABLEAL**

İletileri tüketirken ya da iletirken okuma işlemine izin verilmez.

**XMSC\_WMQ\_READ\_AHEAD\_ALLOWED\_ENABLED**

İlerlemenize izin verilir.

Varsayılan olarak bu özellik XMSC\_WMQ\_READ\_AHEAD\_ALLOWED\_AS\_DEST olarak ayarlıdır.

**XMSC\_WMQ\_READ\_AHEAD\_CLOSE\_POLICY**

**Veri tipi:**

System.Int32

**Özellik:**

Hedef

Bu özellik, zamanuyumsuz bir ileti dinleyicisine teslim edilen iletiler için, ileti tüketicisi kapatıldığında iç önden okuma arabelleğindeki iletilere ne olacağını belirler.

Bu özellik, bir hedeften iletiler tüketirken ve bir hedefe ileti gönderirken uygulanmadığında, kapanış kuyruğu seçeneklerinin belirlenmesi için geçerlidir.

Bu özellik, kuyruğa göz atma işlemi sırasında kuyrukta bulunmaya devam eden kuyruklar için, kuyruk tarayıcılarında yoksayılr.

Bu özelliğe ilişkin geçerli değerler şunlardır:

**XMSC\_WMQ\_READ\_AHEAD\_CLOSE\_POLICY\_DELIVER\_CURRENT**

Geri dönmeden önce yalnızca yürürlükteki ileti dinleyicisi çağrısı tamamlanır, iç okuma arabelleğindeki iletiler bırakılır ve bu iletiler atılır.

**XMSC\_WMQ\_READ\_AHEAD\_CLOSE\_POLICY\_DELIVER\_ALL**

İç okuma yazma arabelleğindeki tüm iletiler, geri dönmeden önce uygulama iletilisi dinleyicisine teslim edilir.

Varsayılan olarak bu özellik XMSC\_WMQ\_READ\_AHEAD\_CLOSE\_POLICY\_DELIVER\_CURRENT olarak ayarlıdır.

**Not:****Olağandışı uygulama sonlandırması**

Bir XMS uygulaması aniden sona erdiğinde, ileriye okuma arabelleğindeki tüm iletiler kaybedilir.

**Hareketlere ilişkin etkiler**

Uygulamalar hareketleri kullandığında, okuma tamamlama özelliği devre dışı bırakılır. Yani, uygulama, hareket eden seansları kullandıklarında hiçbir şekilde bir fark görmeyecek.

**Oturum scknowledgement kiplerinin etkileri**

Alındı bildirim kipleri XMSC\_AUTO\_RELSE ya da XMSSC\_DUPS\_OK\_RELSE olduğunda, okuma öncesinde okuma yazma işlemi etkin olmayan bir oturumda etkindir. Oturum onaylaması kipi XMSSC\_CLIENT\_RENDE, hareket eden ya da çevrilemeyen oturumlardan bağımsız olarak XMSSC\_CLIENT\_RENDE, okuma öncesinde okuma işlemi geçersiz kılınır.

**Kuyruk tarayıcıları ve kuyruk tarayıcısı seçicileri için etkileri**

The queue browsers and queue browser selectors, used in XMS applications, get the performance advantage from read ahead. Kuyruk tarayıcısının kapatılması başarıyı düşürmez; çünkü ileti kuyrukta devam eder ve daha sonraki işlemler için kuyrukta kullanılabilir. Kuyruk tarayıcılarının ve kuyruk tarayıcısı seçicilerinin, okuma önlerindeki performans avantajlarından ayrı olarak başka hiçbir etkisi yoktur.

**XMLC\_WMQ\_HOST\_NAME****Veri tipi:**

Dizgi

**Özellik:**

ConnectionFactory

**Uygulanabilir nesnelere:**

JMS yönetim aracı uzun adı: HOSTNAME

JMS yönetim aracı kısa adı: HOST

Kuyruk yöneticisinin çalıştığı sistemin anasistem adı ya da IP adresi.

Bu özellik, yalnızca bir uygulama istemci kipinde bir kuyruk yöneticisine bağlandığında kullanılır. Bu özellik, kuyruk yöneticisini tanımlamak için [XMLC\\_WMQ\\_PORT](#) özelliğiyle birlikte kullanılır.

Özelliğin varsayılan değeri localhost' dir.

## ***XMLC\_WMQ\_LOCAL\_ADRESİ***

### **Veri tipi:**

Dizgi

### **Özellik:**

ConnectionFactory

### **Uygulanabilir nesnelere:**

JMS yönetim aracı uzun adı: LOCALADDRESS

JMS yönetim aracı kısa adı: LA

Kuyruk yöneticisine yönelik bir bağlantı için bu özellik, kullanılacak yerel ağ arabirimini ya da kullanılacak yerel kapı ya da yerel kapı aralığını ya da her ikisini de belirtir.

Özelliğin değeri, aşağıdaki biçimi taşıyan bir dizgidir:

[*anasistem\_adi*] [ (*düşük\_kapısı*) [,*yükseklik\_kapısı*]]

Değişkenlerin anlamları aşağıdaki gibidir:

### ***anasistem\_adi***

Bağlantı için kullanılacak yerel ağ arabiriminin anasistem adı ya da IP adresi.

Bu bilgileri yalnızca, uygulamanın çalıştığı sistemin iki ya da daha çok ağ arabirimi varsa ve bağlantı için hangi arabirimin kullanılması gerektiğini belirtmeniz gerekiyorsa gereklidir. Sistemde yalnızca bir ağ arabirimi varsa, yalnızca o arabirim kullanılabilir. Sistemde iki ya da daha çok ağ arabirimi varsa ve hangi arabirimin kullanılması gerektiğini belirtmezseniz, arabirim rasgele seçilir.

### ***düşük\_kapı***

Bağlantı için kullanılacak yerel kapı numarası.

*high\_port* da belirtilirse, *low\_port* , bir kapı numaralarındaki en düşük kapı numarasını yorumlanır.

### ***yükseklik\_kapısı***

Bir kapı numaralarındaki en yüksek kapı numarası. Bağlantı için belirtilen aralıktaki kapılardan biri kullanılmalıdır.

Dizilimin uzunluk üst sınırı 48 karakterdir.

Aşağıda, özelliğin geçerli değerlerine ilişkin bazı örnekler bulunmaktadır:

JUPITER  
9.20.4.98  
JÜPITER (1000)  
9.20.4.98(1000,2000)  
(1000)  
(1000,2000)

Varsayılan olarak özellik ayarlanmaz.

Bu özellik, yalnızca bir uygulama istemci kipinde bir kuyruk yöneticisine bağlandığında anlamlıdır.

## ***XMSC\_WMQ\_MESSAGE\_SELECTION***

### **Veri tipi:**

System.Int32

### **Özellik:**

ConnectionFactory

İleti seçiminin XMS istemcisi tarafından mı, yoksa aracı tarafından mı yapılacağını belirler.

Özelliğin geçerli değerleri şunlardır:

<b>Geçerli değer</b>	<b>Anlamı</b>
XMLC_WMQ_MSEL_İSTEMCESİ	İleti seçimi XMS istemcisi tarafından yapılır.



<b>Geçerli değer</b>	<b>Anlamı</b>
XMLC_WMQ_MSEL_BROKER	İleti seçimi aracı tarafından yapılır.

Varsayılan değer XMSC\_WMQ\_MSEL\_Client değeridir.

Bu özellik, yalnızca yayınlama/abone olma etki alanında anlamlıdır. Message selection by the broker is not supported if the XMLC\_WMQ\_BROKER\_VERSION property is set to XMSC\_WMQ\_BROKER\_V1.

### ***XMLC\_WMQ\_MSG\_BATCH\_SIZE***

**Veri tipi:**

System.Int32

**Özellik:**

ConnectionFactory

Zamanuyumsuz ileti teslimi kullanılırken bir toplu işte kuyruktan alınacak ileti sayısı üst sınırı.

Bir uygulama, belirli koşullar altında zamanuyumsuz ileti teslimi kullanırken, XMS istemcisi her iletiyi uygulamaya tek tek iletmeye önce bir kuyruktan ileti toplu iş alır. Bu özellik, toplu işte olabilecek ileti sayısı üst sınırını belirtir.

Özelliğin değeri pozitif bir tamsayıdır ve varsayılan değer 10 'tır. Yalnızca ele almanız gereken belirli bir başarımlı sorunuz varsa, özelliği farklı bir değere ayarlamayı düşünün.

Bir uygulama, bir ağ üzerinden bir kuyruk yöneticisine bağlıysa, bu özelliğin değerini yükseltmek ağ başlarını ve yanıt sürelerini azaltabilir, ancak iletilerin istemci sisteminde saklanabilmesi için gereken bellek miktarını artırır. Bunun tersine, bu özelliğin değerinin düşürülmesi ağı, ağ kafalarını ve yanıt sürelerini artırabilir, ancak iletilerin depolması için gereken bellek miktarını azaltabilir.

### ***XMLC\_WMQ\_POLLING\_INTERVAL***

**Veri tipi:**

System.Int32

**Özellik:**

ConnectionFactory

Bir oturum içindeki her ileti dinleyicinin kuyruğunda uygun bir ileti yoksa, bu değer, her ileti dinleyicisinin kuyruğundan ileti almaya yeniden çalışmadan önce geçmesi gereken milisaniye cinsinden aralık üst sınırıdır.

Bir oturumda ileti dinleyicilerinin herhangi biri için uygun bir ileti bulunmuyorsa, bu özelliğin değerini artırmayı düşünün.

Özelliğin değeri pozitif bir tamsayıdır. Varsayılan değer 5000 'dir.

### ***XMLC\_WMQ\_PORT***

**Veri tipi:**

System.Int32

**Özellik:**

ConnectionFactory

**Uygulanabilir nesnelere:**

JMS yönetim aracı uzun adı: PORT

JMS yönetim aracı kısa adı: PORT

Kuyruk yöneticisinin gelen istekleri dinlediği kapının numarası.

Bu özellik, yalnızca bir uygulama istemci kipinde bir kuyruk yöneticisine bağlandığında kullanılır. Bu özellik, kuyruk yöneticisini tanıtmak için XMLC\_WMQ\_HOST\_NAME özelliği ile kullanılır.

Özelliğin varsayılan değeri XMLC\_WMQ\_DEFAULT\_CLIENT\_PORT ya da 1414 'tür.

## **XMLC\_WMQ\_PROVIDER\_VERSION**

### **Veri tipi:**

Dizgi

### **Özellik:**

ConnectionFactory

Uygulamanın bağlanmayı planladığı kuyruk yöneticisinin sürümü, yayını, değişiklik düzeyi ve düzeltme paketi. Bu özelliğe ilişkin geçerli değerler şunlardır:

- Belirtilmedi

Ya da aşağıdaki biçimlerden birindeki bir dizgi

- V.R.M.F
- V.R.M
- V.R
- V

Burada V, R, M ve F, sıfıra eşit ya da sıfırdan büyük tamsayı değerleridir.

7 ya da daha büyük bir değer, bu sürümün bir IBM WebSphere MQ 7.0 kuyruk yöneticisine bağlantı için bir IBM WebSphere MQ 7.0 ConnectionFactory olarak amaçlandığı anlamına gelir. A value earlier than 7 (for example "6.0.2.0"), indicates that it is intended for use with queue managers earlier than Version 7.0. Varsayılan değer, belirlenmemiş, kuyruk yöneticisinin herhangi bir düzeylerine bağlantı sağlar ve kuyruk yöneticisinin yeteneklerine dayalı olarak uygulanabilir özellikleri ve işlevleri belirler.

Varsayılan olarak bu özellik "belirtilmemiş" olarak ayarlıdır.

### **Not:**

- XMLC\_WMQ\_PROVIDER\_VERSION 6 olarak ayarlandıysa, yuva paylaşımı olmaz. 2.
- XMLC\_WMQ\_PROVIDER\_VERSION 7 değerine ayarlıysa ve kanal için SHARECNV sunucusunda 0 değeri belirlendiğinde bağlantı başarısız olur.
- XMLC\_WMQ\_PROVIDER\_VERSION BELIRTILMEYEN ve SHARECNV olarak ayarlandıysa, IBM WebSphere MQ 7.0 ' a özgü özellikler 0 olarak ayarlanır.

IBM MQ Client 'in sürümü, bir XMS istemci uygulamasının IBM WebSphere MQ 7.0 ' in özel özelliklerini kullanıp kullanamayacağı konusunda önemli rol oynar. Aşağıdaki tablo, davranışı açıklamalı.

**Not:** XMLC\_WMQ\_OVERRIDEPROVIDERVERSION sistem özelliği XMLC\_WMQ\_PROVIDER\_VERSION özelliğini geçersiz kılar. Bağlantı üreticisi ayarını değiştiremiyorsanız bu özellik kullanılabilir.

<i>Çizelge 881. XMS istemcisi- IBM WebSphere MQ 7.0 ' ye özgü özellikleri kullanma yeteneği.</i>			
#	<b>XMLC_WMQ_PROVIDER_VERSION</b>	<b>IBM MQ İstemci sürümü</b>	<b>IBM WebSphere MQ 7.0 özellikler</b>
1	belirlenmedi	7	AÇIK
2	belirlenmedi	6	KAPALI
3	7	7	AÇIK
4	7	6	Özel durum
5	6	6	KAPALI
6	6	7	KAPALI

## **XMLC\_WMQ\_PUB\_ACK\_INTERVAL**

### **Veri tipi:**

System.Int32

**Özellik:**

ConnectionFactory

XMS istemcisi aracından bir alındı bildirimini istemeden önce bir yayıncı tarafından yayınlanan iletilerin sayısı.

Bu özelliğin değerini azalrsanız, istemci daha sık kabul eder ve dolayısıyla yayıncının performansı azalır. Değeri yükseltiyorsanız, aracı başarısız olursa, istemci daha uzun süre kural dışı durum yayınlamaya devam eder.

Özelliğin değeri pozitif bir tamsayıdır. Varsayılan değer 25 'tür.

***XMSC\_WMQ\_QMGR\_CCSID*****Veri tipi:**

System.Int32

**Bu özelliğin özelliği:**

ConnectionFactory

İleti Kuyruğu Arabirimi 'nde (MQI) tanımlanan karakter verileri alanlarının XMS istemcisi ile IBM MQ istemcisi arasında değiş tokuş edildiği, kodlanmış karakter takımının ya da kod sayfasının tanıtıcısı (CCSID). Bu özellik, iletilerin gövdelerindeki karakter verisi dizgileri için geçerli değildir.

XMS uygulaması istemci kipinde bir kuyruk yöneticisine bağlandığında, XMS istemcisi IBM MQ istemcisine bağlanır. İki istemci arasında değiş tokuş edilen bilgiler, MQI 'da tanımlı karakter verileri alanlarını içerir. Olağan koşullarda, IBM MQ istemcisi bu alanların istemcilerin çalıştığı sistemin kod sayfasında olduğunu varsayar. XMS istemcisi bu alanları farklı bir kod sayfasında sağlar ve almayı bekliyorsa, IBM MQ istemcisini bilgilendirmek için bu özelliği ayarlamamız gerekir.

IBM MQ istemcisi bu karakter verileri alanlarını kuyruk yöneticisine ilettiğinde, gerekiyorsa bu alanlardaki veriler kuyruk yöneticisi tarafından kullanılan kod sayfasına dönüştürülmelidir. Benzer şekilde, IBM MQ istemcisi kuyruk yöneticisinden bu alanları aldığıında, gerekirse bu alanlardaki veriler XMS istemcisinin verileri almayı beklediği kod sayfasına dönüştürülmelidir. IBM MQ istemcisi bu veri dönüştürmelerini gerçekleştirmek için bu özelliği kullanır.

Varsayılan olarak özellik ayarlanmaz.

Bu özelliğin ayarlanması, yerli IBM MQ istemci uygulamalarını destekleyen bir IBM MQ istemcisi için MQCCSID ortam değişkeninin ayarlanmasıyla eşdeğerdir. Bu ortam değişkeniyle ilgili daha fazla bilgi için bkz. [MQCCSID](#).

***XMLC\_WMQ\_QUEUE\_YÖNETİCİSİ*****Veri tipi:**

Dizgi

**Özellik:**

ConnectionFactory

**Uygulanabilir nesnelere:**

JMS yönetim aracı uzun adı: QYORDER

JMS yönetim aracı kısa adı: QMGR

Bağlanılacak kuyruk yöneticisinin adı.

Varsayılan olarak özellik ayarlanmaz.

***XMLC\_WMQ\_RECEIVE\_CCSID***

Kuyruk yöneticisi ileti dönüştürmesi için hedef CCSID ' yi belirleyen hedef özellik.

XMLC\_WMQ\_RECEIVE\_CONVERSION WMQ\_RECEIVE\_CONVERSION\_QMGR değerine ayarlanmıyorsa, değer yoksayılr.

**Veri tipi:**

Tamsayı

**Değer:**

Herhangi bir pozitif tamsayı.

Varsayılan değer 1208 'dir.

İletide GMO\_CONVERT değerinin belirtilmesi isteğe bağlıdır. Bir GMO\_CONVERT değeri belirtilirse, dönüştürme, belirtilen değere göre gerçekleşir.

***XMLC\_WMQ\_RECEIVE\_CONVERSION***

Veri dönüştürme işleminin kuyruk yöneticisi tarafından gerçekleştirilip gerçekleştirilmeyeceğini belirleyen hedef özellik.

**Veri tipi:**

Tamsayı

**Değerler:**

**XMLC\_WMQ\_RECEIVE\_CONVERSION\_CLIENT\_MSG (DEFAULT):** Yalnızca XMS istemcisinde veri dönüştürme işlemi gerçekleştirir. Dönüştürme her zaman 1208 kod sayfası kullanılarak yapılır.

**XMLC\_WMQ\_RECEIVE\_CONVERSION\_QMGR:** XMS istemcisine bir ileti göndermeden önce kuyruk yöneticisine veri dönüştürme işlemi gerçekleştirir.

***XMLC\_WMQ\_RECEIVE\_EXIT*****Veri tipi:**

Dizgi

**Özellik:**

ConnectionFactory

çalıştırılacak kanal alma çıkışı tanımlar.

Özelliğin değeri, bir kanal alma çıkışı tanımlayan bir dizgidir ve aşağıdaki biçime sahiptir:

**libraryName**(entryPointAd)

Burada,

- **libraryName** , yönetilen çıkışa ilişkin .dll' in tam yoludur.
- **entryPointAd** , ad alanı tarafından nitelenen sınıf adıdır

Örneğin, C:\MyReceiveExit.dll(MyReceiveExitNameSpace.MyReceiveExitClassName)

Varsayılan olarak özellik ayarlanmaz.

Bu özellik yalnızca, bir uygulama yönetilen istemci kipindeki bir kuyruk yöneticisine bağlandığında anlamlıdır. Ayrıca yalnızca yönetilen çıkışlar desteklenir.

***XMLC\_WMQ\_RECEIVE\_EXIT\_INIT*****Veri tipi:**

Dizgi

**Özellik:**

ConnectionFactory

Bir kanala geçirilen kullanıcı verileri, çağrıldığında çıkış alır.

Özelliğin değeri bir dizgidir. Varsayılan olarak özellik ayarlanmaz.

Bu özellik yalnızca, bir uygulama yönetilen istemci kipindeki bir kuyruk yöneticisine bağlandığında ve ["XMLC\\_WMQ\\_RECEIVE\\_EXIT"](#) sayfa 2048 özelliği ayarlandığında anlamlıdır.

***XMLC\_WMQ\_RESOLVED\_QUEUE\_YÖNETİCELERİ*****Veri tipi:**

Dizgi

**Özellik:**

ConnectionFactory

Bu özellik, bağlı olduğu kuyruk yöneticisinin adını elde etmek için kullanılır.

Bir CCDT (İstemci Kanalı Tanımlama Çizelgesi) ile kullanıldığında, bu ad, Bağlantı Üreticisi 'nde belirtilen kuyruk yöneticisi adından farklı olabilir.

***XMLC\_WMQ\_RESOLVED\_QUEUE\_MANAGER\_ID*****Veri tipi:**

Dizgi

**Özellik:**

ConnectionFactory

Bu özellik, bağlantı tamamlandıktan sonra kuyruk yöneticisinin tanıtıcısıyla doldurulur.

***XMLC\_WMQ\_SECURITY\_EXIT*****Veri tipi:**

Dizgi

**Özellik:**

ConnectionFactory

Bir kanal güvenlik çıkışı tanımlar.

Özelliğin değeri, kanal güvenliği çıkışı tanımlayan ve aşağıdaki biçime sahip bir dizgidir:

**libraryName**(entryPointAd)

Burada,

- **libraryName** , yönetilen çıkışa ilişkin .dll dosyasının tam yoludur.
- **entryPointAd** , ad alanı tarafından nitelenen sınıf adıdır

Örneğin, C:\MySecurityExit.dll(MySecurityExitNamespace.MySecurityExitClassName)

Dizilimin uzunluk üst sınırı 128 karakterdir.

Varsayılan olarak özellik ayarlanmaz.

Bu özellik yalnızca, bir uygulama yönetilen istemci kipindeki bir kuyruk yöneticisine bağlandığında anlamlıdır. Ayrıca yalnızca yönetilen çıkışlar desteklenir.

***XMLC\_WMQ\_SECURITY\_EXIT\_INIT*****Veri tipi:**

Dizgi

**Özellik:**

ConnectionFactory

Bir kanal güvenlik çıkışına çağrıldığında geçirilen kullanıcı verileri.

Kullanıcı verisi dizgisinin uzunluk üst sınırı 32 karakterdir.

Varsayılan olarak özellik ayarlanmaz.

Bu özellik yalnızca, bir uygulama yönetilen istemci kipindeki bir kuyruk yöneticisine bağlandığında ve [“XMLC\\_WMQ\\_SECURITY\\_EXIT” sayfa 2049](#) özelliği ayarlandığında anlamlıdır.

***XMLC\_WMQ\_SEND\_EXIT*****Veri tipi:**

Dizgi

**Özellik:**

ConnectionFactory

Bir kanal gönderme çıkışı tanımlar.

Özelliğin değeri bir dizgidir. Kanal gönderme çıkışta aşağıdaki biçim bulunur:

**libraryName**(entryPointAd)

Burada,

- **libraryName** , yönetilen çıkışa ilişkin .dll dosyasının tam yoludur.
- **entryPointAd** , ad alanı tarafından nitelenen sınıf adıdır

Örneğin, C:\MySendExit.dll(MySendExitNameSpace.MySendExitClassName)

Varsayılan olarak özellik ayarlanmaz.

Bu özellik yalnızca, bir uygulama yönetilen istemci kipindeki bir kuyruk yöneticisine bağlandığında anlamlıdır. Ayrıca yalnızca yönetilen çıkışlar desteklenir.

## ***XMLC\_WMQ\_SEND\_EXIT\_INIT***

### **Veri tipi:**

Dizgi

### **Özellik:**

ConnectionFactory

Çağrıldığında kanal gönderme çıkışlarına geçirilen kullanıcı verileri.

Özelliğin değeri, kullanıcı verilerinin virgülle ayrılmış bir ya da daha fazla ögesinin dizgisidir. Varsayılan olarak özellik ayarlanmaz.

Kanal gönderme çıkışlarına geçirilen kullanıcı verilerinin belirtilmesine ilişkin kurallar, kanal alma çıkışlarına geçirilen kullanıcı verilerinin belirtilmesine ilişkin kurullarla aynıdır. Bu nedenle kurallar için bkz. [“XMLC\\_WMQ\\_RECEIVE\\_EXIT\\_INIT” sayfa 2048.](#)

Bu özellik yalnızca, bir uygulama yönetilen istemci kipindeki bir kuyruk yöneticisine bağlandığında ve [“XMLC\\_WMQ\\_SEND\\_EXIT” sayfa 2049](#) özelliği ayarlandığında anlamlıdır.

## ***XMLC\_WMQ\_SEND\_CHECK\_COUNT***

### **Veri tipi:**

System.Int32

### **Özellik:**

ConnectionFactory

Tek bir hareket etmeyen XMS oturumunda, zamanuyumsuz koyma hatalarının denetlenmesi arasında izin verilecek gönderme çağrılarının sayısı.

Varsayılan olarak bu özellik 0 değerine ayarlıdır.

## ***XMLC\_WMQ\_SHARE\_CONV\_ALLOWED***

### **Veri tipi:**

System.Int32

### **Özellik:**

ConnectionFactory

### **Uygulanabilir nesnelere:**

JMS denetim aracı uzun adı: SHARECVALLED

JMS yönetim aracı kısa adı: SCALD

Bir istemci bağlantısının, kanal tanımlamaları eşleşiyorsa, aynı işlemde aynı kuyruk yöneticisine yapılan diğer üst düzey XMS bağlantılarıyla yuvasını paylaşıp paylaşamayabileceğini belirler. Bu özellik, uygulama geliştirme, bakım ya da işletim nedenleriyle gerektiğinde Connections 'ın ayrı yuvalarda tam olarak yalıtılmasına olanak sağlamak için sağlanır. Bu özelliğin ayarlanması yalnızca, temeldeki yuvanın paylaşılması için XMS 'a işaret eder. Bu, tek bir yuvayla kaç bağlantının paylaşılacağını göstermez. The

number of connections sharing a socket is determined by SHARECNV value which is negotiated between IBM MQ client and IBM MQ server.

Bir uygulama, özelliği ayarlamak için aşağıdaki adlandırılmış değişmezleri ayarlayabilir:

- XMLC\_WMQ\_SHARE\_CONV\_ALLOWED\_YANR-Bağlantılar bir yuvayı paylaşmaz.
- XMLC\_WMQ\_SHARE\_CONV\_ALLOWED\_TRUE-Bağlantılar bir yuvayı paylaşır.

Varsayılan olarak özellik, XMSSC\_WMQ\_SHARE\_CONV\_ALLOWED\_ENABLED olarak ayarlanır.

Bu özellik, yalnızca bir uygulama istemci kipinde bir kuyruk yöneticisine bağlandığında anlamlıdır.

## ***XMLC\_WMQ\_SSL\_CERT\_STORES***

### **Veri tipi:**

Dizgi

### **Özellik:**

ConnectionFactory

Kuyruk yöneticisine SSL bağlantısında kullanılacak sertifika iptal listelerini (CRL) bulduran sunucuların yerleri.

Özelliğin değeri, virgüllerle ayrılan bir ya da daha fazla URL ' nin bir listesidir. Her URL ' de aşağıdaki biçim vardır:

```
[user[/password]@]ldap://[serveraddress][:portnum][, ...]
```

Bu biçim, temel MQJMS biçimiyle uyumlu, ancak genişletilmiş biçimiyle uyumludur.

Boş bir serveraddress olması geçerlidir. Bu durumda XMS , değerin "localhost" dizgisinin olduğunu varsayar.

Örnek bir liste şöyledir:

```
myuser/mypassword@ldap://server1.mycom.com:389  
ldap://server1.mycom.com  
ldap://  
ldap://:389
```

Yalnızca .NET için: IBM MQ 8.0' tan, IBM MQ (WMQ\_CM\_Client) ile yönetilen bağlantılardan ve IBM MQ (WMQ\_CM\_CLIENT\_UNManaged) ile yönetilen bağlantılardan (WMQ\_CM\_CLIENT\_UNMAED) her ikisi de TLS/SSL bağlantılarını destekler.

Varsayılan olarak özellik ayarlanmaz.

### **İlgili kavramlar**

[Yönetilmeyen .NET istemcisi için SSL ve TLS desteği](#)

[Yönetilen .NET istemcisi için SSL ve TLS desteği](#)

## ***XMLC\_WMQ\_SSL\_CIPHER\_SPEC***

### **Veri tipi:**

Dizgi

### **Özellik:**

ConnectionFactory

Bir kuyruk yöneticisine güvenli bir bağlantıda kullanılacak CipherSpec ' in adı.

IBM MQ TLS desteği ile kullanabileceğiniz şifre belirteçleri aşağıdaki çizelgede listelenir. Kişisel bir sertifika istediğinizde, genel ve özel anahtar çifti için bir anahtar boyutu belirtiyorsunuz. SSL el sıkışması sırasında kullanılan anahtar büyüklüğü, çizelgede belirtildiği gibi CipherSpec tarafından belirlenmediği sürece sertifikada saklanan boyuttur. Varsayılan olarak bu özellik belirlenmez.

CipherSpec adı	Kullanılan iletişim kuralı	Hash algoritması	Şifreleme Algoritması	Şifreleme bitleri	FIPS <sup>1</sup>	Suite B 128 bit	Suite B 192 bit
TLS_RSA_WITH_AES_128_CBC_SHA	TLS 1.0	SHA-1	AES	128	Evet	Hayır	Hayır
TLS_RSA_WITH_AES_256_CBC_SHA <sup>2</sup>	TLS 1.0	SHA-1	AES	256	Evet	Hayır	Hayır
TLS_RSA_WITH_DES_CBC_SHA	TLS 1.0	SHA-1	DES	56	Hayır	Hayır	Hayır
TLS_RSA_WITH_3DES_EDE_CBC_SHA <sup>4</sup>	TLS 1.0	SHA-1	3DES	168	Evet	Hayır	Hayır
TLS_RSA_WITH_AES_128_GCM_SHA256	TLS 1.2	SHA-256	AES	128	Evet	Hayır	Hayır
TLS_RSA_WITH_AES_256_GCM_SHA384	TLS 1.2	SHA-384	AES	256	Evet	Hayır	Hayır
TLS_RSA_WITH_AES_128_CBC_SHA256	TLS 1.2	SHA-256	AES	128	Evet	Hayır	Hayır
TLS_RSA_WITH_AES_256_CBC_SHA256	TLS 1.2	SHA-256	AES	256	Evet	Hayır	Hayır
ECDHE_ECDSA_RC4_128_SHA256	TLS 1.2	SHA-256	RC4	128	Hayır	Hayır	Hayır
ECDHE_ECDSA_3DES_EDE_CBC_SHA256	TLS 1.2	SHA-256	3DES	168	Evet	Hayır	Hayır
ECDHE_RSA_RC4_128_SHA256	TLS 1.2	SHA-256	RC4	128	Hayır	Hayır	Hayır
ECDHE_RSA_3DES_EDE_CBC_SHA256	TLS 1.2	SHA-256	3DES	168	Evet	Hayır	Hayır
ECDHE_ECDSA_AES_128_CBC_SHA256	TLS 1.2	SHA-256	AES	128	Evet	Hayır	Hayır
ECDHE_ECDSA_AES_256_CBC_SHA384	TLS 1.2	SHA-384	AES	256	Evet	Hayır	Hayır
ECDHE_RSA_AES_128_CBC_SHA256	TLS 1.2	SHA-256	AES	128	Evet	Hayır	Hayır
ECDHE_RSA_AES_256_CBC_SHA384	TLS 1.2	SHA-384	AES	256	Evet	Hayır	Hayır
ECDHE_ECDSA_AES_128_GCM_SHA256	TLS 1.2	SHA-256	AES	128	Evet	Evet	Hayır
ECDHE_ECDSA_AES_256_GCM_SHA384	TLS 1.2	SHA-384	AES	256	Evet	Hayır	Evet
ECDHE_RSA_AES_128_GCM_SHA256	TLS 1.2	SHA-256	AES	128	Evet	Hayır	Hayır
ECDHE_RSA_AES_256_GCM_SHA384	TLS 1.2	SHA-384	AES	256	Evet	Hayır	Hayır
TLS_RSA_WITH_NULL_SHA256	TLS 1.2	SHA-256	Yok	0	Hayır	Hayır	Hayır
ECDHE_RSA_NULL_SHA256	TLS 1.2	SHA-256	Yok	0	Hayır	Hayır	Hayır



CipherSpec adı	Kullanılan iletişim kuralı	Hash algoritması	Şifreleme Algoritması	Şifreleme bitleri	FIPS <sup>1</sup>	Suite B 128 bit	Suite B 192 bit
ECDHE_ECDSA_NULL_SHA256	TLS 1.2	SHA-256	Yok	0	Hayır	Hayır	Hayır
TLS_RSA_WITH_NULL_NULL	TLS 1.2	Yok	Yok	0	Hayır	Hayır	Hayır
TLS_RSA_WITH_RC4_128_SHA256	TLS 1.2	SHA-256	RC4	128	Hayır	Hayır	Hayır

#### Notlar:

1. CipherSpec ' in Federal Information Processing Standards (FIPS) 140-2 ile uyumlu olup olmadığını belirtir. FIPS 'nin bir açıklaması ve FIPS 140-2 uyumlu işlem için IBM MQ ' nin nasıl yapılandırılacağı hakkında bilgi için bkz. [Federal Information Processing Standards \(FIPS\)](#).
2. This CipherSpec cannot be used to secure a connection from the IBM MQ Explorer to a queue manager unless the appropriate unrestricted policy files are applied to the JRE used by the IBM MQ Explorer.
3. Bu CipherSpec , 19 Mayıs 2007 tarihinden önce FIPS 140-2 sertifikasına sahiptir.
4. When IBM MQ is configured for FIPS 140-2 compliant operation, this CipherSpec can be used to transfer up to 32 GB of data before the connection is terminated with error AMQ9288. Bu hatayı önlemek için, üçlü DES kullanmaktan kaçının (kullanımdan kaldırılacaktır) ya da FIPS 140-2 yapılandırmasındaki bu CipherSpec komutunu kullanırken gizli anahtar sınırlamayı etkinleştirin.

#### İlgili kavramlar

[İletilerin veri bütünlüğü](#)

#### İlgili görevler

[güvenlik](#)

[CipherSpecsbelirtme](#)

#### ***XMLC\_WMQ\_SSL\_CIPHER\_SUIT***

#### Veri tipi:

Dizgi

#### Özellik:

ConnectionFactory

Bir kuyruk yöneticisine TLS bağlantısında kullanılacak CipherSuite ' in adı. Güvenli bağlantının kararlaştırılmasında kullanılan protokol, belirtilen CipherSuite' e bağlıdır.

Bu özellik aşağıdaki kurallı değerlere sahiptir:

- SSL\_RSA\_WITH\_DES\_CBC\_SHA
- SSL\_RSA\_EXPORT1024\_WITH\_DES\_CBC\_SHA
- SSL\_RSA\_EXPORT1024\_WITH\_RC4\_56\_SHA
- SSL\_RSA\_EXPORT\_WITH\_RC4\_40\_MD5
- SSL\_RSA\_WITH\_RC4\_128\_MD5
- SSL\_RSA\_WITH\_RC4\_128\_SHA
- SSL\_RSA\_WITH\_3DES\_EDE\_CBC\_SHA
- SSL\_RSA\_WITH\_AES\_128\_CBC\_SHA
- SSL\_RSA\_WITH\_AES\_256\_CBC\_SHA
- SSL\_RSA\_WITH\_DES\_CBC\_SHA
- SSL\_RSA\_WITH\_3DES\_EDE\_CBC\_SHA

Bu deęer, `XMLC_WMQ_SSL_CIPHER_SPEC` için bir seçenek olarak sağlanabilir.

`XMSC_WMQ_SSL_CIPHER_SPEC` için boş olmayan bir deęer belirtilirse, bu deęer `XMLC_WMQ_SSL_CIPHER_SUITE` ayarını geçersiz kılar. `XMLC_WMQ_SSL_CIPHER_SPEC` bir deęeri yoksa, GSKit 'e verilecek şifreleme takımı olarak `XMLC_WMQ_SSL_CIPHER_SUITE` deęeri kullanılır. Bu durumda deęer, CipherSuite and CipherSpec name mappings for XMS connections to an IBM MQ queue manager içinde açıklandığı şekilde, eşdeęer CipherSpec deęeriyle eşleştirilir.

Hem `XMSC_WMQ_SSL_CIPHER_SPEC` hem de `XMLC_WMQ_SSL_CIPHER_SUITE` boşsa, pChDef->SSLCipherSpec alanı boşluklarla doldurulur.

Yalnızca .NET için: IBM MQ 8.0' tan, IBM MQ (`WMQ_CM_Client`) ile yönetilen bağlantılardan ve IBM MQ (`WMQ_CM_CLIENT_UNManaged`) ile yönetilen bağlantılardan (`WMQ_CM_CLIENT_UNMAED`) her ikisi de TLS/SSL bağlantılarını destekler.

Varsayılan olarak özellik ayarlanmaz.

### **İlgili kavramlar**

[Yönetilmeyen .NET istemcisi için SSL ve TLS desteęi](#)

[Yönetilen .NET istemcisi için SSL ve TLS desteęi](#)

## ***XMLC\_WMQ\_SSL\_CRYPTO\_HW***

### **Veri tipi:**

Dizgi

### **Özellik:**

ConnectionFactory

İstemci sistemine baęlı şifreleme donanımına ilişkin yapılandırma ayrıntıları.

Bu özellik aşağıdaki kurallı deęerlere sahiptir:

- GSK\_ACCELERATOR\_RAINBOW\_CS\_OFF
- GSK\_ACCELERATOR\_RAINBOW\_CS\_ON
- GSK\_ACCELERATOR\_NCIPHER\_NF\_OFF
- GSK\_ACCELERATOR\_NCIPHER\_NF\_ON

PKCS11 şifreleme donanımları için özel bir biçim vardır (burada DriverPath, TokenLabel ve TokenPassword , kullanıcı tarafından belirlenen dizgilerdir):

```
GSK_PKCS11=PKCS#11 DriverPath; PKCS#11 TokenLabel;PKCS#11 TokenPassword
```

XMS , dizginin içeriğini yorumlamıyor ya da deęiştirmiyor. Sağlanan deęeri, en çok 256 tek baytlık karakter üst sınırı olarak MQSCO.CryptoHardware (Şifreleme) alanı

Yalnızca .NET için: IBM MQ 8.0' tan, IBM MQ (`WMQ_CM_Client`) ile yönetilen bağlantılardan ve IBM MQ (`WMQ_CM_CLIENT_UNManaged`) ile yönetilen bağlantılardan (`WMQ_CM_CLIENT_UNMAED`) her ikisi de TLS/SSL bağlantılarını destekler.

Varsayılan olarak özellik ayarlanmaz.

### **İlgili kavramlar**

[Yönetilmeyen .NET istemcisi için SSL ve TLS desteęi](#)

[Yönetilen .NET istemcisi için SSL ve TLS desteęi](#)

## ***XMLC\_WMQ\_SSL\_FIPS\_REQUIRED***

### **Veri tipi:**

Boole

### **Özellik:**

ConnectionFactory

Bu özelliğin değeri, bir uygulamanın FIPS uyumlu olmayan şifreleme takımlarını kullanıp kullanamayacağını belirler. Bu özellik true olarak ayarlanırsa, istemci-sunucu bağlantısı için yalnızca FIPS algoritmaları kullanılır.

Bu özellik, MQSCO.FipsRequired:

<i>Çizelge 882. MQSCO.FlipsRequired özelliği</i>		
<b>Değer</b>	<b>Tanım</b>	<b>Karşılık gelen MQSCO.FipsRequired</b>
yanlış	Herhangi bir CipherSpec kullanılabilir.	MQSSL_FIPS_NO (varsayılan değer)
doğru	Bu istemci bağlantısına uygulanan CipherSpec ' de yalnızca FIPS onaylı şifreleme algoritmaları kullanılabilir.	MQSSL_FIPS_YES

XMS , ilgili değeri MQSCO.FipsRequired , MQCONNX çağrılmadan önce.

MQSCO.FipsRequired , yalnızca IBM WebSphere MQ 6.0' den edinilebilir. IBM WebSphere MQ 5.3 için bu özellik belirlendiye, XMS kuyruk yöneticisiyle bağlantı kurma girişiminde bulunmaz ve bunun yerine uygun bir kural dışı durum yayınlar.

Yalnızca .NET için: IBM MQ 8.0' tan, IBM MQ (WMQ\_CM\_Client) ile yönetilen bağlantılardan ve IBM MQ (WMQ\_CM\_CLIENT\_UNManaged) ile yönetilen bağlantılardan (WMQ\_CM\_CLIENT\_UNMAED) her ikisi de TLS/SSL bağlantılarını destekler.

#### **İlgili kavramlar**

Yönetilmeyen .NET istemcisi için SSL ve TLS desteği

Yönetilen .NET istemcisi için SSL ve TLS desteği

#### ***XMLC\_WMQ\_SSL\_KEY\_REPOSITORY***

##### **Veri tipi:**

Dizgi

##### **Özellik:**

ConnectionFactory

Anahtarların ve sertifikaların saklandığı anahtar veritabanı dosyasının konumu.

XMS , dizeyi 256 tek baytlık karakter sınırına kadar, MQSCO.KeyRepository alanı. IBM MQ , bu dizeyi tam yol da dahil olmak üzere bir dosya adı olarak yorumlar.

Yalnızca .NET için: IBM MQ 8.0' tan, IBM MQ (WMQ\_CM\_Client) ile yönetilen bağlantılardan ve IBM MQ (WMQ\_CM\_CLIENT\_UNManaged) ile yönetilen bağlantılardan (WMQ\_CM\_CLIENT\_UNMAED) her ikisi de TLS/SSL bağlantılarını destekler.

Varsayılan olarak özellik ayarlanmaz.

#### **İlgili kavramlar**

Yönetilmeyen .NET istemcisi için SSL ve TLS desteği

Yönetilen .NET istemcisi için SSL ve TLS desteği

#### ***XMLC\_WMQ\_SSL\_KEY\_RESETCOUNT***

##### **Veri tipi:**

System.Int32

##### **Özellik:**

ConnectionFactory

KeyResetSayısı, gizli anahtar yeniden anlaşılmadan önce SSL etkileşimi içinde gönderilen ve alınan toplam şifrelenmemiş bayt sayısını temsil eder. Bayt sayısı, MCA tarafından gönderilen denetim bilgilerini içerir.

XMS , bu özellik için sağladığınız değeri MQSCO.KeyResetCount , MQCONNX çağrılmadan önce.

MQSCO.KeyRestCount , yalnızca IBM MQ sürüm 6 'dan kullanılabilir. IBM MQ sürüm 5.3 ise, bu özellik ayarlandıysa, XMS kuyruk yöneticisiyle bağlantı kurma girişiminde bulunmaz ve bunun yerine uygun bir kural dışı durum yayınlar.

Yalnızca .NET için: IBM MQ 8.0' tan, IBM MQ (WMQ\_CM\_Client) ile yönetilen bağlantılardan ve IBM MQ (WMQ\_CM\_CLIENT\_UNManaged) ile yönetilen bağlantılardan (WMQ\_CM\_CLIENT\_UNMAED) her ikisi de TLS/SSL bağlantılarını destekler.

Bu özelliğin varsayılan değeri sıfır, bu da gizli anahtarların hiçbir zaman yeniden görüşmemekte olduğunu gösterir.

### **İlgili kavramlar**

[Yönetilmeyen .NET istemcisi için SSL ve TLS desteği](#)

[Yönetilen .NET istemcisi için SSL ve TLS desteği](#)

## ***XMLC\_WMQ\_SSL\_PEER\_NAME***

### **Veri tipi:**

Dizgi

### **Özellik:**

ConnectionFactory

Kuyruk yöneticisine SSL bağlantısında kullanılacak eşdüzey ad.

Bu özellik için kurallı değerler listesi yok. Bunun yerine, bu dizeyi SSLPEERkurallarına göre oluşturmanız gerekir.

Eşdüzey bir ad örneği:

```
"CN=John Smith, O=IBM ,OU=Test , C=GB"
```

XMS , dizgiyi doğru tek byte kod sayfasına kopyalar ve MQCONNX çağrılmadan önce doğru değerleri MQCD.SSLPeerNamePtr ve MQCD.SSLPeerNameLength içine yerleştirir.

Bu özellik, uygulamanın istemci kipinde bir kuyruk yöneticisine bağlandığı durumlarda anlamlıdır.

Yalnızca .NET için: IBM MQ 8.0' tan, IBM MQ (WMQ\_CM\_Client) ile yönetilen bağlantılardan ve IBM MQ (WMQ\_CM\_CLIENT\_UNManaged) ile yönetilen bağlantılardan (WMQ\_CM\_CLIENT\_UNMAED) her ikisi de TLS/SSL bağlantılarını destekler.

Varsayılan olarak özellik ayarlanmaz.

### **İlgili kavramlar**

[Yönetilmeyen .NET istemcisi için SSL ve TLS desteği](#)

[Yönetilen .NET istemcisi için SSL ve TLS desteği](#)

### **İlgili başvurular**

[SLAYICI ADI](#)

## ***XMSC\_WMQ\_SYNCPOINT\_ALL\_GETS***

### **Veri tipi:**

System.Boolean

### **Özellik:**

ConnectionFactory

Eşitleme noktası denetimi içindeki kuyruklardan tüm iletilerin alınıp alınmayacağı.

Özelliğin geçerli değerleri şunlardır:

### **Geçerli değer**

yanlış

### **Anlamı**

Koşullar uygun olduğunda, XMS istemcisi, iletileri eşitleme noktası denetimi dışında kuyruklardan alabilir.

**Geçerli değer**

doğru

**Anlamı**

XMS istemcisi, eşitleme noktası denetimi içindeki kuyruklardan tüm iletileri almalıdır.

Varsayılan değer: false.

***XMLC\_WMQ\_TARGET\_İSTEMCENI*****Veri tipi:**

System.Int32

**Özellik:**

Hedef

**Bir URI ' de kullanılan ad:**

targetClient

Hedefe gönderilen iletilerin bir MQRFH2 üstbilgisi içerip içermediğini belirler.

Bir uygulama MQRFH2 üstbilgisi içeren bir ileti gönderirse, alıcı uygulamanın üstbilgiyi işleyebilmelidir.

Özelliğin geçerli değerleri şunlardır:

**Geçerli değer****Anlamı**

XMLC\_WMQ\_TARGET\_DEST\_JMS

Hedefte gönderilen iletiler bir MQRFH2 üstbilgisi içerir. Uygulama, iletileri başka bir XMS uygulamasına, bir IBM MQ classes for JMS uygulamasına ya da bir MQRFH2 üstbilgisini işlemek üzere tasarlanmış bir yerel IBM MQ uygulamasına gönderiyorsa bu değeri belirtin.

XMLC\_WMQ\_TARGET\_DEST\_MQ

Hedefte gönderilen iletiler bir MQRFH2 üstbilgisi içermiyor. Uygulama, iletileri bir MQRFH2 üstbilgisini işlemek üzere tasarlanmamış bir yerel IBM MQ uygulamasına gönderiyorsa, bu değeri belirtin.

Varsayılan değer XMSC\_WMQ\_TARGET\_DEST\_JMS ' dir.

***XMLC\_WMQ\_TEMP\_Q\_PREFIX*****Veri tipi:**

Dizgi

**Özellik:**

ConnectionFactory

Uygulama bir XMS geçici kuyruğu oluşturduğunda oluşturulan IBM MQ dinamik kuyruğunun adını oluşturmak için kullanılan önek.

The rules for forming the prefix are the same as the rules for forming the contents of the **DynamicQName** field in an object descriptor, but the last non-blank character must be an asterisk(\*). If the property is not set, the value used is CSQ.\* on z/OS and AMQ.\* on the other platforms. Varsayılan olarak özellik ayarlanmaz.

Bu özellik, yalnızca noktadan noktaya iletişim alanında ilişkilidir.

***XMSC\_WMQ\_TEMP\_TOPIC\_PREFIX*****Veri tipi:**

Dizgi

**Özellik:**

ConnectionFactory, Hedef

Geçici konular oluştururken XMS , "TEMP/TEMPTOPICPREFIX/unique\_id" biçiminde bir konu dizgisi oluşturur ya da bu özellik varsayılan değeri içeriyorsa, bu dizgi ("TEMP/unique\_id") oluşturulur. Boş olmayan bir değer belirtilmesi, abonelerin bu bağlantı altında yaratılan geçici konulara ilişkin yönetilen kuyruklarının yaratılması için belirli model kuyruklarının tanımlanmasına olanak sağlar.

Bir IBM MQ konu dizgisi için yalnızca geçerli karakterlerden oluşan, boş değerli olmayan dizgi, bu özellik için geçerli bir değerdir.

Varsayılan olarak bu özellik "" (boş dizgi) olarak ayarlıdır.

**Not:** Bu özellik, yalnızca yayınlama/abone olma etki alanında anlamlıdır.

### ***XMLC\_WMQ\_TEMPORARY\_MODEL***

**Veri tipi:**

Dizgi

**Özellik:**

ConnectionFactory

Uygulama bir XMS geçici kuyruğu yarattığında dinamik kuyruğun yaratıldığı IBM MQ model kuyruğunun adı.

Özelliğin varsayılan değeri SYSTEM.DEFAULT.MODEL.QUEUE.

Bu özellik, yalnızca noktadan noktaya iletişim alanında ilişkilidir.

### ***XMLC\_WMQ\_WILDCARD\_BIÇIMI***

**Veri tipi:**

System.Int32

**Özellik:**

ConnectionFactory, Hedef

Bu özellik, genel arama karakteri sözdiziminin hangi sürümünü kullanılacağını belirler.

Yayınlama/abone olma özelliğini IBM MQ '\*' ve '?' ile kullanırken genel arama karakteri olarak kabul edilir. Yayınlama kullanılırken '#' ve '+' karakterleri, IBM Integration Bus ile abone olma özelliğini kullanırken joker karakterler olarak ele alınır. Bu özellik, XMLC\_WMQ\_ROKER\_VERSION özelliğinin yerini alır.

Bu özelliğe ilişkin geçerli değerler şunlardır:

#### ***XMSC\_WMQ\_WILDCARD\_TOPIC\_ONLY***

Konu düzeyi genel arama karakterlerini yalnızca başka bir deyişle tanıır. '#' ve '+' karakteri joker karakter olarak kabul edilir. Bu değer XMSC\_WMQ\_BROKER\_V2 ile aynıdır.

#### ***XMLC\_WMQ\_WILDCARD\_CHAR\_ONLY***

Karakter genel arama karakterlerini tanıır, başka bir deyişle '\*' ve '?' genel arama karakteri olarak kabul edilir. Bu değer XMSC\_WMQ\_BROKER\_V1 ile aynıdır.

Varsayılan olarak bu özellik XMSSC\_WMQ\_WILDCARD\_TOPIC\_TOPIC\_TOPIC\_ONLY olarak ayarlanır.

### ***XMLC\_WPM\_BUSU\_ADI***

**Veri tipi:**

Dizgi

**Özellik:**

ConnectionFactory ve Hedef

**Bir URI ' de kullanılan ad:**

busName

Bir bağlantı üreticisi için, uygulamanın bağlandığı hizmet tümleştirme veriyolunun adı ya da hedef için, hedefin bulunduğu hizmet tümleştirme veriyolunun adı.

Konu olan bir hedef için bu özellik, ilişkili konu alanının bulunduğu hizmet tümleştirme veriyolunun adıdır. Bu konu alanı, [XMSSC\\_WPM\\_TOPIC\\_SPACE](#) özelliği tarafından belirtilir.

Özellik bir hedef için ayarlanmazsa, kuyruğun ya da ilişkili konu alanının, uygulamanın bağlandığı hizmet tümleştirme veriyolunda var olduğu varsayılır.

Varsayılan olarak özellik ayarlanmaz.

## ***XMSC\_WPM\_CONNECTION\_PROTOCOL***

### **Veri tipi:**

System.Int32

### **Özellik:**

Bağlantı

İleti alışverişi altyapısına bağlantı için kullanılan iletişim protokolü. Bu özellik salt okunur.

Özellikğin olası değerleri aşağıdaki gibidir:

<b>Değer</b>	<b>Anlamı</b>
XMLC_WPM_CP_HTTP	Bağlantı TCP/IP üzerinden HTTP kullanır.
XMLC_WPM_CP_TCP	Başlantı TCP/IP ' yi kullanır.

## ***XMSC\_WPM\_CONNECTION\_PROXIMITY***

### **Veri tipi:**

System.Int32

### **Özellik:**

ConnectionFactory

Bağlantı için bağlantı yakınlığı ayarı. Bu özellik, uygulamanın bağlanacağı ileti alışverişi altyapısının önyükleme sunucusuna ne kadar yakın olduğunu belirler.

Özellikğin geçerli değerleri şunlardır:

<b>Geçerli değer</b>	<b>Bağlantı yakınlığı ayarı</b>
XMSC_WPM_CONNECTION_PROXIMITY_BUS	Veriyolu
XMSC_WPM_CONNECTION_PROXIMITY_CLUSTER	Küme
XMSC_WPM_CONNECTION_PROXPROTY_HOST	Anasistem
XMSC_WPM_CONNECTION_PROXIMITY_SERVER	Sunucu

Varsayılan değer XMSSC\_WPM\_CONNECTION\_PROXIMITY\_BUS değeridir.

## ***XMLC\_WPM\_DUR\_SUB\_HOME***

### **Veri tipi:**

Dizgi

### **Özellik:**

ConnectionFactory

### **Bir URI ' de kullanılan ad:**

durableSubscriptionAna Sayfası

Dayanıklı abonelere teslim edilecekBir bağlantıya ya da hedefe ilişkin tüm sürekli aboneliklerin yönetildiği ileti alışverişi altyapısının adı. iletileri, aynı ileti alışverişi işlemcisinin yayın noktasında saklanır.

Bir uygulamanın bağlantıyı kullanan dayanıklı bir abone oluşturabilmesi için, bağlantı için sürekli abonelik ana dizini belirtilmelidir. Bir hedef için belirtilen herhangi bir değer, bağlantı için belirtilen değeri geçersiz kılar.

Varsayılan olarak özellik ayarlanmaz.

Bu özellik, yalnızca yayınlama/abone olma etki alanında anlamlıdır.

## ***XMLC\_WPM\_HOST\_NAME***

### **Veri tipi:**

Dizgi

### **Özellik:**

Bağlantı

Uygulamanın bağlı olduğu ileti alışverişi altyapısını içeren sistemin anasistem adı ya da IP adresi. Bu özellik salt okunur.

## ***XMLC\_WPM\_LOCAL\_ADRESİ***

### **Veri tipi:**

Dizgi

### **Özellik:**

ConnectionFactory

Bir hizmet bütünleştirme veriyolu bağlantısı için bu özellik, kullanılacak yerel ağ arabirimini ya da kullanılacak yerel kapı ya da yerel kapı aralığını ya da her ikisini de belirtir.

Özelliğin değeri, aşağıdaki biçimi taşıyan bir dizgidir:

[*anasistem\_adi*] [ (*düşük\_kapısı*) [,*yükseklik\_kapısı*!]]

Değişkenlerin anlamları aşağıdaki gibidir:

### ***anasistem\_adi***

Bağlantı için kullanılacak yerel ağ arabiriminin anasistem adı ya da IP adresi.

Bu bilgileri yalnızca, uygulamanın çalıştığı sistemin iki ya da daha çok ağ arabirimi varsa ve bağlantı için hangi arabirimin kullanılması gerektiğini belirtmeniz gerekiyorsa gereklidir. Sistemde yalnızca bir ağ arabirimi varsa, yalnızca o arabirim kullanılabilir. Sistemde iki ya da daha çok ağ arabirimi varsa ve hangi arabirimin kullanılması gerektiğini belirtmezseniz, arabirim rasgele seçilir.

### ***düşük\_kapı***

Bağlantı için kullanılacak yerel kapı numarası.

*high\_port* da belirtilirse, *low\_port* , bir kapı numaralarındaki en düşük kapı numarasını yorumlanır.

### ***yükseklik\_kapısı***

Bir kapı numaralarındaki en yüksek kapı numarası. Bağlantı için belirtilen aralıktaki kapılardan biri kullanılmalıdır.

Aşağıda, özelliğin geçerli değerlerine ilişkin bazı örnekler bulunmaktadır:

```
JUPITER
9.20.4.98
JUPITER (1000)
9.20.4.98(1000,2000)
(1000)
(1000,2000)
```

Varsayılan olarak özellik ayarlanmaz.

## ***XMLC\_WPM\_ME\_ADI***

### **Veri tipi:**

Dizgi

### **Özellik:**

Bağlantı

Uygulamanın bağlı olduğu ileti alışverişi altyapısının adı. Bu özellik salt okunur.



## ***XMLC\_WPM\_NON\_PERSISTENT\_MAP***

### **Veri tipi:**

System.Int32

### **Özellik:**

ConnectionFactory

Bağlantı kullanılarak gönderilen kalıcı olmayan iletilerin güvenilirlik düzeyi.

Özelliğın geçerli değeri şunlardır:

### **Geçerli değeri**

XMSC\_WPM\_MAPPING\_AS\_DESTIND

XMSSC\_WPM\_MAPPING\_BEST\_ZAHMET\_NON\_  
Kalıcı

XMSSC\_WPM\_MAPPING\_EXPRESS\_NON\_  
Kalıcı

XMSSC\_WPM\_MAPPING\_RELIABLE\_NON\_  
Kalıcı

XMSC\_WPM\_MAPPING\_RELIABLE\_PERSISTENT

XMSC\_WPM\_MAPPING\_ASSURED\_PERSISTENT

Varsayılan değeri XMSSC\_WPM\_MAPPING\_EXPRESS\_NON\_PERSISTENT değeri.

## ***XMLC\_WPM\_PERSISTENT\_MAP***

### **Veri tipi:**

System.Int32

### **Özellik:**

ConnectionFactory

Bağlantı kullanılarak gönderilen kalıcı iletilerin güvenilirlik düzeyi.

Özelliğın geçerli değeri şunlardır:

### **Geçerli değeri**

XMSC\_WPM\_MAPPING\_AS\_DESTIND

XMSSC\_WPM\_MAPPING\_BEST\_ZAHMET\_NON\_  
Kalıcı

XMSSC\_WPM\_MAPPING\_EXPRESS\_NON\_  
Kalıcı

XMSSC\_WPM\_MAPPING\_RELIABLE\_NON\_  
Kalıcı

### **Güvenilirlik düzeyi**

Hizmet tümleştirme veriyolundaki kuyruk ya da konu alanı için belirtilen varsayılan güvenilirlik düzeyine göre belirlenir.

Kalıcı olmayan en iyi çalışma

Express kalıcı olmayan

Güvenilir olmayan güvenilir

Güvenilir kalıcı

Güvenli kalıcı

### **Güvenilirlik düzeyi**

Hizmet tümleştirme veriyolundaki kuyruk ya da konu alanı için belirtilen varsayılan güvenilirlik düzeyine göre belirlenir.

Kalıcı olmayan en iyi çalışma

Express kalıcı olmayan

Güvenilir olmayan güvenilir

**Geçerli değer**

XMSC\_WPM\_MAPPING\_RELIABLE\_PERSISTENT

XMSC\_WPM\_MAPPING\_ASSURED\_PERSISTENT

**Güvenilirlik düzeyi**

Güvenilir kalıcı

Güvenli kalıcı

Varsayılan değer XMSSC\_WPM\_MAPPING\_RELIABLE\_PERSISTENT değeridir.

***XMLC\_WPM\_KAPISI*****Veri tipi:**

System.Int32

**Özellik:**

Bağlantı

Uygulamanın bağlı olduğu ileti alışverişi altyapısı tarafından dinlenen kapının sayısı. Bu özellik salt okunur.

***XMSC\_WPM\_PROVIDER\_UÇ Noktaları*****Veri tipi:**

Dizgi

**Özellik:**

ConnectionFactory

Önyükleme sunucularının bir ya da daha çok uç noktası adresi sırası. Uç nokta adresleri virgüllerle ayrılır.

Önyükleme sunucusu, uygulamanın bağlandığı ileti alışverişi altyapısının seçilmesinden sorumlu olan bir uygulama sunucudur. Bir önyükleme sunucusunun uç noktası adresi şu biçimde olmalıdır:

*anasistem\_adı:kapı\_numarası:chain\_name*

Bir uç nokta adresinin bileşenlerinin anlamları aşağıdaki gibidir:

***anasistem\_adı***

Önyükleme sunucusunun yer aldığı sistemin anasistem adı ya da IP adresi. Anasistem adı ya da IP adresi belirlenmezse, varsayılan değer localhost olur.

***kapı\_numarası***

Önyükleme sunucusunun gelen istekleri dinlediği kapının numarası. Kapı numarası belirtilmezse, varsayılan değer 7276 'tır.

***chain\_name***

Önyükleme sunucusu tarafından kullanılan bir önyükleme iletim zincirinin adı. Geçerli değerler şunlardır:

**Geçerli değer**

XMSC\_WPM\_BOOTSTRAP\_HTTP

XMSSC\_WPM\_BOOTSTRAP\_HTTPS

XMSC\_WPM\_BOOTSTRAP\_SSL

XMLC\_WPM\_BOOTSTRAP\_TCP

**Önyükleme iletimi zincirinin adı**

BootstrapTunneledİleti Sistemi

BootstrapTunneledSecureMessaging

BootstrapSecureİleti Sistemi

BootstrapBasicİleti Sistemi

Ad belirtilmezse, varsayılan değer XMSC\_WPM\_BOOTSTRAP\_TCP ' dir.

Herhangi bir uç nokta adresi belirtilmezse, varsayılan değer olarak localhost:7276:BootstrapBasicMessagingvarsayılan değeri olur.

***XMLC\_WPM\_SSL\_CIPHER\_SUIT*****Veri tipi:**

Dizgi

**Özellik:**

ConnectionFactory

Bir WebSphere Application Server service integration bus ileti alışverişi altyapısına TLS bağlantısında kullanılacak CipherSuite ' in adı. Güvenli bağlantının kararlaştırılmasında kullanılan protokol, belirtilen CipherSuite' e bağlıdır.

<i>Çizelge 883. WebSphere Application Server service integration bus ileti alışverişi altyapısıyla bağlantı içinCipherSuite seçenekleri</i>	
<b>şifre paketi</b>	<b>Kullanılan iletişim kuralı</b>
TLS_RSA_WITH_DES_CBC_SHA	TLSv1
TLS_RSA_WITH_3DES_EDE_CBC_SHA	TLSv1
TLS_RSA_WITH_AES_128_CBC_SHA	TLSv1
TLS_RSA_WITH_AES_256_CBC_SHA	TLSv1

#### **Notlar:**

1. **Windows** TLS\_RSA\_WITH\_AES\_128\_CBC\_SHA ve TLS\_RSA\_WITH\_AES\_256\_CBC\_SHA CipherSuites , yalnızca Windows üzerinde desteklenir. (Bu, GSKit tarafından dikte edilir.)
2. TLS\_RSA\_WITH\_3DES\_EDE\_CBC\_SHA kullanımdan kaldırıldı. However, it can still be used to transfer up to 32 GB of data before the connection is terminated with error AMQ9288. Bu hatayı önlemek için, üçlü DES kullanmaktan kaçınmanız ya da bu CipherSpeckomutunu kullanırken gizli anahtar sınırlamayı etkinleştirmeniz gerekir.

Bu özellik için varsayılan bir değer yoktur. SSL ya da TLS kullanmak istiyorsanız, bu özellik için bir değer belirlemeniz gerekir; tersi durumda, uygulamanız sunucuya başarıyla bağlanamıyor.

#### ***XMSC\_WPM\_SSL\_FIPS\_REQUIRED***

**Not:** AIX, Linux, and Windows işletim sistemlerinde IBM MQ , "IBM Crypto for C" şifreleme modülü aracılığıyla FIPS 140-2 uyumluluğu sağlar. Bu modüle ilişkin sertifika Geçmiş durumuna taşındı. Müşteriler, IBM Crypto for C certificate belgesini görüntüleyip NIST tarafından sağlanan tüm tavsiyeleri dikkate almalıdır. Yeni bir FIPS 140-3 modülü şu anda devam ediyor ve durumu [İşlem listesindeki NIST CMVP modüllerinde](#) aranarak görüntülenebilir.

#### **Veri tipi:**

Boole

#### **Bu özelliğin özelliği:**

ConnectionFactory

Bu özelliğin değeri, bir uygulamanın FIPS uyumlu olmayan şifreleme takımlarını kullanıp kullanamayacağını belirler. Bu özellik true olarak ayarlanırsa, istemci-sunucu bağlantısı için yalnızca FIPS algoritmaları kullanılır. Bu özelliğin değerinin TRUE olarak ayarlanması, uygulamanın FIPS uyumlu olmayan şifreleme takımlarını kullanmasını önler.

Varsayılan olarak, özellik FALSE (FIPS kipi kapalı) olarak ayarlanır.

#### ***XMSC\_WPM\_SSL\_KEY\_REPOSITORY***

#### **Veri tipi:**

Dizgi

#### **Özellik:**

ConnectionFactory

Güvenli bağlantıda kullanılacak genel ya da özel anahtarları içeren anahtarlık dosyası olan dosyanın yolu.

Anahtarlık dosyası özelliğinin, XMSSC\_WPM\_SSL\_MS\_CERTIFICATE\_STORE özel değerine ayarlanması, Microsoft Windows anahtar veri tabanının kullanılmasını belirtir. **Control Panel > Internet Options > Content > Certificates**(Denetim Masası-İnternet Seçenekleri-> İçerik-> Sertifikalar)underaltında bulunan Microsoft Windows anahtar veritabanını kullanarak, ayrı bir anahtar dosyası veritabanı gereksinmesini kaldırır. Bu deęişmezin Windows x64 üzerinde ve dięer platformlarda kullanılmasına izin verilmez.

Varsayılan olarak özellik ayarlanmaz.

### ***XMLC\_WPM\_SSL\_KEYRING\_LABEL***

**Veri tipi:**

Dizgi

**Özellik:**

ConnectionFactory

Sunucuyla kimlik doğrulaması yaparken kullanılacak sertifika. Değer belirlenmemişse, varsayılan sertifika kullanılır.

Varsayılan olarak özellik ayarlanmaz.

### ***XMLC\_WPM\_SSL\_KEYRING\_PW***

**Veri tipi:**

Dizgi

**Özellik:**

ConnectionFactory

Anahtarlık dosyasına ilişkin parola.

Bu özellik, anahtarlık dosyasına ilişkin parolayı yapılandırmak için XMLC\_WPM\_SSL\_KEYRING\_STASH\_FILE dosyasını kullanmak için bir alternatif olarak kullanılabilir.

Varsayılan olarak özellik ayarlanmaz.

### ***XMLC\_WPM\_SSL\_KEYRING\_STASH\_FILE***

**Veri tipi:**

Dizgi

**Özellik:**

ConnectionFactory

Anahtar havuzu dosyasının parolasını içeren ikili dosyanın adı.

Bu özellik, anahtarlık dosyasına ilişkin parolayı yapılandırmak için XMLC\_WPM\_SSL\_KEYRING\_PW dosyasının kullanılması için bir alternatif olarak kullanılabilir.

Varsayılan olarak özellik ayarlanmaz.

### ***XMLC\_WPM\_TARGET\_GROUP***

**Veri tipi:**

Dizgi

**Özellik:**

ConnectionFactory

İleti alışverişi altyapılarının hedef grubunun adı. Hedef grubun niteliği XMLC\_WPM\_TARGET\_TYPE özelliği tarafından belirlenir.

Bir ileti alışverişi altyapısıyla ilgili aramayı, hizmet tümleştirme veriyolundaki ileti alışverişi altyapılarının bir alt grubuna sınırlamak için bu özelliği ayarlayın. Uygulamanızın hizmet tümleştirme veriyolundaki herhangi bir ileti alışverişi işlemcisine bağlanabilmesini istiyorsanız, bu özelliği ayarlamayın.

Varsayılan olarak özellik ayarlanmaz.

### ***XMSC\_WPM\_HEDEF\_ANLAM***

**Veri tipi:**

System.Int32

**Özellik:**

ConnectionFactory

İleti alışverişi altyapılarının hedef grubunun önemi.

Özelliğın geçerli değeri şunlardır:

**Geçerli değeri**

XMSSC\_WPM\_TARGET\_SIGNANCE\_  
Tercih Edilen

XMSSC\_WPM\_TARGET\_SIGNANCE\_  
ZORUNLU

**Anlamı**

Hedef gruptaki bir ileti alışverişi altyapısı varsa, hedef gruptaki bir ileti alışverişi altyapısı seçilir. Ters durumda, aynı hizmet tümleştirme veriyolunda olması koşuluyla, hedef grubun dışındaki bir ileti alışverişi altyapısı seçilirdir.

Seçilen ileti alışverişi altyapısının hedef grupta olması gerekir. Hedef gruptaki bir ileti alışverişi altyapısı kullanılmıyorsa, bağlantı işlemi başarısız olur.

Özelliğın varsayılan değeri XMSSC\_WPM\_TARGET\_SENCEANCE\_PREFER\_PRECIENCE'DIR.

***XMSC\_WPM\_TARGET\_TRANSPORT\_CHAIN***

**Veri tipi:**

Dizgi

**Özellik:**

ConnectionFactory

Uygulamanın bir ileti alışverişi altyapısına bağlanmak için kullanması gereken gelen iletim zincirinin adı.

Özelliğın değeri, ileti alışverişi altyapısını barındıran uygulama sunucusunda var olan herhangi bir gelen iletim zincirinin adı olabilir. Önceden tanımlanmış gelen iletim zincirlerinden biri için aşağıdaki adlandırılmış sabit değeri sağlar:

**Adlandırılmış sabit**

XMSC\_WPM\_TARGET\_TRANSPORT\_CHAIN\_BASIC

**İletim zincirinin adı**

InboundBasicİleti Sistemi

Özelliğın varsayılan değeri XMSSC\_WPM\_TARGET\_TRANSPORT\_CHAIN\_BASIC değeri'dir.

***XMLC\_WPM\_TARGET\_TIP***

**Veri tipi:**

System.Int32

**Özellik:**

ConnectionFactory

İleti alışverişi altyapılarının hedef grubunun tipi. Bu özellik, XMLC\_WPM\_TARGET\_GROUP özelliğı tarafından tanımlanan hedef grubun doğasını belirler.

Özelliğın geçerli değeri şunlardır:

**Geçerli değeri**

XMLC\_WPM\_TARGET\_TYPE\_KULLANICILARI

XMSC\_WPM\_TARGET\_TYPE\_CUSTOM

XML\_WPM\_TARGET\_TYPE\_ME

**Anlamı**

Hedef grubun adı, bir veriyolu üyesinin adıdır. Hedef grup, veriyolu üyesindeki tüm ileti alışverişi altyapıları.

Hedef grubun adı, kullanıcı tanımlı ileti alışverişi altyapıları grubunun adıdır. Hedef grup, kullanıcı tanımlı gruba kayıtlı tüm ileti alışverişi altyapılarıdır.

Hedef grubun adı, bir ileti alışverişi altyapısının adıdır. Hedef grup, belirtilen ileti alışverişi altyapısıdır.

Varsayılan olarak özellik ayarlanmaz.

### ***XMSC\_WPM\_TEMP\_Q\_PREFIX***

**Veri tipi:**

Dizgi

**Özellik:**

ConnectionFactory

Uygulama bir XMS geçici kuyruğu yarattığında hizmet tümleştirme veriyolunda yaratılan geçici kuyruğun adını oluşturmak için kullanılan önek. Önek en fazla 12 karakter içerebilir.

Geçici bir kuyruğun adı, "\_Q" karakteriyle ve ardından önek ile başlar. Adın geri kalan kısmı sistem tarafından oluşturulan karakterlerden oluşur.

Varsayılan olarak, özellik belirlenmez; bu, geçici bir kuyruğun adının bir öneki olmadığı anlamına gelir.

Bu özellik, yalnızca noktadan noktaya iletişim alanında ilişkilidir.

### ***XMSC\_WPM\_TEMP\_TOPIC\_PREFIX***

**Veri tipi:**

Dizgi

**Özellik:**

ConnectionFactory

Uygulama tarafından oluşturulan geçici bir konunun adını oluşturmak için kullanılan önek. Önek en fazla 12 karakter içerebilir.

Geçici bir konunun adı, "\_T" karakterleriyle ve ardından önek ile başlar. Adın geri kalan kısmı sistem tarafından oluşturulan karakterlerden oluşur.

Varsayılan olarak, özellik belirlenmez; bu, geçici bir konunun adının bir öneki olmadığı anlamına gelir.

Bu özellik, yalnızca yayınlama/abone olma etki alanında anlamlıdır.

### ***XMSC\_WPM\_TOPIC\_SPACE***

**Veri tipi:**

Dizgi

**Özellik:**

Hedef

**Bir URI ' de kullanılan ad:**

topicSpace

Konuyu içeren konu alanının adı. Yalnızca konu olan bir hedef bu özelliğe sahip olabilir.

Varsayılan değer olarak, özellik ayarlanmaz, varsayılan konu alanı varsayıldığı anlamına gelir.

Bu özellik, yalnızca yayınlama/abone olma etki alanında anlamlıdır.

## **Managed File Transfer uygulama başvurularını geliştirme**

Reference information to help you develop applications for Managed File Transfer.

### **fteCreateAktarımı programını başlatmak için kullanma örnekleri**

Bir aktarımdan önce ya da sonra çalıştırılacak programları belirlemek için **fteCreateTransfer** komutunu kullanabilirsiniz.

**fteCreateTransfer**' un kullanılmasının yanı sıra, bir programı aktarmadan önce ya da sonra çağırmanın başka yolları da vardır. Daha fazla bilgi için [MFTile çalıştırılacak programları belirleme](#) başlıklı konuya bakın.

Bu örneklerin tümü, bir programı belirtmek için aşağıdaki sözdizimini kullanır:

```
[type:]commandspec[, [retrycount][, [retrywait][, successsrc]]]
```

Bu sözdizimiyle ilgili daha fazla bilgi için bkz. **[fteCreateTransfer](#)**: yeni bir dosya aktarımı başlatın.

### Yürütülür programın çalıştırılması

Aşağıdaki örnek, mycommand adlı yürütülebilir bir programı belirtir ve programa iki bağımsız değişken ( a ve b) geçer.

```
mycommand(a,b)
```

Aktarma işlemi başlamadan önce bu programı kaynak aracıda AGENT1 kaynak aracısında çalıştırmak için aşağıdaki komutu kullanın:

```
fteCreateTransfer -sa AGENT1 -da AGENT2 -presrc mycommand(a,b)  
destinationSpecification sourceSpecification
```

### Yürütülebilir bir program çalıştırılıyor ve yeniden deniyor

Aşağıdaki örnek, herhangi bir bağımsız değişken almayan simpleadlı yürütülebilir bir programı belirtir. retrycount için 1 değeri belirlendi ve retrywait için 5 değeri belirtildi. Bu değerler, beş saniye bekledikten sonra, başarılı bir dönüş kodu döndürmezse, bu programın yeniden deneneceği anlamına gelir. successsrc için değeri belirtilmedi; bu nedenle, tek başarılı dönüş kodu varsayılan değeri olan 0 değeridir.

```
executable:simple,1,5
```

Aktarma işlemi tamamlandıktan sonra bu programı kaynak aracıda AGENT1 kaynak aracısında çalıştırmak için aşağıdaki komutu kullanın:

```
fteCreateTransfer -sa AGENT1 -da AGENT2 -postsrc executable:simple,1,5  
destinationSpecification sourceSpecification
```

### Bir Ant komut dosyasını çalıştırma ve başarılı dönüş kodları belirtme

The following example specifies an Ant script called myscript and passes two properties to the script. Komut dosyası, **fteAnt** komutu kullanılarak çalıştırılır. successsrc değeri, 0, 3, 4, 6 ve 14 dönüş kodlarının başarılı olduğunu belirten >2&<7&!5|0|14 olarak belirtilir.

```
antscript:myscript(prop1=fred,prop2=bob),,,>2&<7&!5|0|14
```

Aktarma işlemi başlamadan önce bu programı hedef aracıda AGENT2 hedef aracısında çalıştırmak için aşağıdaki komutu kullanın:

```
fteCreateTransfer -sa AGENT1 -da AGENT2 -predst  
"antscript:myscript(prop1=fred,prop2=bob),,,>2&<7&!5|0|14"destinationSpecification sourceSpecification
```

### Bir Ant komut dosyasının çalıştırılması ve çağrılacak hedefleri belirtme

The following example specifies an Ant script called script2 and two targets, target1 and target2, to call. prop1 özelliği de, recmfm(F,B) değeriyle birlikte iletilir. Bu değerdeki virgül (,) ve parantezler, ters eğik çizgi karakteri (\) kullanılarak kaçılır.

```
antscript:script2(target1,target2,prop1=recmfm(F,B\)),,,>2&<7&!5|0|14
```

Aktarma işlemi tamamlandıktan sonra bu programı hedef aracıda AGENT2 hedef aracısında çalıştırmak için aşağıdaki komutu kullanın:

```
fteCreateTransfer -sa AGENT1 -da AGENT2
-postdst "antscript:script2(target1,target2,prop1=recmfm\F\B\),,,>2&<7&!5|0|14"
destinationSpecification sourceSpecification
```

### Ant komut dosyasında meta verileri kullanma

Bir aktarma işlemi için aşağıdaki çağrılar herhangi biri olarak bir Ant görevi belirtebilirsiniz:

- ön kaynak
- kaynak gönderi
- ön hedef
- gönderi hedefi

Ant görevi çalıştırıldığında, ortam değişkenleri kullanılarak aktarımın kullanıcı meta verileri kullanılabilir olur. Örneğin, aşağıdaki kodu kullanarak bu verilere erişebilirsiniz:

```
<property environment="environment" />
<echo>${environment.mymetadata}</echo>
```

Burada mymetadata , aktarıma eklenen bazı meta verilerin adıdır.

### JCL komut dosyası çalıştırılıyor

Aşağıdaki örnekte, ZOSBATCH'lı bir JCL komut dosyası belirtilmektedir. retrycount için 3 değeri belirtildi, retrywait için 30 değeri belirlendi ve successrc için 0 değeri belirtildi. Bu değerler, komut dosyasının başarılı bir dönüş kodu döndürmemesi durumunda üç kez yeniden denendiği anlamına gelir. Bu, her bir deneme için otuz saniye beklemeyle birlikte, 0 'a kadar başarılı bir dönüş kodu döndürmez.

```
jcl:ZOSBATCH,3,30,0
```

where ZOSBATCH is a member of a PDS called MYSYS.JCL, and the agent.properties file contains the line commandPath=...:/'MYSYS.JCL':...

Aktarma işlemi tamamlandıktan sonra bu programı kaynak aracıda AGENT1 kaynak aracısında çalıştırmak için aşağıdaki komutu kullanın:

```
fteCreateTransfer -sa AGENT1 -da AGENT2 -postsrc jcl:ZOSBATCH,3,30,0
destinationSpecification sourceSpecification
```

### İlgili görevler

[MFT ile çalışmak üzere programların belirtilmesi](#)

### İlgili başvurular

[fteCreateTransfer: yeni bir dosya aktarımı başlatır](#)

## fteAnt: run Ant tasks in MFT

The **fteAnt** command runs Ant scripts in an environment that has Managed File Transfer Ant tasks available. Standart **ant** komutunun aksine, **fteAnt** , bir komut dosyası tanımlamanızı gerektirir.

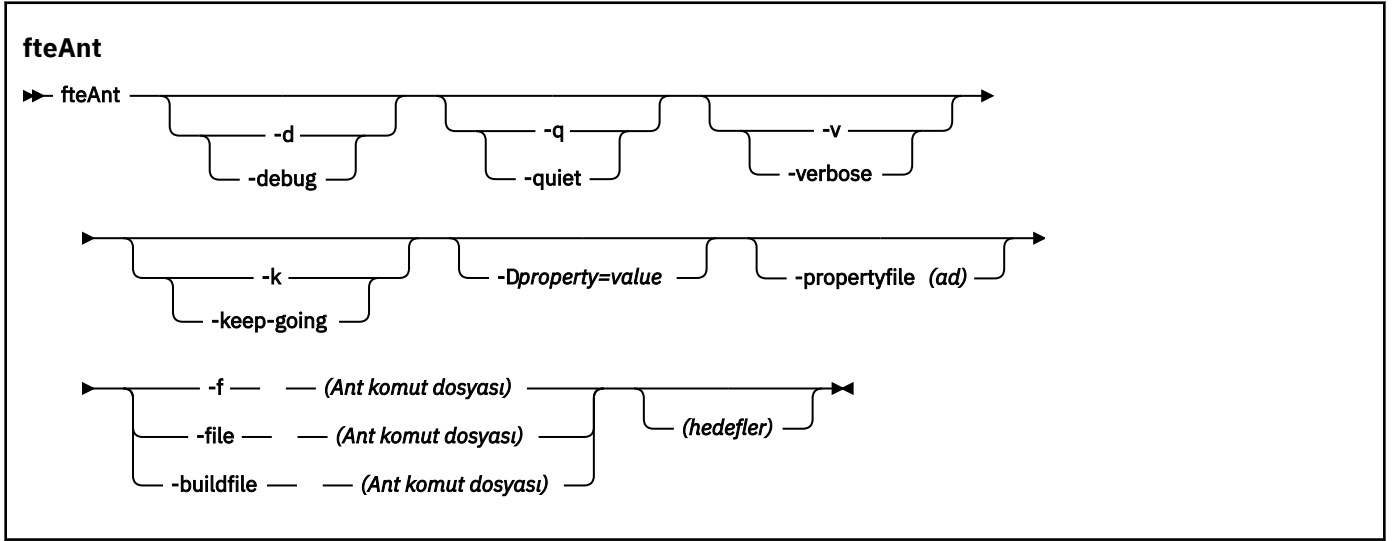
### MFT Ant görevleri ve iç içe yerleşimli parametreler

Managed File Transfer , dosya aktarım yeteneklerine erişmek için kullanabileceğiniz bir dizi Ant görevi sağlar. Ayrıca, kullanılabilir bir iç içe yerleşimli parametreler kümesi de vardır; bu parametreler, sağlanan Ant görevlerinin birkaç tanesi boyunca ortak olan iç içe yerleştirilmiş öge kümelerini açıklar.

**fteAnt** komutunun sözdizimi, parametreleri, kullanım örneği ve dönüş kodları bu konunun geri kalan bölümüne açıklanmıştır. MFT tarafından sağlanan Ant görevlerine ve iç içe yerleşimli parametrelere ilişkin ayrıntılar için: altbaşlıklara bakın.



## fteAnt sözdizimi



### Parametreler

#### -debug ya da

İsteğe Bağlı. Hata ayıklama çıktısı oluşturur.

#### -sessiz ya da -q

İsteğe Bağlı. En düşük çıkış oluşturur.

#### -ayrıntılı ya da -v

İsteğe Bağlı. Ayrıntılı çıkış oluşturur.

#### -devam et ya da -k

İsteğe Bağlı. Başarısız hedeflere bağlı olmayan tüm hedefleri yürütün.

#### -D özellik=değer

İsteğe Bağlı. Belirli bir *özellik* için *değer* değerini kullanın. **-D** ile ayarlanan özellikler, özellikler dosyasındaki ayarlardan öncelikli olarak uygulanır.

Ant görevleri için kullanılan yapılandırma seçenekleri kümesini belirtmek için

**com.ibm.wmqfte.propertyset** özelliğini kullanın. Bu özelliğin değeri olarak, varsayılan olmayan bir koordinasyon kuyruğu yöneticisi adını kullanın. Ant görevleri, daha sonra, bu varsayılan olmayan koordinasyon kuyruğu yöneticisiyle ilişkilendirilmiş yapılandırma seçenekleri kümesini kullanır. Bu özelliği belirtmezseniz, varsayılan koordinasyon kuyruğu yöneticisini temel alan varsayılan yapılandırma seçenekleri kümesi kullanılır. Bir Ant görevi için **cmdqm** özniteliğini belirtirseniz, bu öznitelik, **fteAnt** komutu için belirtilen yapılandırma seçenekleri kümesine göre öncelik kazanır. Bu davranış, varsayılan yapılandırma seçenekleri kümesini kullanıp kullanmadığınız ya da **com.ibm.wmqfte.propertyset** özelliği ile bir küme belirtme özelliğinin geçerli olup olmamasından bağımsız olarak uygulanır.

#### -propertyfile (ad)

İsteğe Bağlı. Load all properties from a file with **-D** properties taking precedence.

#### -f (Ant komut dosyası), -file (Ant komut dosyası) ya da -buildfile (Ant komut dosyası)

Gereklidir. Çalıştırılacak Ant komut dosyasının adını belirtir.

#### hedefler

İsteğe Bağlı. Ant komut dosyasından çalıştırılacak bir ya da daha fazla hedefin adı. Bu parametre için bir değer belirtmezseniz, komut dosyası için varsayılan hedef çalıştırılır.

#### -version

İsteğe Bağlı. Managed File Transfer komutunu ve Ant sürümlerini görüntüler.

#### -Ne? ya da -h

İsteğe Bağlı. Komut sözdizimini görüntüler.

## Örnek

In this example, the target **copy** in Ant script `fte_script.xml` is run and the command writes debugging output to standard out.

```
fteAnt -d -f fte_script.xml copy
```

## Dönüş kodları

**0**

Komut başarıyla tamamlandı.

**1**

Komut başarısız oldu.

Diğer durum dönüş kodları da Ant komut dosyalarından da belirtilebilir; örneğin, Ant başarısız görevi kullanılarak.

Daha fazla bilgi için bkz. [Başarısız](#).

## fte: awaitsonucun Ant görevi

Waits for a **fte:filecopy**, **fte:filemove**, or **fte:call** operation to complete.

## Öznitelikler

### tanıtıcı

Gereklidir. Bir sonucun bekleneceği aktarımı tanımlar. Genellikle, bu özellik, [fte:filecopy](#), [fte:filemove](#)ya da [fte:call](#) görevlerinin `idProperty` öznelimesiyle ayarlanır.

### rcproperty

Gereklidir. **fte:awaitoutcome** görevinin dönüş kodunu saklamak için bir özellik adı verir.

### zaman aşımı

İsteğe Bağlı. İşlemin tamamlanmak üzere beklenmesi için saniye cinsinden süre üst sınırı.

Zamanaşımı alt sınırı bir saniyedir. Bir zamanaşımı değeri belirtmezseniz, **fte:awaitoutcome** görevi işlemin sonucunun belirlenmesini sonsuza kadar bekler.

## Örnek

Bu örnekte bir dosya kopyası başlatılır ve tanıtıcısı `copy.id` özelliğinde saklanır. Kopyalama işlemi devam ederken, diğer işlemler de gerçekleşebilir. The **fte:awaitoutcome** statement is used to wait until the copy operation completes. **fte:awaitoutcome** deyimini, `copy.id` özelliğinde saklanan tanıtıcıyı kullanmak için hangi işlemin bekleneceğini belirler. **fte:awaitoutcome**, kopyalama işleminin sonucunu `copy.result` adlı bir özelliğe belirten bir dönüş kodu saklar.

```
<!-- issue a file copy request -->
<fte:filecopy
  src="AGENT1@QM1"
  dst="AGENT2@QM2"
  idproperty="copy.id"
  outcome="defer">

  <fte:filespec
    srcfilespec="/home/fteuser1/file.bin"
    dstdir="/home/fteuser2"/>

  </fte:filecopy>

  <fte:awaitoutcome id="{copy.id}" rcProperty="copy.rc"/>

  <echo>Copy id={copy.id} rc={copy.rc}</echo>
```

## İlgili görevler

[Apache Ant ile MFTkomutunu kullanma](#)

## fte: Ant görevini çağır

Komut dosyalarını ve programları uzaktan çağırmak için **fte:call** görevini kullanabilirsiniz.

Bu görev, bir aracıya **fte:call** isteği göndermenizi sağlar. Aracı, bir komut dosyası ya da program çalıştırarak ve sonucu döndürerek bu isteği işler. Çağrıya verilen komutların aracıya erişilebilir olması gerekir. `agent.properties` dosyasındaki `commandPath` özellik değerinin, çağrılacak komutların yerini içerdiğinden emin olun. Komut içiçe yerleştirilmiş öge tarafından belirtilen yol bilgilerinin, `commandPath` özelliği tarafından belirlenen konumlarla görelili olması gerekir. Varsayılan olarak `commandPath`, aracının herhangi bir komut çağırılmayacağı için boştur. Bu özellik hakkında daha fazla bilgi için bkz. [commandPath MFT property](#).

`agent.properties` dosyası hakkında daha fazla bilgi için bkz. [MFT agent.properties dosyası](#).

## Öznitelikler

### aracı

Gereklidir. **fte:call** isteğinin gönderileceği aracıyı belirtir. Aracı bilgilerini şu biçimde belirtin: `agentname@qmgrname`; burada `agentname`, aracının adıdır ve `qmgrname`, bu aracının doğrudan bağlı olduğu kuyruk yöneticisinin adıdır.

### cmdqm

İsteğe Bağlı. İsteği teslim etmek için komut kuyruğu yöneticisi. Bu bilgileri `qmgrname@host@port@channel` biçiminde belirtin; burada:

- `qmgrname`, kuyruk yöneticisinin adıdır.
- `host`, kuyruk yöneticisinin çalıştırıldığı sistemin isteğe bağlı anasistem adıdır.
- `port`, kuyruk yöneticisinin dinlediği isteğe bağlı kapı numarasıdır.
- `channel`, kullanılacak isteğe bağlı SVRCONN kanalına

Komut kuyruğu yöneticisi için `host`, `port` ya da `channel` bilgilerini çıkarırsanız, `command.properties` dosyasında belirtilen bağlantı bilgileri kullanılır.



**Uyarı:** Şu öge için bir değer belirtilmediyse:

- `anasistem` değişkeni, bağ tanımları kipi kullanılır
- `port` değişkeni, 1414 değeri kullanılır
- `kanal` değişkeni, SYSTEM.DEF.SVRCONN kullanılır.

Ek bilgi için [MFT command.properties dosyası](#) başlıklı konuya bakın.

Ancak, ortadaki öznitelikleri atlayamazsınız; örneğin, `qmgrname@host@@channel`. Sahip olabilirsiniz; örneğin, `qmgrname@host` ya da `qmgrname@host@port` ya da `qmgrname@hostport@@channel`.

MFT, belirtilen özniteliği @ sınırlayıcısını kullanarak böler. Bulunan simgelerin sayısına bağlı olarak, ilk simgeyi `qmgrname`, ikincisini `anasistem`, üçüncüsü `kapı` ve son olarak da `kanal` olarak alır.

Daha fazla bilgi için [MFT command.properties dosyası](#) başlıklı konuya bakın.

You can use the **com.ibm.wmqfte.propertySet** property to specify which `command.properties` file to use. Daha fazla bilgi için bkz. [com.ibm.wmqfte.propertySet](#).

`cmdqm` özniteliğini kullanmayacaksa, bu özellik ayarlandıysa, görev varsayılan olarak `com.ibm.wmqfte.ant.commandQueueManager` özelliğini kullanır. `com.ibm.wmqfte.ant.commandQueueManager` özelliği ayarlanmadıysa, `command.properties` dosyasında tanımlı olan varsayılan kuyruk yöneticisine yönelik bir bağlantı denenir. `com.ibm.wmqfte.ant.commandQueueManager` özelliğinin biçimi, `cmdqm` öznitelimesiyle aynıdır; bu, `qmgrname@host@port@channel`.

### idproperty

Optional unless you have specified an outcome of `defer`. Aktarma tanıtıcısını atamak için bir özelliğin adını belirtir. Aktarma tanıtıcıları, aktarma isteği sunulmakta ve aktarma işleminin ilerleyişini

izlemek, sorunları bir aktarma ile tanılamak ve aktarma işlemini iptal etmek için aktarma tanıtıcılarını kullanabilirsiniz.

`ignore`' un `outcome` özelliğini de belirtmiş bulursanız, bu özelliği belirtemezsiniz. Ancak, `defer`' un `outcome` özelliğini de belirtmiş olmanız durumunda, `idproperty` belirtmeniz gerekir.

### **jobName**

İsteğe Bağlı. **fte:call** isteğine bir iş adı atar. Mantıksal aktarımlar grupları oluşturmak için iş adlarını kullanabilirsiniz. Sözde benzersiz iş adları oluşturmak için “`fte: uuid Ant görevi`” [sayfa 2083](#) görevini kullanın. `jobname` özneliğini kullanmayarsanız, bu özellik ayarlandıysa, görevin varsayılan değer olarak `com.ibm.wmqfte.ant.jobName` özellik değerini kullanması gerekir. Bu özelliği ayarlamadıysanız, **fte:call** isteğiyle hiçbir iş adı ilişkilendirilemez.

### **ilk kullanıcı**

İsteğe Bağlı. **fte:call** isteğiyle ilişkilendirilecek kaynak kullanıcı kimliğini belirtir. `originuser` özneliğini kullanmayarsanız, Ant komut dosyasını çalıştırmak için kullanılan kullanıcı kimliğini kullanarak varsayılan olarak görev varsayılan değerini kullanır.

### **sonuç**

İsteğe Bağlı. Determines whether the task waits for the **fte:call** operation to complete before returning control to the Ant script. Aşağıdaki seçeneklerden birini belirleyin:

#### **bekliyor**

Görev, geri dönmeye önce **fte:call** işleminin tamamlanmasını bekler. `await` için `outcome` belirtildiğinde, `idproperty` özneliği isteğe bağlıdır.

#### **Ertele**

Görev, **fte:call** isteği sunulmaz ve çağrı işleminin sonucunun daha sonra uyuyacağı ya da `ignore` sonuçları görevleri kullanılarak ele alınması varsayıldığında geri döner. When an outcome of `defer` is specified the `idproperty` attribute is required.

#### **yoksay**

**fte:call** işleminin sonucu önemli değilse, `ignore` için bir değer belirtebilirsiniz. Daha sonra görev, komutun sonucunu izlemek için herhangi bir kaynak ayrılmadan **fte:call** isteği gönderildikten hemen sonra geri döner. When an outcome of `ignore` is specified the `idproperty` attribute cannot be specified.

Sonuç özneliğini belirtmezseniz, görev varsayılan olarak `await` değerini kullanacak şekilde ayarlanır.

### **rcproperty**

İsteğe Bağlı. **fte:call** isteğinin sonuç kodunu atamak için bir özelliğin adını belirtir. Sonuç kodu, **fte:call** isteğinin genel sonucunu yansıtır.

You cannot specify this property if you have also specified an outcome property of `ignore` or `defer`. Ancak, `await` sonucunun bir sonucunu belirtmiş olmanız durumunda `rcproperty` değerini belirtmeniz gerekir.

## **İç içe geçmiş öğeler olarak belirtilen parametreler**

### **fte: komut**

Aracı tarafından çağrılacak komutu belirtir. Tek bir `fte: command` öğesini yalnızca belirli bir **fte:call** işlemi ile ilişkilendirebilirsiniz. Çağrılacak komut, aracının `agent.properties` dosyasındaki `commandPath` özelliği tarafından belirtilen yolda bulunmalıdır.

### **fte: metadata**

Arama işlemiyle ilişkilendirilecek meta verileri belirtebilirsiniz. Bu meta veri, çağrı işlemi tarafından oluşturulan günlük iletilerine kaydedilir. Belirli bir aktarım öğesiyle yalnızca tek bir meta veri bloğunu ilişkilendirebilirsiniz; ancak bu blok birçok meta veri parçası içerebilir.

### **Örnek**

This example shows how to call a command at AGENT1 running on queue manager QM1. Çağrılacak komut, `command.shkomut` dosyasıdır ve komut dosyası, `xyz`' un tek bir bağımsız değişkeniyle çağrılır.

command.sh komutu, aracının agent.properties dosyasında commandPath özelliği tarafından belirtilen yolda bulunur.

```
<fte:call cmdqm="QM0@localhost@1414@SYSTEM.DEF.SVRCONN"
  agent="AGENT1@QM1"
  rcproperty="call.rc"
  origuser="bob"
  jobname="{job.id}">

  <fte:command command="command.sh" successrc="1" retrycount="5" retrywait="30">
    <fte:arg value="xyz" />
  </fte:command>

  <fte:metadata>
    <fte:entry name="org.foo.accountName" value="BDG3R" />
  </fte:metadata>

</fte:call>
```

## İlgili görevler

[Apache Ant ile MFTkomutunu kullanma](#)

## fte: Ant görevini iptal et

Managed File Transfer tarafından yönetilen bir aktarımı ya da yönetilen aramayı iptal eder. A managed transfer might have been created using the **fte:filecopy** or **fte:filemove** tasks. A managed call might have been created using the **fte:call** task.

## Öznitelikler

### aracı

Gereklidir. **fte:cancel** isteğinin gönderileceği aracıyı belirtir. Değer şu biçimdeki biçimdir: *agentname@qmgrname* ; burada *agentname* aracının adı ve *qmgrname* bu aracının doğrudan bağlı olduğu kuyruk yöneticisinin adıdır.

### cmdqm

İsteğe Bağlı. İsteği teslim etmek için komut kuyruğu yöneticisi. Bu bilgileri *qmgrname@host@port@channel* biçiminde belirtin; burada:

- *qmgrname* , kuyruk yöneticisinin adıdır.
- *host* , kuyruk yöneticisinin çalıştırıldığı sistemin isteğe bağlı anasistem adıdır.
- *port* , kuyruk yöneticisinin dinlediği isteğe bağlı kapı numarasıdır.
- *channel* , kullanılacak isteğe bağlı SVRCONN kanalına

Komut kuyruğu yöneticisi için *host*, *port*ya da *channel* bilgilerini çıkarırsanız, *command.properties* dosyasında belirtilen bağlantı bilgileri kullanılır.



**Uyarı:** Şu öge için bir değer belirtilmediyse:

- *anasistem* değişkeni, bağ tanımları kipi kullanılır
- *port* değişkeni, 1414 değeri kullanılır
- *kanal* değişkeni, SYSTEM.DEF.SVRCONN kullanılır.

Ek bilgi için [MFT command.properties dosyası](#) başlıklı konuya bakın.

Ancak, ortadaki öznitelikleri atlayamazsınız; örneğin, *qmgrname@host@@channel*. Sahip olabilirsiniz; örneğin, *qmgrname@host*ya da *qmgrname@host@port*ya da *qmgrname@hostport@@channel*.

MFT , belirtilen özniteliği @ sınırlayıcısını kullanarak böler. Bulunan simgelerin sayısına bağlı olarak, ilk simgeyi *qmgrname*, ikincisini *anasistem*, üçüncüsü *kapı* ve son olarak da *kanal*olarak alır.

Daha fazla bilgi için [MFT command.properties dosyası](#) başlıklı konuya bakın.

You can use the **com.ibm.wmqfte.propertySet** property to specify which `command.properties` file to use. Daha fazla bilgi için bkz. `com.ibm.wmqfte.propertySet`.

`cmdqmc` özneliğini kullanmayacaksa, bu özellik ayarlandıysa, görev varsayılan olarak `com.ibm.wmqfte.ant.commandQueueManager` özneliğini kullanır. `com.ibm.wmqfte.ant.commandQueueManager` özneliği ayarlanmadıysa, `command.properties` dosyasında tanımlı olan varsayılan kuyruk yöneticisine yönelik bir bağlantı denir. `com.ibm.wmqfte.ant.commandQueueManager` özneliğinin biçimi, `cmdqmc` öznelimesiyle aynıdır; bu, `qmgrname@host@port@channel`.

### tanıtıcı

Gereklidir. İptal etmek için aktarmanın aktarma tanıtıcısını belirtir. Aktarma tanıtıcıları, aktarma isteğinin hem `fte:filecopy` hem de `fte:filemove` görevleri tarafından sunulmakta olduğu noktada oluşturulur.

### başlangıç kullanıcı

İsteğe Bağlı. **cancel** isteğiyle ilişkilendirilecek kaynak kullanıcı kimliğini belirtir. `originser` özneliği kullanılmazsa, görev varsayılan olarak Ant komut dosyasını çalıştırmak için kullanılan kullanıcı kimliğini kullanır.

### Örnek

The example sends a **fte:cancel** request to the command queue manager `qm0`. The **fte:cancel** request is targeted at `agent1` on queue manager `qm1` for the transfer identifier populated by the `transfer.id` variable. İstek, "bob" kullanıcı kimliği kullanılarak çalıştırılır.

```
<fte:cancel cmdqmc="qm0@localhost@1414@SYSTEM.DEF.SVRCONN"
  agent="agent1@qm1"
  id="{transfer.id}"
  origuser="bob"/>
```

### İlgili görevler

[Apache Ant ile MFTkomutunu kullanma](#)

## fte:filecopy Ant görevi

**fte:filecopy** görevi, dosyaları Managed File Transfer araçları arasında kopyalar. Dosya, kaynak araçtan silinmez.

## Öznelikler

### cmdqmc

İsteğe Bağlı. İsteği teslim etmek için komut kuyruğu yöneticisi. Bu bilgileri `qmgrname@host@port@channel` biçiminde belirtin; burada:

- `qmgrname` , kuyruk yöneticisinin adıdır.
- `host` , kuyruk yöneticisinin çalıştırıldığı sistemin isteğe bağlı anasistem adıdır.
- `port` , kuyruk yöneticisinin dinlediği isteğe bağlı kapı numarasıdır.
- `channel` , kullanılacak isteğe bağlı SVRCONN kanalına

Komut kuyruğu yöneticisi için `host`, `port`ya da `channel` bilgilerini çıkarırsanız, `command.properties` dosyasında belirtilen bağlantı bilgileri kullanılır.



**Uyarı:** Şu öge için bir değer belirtilmediyse:

- `anasistem` değişkeni, bağ tanımları kipi kullanılır
- `port` değişkeni, 1414 değeri kullanılır
- `kanal` değişkeni, SYSTEM.DEF.SVRCONN kullanılır.

Ek bilgi için [MFT command.properties dosyası](#) başlıklı konuya bakın.

Ancak, ortadaki öznitelikleri atlayamazsınız; örneğin, `qmgrname@host@@channel`. Sahip olabilirsiniz; örneğin, `qmgrname@hostya` da `qmgrname@host@portya` da `qmgrname@hostport@@channel`.

MFT , belirtilen özniteliği @ sınırlayıcısını kullanarak böler. Bulunan simgelerin sayısına bağlı olarak, ilk simgeyi *qmgrname*, ikincisini *anasistem*, üçüncüsü *kapı* ve son olarak da *kanal* olarak alır.

Daha fazla bilgi için [MFT command.properties](#) dosyasıbaşlıklı konuya bakın.

You can use the **com.ibm.wmqfte.propertySet** property to specify which `command.properties` file to use. Daha fazla bilgi için bkz. [com.ibm.wmqfte.propertySet](#).

`cmdqmc` özniteliğini kullanmayacaksa, bu özellik ayarlandıysa, görev varsayılan olarak `com.ibm.wmqfte.ant.commandQueueManager` özelliğini kullanır. `com.ibm.wmqfte.ant.commandQueueManager` özelliği ayarlanmadıysa, `command.properties` dosyasında tanımlı olan varsayılan kuyruk yöneticisine yönelik bir bağlantı denenir. `com.ibm.wmqfte.ant.commandQueueManager` özelliğinin biçimi, `cmdqmc` öznitelimesiyle aynıdır; bu, `qmgrname@host@port@channel`.

#### **dst**

Gereklidir. Kopyalama işlemine ilişkin hedef aracıyı belirtir. Bu bilgileri şu biçimde belirtin: `agentname@qmgrname` ; burada `agentname` hedef aracısının adıdır ve `qmgrname` , bu aracının doğrudan bağlı olduğu kuyruk yöneticisinin adıdır.

#### **idproperty**

Optional unless you have specified an outcome of `defer`. Aktarma tanıtıcısını atamak için bir özelliğin adını belirtir. Aktarma tanıtıcıları, aktarma isteği sunulmakta ve aktarma işleminin ilerleyişini izlemek, sorunları bir aktarma ile tanılamak ve aktarma işlemini iptal etmek için aktarma tanıtıcılarını kullanabilirsiniz.

`ignore`' un outcome özelliğini de belirtmiş bulursanız, bu özelliği belirtemezsiniz. Ancak, `defer`' un outcome özelliğini de belirtmiş olmanız durumunda, `idproperty` belirtmeniz gerekir.

#### **jobName**

İsteğe Bağlı. Kopyalama isteğine bir iş adı atar. Mantıksal aktarımlar grupları oluşturmak için iş adlarını kullanabilirsiniz. Sözde benzersiz iş adları oluşturmak için "[fte: uuid Ant görevi](#)" sayfa 2083 görevini kullanın. `jobname` özniteliğini kullanmayarsanız, bu özellik ayarlandıysa, görevin varsayılan değer olarak `com.ibm.wmqfte.ant.jobName` özellik değerini kullanması gerekir. Bu özellik ayarlamadıysanız, kopyalama isteğiyle ilişkilendirilmiş bir iş adı yok.

#### **ilk kullanıcı**

İsteğe Bağlı. Kopyalama isteğiyle ilişkilendirilecek kaynak kullanıcı kimliğini belirtir. `originuser` özniteliğini kullanmayacaksa, görev varsayılan olarak Ant komut dosyasını çalıştırmak için kullanılan kullanıcı kimliğini kullanır.

#### **sonuç**

İsteğe Bağlı. Denetimi Ant komut dosyasına geri döndürmeden önce, kopyalama işleminin tamamlanıp tamamlanmayacağını belirler. Aşağıdaki seçeneklerden birini belirleyin:

##### **bekliyor**

Görev, geri dönmeye başlamadan önce kopyalama işleminin tamamlanmasını bekler. `await` için outcome belirtildiğinde, `idproperty` özniteliği isteğe bağlıdır.

##### **Ertele**

Bu görev, kopyalama isteği gönderilip gönderilmez ve kopyalama işleminin sonucunun "[fte: awaitsonucun Ant görevi](#)" sayfa 2070 ya da "[fte: ignoreoutcome Ant görevi](#)" sayfa 2081 görevleri kullanılarak daha sonra ele alınması sonucuna vardığı varsayıldığında geri döner. When an outcome of `defer` is specified the `idproperty` attribute is required.

##### **yoksay**

Kopyalama işleminin sonucu önemli değilse, `ignore` için bir değer belirtebilirsiniz. Daha sonra, görev, aktarımın sonucunu izlemek için herhangi bir kaynak ayrılmadan, kopyalama isteği teslim edildikten sonra geri döner. When an outcome of `ignore` is specified the `idproperty` attribute cannot be specified.

Sonuç özniteliğini belirtmezseniz, görev varsayılan olarak `await` değerini kullanacak şekilde ayarlanır.

### öncelik

İsteğe Bağlı. Kopyalama isteğiyle ilişkilendirilecek önceliği belirtir. Genel olarak, daha yüksek öncelikli aktarma istekleri, düşük öncelikli isteklere göre öncelikli olarak uygulanır. Öncelik değeri 0-9 aralığında (bu değerler de içinde olmak üzere) olmalıdır. 0 öncelik değeri, en düşük öncelik ve 9 değerinin en yüksek önceliğe sahip olduğu bir değerdir. `priority` özniteliğini belirtmezseniz, aktarma varsayılan olarak 0 önceliğine ayarlanır.

### rcproperty

İsteğe Bağlı. Kopyalama isteğinin sonuç kodunu atamak için bir özelliğin adını belirtir. Sonuç kodu, kopyalama isteğinin genel sonucunu yansıtır.

You cannot specify this property if you have also specified an outcome property of `ignore` or `defer`. Ancak, `awaitsonucunu` belirtirseniz, `rcproperty` değerini belirtmeniz gerekir.

### transferRecoveryZamanaşımı

İsteğe Bağlı. Bir kaynak aracının durdurulan bir dosya aktarımı kurtarmaya çalıştığı süreyi saniye cinsinden belirler. Aşağıdaki seçeneklerden birini belirleyin:

#### -1

Aracı, aktarma tamamlanıncaya kadar, durdurulan aktarımı kurtarmayı denemeye devam eder. Bu seçeneği kullanmak, özellik ayarlanmadığında aracının varsayılan davranışına eşdeğerdir.

#### 0

Aracı, kurtarma işlemini girer girmez dosya aktarımlarını durdurur.

#### >0

Aracı, belirtilen pozitif tamsayı değeri tarafından belirlenen süre saniye cinsinden, durdurulan aktarım işlemini kurtarma girişiminde bulunmaya devam eder. Örneğin,

```
<fte:filecopy cmdqm="qm0@localhost@1414@SYSTEM.DEF.SVRCONN"
  src="agent1@qm1" dst="agent2@qm2"
  rcproperty="copy.result" transferRecoveryTimeout="21600">
  <fte:filespec srcfilespec="/home/fteuser1/file.bin" dstfile="/home/fteuser2/
file.bin"/>
</fte:filecopy>
```

aracının, iyileşmeye girdiği zamandan itibaren 6 saat boyunca transferi kurtarmaya çalıştığıda olduğunu belirtir. Bu öznitelik için üst sınır değeri '999999999' dir.

Aktarma kurtarma zamanlaşımı değerinin bu şekilde belirtilmesi, aktarma temelinde bu değeri belirler. Bir Managed File Transfer ağındaki tüm aktarımlar için genel bir değer ayarlamak üzere, [Aktarma kurtarma zamanlaşımı özellikleri](#)' ne bir özellik ekleyebilirsiniz. Daha fazla bilgi için bakınız: [Timeout option for transfers in recovery](#).

### src

Gereklidir. Kopyalama işlemine ilişkin kaynak aracıyı belirtir. Bu bilgileri şu biçimde belirtin: `agentname@qmgrname` ; burada `agentname` kaynak aracısının adı ve `qmgrname` , bu aracının doğrudan bağlı olduğu kuyruk yöneticisinin adıdır.

## İç içe geçmiş öğeler olarak belirtilen parametreler

### fte: filespec

Gereklidir. Kopyalanacak dosyaları tanımlayan en az bir dosya belirtimi belirlemelisiniz. Gerekirse birden çok dosya belirtimi belirtebilirsiniz. Ek bilgi için [“fte: filespec Ant içiçe yerleştirilmiş öğe” sayfa 2084](#) başlıklı konuya bakın.

### fte: meta veri

Kopyalama işlemiyle ilişkilendirilecek meta verileri belirtebilirsiniz. Bu meta veri aktarımla birlikte taşınır ve aktarma tarafından oluşturulan günlük iletilerine kaydedilir. Belirli bir aktarım ögesiyle yalnızca tek bir meta veri bloğunu ilişkilendirebilirsiniz; ancak bu blok birçok meta veri parçası içerebilir. Ek bilgi için [fte: metadata](#) başlıklı konuya bakın.



**fte: prinrc**

Aktarma işlemi başlamadan önce, kaynak aracıda yer alacak bir program çağrısını belirtir. Tek bir fte : presrc ögesini yalnızca belirli bir aktarımla ilişkilendirebilirsiniz. Ek bilgi için [program çağırma](#) konusuna bakın.

**fte: predst**

Aktarma işlemi başlamadan önce hedef aracıda yer alacak bir program çağrısını belirtir. Tek bir fte : predst ögesini yalnızca belirli bir aktarımla ilişkilendirebilirsiniz. Ek bilgi için [program çağırma](#) konusuna bakın.

**fte: postsrc**

Aktarma tamamlanınca, kaynak aracıda yer alacak bir program çağrısını belirtir. Tek bir fte : postsrc ögesini yalnızca belirli bir aktarımla ilişkilendirebilirsiniz. Ek bilgi için [program çağırma](#) konusuna bakın.

**fte: postdst**

Aktarma tamamlanınca hedef aracıda yer alacak bir program çağrısını belirtir. Tek bir fte : postdst ögesini yalnızca belirli bir aktarımla ilişkilendirebilirsiniz. Ek bilgi için [program çağırma](#) konusuna bakın.

fte:presrc, fte:predst, fte:postsrc, fte:postdst ve çıkışlar başarılı bir durum döndürmezse, kurallar belirtilen sırada aşağıdaki gibidir:

1. Kaynak başlatma çıkışlarını çalıştırın. Kaynak başlatma çıkışları başarısız olursa, aktarma başarısız olur ve başka bir işlem başlatılamaz.
2. Kaynak öncesi aramayı çalıştırın (varsa). Kaynak öncesi arama başarısız olursa, aktarma başarısız olur ve başka bir şey çalıştırılmaz.
3. Hedef başlangıç çıkışlarını çalıştırın. Hedef başlangıç çıkışları aktarma başarısız olursa, aktarma başarısız olur ve başka bir işlem yürütülemez.
4. Hedef çağrı öncesi aramayı çalıştırın (varsa). Hedef öncesi arama başarısız olursa, aktarma başarısız olur ve başka bir şey çalıştırılmaz.
5. Dosya aktarımlarını gerçekleştir.
6. Hedef uç çıkışlarını çalıştırın. Bu çıkışlar için herhangi bir arıza durumu yok.
7. Aktarma başarılı olursa (bazı dosyalar başarıyla aktarılırsa, başarılı olarak değerlendirilir) hedef sonrası çağrıyı (varsa) çalıştırın. Hedef sonrası arama başarısız olursa, aktarma başarısız olur.
8. Kaynak uç çıkışlarını çalıştırın. Bu çıkışlar için herhangi bir arıza durumu yok.
9. Aktarım başarılı olursa, kaynak sonrası arama (varsa) çalıştırılır. Kaynak sonrası arama başarısız olursa, aktarma başarısız olur.

**Örnekler**

Bu örnek, agent1 ile agent2 arasında bir temel dosya aktarımı gösterir. Kütük aktarımı başlatmak için kullanılan komut, istemci aktarım kipi bağlantısı kullanılarak qm0, adlı bir kuyruk yöneticisine gönderilir. Dosya aktarma işleminin sonucu, copy .result adlı özelliğe atanır.

```
<fte:filecopy cmdqm="qm0@localhost@1414@SYSTEM.DEF.SVRCONN"
  src="agent1@qm1" dst="agent2@qm2"
  rcproperty="copy.result">
  <fte:filespec srcfilespec="/home/fteuser1/file.bin" dstfile="/home/fteuser2/file.bin"/>
</fte:filecopy>
```

Bu örnek, aynı dosya aktarımı gösterir, ancak meta veri eklenmesiyle ve bir program, aktarma tamamlandıktan sonra kaynak aracıda yer almaya başlar.

```
<fte:filecopy cmdqm="qm0@localhost@1414@SYSTEM.DEF.SVRCONN"
  src="agent1@qm1" dst="agent2@qm2"
  rcproperty="copy.result">
```

```
<fte:metadata>
  <fte:entry name="org.example.departId" value="ACCOUNTS"/>
  <fte:entry name="org.example.batchGroup" value="A1"/>
</fte:metadata>

<fte:filespec srcfilespec="/home/fteuser1/file.bin" dstfile="/home/fteuser2/file.bin"/>

  <fte:postsrc command="/home/fteuser2/scripts/post.sh" successsrc="1" >
    <fte:arg value="/home/fteuser2/file.bin"/>
  </fte:postsrc>
</fte:filecopy>
```

## İlgili kavramlar

[Kurtarma işlemi sırasında dosya aktarımları için zamanaşımı seçeneği](#)

## İlgili görevler

[Apache Ant ile MFTkomutunu kullanma](#)

## fte: filemove Ant görevi

**fte: filemove** görevi, dosyaları Managed File Transfer araçları arasında taşır. Bir dosya kaynak araçından hedef aracıya başarıyla aktarıldığında, dosya kaynak araçından silinir.

## Öznitelikler

### cmdqm

İsteğe Bağlı. İsteği teslim etmek için komut kuyruğu yöneticisi. Bu bilgileri *qmgrname@host@port@channel* biçiminde belirtin; burada:

- *qmgrname* , kuyruk yöneticisinin adıdır.
- *host* , kuyruk yöneticisinin çalıştırıldığı sistemin isteğe bağlı anasistem adıdır.
- *port* , kuyruk yöneticisinin dinlediği isteğe bağlı kapı numarasıdır.
- *channel* , kullanılacak isteğe bağlı SVRCONN kanalına

Komut kuyruğu yöneticisi için *host*, *port* ya da *channel* bilgilerini çıkarırsanız, `command.properties` dosyasında belirtilen bağlantı bilgileri kullanılır.



**Uyarı:** Şu öğe için bir değer belirtilmediyse:

- *anasistem* değişkeni, bağ tanımları kipi kullanılır
- *port* değişkeni, 1414 değeri kullanılır
- *kanal* değişkeni, SYSTEM.DEF.SVRCONN kullanılır.

Ek bilgi için [MFT command.properties dosyası](#) başlıklı konuya bakın.

Ancak, ortadaki öznitelikleri atlayamazsınız; örneğin, *qmgrname@host@@channel*. Sahip olabilirsiniz; örneğin, *qmgrname@hostya* da *qmgrname@host@portya* da *qmgrname@hostport@@channel*.

MFT , belirtilen özniteliği @ sınırlayıcısını kullanarak böler. Bulunan simgelerin sayısına bağlı olarak, ilk simgeyi *qmgrname*, ikincisini *anasistem*, üçüncüsü *kapı* ve son olarak da *kanal* olarak alır.

Daha fazla bilgi için [MFT command.properties dosyası](#) başlıklı konuya bakın.

You can use the **com.ibm.wmqfte.propertySet** property to specify which `command.properties` file to use. Daha fazla bilgi için bkz. [com.ibm.wmqfte.propertySet](#).

`cmdqm` özniteliğini kullanmayacaksa, bu özellik ayarlandıysa, görev varsayılan olarak `com.ibm.wmqfte.ant.CommandQueueManager` özelliğini kullanır. `com.ibm.wmqfte.ant.CommandQueueManager` özelliği ayarlanmadıysa, `command.properties` dosyasında tanımlı olan varsayılan kuyruk yöneticisine yönelik bir bağlantı denenir. `com.ibm.wmqfte.ant.CommandQueueManager` özelliğinin biçimi, `cmdqm` öznitelimesiyle aynıdır; bu, *qmgrname@host@port@channel*.

## dst

Gereklidir. Kopyalama işlemine ilişkin hedef aracıyı belirtir. Bu bilgileri şu biçimde belirtin: *agentname@qmgrname* ; burada *agentname* hedef aracısının adıdır ve *qmgrname* , bu aracının doğrudan bağlı olduğu kuyruk yöneticisinin adıdır.

## idproperty

Optional unless you have specified an outcome of *defer*. Aktarma tanıtıcısını atamak için bir özelliğin adını belirtir. Aktarma tanıtıcıları, aktarma isteği sunulmakta ve aktarma işleminin ilerleyişini izlemek, sorunları bir aktarma ile tanılamak ve aktarma işlemini iptal etmek için aktarma tanıtıcılarını kullanabilirsiniz.

*ignore*' un outcome özelliğini de belirtmiş bulursanız, bu özelliği belirtebilirsiniz. Ancak, *defer*' un outcome özelliğini de belirtmiş olmanız durumunda, *idproperty* belirtmeniz gerekir.

## jobName

İsteğe Bağlı. Taşıma isteğine bir iş adı atar. Mantıksal aktarımlar grupları oluşturmak için iş adlarını kullanabilirsiniz. Sözde benzersiz iş adları oluşturmak için *fte: uuid* görevini kullanın. *jobname* özneliğini kullanmayarsanız, bu özellik ayarlandıysa, görevin varsayılan değeri olarak *com.ibm.wmqfte.ant.jobName* özellik değerini kullanması gerekir. Bu özelliği ayarlamadıysanız, taşıma isteğiyle hiçbir iş adı ilişkilendirilemez.

## ilk kullanıcı

İsteğe Bağlı. Taşıma isteğiyle ilişkilendirilecek kaynak kullanıcı tanıtıcısını belirtir. *originuser* özneliğini kullanmayacaksa, görev varsayılan olarak Ant komut dosyasını çalıştırmak için kullanılan kullanıcı kimliğini kullanır.

## sonuç

İsteğe Bağlı. Denetim, denetimi Ant komut dosyasına döndürmeden önce taşıma işlemini tamamlamak için bekleyeceğini belirler. Aşağıdaki seçeneklerden birini belirleyin:

### bekliyor

Görev, geri dönmeye kadar taşıma işleminin tamamlanmak üzere beklenmesini bekler. *await* için outcome belirtildiğinde, *idproperty* özneliği isteğe bağlıdır.

### Ertele

Bu görev, taşıma isteği gönderilip gönderilmez ve taşıma işleminin sonucunun "*fte: awaitsonucun Ant görevi*" sayfa 2070 ya da "*fte: ignoreoutcome Ant görevi*" sayfa 2081 görevi kullanılarak daha sonra ele alınması sonucuna vardığı varsayıldığında geri döner. When an outcome of *defer* is specified the *idproperty* attribute is required.

### yoksay

Taşıma işleminin sonucu önemli değilse, *ignore* için bir değer belirtebilirsiniz. Daha sonra, aktarma isteği gönderildikten sonra, aktarma işleminin sonucunu izlemek için kaynak ayrılmadan önce, bu görev geri döndürülür. When an outcome of *ignore* is specified the *idproperty* attribute cannot be specified.

Sonuç özneliğini belirtmezseniz, görev varsayılan olarak *await* değerini kullanacak şekilde ayarlanır.

## öncelik

İsteğe Bağlı. Taşıma isteğiyle ilişkilendirilecek önceliği belirtir. Genel olarak, daha yüksek öncelikli aktarma istekleri, düşük öncelikli isteklere göre öncelikli olarak uygulanır. Öncelik değeri 0-9 aralığında (bu değerler de içinde olmak üzere) olmalıdır. 0 öncelik değeri, en düşük öncelik ve 9 değerinin en yüksek önceliğe sahip olduğu bir değerdir. *priority* özneliğini belirtmezseniz, aktarma varsayılan olarak 0 önceliğine ayarlanır.

## rcproperty

İsteğe Bağlı. Taşıma isteğinin sonuç kodunu atamak için bir özelliğin adını belirtir. Sonuç kodu, taşıma isteğinin genel sonucunu yansıtır.

You cannot specify this property if you have also specified an outcome property of *ignore* or *defer*. Ancak, *awaitsonucunun* bir sonucunu belirtmiş olmanız durumunda *rcproperty* değerini belirtmeniz gerekir.

## transferRecoveryZamanaşımı

İsteğe Bağlı. Bir kaynak aracının durdurulan bir dosya aktarımı kurtarmaya çalıştığı süreyi saniye cinsinden belirler. Aşağıdaki seçeneklerden birini belirleyin:

-1

Aracı, aktarma tamamlanıncaya kadar, durdurulan aktarımı kurtarmayı denemeye devam eder. Bu seçeneği kullanmak, özellik ayarlanmadığında aracının varsayılan davranışına eşdeğerdir.

0

Aracı, kurtarma işlemini girer girmez dosya aktarımlarını durdurur.

>0

Aracı, belirtilen pozitif tamsayı değeri tarafından belirlenen süre saniye cinsinden, durdurulan aktarım işlemini kurtarma girişiminde bulunmaya devam eder. Örneğin,

```
<fte:filemove cmdqm="qm0@localhost@1414@SYSTEM.DEF.SVRCONN"
src=agent1@qm1 dst="agent2@qm2"
rcproperty="move.result" transferRecoveryTimeout="21600">

  <fte:filespec srcfilespec="/home/fteuser1/file.bin" dstfile="/home/fteuser2/
file.bin"/>

</fte:filemove
```

aracının, iyileşmeye girdiği zamandan itibaren 6 saat boyunca transferi kurtarmaya çalıştığıda olduğunu belirtir. Bu öznelik için üst sınır değeri 999999999' dir.

Aktarma kurtarma zamanasını değerinin bu şekilde belirtilmesi, aktarma temelinde bu değeri belirler. Bir Managed File Transfer ağındaki tüm aktarımlar için genel bir değer ayarlamak üzere, Aktarma kurtarma zamanasını özellikleri' ne bir özellik ekleyebilirsiniz. Daha fazla bilgi için bakınız: [Timeout option for transfers in recovery](#).

**src**

Gereklidir. Taşıma işlemine ilişkin kaynak aracıyı belirtir. Bu bilgileri şu biçimde belirtin: *agentname@qmgrname* ; burada *agentname* kaynak aracısının adıdır ve *qmgrname* , bu aracının doğrudan bağlı olduğu kuyruk yöneticisinin adıdır.

## İç içe geçmiş öğeler olarak belirtilen parametreler

**fte: filespec**

Gereklidir. Taşınmak üzere dosyaları tanımlayan en az bir dosya belirtimi belirlemelisiniz. Gerekirse birden çok dosya belirtimi belirtebilirsiniz. Ek bilgi için [“fte: filespec Ant içiçe yerleştirilmiş öğe” sayfa 2084](#) başlıklı konuya bakın.

**fte: meta veri**

İsteğe Bağlı. Dosya taşıma işlemiyle ilişkilendirilecek meta verileri belirtebilirsiniz. Bu meta veri aktarımla birlikte taşınır ve aktarma tarafından oluşturulan günlük iletilerine kaydedilir. Belirli bir aktarım öğesiyle yalnızca tek bir meta veri bloğunu ilişkilendirebilirsiniz; ancak bu blok birçok meta veri parçası içerebilir. Ek bilgi için [fte: metadata](#) başlıklı konuya bakın.

**fte: princ**

İsteğe Bağlı. Aktarma işlemi başlamadan önce, kaynak aracıda yer alacak bir program çağrısını belirtir. Tek bir `fte: princ` öğesini yalnızca belirli bir aktarımla ilişkilendirebilirsiniz. Ek bilgi için [program çağırma](#) konusuna bakın.

**fte: predst**

İsteğe Bağlı. Aktarma işlemi başlamadan önce hedef aracıda yer alacak bir program çağrısını belirtir. Tek bir `fte: predst` öğesini yalnızca belirli bir aktarımla ilişkilendirebilirsiniz. Ek bilgi için [program çağırma](#) konusuna bakın.

**fte: postsrc**

İsteğe Bağlı. Aktarma tamamlanınca, kaynak aracıda yer alacak bir program çağrısını belirtir. Tek bir `fte: postsrc` öğesini yalnızca belirli bir aktarımla ilişkilendirebilirsiniz. Ek bilgi için [program çağırma](#) konusuna bakın.

**fte: postdst**

İsteğe Bağlı. Aktarma tamamlanınca hedef aracıda yer alacak bir program çağrısını belirtir. Tek bir `fte: postdst` öğesini yalnızca belirli bir aktarımla ilişkilendirebilirsiniz. Ek bilgi için [program çağırma](#) konusuna bakın.

fte:presrc, fte:predst, fte:postsrc, fte:postdst ve çıkışlar başarılı bir durum döndürmezse, kurallar belirtilen sırada aşağıdaki gibidir:

1. Kaynak başlatma çıkışlarını çalıştırın. Kaynak başlatma çıkışları başarısız olursa, aktarma başarısız olur ve başka bir işlem başlatılamaz.
2. Kaynak öncesi aramayı çalıştırın (varsa). Kaynak öncesi arama başarısız olursa, aktarma başarısız olur ve başka bir şey çalıştırılmaz.
3. Hedef başlangıç çıkışlarını çalıştırın. Hedef başlangıç çıkışları aktarma başarısız olursa, aktarma başarısız olur ve başka bir işlem yürütülemez.
4. Hedef çağrı öncesi aramayı çalıştırın (varsa). Hedef öncesi arama başarısız olursa, aktarma başarısız olur ve başka bir şey çalıştırılmaz.
5. Dosya aktarımlarını gerçekleştir.
6. Hedef uç çıkışlarını çalıştırın. Bu çıkışlar için herhangi bir arıza durumu yok.
7. Aktarım başarılı olursa (bazı dosyalar başarıyla aktarılırsa, aktarma başarılı olarak kabul edilir), hedef sonrası aramayı (varsa) çalıştırın. Hedef sonrası arama başarısız olursa, aktarma başarısız olur.
8. Kaynak uç çıkışlarını çalıştırın. Bu çıkışlar için herhangi bir arıza durumu yok.
9. Aktarma başarılı olursa, kaynak sonrası çağrısı (varsa) çalıştırın. Kaynak sonrası arama başarısız olursa, aktarma başarısız olur.

## Örnekler

Bu örnekte, agent1 ile agent2 arasında bir temel dosya taşıma işlemi gösterilmektedir. Dosya taşıma işlemi başlatmak için kullanılan komut, istemci aktarım kipi bağlantısı kullanılarak qm0, adlı bir kuyruk yöneticisine gönderilir. Dosya aktarma işleminin sonucu, move.result adlı özelliğe atanır.

```
<fte:filemove cmdqm="qm0@localhost@1414@SYSTEM.DEF.SVRCONN"
  src="agent1@qm1" dst="agent2@qm2"
  rcproperty="move.result">
  <fte:filespec srcfilespec="/home/fteuser1/file.bin" dstfile="/home/fteuser2/file.bin"/>
</fte:filemove>
```

## İlgili kavramlar

[Kurtarma işlemi sırasında dosya aktarımları için zamanaşımı seçeneği](#)

## İlgili görevler

[Apache Ant ile MFTkomutunu kullanma](#)

## fte:ignoreoutcome Ant görevi

Bir **fte:filecopy**, **fte:filemove** ya da **fte:call** komutunun sonucunu yoksayın. defersonucuna sahip olmak için bir **fte:filecopy**, **fte:filemove** ya da **fte:call** görevi belirttiğinizde, Ant görevi bu sonucu izlemek için kaynakları ayırır. Sonuçla artık ilgilenmediyseniz, bu kaynakları serbest olarak kullanmak için **fte:ignoreoutcome** görevini kullanabilirsiniz.

## Öznitelikler

### tanıtıcı

Gereklidir. Artık ilgi çekmeyecek olan sonucu tanımlar. Genellikle bu tanıtıcıyı, "[fte:filecopy Ant görevi](#)" sayfa 2074, "[fte:filemove Ant görevi](#)" sayfa 2078 ya da "[fte: Ant görevini çağır](#)" sayfa 2071 görevinin idproperty özniteliğini kullanarak ayarladığınız bir özelliği kullanarak belirtiyorsunuz.

## Örnek

Bu örnek, önceki “[fte: filecopy Ant görevi](#)” sayfa 2074 görevinin sonucunu izlemek için ayrılan kaynakları serbest olarak kullanmak için `fte: ignoreoutcome` görevini nasıl kullanabileceğinin gösterilmektedir.

```
<!-- issue a file copy request -->
<fte:filecopy cmdqm="qm1@localhost@1414@SYSTEM.DEF.SVRCONN"
             src="agent1@qm1" dst="agent1@qm1"
             idproperty="copy.id"
             outcome="defer"/>

<!-- do some other things -->

<!-- decide that the result of the copy is not interesting -->
<fte:ignoreoutcome id="{copy.id}"/>
```

## İlgili görevler

[Apache Ant ile MFTkomutunu kullanma](#)

## fte: ping Ant görevi

Bu IBM MQ Managed File Transfer Ant görevi, bir aracıyı yanıtlamak için bir aracıyı ping eder ve aracının aktarımları işleyip işleyemediğini belirler.

**Not:** IBM WebSphere MQ File Transfer Edition (FTE) artık desteklenen bir ürün değildir. IBM MQ'da FTE'den Managed File Transfer bileşenine geçiş için [Migrating Managed File Transfer](#) başlıklı konuya bakın.

## Öznitelikler

### aracı

Gereklidir. **fte:ping** isteğinin gönderileceği aracıyı belirtir. Değer şu biçimdeki biçimdir: `agentname@qmgrname` ; burada `agentname` aracının adı ve `qmgrname` bu aracının doğrudan bağlı olduğu kuyruk yöneticisinin adıdır.

### cmdqm

İsteğe Bağlı. İsteği teslim etmek için komut kuyruğu yöneticisi. Bu bilgileri `qmgrname@host@port@channel` biçiminde belirtin; burada:

- `qmgrname` , kuyruk yöneticisinin adıdır.
- `host` , kuyruk yöneticisinin çalıştırıldığı sistemin isteğe bağlı anasistem adıdır.
- `port` , kuyruk yöneticisinin dinlediği isteğe bağlı kapı numarasıdır.
- `channel` , kullanılacak isteğe bağlı SVRCONN kanalına

Komut kuyruğu yöneticisi için `host`, `port` ya da `channel` bilgilerini çıkarırsanız, `command.properties` dosyasında belirtilen bağlantı bilgileri kullanılır.



**Uyarı:** Şu öğe için bir değer belirtilmediyse:

- `anasistem` değişkeni, bağ tanımları kipi kullanılır
- `port` değişkeni, 1414 değeri kullanılır
- `kanal` değişkeni, SYSTEM.DEF.SVRCONN kullanılır.

Ek bilgi için [MFT command.properties dosyası](#) başlıklı konuya bakın.

Ancak, ortadaki öznitelikleri atlayamazsınız; örneğin, `qmgrname@host@@channel`. Sahip olabilirsiniz; örneğin, `qmgrname@hostya` da `qmgrname@host@portya` da `qmgrname@hostport@@channel`.

MFT , belirtilen özniteliği @ sınırlayıcısını kullanarak böler. Bulunan simgelerin sayısına bağlı olarak, ilk simgeyi `qmgrname`, ikincisini `anasistem`, üçüncüsü `kapı` ve son olarak da `kanal` olarak alır.

Daha fazla bilgi için [MFT command.properties dosyası](#) başlıklı konuya bakın.

You can use the **com.ibm.wmqfte.propertySet** property to specify which `command.properties` file to use. Daha fazla bilgi için bkz. `com.ibm.wmqfte.propertySet`.

`cmdqmc` öznelikliğini kullanmayacaksa, bu özellik ayarlandıysa, görev varsayılan olarak `com.ibm.wmqfte.ant.commandQueueManager` öznelikliğini kullanır. `com.ibm.wmqfte.ant.commandQueueManager` öznelikliğini ayarlanmadıysa, `command.properties` dosyasında tanımlı olan varsayılan kuyruk yöneticisine yönelik bir bağlantı denir. `com.ibm.wmqfte.ant.commandQueueManager` öznelikliğinin biçimi, `cmdqmc` öznelikmesiyle aynıdır; bu, `qmgrname@host@port@channel`.

### rcproperty

Gereklidir. **ping** işleminin dönüş kodunu depolamak için bir öznelikliğini adlandırır.

### zaman aşımı

İsteğe Bağlı. Görevin aracının yanıt vermesinin beklenmesi için saniye cinsinden süre üst sınırı. Zamanaşımı alt sınırı sıfır saniyedir; ancak, komutun, aracının yanıt vermesi için sonsuza kadar bekleyeceği gibi, bir zamanaşımı eksi bir saniye de belirtilebilir. `timeout` için değer belirtilmezse, varsayılan değer, aracının yanıt vermesi için en çok 5 saniye beklemektedir.

### Örnek

This example sends a **fte:ping** request to `agent1` hosted by `qm1`. **fte:ping** isteği, aracının yanıt vermeleri için 15 saniye bekler. **fte:ping** isteğinin sonucu, `ping.rc` adlı bir öznelikte saklanır.

```
<fte:ping agent="agent1@qm1" rcproperty="ping.rc" timeout="15"/>
```

### Dönüş kodları

0

Komut başarıyla tamamlandı.

2

Komut zamanaşımına uğradı.

### İlgili görevler

[Apache Ant ile MFTkomutunu kullanma](#)

### fte: uuid Ant görevi

Sözde rasgele benzersiz bir tanıtıcı oluşturur ve bu tanıtıcıyı verili bir özneliğe atar. Örneğin, diğer dosya aktarma işlemleri için iş adları oluşturmak üzere bu tanıtıcıyı kullanabilirsiniz.

### Öznelikler

#### uzunluk

Gereklidir. Oluşturulacak UUID 'nin sayısal uzunluğu. Bu uzunluk değeri, **prefix** parametresiyle belirtilen herhangi bir önekin uzunluğunu içermez.

#### özellik

Gereklidir. Oluşturulan UUID 'ye atanabilmek için özneliğinin adı.

#### Önek

İsteğe Bağlı. Oluşturulan UUID 'ye eklemek için bir önek. Bu önek, **length** parametresi tarafından belirtildiği gibi, UUID 'nin uzunluğunun bir parçası olarak sayılmaz.

### Örnek



Bu örnek, ABC harfleriyle başlayan bir UUID 'yi ve 16 adet sözde rasgele onaltılı karakterin izlediği bir UUID tanımlar. UUID, `uuid.property` adlı bir özneliğe atanmaktadır.

```
<fte:uuid length="16" property="uuid.property" prefix="ABC"/>
```

## İlgili görevler

[Apache Ant ile MFTkomutunu kullanma](#)

## fte: filespec Ant içiçe yerleştirilmiş öge

**fte:filespec** parametresi, diğer görevlerde içiçe yerleşimli bir öge olarak kullanılır. Bir ya da daha fazla kaynak dosya, dizin  ya da veri kümesi ile bir hedef arasındaki eşlemeyi açıklamak için **fte:filespec** seçeneğini kullanın. Typically this element is used when expressing a set of files or directories  ya da veri kümeleri to move or copy.

## İçiçe geçmiş:

- [fte: filecopy](#) görevi
- [fte: filemove](#) görevi

## Kaynak belirtim öznitelikleri

srcfilespec ya da srcqueuebelirtimlerinden birini belirtmelisiniz.

### srcfilespec

Dosya işleminin kaynağını belirtir. Bu özniteliğin değeri joker karakter içerebilir.

### srcqueue

Aktarmanın kaynağının bir kuyruk olduğunu belirtir. Aktarma işlemi, verileri bu özniteliğe göre belirlenen kuyruktan taşıyan iletilerden taşır. **fte:filespec** görevi, **fte:filecopy** görevi içinde içiçe yerleştirilmişse, bu özniteliği belirtmezsiniz.

Kaynak aracı bir iletişim kuralı köprüsü aracıysa, srcqueue özniteliği desteklenmez.

## Hedef belirtim öznitelikleri

dstdir, dstds, dstfilespace, dstfile, dstqueue ya da dstpdsbelirtimlerinden birini belirtmelisiniz.

### dstdir

Bir dosya işlemi için hedef olarak bir dizin belirtir.

### dstds

Bir dosya işlemi için hedef olarak ayarlanan bir veri kümesini belirtir.

Bu öznitelik yalnızca hedef aracı z/OS altyapısında çalışıyorsa desteklenir.

### dstfile

Bir dosya işlemi için hedef olarak bir dosya belirtir.

### dstfilespace

Dosya işlemi için hedef olarak bir dosya alanı belirtir.

Bu öznitelik, hedef aracı, web ağ geçidi dosya alanına erişimi olan bir IBM MQ 8.0 web aracıysa geçerlidir.

### dstpds

Bir dosya işlemi için hedef olarak ayarlanmış bölümlenmiş bir veri kümesini belirtir.

Bu öznitelik yalnızca hedef aracı z/OS altyapısında çalışıyorsa desteklenir.

### dstqueue

Bir dosyanın ileti işlemi için hedefi olarak bir kuyruk belirtir. İsteğe bağlı olarak, QUEUE@QUEUEMANAGER biçimini kullanarak bu belirtim için bir kuyruk yöneticisi adı ekleyebilirsiniz. Bir kuyruk yöneticisi adı belirtmezseniz, enableClusterQueueInputOutput Agent özelliğini true değerine ayarlamadıysanız, hedef aracı kuyruk yöneticisi kullanılır. enableClusterQueueInputOutput özelliği true olarak ayarlanırsa, hedef aracı, kuyruğun nerede olduğunu belirlemek için standart IBM MQ yordamlarını kullanır. Kuyruk yöneticisinde varolan geçerli bir kuyruk adı belirtmelisiniz.



dstqueue özniteliğini belirtirseniz, bu öznitelikler karşılıklı olarak birbirini dışladığından, srcqueue özniteliklerini belirleyemezsiniz.

Hedef aracı bir protokol köprüsü aracıysa, dstqueue özniteliği desteklenmez.

## Kaynak seçeneği öznitelikleri

### srcencoding

İsteğe Bağlı. Aktarmak için dosya tarafından kullanılan karakter kümesi kodlaması.

Bu özniteliği yalnızca, conversion özniteliği text . değerine ayarlandığında belirtebilirsiniz.

srcencoding özniteliğini belirtmezseniz, metin aktarımları için kaynak sistemin karakter takımı kullanılır.

### srceol

İsteğe Bağlı. Aktarılmakta olan dosya tarafından kullanılan satır sınırlayıcısının sonu. Geçerli değerler şunlardır:

- CRLF -Satır sonu sınırlayıcısının sonu olarak bir satır başı karakteri ve ardından satırbaşı karakteri kullanın. Bu kural, Windows sistemleri için tipik bir kuraldır.
- LF -Satır sonu sınırlayıcısının sonu olarak bir satır besleme karakteri kullanın. Bu kural, UNIX sistemleri için tipik bir kuraldır.

Bu özniteliği yalnızca, conversion özniteliği text değerine ayarlandığında belirtebilirsiniz. srceol özniteliğini belirlemezseniz, metin aktarımları kaynak aracının işletim sistemine dayalı olarak doğru değeri otomatik olarak belirler.

### z/OS

### srckeeptrailingspaces

İsteğe Bağlı. Kaynak kayıtlarda, metin kipi aktarımın bir parçası olarak ayarlanmış sabit uzunluklu veri kümesinden okunan boşlukların tutulup tutulmayacağını belirler. Geçerli değerler şunlardır:

- true -sondaki boşluklar korunur.
- false -sondaki boşluklar çıkarılır.

srckeeptrailingspaces özniteliğini belirtmezseniz, varsayılan bir false değeri belirtilir.

Bu özniteliği yalnızca, srcfilespec özniteliğini de belirtirseniz ve conversion özniteliğini text . değerine ayarladıysanız belirtebilirsiniz.

### srcmsgdelimbyte

İsteğe Bağlı. Bir ikili dosyaya birden çok ileti eklenirken sınırlayıcı olarak eklenecek bir ya da daha çok bayt değerini belirtir. Her değer, çift başında x başında olmak üzere, 00-FF aralığında iki onaltılı sayı olarak belirlenmelidir. Birden çok byte 'ın virgülle ayrılmış olması gerekir. Örneğin, srcmsgdelimbytes="x08, xA4". srcmsgdelimbytes özniteliğini yalnızca srcqueue özniteliğini belirtmiş olarak belirleyebilirsiniz. dönüştürme özniteliği için text değerini de belirlediyseniz, srcmsgdelimbytes özniteliğini belirleyemezsiniz.

### srcmsgdelimtext

İsteğe Bağlı. Bir metin dosyasına birden çok ileti eklenirken sınırlayıcı olarak eklenecek metin dizisini belirtir. Sınırlayıcıda dizgi hazır bilgileri için Java çıkış sıralarını ekleyebilirsiniz. Örneğin, srcmsgdelimtext="\u007d\n". Metin sınırlayıcı, kaynak aracıya göre her iletinin ardından eklenir. Metin sınırlayıcı, aktarımın kaynak kodlamasını kullanarak ikili biçime kodlanır. İletilerin her biri ikili biçimde okunur, kodlanmış sınırlayıcı iletiye ikili biçimde eklenir ve sonuç hedef aracıya ikili biçimde aktarılır. Kaynak aracı kod sayfası, çift bayt dizilimi başlangıç ve bitiş durumları içeriyorsa, aracı, iletinin sonunda her iletinin çıkış durumunda olduğunu varsayar. Hedef aracıda ikili veri, dosya metin aktarımında bir dosyayla aynı şekilde dönüştürülür. srcmsgsinirlaması özniteliğini yalnızca srckkuyruğu özniteliği için de belirtmiş ve dönüştürme özniteliği için text değerini belirlediyseniz belirtebilirsiniz.

### srcmsgsinirlaması

İsteğe Bağlı. Metin ya da ikili sınırlayıcının içine yerleştirdiği konumu belirler. Geçerli değerler şunlardır:

- **prefix** -Sınırlar, her iletiden gelen verilerden önce hedef dosyaya eklenir.
- **postfix** -Sınırlayıcılar, her iletiden gelen veriler sonrasında hedef dosyaya eklenir.

srcmsgdelimposition özniteliğini, yalnızca srcmsgdelimbytes ya da srcmsgdelimtext özniteliklerinden birini de belirtmiş durumunuz varsa belirtebilirsiniz.

### **srcmsggruplar**

İsteğe Bağlı. İletilerin, IBM MQ grup tanıtıcısına göre gruplandırıldığını belirtir. İlk tamamlanan grup hedef dosyaya yazılır. Bu öznitelik belirlenmezse, kaynak kuyruktaki tüm iletiler hedef dosyaya yazılır. srcmsggrousozniteliğini yalnızca srcqueue özniteliğini belirtmiş olarak belirleyebilirsiniz.

### **srcqueuetimeout**

İsteğe Bağlı. Aşağıdaki koşullardan birinin karşılanması için beklenecek süreyi (saniye cinsinden) belirtir:

- Kuyruğa yeni bir ileti yazılacak.
- kaynak grubu grupları özniteliği belirtilmişse, tam bir grup için kuyruğun üzerine yazılır.

Bu koşullardan hiçbiri srcqueuetimeoutdeğeriyle belirlenen süre içinde karşılanırsa, kaynak aracı kuyruktan okumayı durdurur ve aktarımı tamamlar. srcqueuetimeout özniteliği belirtilmezse, kaynak kuyruk boşsa ya da srcmsggrousozniteliğinin belirtildiği durumda, kuyruktaki tamamlanmış bir grup yoksa, kaynak aracı kaynak kuyruktan okumayı hemen durdurur. srcqueuetimeout özniteliğini ancak srcqueue özniteliğini belirtmiş olarak belirtebilirsiniz.

srcqueuetimeout değerini ayarlamaya ilişkin bilgi için [Guidance for belirtme a wait time on a message-to-file transfer](#) (İletiyeye dosya aktarımını bekleme süresi) başlıklı konuya bakın.

### **z/OS srcrcdelim**

İsteğe Bağlı. Kayıt odaklı bir kaynak dosyadan bir ikili dosyaya birden çok kayıt eklenirken sınırlayıcı olarak eklenecek bir ya da daha çok bayt değerini belirtir. Her bir değeri, çift olarak x başında olmak üzere, 00-FF aralığında iki onaltılı sayı olarak belirlemeniz gerekir. Birden çok byte 'in virgülle ayrılmış olması gerekir. Örneğin:

```
srcrcdelimbytes="x08,xA4"
```

srcrcdelimbytes özniteliğini yalnızca, aktarma kaynak dosyası kayıt odaklı ise, örneğin bir z/OS veri kümesi ve hedef dosya, kayıt odaklı olmayan normal bir dosya olduğunda belirleyebilirsiniz. dönüştürme özniteliği için text değerini de belirlediyseniz, srcrcdelimbytes özniteliğini belirtebilirsiniz.

### **srcrcdelimpos**

İsteğe Bağlı. İkili sınırlayıcının yerleştirileceği konumu belirtir. Geçerli değerler şunlardır:

- **prefix** (önek)-her kaynak kayıt odaklı dosya kaydından önce, hedef dosyaya sınırlayıcılar eklenir.
- **postfix**-sınırlayıcılar, her kaynak kayıt odaklı dosya kaydından gelen verilerden sonra hedef dosyaya eklenir.

srcrcdelimpos özniteliğini, yalnızca srcrcdelim özniteliğini de belirtmiş olduğunuz takdirde belirtebilirsiniz.

## **Hedef seçenek öznitelikleri**

### **dstencoding**

İsteğe Bağlı. Aktarılan dosya için kullanılacak karakter takımı kodlaması.

Bu özniteliği yalnızca, conversion özniteliği text . değerine ayarlandığında belirtebilirsiniz.

dstencoding özniteliği belirtilmediyse, metin aktarımları için hedef sistemin karakter takımı kullanılır.

### **dsteol**

İsteğe Bağlı. Aktarılan dosya için kullanılacak satır sınırlayıcısının sonu. Geçerli değerler şunlardır:

- CRLF -Satır sonu sınırlayıcısının sonu olarak bir satır başı karakteri ve ardından satırbaşı karakteri kullanın. Bu kural, Windows sistemleri için tipik bir kuraldır.
- LF -Satır sonu sınırlayıcısının sonu olarak bir satır besleme karakteri kullanın. Bu kural, UNIX sistemleri için tipik bir kuraldır.

Bu özneliği yalnızca, conversion özneliği text . değerine ayarlandığında belirtebilirsiniz.

dsteol özneliğini belirlemezseniz, metin aktarımları hedef aracının işletim sistemine dayalı olarak doğru değeri otomatik olarak belirler.

### **dstmsgdelimbytes**

İsteğe Bağlı. Bir ikili dosyayı birden çok iletiye bölürken kullanılacak onaltılı sınırlayıcıyı belirler. Tüm iletiler aynı IBM MQ grup tanıtıcısına sahiptir; gruptaki son ileti, IBM MQ LAST\_MSG\_IN\_GROUP işaret kümesine sahiptir. Sınırlayıcı olarak onaltılı byte belirtme biçimi şöyledir: xNN; burada N, 0-9 aralığındaki bir karakterdir ya da a-f. Onaltılı baytların virgülle ayrılmış bir listesini belirterek, sınırlayıcı olarak onaltılı byte 'lık bir bayt dizisi belirtebilirsiniz; örneğin: x3e , x20 , x20 , xbf.

dstmsgdelimbytes özneliğini yalnızca, dstqueue özneliğini de belirlediyseniz ve aktarma ikili kipte belirtilmişse belirtebilirsiniz. dstmsgsize, dstmsgdelimbyteve dstmsgdelimpattern özneliklerinden yalnızca birini belirtebilirsiniz.

### **dstmsgdelimpattern**

İsteğe Bağlı. Bir metin dosyasını birden çok iletiye bölürken kullanılacak Java düzenli ifadesini belirtir. Tüm iletiler aynı IBM MQ grup tanıtıcısına sahiptir; gruptaki son ileti, IBM MQ LAST\_MSG\_IN\_GROUP işaret kümesine sahiptir. Sınırlayıcı olarak düzenli ifade belirtilmek için kullanılan biçim, parantez içindeki, (*regular\_expression*) ya da çift tırnak içine alınmış düzenli bir ifadedir. "*regular\_expression*". Daha fazla bilgi için bkz. [MFT tarafından kullanılan düzenli ifadeler](#).

Varsayılan olarak, düzenli ifadenin eşleşebileceği dizginin uzunluğu hedef aracıyla beş karakter arasında sınırlanır. Bu davranışı, **maxDelimiterMatchLength** aracı özelliğini kullanarak değiştirebilirsiniz. Ek bilgi için [MFT gelişmiş aracı özellikler](#) başlıklı konuya bakın.

dstmsgdelimpattern özneliğini yalnızca dstqueue özneliğini belirtmiş ve aktarma metin kipinde olduğunda belirtebilirsiniz. dstmsgsize, dstmsgdelimbyteve dstmsgdelimpattern özneliklerinden yalnızca birini belirtebilirsiniz.

### **dstmsgdelimposition**

İsteğe Bağlı. Metin ya da ikili sınırlayıcının içinde olması beklenen konumu belirtir. Geçerli değerler şunlardır:

- **prefix** -Sınırlayıcıların her satırın başında beklenmesi beklenir.
- **postfix** -Sınırlayıcıların her satırın sonunda olması beklenir.

dstmsgdelimposition özneliğini yalnızca dstmsgdelimpattern özneliğini belirtmiş durumunuz da belirtmiş olabilirsiniz.

### **dstmsgincludedelim**

İsteğe Bağlı. Dosyayı, iletilerde birden çok iletiye bölmek için kullanılan sınırlayıcının eklenip eklenmeyeceğini belirtir. dstmsgincludedelim özneliği belirtilirse, sınırlayıcıdan önce gelen dosya verilerini içeren iletinin sonuna sınırlayıcı eklenir. Varsayılan olarak, sınırlayıcı iletilerde yer almaz. dstmsgincludedelim özneliğini yalnızca, dstmsgdelimpattern ve dstmsgdelimbytes özneliklerinden birini de belirtmiş olarak belirleyebilirsiniz.

### **dstmsgkalıcı saklama**

İsteğe Bağlı. Hedef kuyruğa yazılan iletilerin kalıcı olup olmadığını belirtir. Geçerli değerler şunlardır:

- **true** -Kalıcı iletileri hedef kuyruğa yazın. Bu varsayılan değerdir.
- **false** -Hedef kuyruğa kalıcı olmayan iletileri yazın.
- **qdef** -Kalıcı saklama değeri, hedef kuyruğun DefPersistence özneliğinden alınır.

Bu özneliği yalnızca, dstqueue özneliği de belirtildiğinde belirtebilirsiniz.

## dstmsgprops

İsteğe Bağlı. Aktarıma göre hedef kuyruğa yazılan ilk iletinin IBM MQ ileti özellikleri kümesine sahip olup olmadığını belirtir. Olası değerler şunlardır:

- `true` -Aktarma tarafından yaratılan ilk iletide ileti özelliklerini ayarlayın.
- `false` -Aktarma tarafından yaratılan ilk iletide ileti özelliklerini ayarlamayın. Bu varsayılan değerdir.

Daha fazla bilgi için bkz. [Hedef kuyruklara yazılan iletilerde MFT tarafından ayarlananMQ ileti özellikleri](#).

Bu özneliği yalnızca, `dstqueue` özneliği de belirtildiğinde belirtebilirsiniz.

## dstmsgsize

İsteğe Bağlı. Dosyanın, sabit uzunluklu birden çok iletiye bölünip bölünmeyeceğini belirler. İletilerin tümü aynı IBM MQ grup tanıtıcısına sahiptir; gruptaki son ileti, IBM MQ `LAST_MSG_IN_GROUP` işaret kümesine sahiptir. İletilerin boyutu, `dstmsgsize` değeri tarafından belirtilir. `dstmsgsize` biçimi *uzunlukbirimbiçim*dir; burada *uzunluk* artı bir tamsayı değeri ve *birim* aşağıdaki değerlerden biridir:

- B -Baytlar. İzin verilen alt sınır değeri, hedef iletilerin kod sayfasının karakter başına bayt başına değeri en yüksek olan iki kattır.
- K -Kibibayt. Bu değer 1024 bayttır.
- M -Mebibayt 'lar. Bu, 1024 kibibayt 'a eşdeğerdir.

Dosya metin kipinde aktarılırsa ve çift baytlık karakter takımı ya da çok baytlı karakter takımında ise, dosya, belirlenen ileti boyutuna en yakın karakter sınırındaki iletilere bölünür.

`dstmsgsize` özneliğini yalnızca, `dstqueue` özneliğini de belirtmiş olarak belirtebilirsiniz. `dstmsgsize`, `dstmsgdelimbyte` ve `dstmsgdelimpattern` özneliklerinden yalnızca birini belirtebilirsiniz.

## dstunsupportedcodepage

İsteğe Bağlı. Hedef kuyruk yöneticisi `dstqueue` özneliği tarafından belirtildiği gibi, dosya verilerini bir kuyruğa metin aktarımı olarak aktarırken kullanılan kod sayfasını desteklemiyorsa, yapılacak işlemi belirtir. Bu özneliğe ilişkin geçerli değerler şunlardır:

- `binary` -aktarmaya devam edin, ancak aktarılmakta olan verilere kod sayfası dönüştürme işlemini uygulamayın. Bu değer belirlenmesi, dönüştürme özneliğinin `text` olarak ayarlanmaması için eşdeğerdir.
- `fail` -aktarma işlemine devam edilmeyin. Dosya, aktarılamamış olarak kaydedilir. Bu varsayılandır.

`dstunsuptedcodepage` özneliğini yalnızca `dstqueue` özneliği için de belirtmiş ve dönüştürme özneliği için metin değerini belirlediyseniz belirtebilirsiniz.

## dsttrunkaterecords

İsteğe Bağlı. LRECL veri kümesi öznelikten daha uzun hedef kayıtların kısaltıldığını belirtir. Doğru olarak ayarlanırsa, kayıtlar kısaltılır. "false" olarak ayarlanırsa, kayıtlar kaydırılır. Varsayılan ayar `false` değeridir. Bu parametre yalnızca, hedefin veri kümesi olduğu metin kipi aktarımları için geçerlidir.

## Diğer öznelikler

### sağlama toplamı

İsteğe Bağlı. Aktarılan dosyaları sağlama toplamı için kullanılan algoritmayı belirler.

- MD5 - MD5 hash algoritmasını kullanın.
- NONE -sağlama toplamı algoritması kullanmayın.

Sağlama toplamı özneliğini belirtmezseniz, varsayılan değer olan MD5 kullanılır.

### Dönüştürme

İsteğe Bağlı. Aktarılmakta olan dosya için uygulanacak dönüştürme tipini belirler. Olası değerler şunlardır:

- `binary` -dönüştürme uygulanmaz.

- text -kaynak ve hedef sistemler arasında kod sayfası dönüştürme işlemini uygular. Satır sınırlayıcıların dönüştürülmesini de uygulayın. srcencoding, dstencoding, srceol ve dsteol öznitelikleri, uygulanan dönüştürmeyi etkiler.

dönüştürme özniteliğini belirtmezseniz, varsayılan bir binary değeri belirtilir.

### Üzerine Yaz

İsteğe Bağlı. İşlem tarafından var olan bir hedef dosyanın `z/OS` ya da veri kümesinin üzerine yazılıp yazılamayacağını belirler. Bir `true` değeri belirttiğinizde, var olan herhangi bir hedef dosyası `z/OS` ya da veri kümeleri üzerine yazılır. `false` değerini belirlediğinizde, işlemin hedef sonuçlarındaki yinelenen bir dosyanın `z/OS` ya da veri kümesi varlığı başarısız olur. `overwrite` özniteliği belirtilmediyse, varsayılan bir `false` değeri belirtilir.

### özyinelemeli

İsteğe Bağlı. Dosya aktarma yinelemesinin alt dizinlere yinelenip gönderilmeyeceğini belirler. `true` değerini belirlediğinizde, aktarma özyinelemeli olarak alt dizinlere aktarır. Bir `false` değeri belirlediğinizde, aktarma alt dizinlere özyinelemez. `recurse` özniteliği belirtilmediyse, varsayılan bir `false` değeri belirtilir.

### Örnek

This example specifies a `fte`: filespec with a source file of `file1.bin` and a destination file of `file2.bin`.

```
<fte:filespec srcfilespec="/home/fteuser/file1.bin" dstfile="/home/fteuser/file2.bin"/>
```

### İlgili görevler

[Apache Ant ile MFTkomutunu kullanma](#)

## fte: meta veri Ant içiçe öğeler

Meta veri, dosya aktarma işlemi ile kullanıcı tanımlı ek bilgi taşımak için kullanılır.

Managed File Transfer ' in meta verileri nasıl kullandığı hakkında daha fazla bilgi için [“MFT kullanıcı çıkışları için meta veri” sayfa 2093](#) konusuna bakın.

### İç içe geçmiş:

- [fte: filecopy](#) görevi
- [fte: filemove](#) görevi
- [fte: call](#) görevi

## İç içe geçmiş öğeler olarak belirtilen parametreler

### fte: giriş

`fte:metadata` içiçe yerleşimli öğenin içinde en az bir giriş belirtmelisiniz. Birden fazla giriş belirlemeyi seçebilirsiniz. Girişler, bir anahtar adını bir değerle ilişkilendirir. Tuşlar `fte:metadatablokunda` benzersiz olmalıdır

### Giriş öznitelikleri

#### NAME

Gereklidir. Bu girişe ait anahtarın adı. Bu ad, bir `fte: metadata` öğesinin içinde içiçe yerleştirilen tüm **entry** deęiřtirgelerinde benzersiz olmalıdır.

#### deęer

Gereklidir. Bu girdiyeye atanacak deęer.

## Örnek

Bu örnekte, iki giriş içeren bir `fte:metadata` tanımı gösterilmektedir.

```
<fte:metadata>
  <fte:entry name="org.foo.partColor" value="red"/>
  <fte:entry name="org.foo.partSize" value="medium"/>
</fte:metadata>
```

## İlgili görevler

[Apache Ant ile MFTkomutunu kullanma](#)

## Program çağırma içiçe yerleştirilmiş öğeler

Şu beş içiçe öğeden biri kullanılarak programlar başlatılabilir: `fte:presrc`, `fte:predst`, `fte:postdst`, `fte:postsrc`, ve `fte:command`. Bu içiçe yerleştirilmiş öğeler, bir aracıya, işleminin bir parçası olarak bir dış program çağırmasını bildirir. Bir programı başlatmadan önce, komutun, komutu çalıştıran aracının `agent.properties` dosyasındaki `commandPath` özelliği tarafından belirtilen konumda olduğundan emin olmanız gerekir.

Her program çağırma öğesinin farklı bir adı olmasına rağmen, aynı öznitelikler kümesini ve içiçe yerleştirilmiş öğeler kümesini aynı şekilde paylaşırlar. Programlar, **`fte:filecopy`**, **`fte:filemove`** ve **`fte:command`** Ant görevleri tarafından başlatılabilir.

Programları bir `Connect:Direct` köprüsü aracısından çağırabilirsiniz.

## Programları çağırabilen Ant görevleri:

- The `fte:filecopy` task nests program invocation parameters using the `fte:predst`, `fte:postdst`, `fte:presrc`, and `fte:postsrc` nested elements.
- The `fte:filemove` task nests program invocation parameters using the `fte:predst`, `fte:postdst`, `fte:presrc`, and `fte:postsrc` nested elements.
- The `fte:çağrı` task nests program invocation parameters using the `fte:command` nested element.

## Öznitelikler

### DELETE ...

Gereklidir. Çağrılacak programı adlandırır. Aracının bir komutu çalıştırabilmesi için, komutun, aracının `agent.properties` dosyasındaki `commandPath` özelliği tarafından belirtilen bir konumda olması gerekir. Daha fazla bilgi için bkz. `commandPath` MFT property. `command` özniteisinde belirtilen yol bilgileri, `commandPath` özelliği tarafından belirlenen bir yere göre kabul edilir. `type` `executable` olduğunda, yürütülebilir bir program beklenir aksi takdirde çağrı türü için uygun bir komut dosyası beklenir.

### yeniden deneme sayısı

İsteğe Bağlı. Program bir başarı dönüş kodu döndürmezse, programı çağırma işlemi yeniden deneme sayısı. `command` özniteliği tarafından adlandırılan program, bu sayı kadar çağrıya kadar çağrılır. Bu özniteliğe atanan değer eksi olmayan bir değer olmalıdır. `retrycount` özniteliğini belirtmezseniz, varsayılan değer olarak sıfır kullanılır.

### yeniden deneme

İsteğe Bağlı. Program çağrısını yeniden denemeden önce beklenecek süre (saniye olarak). `command` özniteliği tarafından adlandırılan program bir başarı dönüş kodu döndürmezse ve `retrycount` özniteliği sıfır dışında bir değer belirtirse, bu parametre yeniden denemeler arasında beklenecek süreyi belirler. Bu özniteliğe atanan değer eksi olmayan bir değer olmalıdır. `retrywait` özniteliğini belirtmezseniz, varsayılan değer olarak sıfır kullanılır.

### başarılı

İsteğe Bağlı. Bu özniteliğin değeri, program çağrısının başarılı bir şekilde ne zaman çalıştırılabileceğini belirlemek için kullanılır. Komut için dönüş kodu bu ifade kullanılarak değerlendirildi. Değer, Boole ORya da ve işareti (&) belirtmek için dikey çubuk karakteri (!) ile birleştirilen bir ya da daha fazla

ifadeden oluşabilir. Boole VE' i belirtmek için kullanılan karakter. Her ifade aşağıdaki ifade tiplerinden biri olabilir:

- Süreç dönüş kodu ile sayı arasındaki eşitlik testini belirtmek için bir sayı.
- A number prefixed with a ">" character to indicate a greater-than test between the number and the process return code.
- Sayı ile işlem dönüş kodu arasındaki testten daha az bir testi belirten bir sayı önek olarak "<" karakteri eklenmiş.
- Başına bir "!" öneki olan bir sayı Sayı ile süreç dönüş kodu arasındaki deneyi eşit olmayan bir testi belirtmek için kullanılan karakter.

Örneğin: >2&<7&!5|0|14 , başarılı olmak için şu dönüş kodları olarak yorumlanır: 0, 3, 4, 6, 14. Diğer tüm dönüş kodları başarısız olduğu şeklinde yorumlanır. successrc özniteliğini belirtmezseniz, varsayılan değer olarak sıfır kullanılır. Bu, komutun, bir sıfır kodunu döndürmesi durumunda başarılı bir şekilde çalıştırılacağına ilişkin olarak değerlendirileceği anlamına gelir.

#### tip

İsteğe Bağlı. Bu özniteliğin değeri, çağrılmakta olan programın tipini belirtir. Aşağıdaki seçeneklerden birini belirleyin:

#### Yürütülebilir

Görev, yürütülebilir bir program çağırır. arg içiçe yerleştirilmiş öge kullanılarak ek bağımsız değişkenler belirtilmiş olabilir. Programın commandPath üzerinde erişilebilir olması ve geçerli olduğu yerlerde, yürütme izin kümesinin erişilebilir olması beklenir. UNIX scripts can be called as long as they specify a shell program (for example, first line of shell script file is: #!/bin/sh). Command output written to stderr or stdout is sent to the Managed File Transfer log for the call. Ancak, veri çıkışı miktarı aracı yapılandırmasıyla sınırlandırılır. Varsayılan değer 10K bayttır, ancak aracı özelliğini kullanarak bu varsayılan değeri geçersiz kılabilirsiniz: maxCommandÇıktısı.

#### antscript

The task runs the specified Ant script, using the **fteAnt** command. Properties can be specified using the özellik nested element. Ant targets can be specified using the hedef nested element. Ant komut dosyasının commandPath üzerinde erişilebilir olması beklenir. stderr ya da stdout 'a yazılanAnt çıkışı, çağrıya ilişkin Managed File Transfer günlüğüne gönderilir. Ancak, veri çıkışı miktarı aracı yapılandırmasıyla sınırlandırılır. Varsayılan değer 10K bayttır, ancak aracı özelliğini kullanarak bu varsayılan değeri geçersiz kılabilirsiniz: maxCommandÇıkış.

#### z/OS jcl

jcl değeri yalnızca z/OS üzerinde desteklenir ve belirtilen z/OS JCL komut dosyasını çalıştırır. JCL, bir iş olarak gönderilir ve iş kartının var olmasını gerektirir. When the job is submitted successfully the JCL command output, written to the Managed File Transfer log, contains the following text: JOB iş\_adi(iş\_tanıtıcısı) where:

- *job\_name* , JCL ' de iş kartı tarafından tanımlanan işin adıdır.
- *job\_id* , z/OS sistemi tarafından oluşturulan iş tanıtıcısıdır.

İş başarıyla teslim edilemezse, JCL komut dosyası başarısız olur ve günlüğe bir ileti yazar ve hatanın nedenini belirtir (örneğin, iş kartı yoktur). İşin başarıyla çalıştırılıp çalıştırılmadığını anlamak için, SDSF gibi bir sistem hizmeti kullanın. Managed File Transfer , yalnızca işi teslim ettiği için bilgileri sağlamaz; daha sonra işin ne zaman çalıştırılacağı ve iş çıkışının nasıl sunulduğu saptanır. Bir JCL komut dosyası toplu iş olarak sunulduğu için, presrc ya da predst içiçe yerleşimli bir öge için jcl belirtmesi önerilmez; ancak, işin başarıyla gönderildiğini ve aktarma işlemi başlamadan önce başarıyla tamamlanıp tamamlanmayacağını bilemezsiniz. jcl tipi ile geçerli olan içiçe yerleştirilmiş öge yok.

Aşağıdaki örnekte bir JCL işi gösterilmektedir:

```
//MYJOB JOB
//*
//MYJOB EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=H
//SYSUT1 DD DSN=FRED.DEMO.TXT,DISP=SHR
```



```
//SYSUT2 DD DSN=BOB.DEMO.TXT,DISP=(NEW,CATLG),
// RECFM=VB,LRECL=133,BLKSIZE=2048,
// SPACE=(TRK,(30,5),RLSE)
//SYSIN DD DUMMY
```

## İç içe geçmiş öğeler olarak belirtilen parametreler

### **fte: arg**

Yalnızca, type özneliğinin değeri yürütüldüğünde geçerlidir. Program çağrısının bir parçası olarak çağrılmakta olan programa ilişkin bağımsız değişkenleri belirtmek için iç içe geçmiş `fte: arg` öğelerini kullanın. The program arguments are built from the values specified by the `fte: arg` elements in the order that the `fte: arg` elements are encountered. Bir program çağrısının iç içe yerleştirilmiş öğeleri olarak sıfır ya da daha fazla `fte: arg` öğesi belirtmeyi seçebilirsiniz.

### **fte: özellik**

Yalnızca, type özneliğinin değeri `antscript` in geçerli olduğu durumlarda geçerlidir. Use the name and value attributes of the nested `fte: property` elements to pass in name-value pairs to the Ant script. Bir program çağrısının iç içe yerleştirilmiş öğeleri olarak sıfır ya da daha fazla `fte: property` öğesi belirtmeyi seçebilirsiniz.

### **fte: hedef**

Yalnızca, type özneliğinin değeri `antscript` in geçerli olduğu durumlarda geçerlidir. Çağrılacak Ant komut dosyasında bir hedef belirtin. Bir program çağrısının iç içe yerleştirilmiş öğeleri olarak sıfır ya da daha fazla `fte: target` öğesi belirtmeyi seçebilirsiniz.

## Arg öznelikleri

### **değer**

Gereklidir. Çağrılmakta olan programa geçirilecek bağımsız değişkenin değeri.

## Özellik öznelikleri

### **ad**

Gereklidir. Ant komut dosyasına iletilecek bir özelliğin adı.

### **değer**

Gereklidir. Ant komut dosyasına iletilmekte olan özellik adıyla ilişkilendirilecek değer.

## Örnekler

Bu örnek, bir `fte: filecopy` görevinin bir parçası olarak belirtilmekte olan bir `fte: postsrc` program çağrısını gösterir. Program çağrısı, `post.sh` adlı bir program içindir ve `/home/fteuser2/file.bin.` in tek bir bağımsız değişkeni olarak sağlanır.

```
<fte:filecopy cmdqm="qm0@localhost@1414@SYSTEM.DEF.SVRCONN"
  src="agent1@qm1" dst="agent2@qm2"
  rcproperty="copy.result">
  <fte:filespec srcfilespec="/home/fteuser1/file.bin" dstfile="/home/fteuser2/file.bin"/>
  <fte:postsrc command="post.sh" successrc="1" >
    <fte:arg value="/home/fteuser2/file.bin"/>
  </fte:postsrc>
</fte:filecopy>
```

Bu örnekte, `fte: call` görevinin bir parçası olarak belirtilmekte olan bir `fte: command` program çağrısını gösterir. Program çağırma, komut satırı bağımsız değişkenlerinden geçirilmeyen `command.sh` adlı yürütülebilir bir yürütülür dosya içindir. `command.sh`, başarı dönüş kodunu 1 döndürmezse, komut 30 saniye sonra yeniden denir.

```
<fte:call cmdqm="qm0@localhost@1414@SYSTEM.DEF.SVRCONN"
  agent="agent1@qm1"
  rcproperty="call.rc"
  origuser="bob">
```



```
        jobname="${job.id}">
    <fte:command command="command.sh" successsrc="1" retrycount="5" retrywait="30"/>
</fte:call>
```

Bu örnekte, `fte: call` görevinin bir parçası olarak belirtilmekte olan bir `fte: command` program çağrısını gösterir. The program invocation is for the copy and compress targets in an Ant script called `script.xml`, which is passed two properties.

```
<fte:call cmdqm="qm0@localhost@1414@SYSTEM.DEF.SVRCONN"
  agent="agent1@qm1"
  rcproperty="call.rc"
  origuser="bob"
  jobname="${job.id}">
  <fte:command command="script.xml" type="antscript">
    <property name="src" value="AGENT5@QM5"/>
    <property name="dst" value="AGENT3@QM3"/>
    <target name="copy"/>
    <target name="compress"/>
  </fte:command>
</fte:call>
```

### İlgili görevler

[MFTile çalışmak üzere programların belirtilmesi](#)

[Apache Ant ile MFTkomutunu kullanma](#)

## Özelleştirme başvurusu için MFT kullanıcı çıkışları

Managed File Transfer için kullanıcı çıkışlarını yapılandırmanıza yardımcı olacak başvuru bilgileri.

### İlgili başvurular

[MFT kaynak ve hedef kullanıcı çıkışları](#)

### MFT kullanıcı çıkışları için meta veri

There are three different types of metadata that can be supplied to user exit routines for Managed File Transfer: environment, transfer, and file metadata. Bu meta veri, Java anahtar-değer çiftleri eşlemleri olarak sunulur.

### Ortam meta verileri

Ortam meta verileri tüm kullanıcı çıkışı yordamlarına geçirilir ve kullanıcı çıkışı yordamından çağrılmakta olan aracı yürütme ortamını tanımlar. Bu meta veri salt okunurdur ve kullanıcı çıkışı yordamlarıyla güncellenemez.

Çizelge 884. Ortam meta verileri	
Anahtar	Tanım
AGENT_CONFIGURATION_DIRECTOR_KEY	Aracının yapılandırma bilgilerini içeren dizinin adı.
AGENT_PRODUCT_DIRECTORY_KEY	Aracı kodunun kurulduğu dizinin adı.
AGENT_VERSION_KEY	Çıkış yordamını çağıran aracı çalıştırma zamanı için sürüm numarası.

Tablo 1 'de verilen anahtar adları ve değer adları, EnvironmentMetaDataConstants arabiriminde tanımlı sabitlerdir.

### Meta verileri aktar

Aktarma meta verileri tüm kullanıcı çıkış yordamlarına iletilir. Meta veriler, sistem tarafından sağlanan değerler ve kullanıcı tarafından sağlanan değerlerden oluşur. Sistem tarafından sağlanan değerleri değiştirirseniz, bu değişiklikler yoksayılar. Kaynak aktarımı başlangıç kullanıcı çıkışa ilişkin ilk kullanıcı tarafından sağlanan değerler, aktarımı tanımlarken sağladığınız değerlere dayalıdır. Kaynak aracı, kullanıcı tarafından sağlanan değerleri, kaynak aktarma başlatma kullanıcı çıkışının işlenmesinin bir parçası olarak

değiştirebilir. Bu kullanıcı çıkışı, tüm dosya aktarımı başlatılmadan önce çağrılır. Bu değişiklikler sonraki çağrılarda, bu aktarıyla ilgili diğer çıkış yordamlarıyla kullanılır. Aktarma meta verileri tüm aktarmaya uygulanır.

Tüm kullanıcı çıkışları veri aktarma meta verilerinden değerleri okuyabilse de, yalnızca kaynak aktarma başlangıç kullanıcı çıkışı aktarma meta verilerini değiştirebilir.

Farklı dosya aktarımları arasında bilgileri yaymak için meta veri aktarımı kullanamazsınız.

Sistem tarafından sağlanan aktarım meta verileri Tablo 2 'de ayrıntılı olarak açıklanmıştır:

<i>Çizelge 885. Meta verileri aktar</i>	
Anahtar	Tanım
AYRINTI_ANAHTARI	Aktarmanın hedefi olan aracının adı.
JOB_NAME_KEY	Aktarma isteğiyle ilişkili iş adı
MQMD_USER_KEY	İleti aktarma isteğini sunmak için kullanılan iletiden MQMD kullanıcı alanı
ORIGINATING_HOST_KEY	Aktarma isteğinde kaynak anasistem adı olarak belirtilen anasistem adı
ORIGINATING_USER_KEY	Aktarma isteğinde kaynak kullanıcı kimliği olarak belirtilen kullanıcı adı
SOURCE_AGENT_KEY	Aktarmanın kaynağı olan aracının adı
TRANSFER_ID_KEY	Aktarmanın tanıtıcısı

Çizelge 2 'de verilen anahtar adları ve değer adları, TransferMetaDataConstants arabiriminde tanımlanan değişmezlerdir.

### Dosya meta verileri

Dosya meta verileri, dosya belirtiminin bir parçası olarak kaynak aktarımı başlangıç çıkışa geçirilir. Kaynak ve hedef dosyalar için ayrı dosya meta verileri vardır.

Farklı dosya aktarımları arasında bilgileri yaymak için dosya meta verilerini kullanamazsınız.

<i>Çizelge 886. Dosya meta verileri</i>		
Anahtar	İzin verilen değerler	Tanım
CONVERT_LINE_SEPARATORS		Kaynak verilerdeki CRLF (satır başı satır besleme) ya da LF (satır besleme) satır ayırıcı sıralarının hedefteki satır ayırıcı düzenine dönüştürülmesini belirtmek için metin aktarımları için kullanılan anahtar değeri.
SINIRLAMA		Kayıt odaklı verileri normal dosyalara aktarırken, kayıt verilerini ayırmak için bir sınırlayıcı tanımlamak için kullanılan anahtar değeri. İletiyeye iletilmekte ve dosyaya yönelik aktarımlar için de kullanılır.
DEĞERLER_ANAHTAR_	DELIMITER_POSITION_PREFIX_VALUE DELIMITER_POSITION_POSTFIX_VALUE	Sınırlayıcının konumunu tanımlamak için DELIMIER_KEY ile birlikte kullanın; önek ya da son düzeltme.

Çizelge 886. Dosya meta verileri (devamı var)

Anahtar	İzin verilen değerler	Tanım
DELIM_TYPE_KEY	DELIMITER_TYPE_BINARY_VALUE DELIMITER_TYPE_TEXT_VALUE DELIM_TYPE_SI_VALUE	Sınırlayıcının tipini tanımlamak için DELIMIER_KEY ile birlikte kullanın.
DESTINATION_EXIST_KEY	DESTINATION_EXIST_KEY_ERROR_VALUE DESTINATION_EXIST_KEY_OVERWRITE_VALUE	Hedef dosya varsa, dosya aktarma davranışını belirler.
FILE_AL_ANAHTARI		Aktarılmakta olan dosyaya ilişkin diğer adı tanımlamak için kullanılan anahtar değeri.
FILE_CHECKSUM_METHOD_KEY	FILE_CHECKSUM_METHOD_NONE_DEĞERİ FILE_CHECKSUM_METHOD_MD5_VALUE	Dosya aktarırken kullanılacak sağlama toplamı yöntemini belirler.
FILE_CONVERSION_KEY	FILE_CONVERSION_TEXT_VALUE FILE_CONVERSION_BINARY_VALUE	Dosya içeriğine uygulanan dönüştürmenin tipini belirler.
FILEN_ENCODING_ANAHTAR		Bir metin dosyası için kullanılan kodlamayı belirler.
FILED_OF_LINE_KEY	FILED_OF_LINE_LF_VALUE FILED_OF_LINE_CRLF_VALUE	Bir satırın sonunu belirten karakter dizisini belirler: < LF > ya da < CR > < LF >.
FILE_SPACE_ALIAS		Dosya alanındaki bir dosyanın diğer adını belirler. <b>Not:</b> Bu meta veri yalnızca FILE_TYPE_KEY FILE_TYPE_FILE_SPACE_VALUE olduğunda kullanılabilir.
File_space_name		Dosya alanının adını belirler. <b>Not:</b> Bu meta veri yalnızca FILE_TYPE_KEY FILE_TYPE_FILE_SPACE_VALUE olduğunda kullanılabilir.
DOSYA_TYPE_KEY	FILE_TYPE_FILE_VALUE FILE_TYPE_DIRECTORY_VALUE FILE_TYPE_DATASET_VALUE FILE_TYPE_PDS_VALUE TYPE_TYPE_QUEUE_VALUE FILE_TYPE_FILE_SPACE_VALUE	Hedef dosya, kuyruk ya da dosya alanı belirtimini belirler.
GRUŞ_TUŞU		Kaynak kuyruktan okunabilmek için ileti grubunu belirlemek üzere, iletiden dosya aktarımları için kullanılan anahtar değeri. Bu öznitelik yalnızca USE_GROUPS_KEY değeri USE_GROUPS_TRUE_VALUE olduğunda geçerlidir.
INCLUDE_DELIMITER_IN_MESSAGE_KEY	INCLUDE_DELIMITER_IN_MESSAGE_TRUE_VALUE INCLUDE_DELIMITER_IN_MESSAGE_FALSE_VALUE	Dosyanın sonundaki birden çok iletiye, birden çok iletiye bölünmek için kullanılan sınırlayıcıların eklenip eklenmeyeceğini belirlemek üzere, dosya için ileti aktarımları için kullanılan anahtar değeri. Bu öznitelik yalnızca DELIMIER_TYPE_KEY değeri DELIMIER_TYPE_BINARY_VALUE DELIMIER_TYPE_TEXT_VALUE olduğunda geçerlidir.
INSERT_RECORD_LINE_SEPARATOR_KEY		Her kayıttan sonra satır ayırıcıların verilere eklenip eklenmeyeceğini belirlemek için kayıt odaklı dosyalardan metin aktarımları için kullanılan anahtar değeri.

Çizelge 886. Dosya meta verileri (devamı var)		
Anahtar	İzin verilen değerler	Tanım
KEEP_TRAILING_SPACE_KEY	KEEP_TRAILING_SPACE_TRUE_VALUE KEEP_TRAILING_SPACES_FALSE_VALUE	Sondaki boşlukların değişmez uzunluklu biçim veri kümelerinden okunan kayıtlardan kaldırılıp kaldırılmayacağını belirlemek için kullanılan anahtar değeri.
NEW_RECORD_ON_LINE_SEPARATOR_KEY		Verilerdeki satır ayırıcılarının kayıt verileri içerip içerilmeyeceğini ya da yeni bir kayıt (ve yazılmamasına) belirtmek için, kayıt odaklı dosyalara metin aktarımları için kullanılan anahtar değeri.
KIŞISEL ANAHTAR	PERSISTENT_TRUE_VALUE PERSISTENT_FALSE_VALUE PERSSTENT_QDEF_VALUE	İletilerin kalıcı olup olmadığını belirlemek için, dosya için aktarımlar için kullanılan anahtar değeri.
SET_MQ_PROPS_ANAHTARI	SET_MQ_PROPS_TRUE_VALUE SET_MQ_PROPS_FALSE_VALUE	Key value used for file-to-message transfers to determine whether IBM MQ message properties are set on the first message in a file, and any messages written to the queue when an error occurs.
TANIMLANMAMIŞ SED_CODE_PA	TANIMLANMAMIŞ SED_CODE_PAGE_FAIL_VALUE UNRECOVERSED_CODE_PAGE_BINARY_VALUE	Verilerin kod sayfası hedef kuyruk yöneticisi tarafından tanınmadığında, metin kipi aktarımın başarısız olup olmadığını ya da dönüştürmenin gerçekleştirilip gerçekleştirilmeyeceğini belirlemek için, dosya için aktarımlar için kullanılan anahtar değeri.
USE_GROUPS_TUŞU	USE_GROUPS_TRUE_VALUE USE_GROUPS_FALSE_VALUE	Kaynak kuyruktan yalnızca eksiksiz bir ileti grubunun aktarılıp aktarılmayacağını belirlemek için, iletiden dosya aktarımları için kullanılan anahtar değeri.
WAIT_TIME_KEY		Kaynak aracının aşağıdaki vakalardan birini bekleyeceği süreyi saniye cinsinden belirlemek üzere, ileti-dosya aktarımları için kullanılan anahtar değeri: <ul style="list-style-type: none"> <li>Kaynak kuyruğunda görüntülenecek ileti; kuyruk boşsa ya da USE_GROUPS_KEY değeri FALSE ise, bu ileti boş olur.</li> <li>USE_GROUPS_KEY değeri TRUE ise, kaynak kuyruğunda görüntülenecek tam bir grup.</li> </ul>

Çizelge 3 'te verilen anahtar adları ve değer adları, FileMetaDataConstants arabiriminde tanımlanan değişmezlerdir.

### MFT kaynak izleyicisi kullanıcı çıkışları

Kaynak izleme programı kullanıcı çıkışları, bir izleme programının tetikleme koşulu yerine getirildiğinde, ilişkili görev başlatılmadan önce çalıştırılacak özel kodu yapılandırmanıza olanak tanır.

Yeni aktarımların doğrudan kullanıcı çıkış kodundan çağırılması önerilmez. Bazı durumlarda, kullanıcı çıkışlarının aracı yeniden başlatılmasına karşı dayanıklı olmadığından, bu durum dosyaların birden çok kez aktarılmasına neden olur.

Kaynak izleyicisi kullanıcı çıkışları, kullanıcı çıkışları için var olan altyapıyı kullanır. İzleme programı kullanıcı çıkışları, bir izleme programı tetiklendikten sonra, ancak ilgili görevin izleme programı tarafından çalıştırılmasından önce çağrılır. Bu, kullanıcı çıkışının çalıştırılacak görevi değiştirmesini ve bir görevin devam edip etmeyeceğini değiştirmesini sağlar. İzleyici meta verilerini güncelleyerek izleme görevini değiştirebilirsiniz; daha sonra, özgün izleme programının yaratılmasıyla yaratılan görev belgesindeki değişken yerine koyma değeri için kullanılır. Diğer bir seçenek olarak, izleme çıkışı, parametre olarak geçirilen görev tanımlaması XML dizesini değiştirebilir ya da güncelleyebilir. İzleme programı çıkışı, görev için 'devam' ya da 'iptal ' sonuç kodunu döndürebilir. İptal işlemi iptal edildiye, görev başlatılmaz ve izlenen kaynak, tetikleme koşullarıyla eşleşinceye kadar izleme programı yeniden başlatılmaz. Kaynak değişmediyse, tetikleyici başlatılmaz. Diğer kullanıcı çıkışlarıyla olduğu gibi, izleme programı çıkışlarını da birlikte yapabilirsiniz. Çıkışlardan biri iptal sonuç kodu döndürürse, genel sonuç iptal olur ve görev başlatılmaz.

- Ortam meta verilerinin bir işlemi (diğer kullanıcı çıkışlarıyla aynı)
- Sabit sistem meta verileri ve mutable kullanıcı meta verileri de dahil olmak üzere, Monitor meta verileri işlemi. Sabitlenebilir sistem meta verileri aşağıdaki gibidir:
  - FILENAME-Tetikleme koşulunu karşılayan dosyanın adı
  - FILEPATH-tetikleme koşulunu karşılayan dosya yolu
  - FILESIZE (bayt olarak-bu meta veriler bulunmayabilir)-tetikleme koşulunu karşılayan dosyanın boyutu
  - LASTMODIFIEDDATE (Yerel)-Tetikleme koşulunu karşılayan dosyanın son olarak değiştiği tarih. Bu tarih, aracının çalıştırıldığı saat diliminin yerel tarihi olarak ifade edilir ve ISO 8601 tarihi olarak biçimlendirilir.
  - LASTMODIFIEDTIME (Yerel)-yerel biçimde, tetikleme koşulunu karşılayan dosyanın son olarak değiştiği süre. Bu süre, aracının çalışmakta olduğu saat diliminin yerel saati olarak ifade edilir ve ISO 8601 saati olarak biçimlendirilir.
  - LASTMODIFIEDDATEUTC-evrensel biçimde, tetikleme koşulunu karşılayan dosyanın son olarak değiştiği tarih. Bu tarih, yerel tarih UTC saat dilimine dönüştürülmüş ve ISO 8601 tarihi olarak biçimlendirilmiş olarak ifade edilir.
  - LASTMODIFIEDTIMEUTC-evrensel biçimde, tetikleme koşulunu karşılayan dosyanın son olarak değiştiği süre. Bu kez yerel saat UTC saat dilimine dönüştürüldü ve ISO 8601 saati olarak biçimlendirilmiş.
  - AGENTNAME-İzleyici aracı adı
- İzleme programı tetikleyicisinin sonucu olarak çalıştırılacak görevi gösteren bir XML dizgisi.

Görüntü birimi çıkışları aşağıdaki verileri döndürür:

- İlerlemenin devam edip etmeyeceğini belirten bir gösterge (devam et ya da iptal et)
- Tetikleyiciye yerleştirilecek günlük iletişime eklenecek dizgi

İzleme programı çıkış kodunu çalıştırmanın bir sonucu olarak, parametre olarak geçirilen Monitor meta verileri ve görev tanımlaması XML dizgisi de güncellenmiş olabilir.

Aracı özelliği monitorExitSınıflarının değeri ( agent . properties dosyasında), hangi izleme çıkış sınıflarının yükleneceğini belirtir; her bir çıkış sınıfı virgülle ayrılır. Örneğin:

```
monitorExitClasses=testExits.TestExit1,testExits.testExit2
```

İzleme programı kullanıcı çıkışa ilişkin arabirim:

```
package com.ibm.wmqfte.exitroutine.api;  
import java.util.Map;
```

```

/**
 * An interface that is implemented by classes that want to be invoked as part of
 * user exit routine processing. This interface defines a method that will be
 * invoked immediately prior to starting a task as the result of a monitor trigger
 */
public interface MonitorExit {

    /**
     * Invoked immediately prior to starting a task as the result of a monitor
     * trigger.
     *
     * @param environmentMetaData
     *      meta data about the environment in which the implementation
     *      of this method is running. This information can only be read,
     *      it cannot be updated by the implementation. The constant
     *      defined in <code>EnvironmentMetaDataConstants</code> class can
     *      be used to access the data held by this map.
     *
     * @param monitorMetaData
     *      meta data to associate with the monitor. The meta data passed
     *      to this method can be altered, and the changes will be
     *      reflected in subsequent exit routine invocations. This map
     *      also contains keys with IBM reserved names. These entries are
     *      defined in the <code>MonitorMetaDataConstants</code> class and
     *      have special semantics. The the values of the IBM reserved names
     *      cannot be modified by the exit
     *
     * @param taskDetails
     *      An XML String representing the task to be executed as a result of
     *      the monitor triggering. This XML string may be modified by the
     *      exit
     *
     * @return
     *      a monitor exit result object which is used to determine if the
     *      task should proceed, or be cancelled.
     */
    MonitorExitResult onMonitor(Map<String, String> environmentMetaData,
                               Map<String, String> monitorMetaData,
                               Reference<String> taskDetails);
}

```

İzleme meta verilerindeki IBM için ayrılan değerlerin değişmezleri aşağıdaki gibidir:

```

package com.ibm.wmqfte.exitroutine.api;

/**
 * Constants for IBM reserved values placed into the monitor meta data
 * maps used by the monitor exit routines.
 */
public interface MonitorMetaDataConstants {

    /**
     * The value associated with this key is the name of the trigger
     * file associated with the monitor. Any modification performed
     * to this property by user exit routines will be ignored.
     */
    final String FILE_NAME_KEY = "FILENAME";

    /**
     * The value associated with this key is the path to the trigger
     * file associated with the monitor. Any modification performed
     * to this property by user exit routines will be ignored.
     */
    final String FILE_PATH_KEY = "FILEPATH";

    /**
     * The value associated with this key is the size of the trigger
     * file associated with the monitor. This will not be present in
     * the cases where the size cannot be determined. Any modification
     * performed to this property by user exit routines will be ignored.
     */
    final String FILE_SIZE_KEY = "FILESIZE";

    /**
     * The value associated with this key is the local date on which
     * the trigger file associated with the monitor was last modified.
     * Any modification performed to this property by user exit routines
     * will be ignored.
     */
}

```

```

*/
final String LAST_MODIFIED_DATE_KEY = "LASTMODIFIEDDATE";

/**
 * The value associated with this key is the local time at which
 * the trigger file associated with the monitor was last modified.
 * Any modification performed to this property by user exit routines
 * will be ignored.
 */
final String LAST_MODIFIED_TIME_KEY = "LASTMODIFIEDTIME";

/**
 * The value associated with this key is the UTC date on which
 * the trigger file associated with the monitor was last modified.
 * Any modification performed to this property by user exit routines
 * will be ignored.
 */
final String LAST_MODIFIED_DATE_KEY_UTC = "LASTMODIFIEDDATEUTC";

/**
 * The value associated with this key is the UTC time at which
 * the trigger file associated with the monitor was last modified.
 * Any modification performed to this property by user exit routines
 * will be ignored.
 */
final String LAST_MODIFIED_TIME_KEY_UTC = "LASTMODIFIEDTIMEUTC";

/**
 * The value associated with this key is the name of the agent on which
 * the monitor is running. Any modification performed to this property by
 * user exit routines will be ignored.
 */
final String MONITOR_AGENT_KEY = "AGENTNAME";
}

```

## Kullanıcı çıkışı örneği örneği

Bu örnek sınıf, MonitorExit arabirimini uygular. Bu örnek, günün saati tek sayıysa LONDON değeriyle doldurulacak *REDIRECTEDAGENT* adlı Monitor meta verilerine özel bir yerine koyma değişkeni ekler ve hatta saat sayısı için PARIS değeri olur. The monitor exit result code is set to always return proceed.

```

package com.ibm.wmqfte.monitor;

import java.util.Calendar;
import java.util.Map;

import com.ibm.wmqfte.exitroutine.api.MonitorExit;
import com.ibm.wmqfte.exitroutine.api.MonitorExitResult;
import com.ibm.wmqfte.exitroutine.api.Reference;

/**
 * Example resource monitor user exit that changes the monitor mutable
 * metadata value between 'LONDON' and 'PARIS' depending on the hour of the day.
 */
public class TestMonitorExit implements MonitorExit {

    // custom variable that will substitute destination agent
    final static String REDIRECTED_AGENT = "REDIRECTEDAGENT";

    public MonitorExitResult onMonitor(
        Map<String, String> environmentMetaData,
        Map<String, String> monitorMetaData,
        Reference<String> taskDetails) {

        // always succeed
        final MonitorExitResult result = MonitorExitResult.PROCEED_RESULT;

        final int hour = Calendar.getInstance().get(Calendar.HOUR_OF_DAY);

        if (hour%2 == 1) {
            monitorMetaData.put(REDIRECTED_AGENT, "LONDON");
        } else {
            monitorMetaData.put(REDIRECTED_AGENT, "PARIS");
        }

        return result;
    }
}

```

```
}  
}
```

**REDIRECTEDAGENT** yerine koyma değişkeninden birini kullanan bir izleme programına ilişkin görev aşağıdakine benzer:

```
<?xml version="1.0" encoding="UTF-8"?>  
<request version="4.00"  
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"  
  xsi:noNamespaceSchemaLocation="FileTransfer.xsd">  
  <managedTransfer>  
    <originator>  
      <hostName>reportserver.com</hostName>  
      <userID>USER1</userID>  
    </originator>  
    <sourceAgent agent="AGENT1"  
      QMgr="QM1"/>  
    <destinationAgent agent="{REIRECTEDAGENT}"  
      QMgr="QM2"/>  
    <transferSet>  
      <item mode="binary" checksumMethod="MD5">  
        <source recursive="false" disposition="delete">  
          <file>c:\sourcefiles\reports.doc</file>  
        </source>  
        <destination type="file" exist="overwrite">  
          <file>c:\destinationfiles\reports.doc</file>  
        </destination>  
      </item>  
    </transferSet>  
  </managedTransfer>  
</request>
```

Bu aktarma başlatılmadan önce, <destinationAgent> ögesinin aracı özniteliğinin değeri, LONDON ya da PARIS ile değiştirilir.

İzleyici çıkış sınıfında ve görev tanımlaması XML 'indeki yerine koyma değişkenini büyük harfle belirtmeniz gerekir.

### **İlgili kavramlar**

[“MFT kullanıcı çıkışları için meta veri” sayfa 2093](#)

There are three different types of metadata that can be supplied to user exit routines for Managed File Transfer: environment, transfer, and file metadata. Bu meta veri, Java anahtar-değer çiftleri eşlemleri olarak sunulur.

[“MFT kullanıcı çıkışları için Java arabirimleri” sayfa 2102](#)

Kullanıcı çıkış yordamları için Java arabirimlerine ilişkin başvuru bilgileri için bu bölümdeki konuları kullanın.

[MFT kaynak ve hedef kullanıcı çıkışları](#)

### **İlgili görevler**

[Kullanıcı çıkışlarıyla MFT 'in uyarlanması](#)

### **İlgili başvurular**

[“Kullanıcı çıkışlarına ilişkin MFT Agent özellikleri” sayfa 2100](#)

agent.properties dosyasındaki standart özelliklerin yanı sıra, özellikle kullanıcı çıkışı yordamlarıyla ilgili olarak birkaç gelişmiş özellik vardır. Bu özellikler varsayılan olarak içerilmez, bu nedenle bunların herhangi birini kullanmak istiyorsanız, agent.properties dosyasını el ile düzenlemeniz gerekir. Aracı çalışırken agent.properties dosyasında bir değişiklik yaparsanız, değişiklikleri almak için aracıyı durdurup yeniden başlatın.

## **Kullanıcı çıkışlarına ilişkin MFT Agent özellikleri**

agent.properties dosyasındaki standart özelliklerin yanı sıra, özellikle kullanıcı çıkışı yordamlarıyla ilgili olarak birkaç gelişmiş özellik vardır. Bu özellikler varsayılan olarak içerilmez, bu nedenle bunların herhangi birini kullanmak istiyorsanız, agent.properties dosyasını el ile düzenlemeniz gerekir. Aracı çalışırken agent.properties dosyasında bir değişiklik yaparsanız, değişiklikleri almak için aracıyı durdurup yeniden başlatın.



IBM WebSphere MQ 7.5 ya da üstü için, ortam değişkenleri, dosya ya da izin konumlarını temsil eden bazı Managed File Transfer özelliklerinde kullanılabilir. Bu, ürünün bir kısmını çalıştırırken kullanılan dosyaların ya da izinlerin konumlarının, kullanıcının süreci çalıştıran kullanıcı gibi değişikliklere bağlı olarak değişiklik göstermesini sağlar. Ek bilgi için [MFT özelliklerindeki ortam değişkenleri](#) başlıklı konuya bakın.

## Kullanıcı çıkışı yordam özellikleri

Kullanıcı çıkışı yordamları, aşağıdaki çizelgede listelenen sırada çağrılır. `agent.properties` dosyası hakkında daha fazla bilgi için bkz. [Gelişmiş aracı özellikleri: Kullanıcı çıkışı yordamı](#).

Özellik adı	Tanım
sourceTransferEndExitSınıfları	Kaynak aktarımı uç çıkış yordamını uygulayan sınıfların virgülle ayrılmış bir listesini belirtir.
sourceTransferStartExitSınıfları	Kaynak aktarımı başlatma çıkış yordamlarından oluşan sınıfların virgülle ayrılmış listesini belirtir.
destinationTransferStartExitSınıfları	Hedef aktarımı başlatma kullanıcı çıkışı yordamını gerçekleştiren sınıfların virgülle ayrılmış bir listesini belirtir.
destinationTransferEndExitSınıfları	Hedef aktarım kullanıcı çıkışı yordamını gerçekleştiren sınıfların virgülle ayrılmış bir listesini belirtir.
exitClassYolu	<p>Kullanıcı çıkışı yordamlarıyla ilgili sınıf yolu olarak işlev gören, platforma özgü, karakterlerle ayrılmış bir dizin listesini belirtir.</p> <p>Bu sınıf yolunda herhangi bir girişten önce aracı çıkış dizini aranır.</p> <p>Bu özelliği Windows üzerinde kullanıyorsanız, yol sınırlayıcı olarak eğik çizgi karakteri (/) kullanın, ters eğik çizgi karakteri (\) değil. Örneğin:</p> <pre>exitClassPath=C:/mycomp/mqft/exits/encryptFileExit.jar; C:/mycomp/mqft/exits/fileFilter.jar.</pre> <p>IBM WebSphere MQ 7.5 ya da sonraki bir yayın düzeyiyle ilgili olarak, bu özelliğin değeri ortam değişkenleri içerebilir.</p>
exitNativeLibraryPath	<p>Kullanıcı çıkış yordamları için yerel kitaplık yolu olarak işlev gösteren, platforma özgü, karakterlerle ayrılmış bir dizin listesini belirtir.</p> <p>IBM WebSphere MQ 7.5 ya da sonraki bir yayın düzeyiyle ilgili olarak, bu özelliğin değeri ortam değişkenleri içerebilir.</p>
monitorExitSınıflar	İzleme programı çıkış yordamını gerçekleştiren sınıfların virgülle ayrılmış listesini belirtir. Daha fazla bilgi için bkz. <a href="#">MFT kaynak izleyicisi kullanıcı çıkışları</a> .
protocolBridgeCredentialExitSınıflar	Protokol köprüsü kimlik bilgileri kullanıcı çıkışı yordamını gerçekleştiren sınıfların virgülle ayrılmış listesini belirtir. Daha fazla bilgi için <a href="#">Çıkış sınıflarını kullanarak dosya sunucusuna ilişkin kimlik bilgilerini eşleme</a> başlıklı konuya bakın.
protocolBridgePropertiesExitSınıflar	Protokol köprüsü sunucusu özellikleri kullanıcı çıkışı yordamını uygulayan sınıfların virgülle ayrılmış listesini belirtir. Daha fazla bilgi için bkz. <a href="#">ProtocolBridgePropertiesExit2: İletişim kuralı dosya sunucusu özelliklerinin araştırılması</a> .
IOExitClasses	G/Ç kullanıcı çıkışı yordamını gerçekleştiren sınıfların virgülle ayrılmış listesini belirtir. Yalnızca IOExit arabirimini gerçekleştiren sınıfları listeleysin; örneğin, IOExitResourcePath ve IOExitChannel gibi diğer G/Ç kullanıcı çıkışı arabirimlerini uygulayan sınıfları listeleme. Daha fazla bilgi için bkz. <a href="#">MFT aktarım G/Ç kullanıcı çıkışlarının kullanılması</a> .

## Çıkış çağrısının sırası

Kaynak ve hedef çıkışlar aşağıdaki sırada çağrılır:

1. SourceTransferStartExit
2. DestinationTransferStartExit

3. DestinationTransferEndExit

4. SourceTransferEndExit

## Zincirleme kaynak ve hedef çıkışları

Birden çok çıkış belirlerseniz, önce listedeki ilk çıkış çağrılır, ikinci çıkıştan sonra ikinci çıkıştan sonra bu şekilde devam eder. İlk çıkışta yapılan değişiklikler, daha sonra çağrılan ve bu şekilde başlatılacak çıkışa giriş olarak geçirilir. Örneğin, iki kaynak aktarımı başlatılırsa, ilk çıkışa göre aktarma meta verilerinde yapılan değişikliklerden ikinci çıkışa giriş yapılır. Her çıkış kendi sonucunu döndürür. Verili bir tipteki tüm çıkışlar bir aktarma sonucu kodu olarak DEVAM EDİLSİN olursa, genel sonuç DEVAM EDER. Bir ya da daha çok çıkış CANCEL\_TRANSFER sonucunu döndürürse, genel sonuç CANCEL\_TRANSFER olur. Çıkışların döndürdüğü sonuç kodlarının ve dizelerin tümü, aktarma günlüğünde çıkışa neden olur.

Kaynak aktarımı başlangıç çıkıştan gelen genel sonuç DEVAM ederse, aktarma, çıkışlar tarafından yapılan değişiklikleri kullanarak devam eder. Genel sonuç CANCEL\_TRANSFER ise, kaynak aktarma uç çıkışları çağrılır ve aktarma işlemi iptal edilir. Aktarma günlüğündeki tamamlanma durumu "iptal edildi".

Hedef aktarma başlangıç çıkışlarından gelen genel sonuç DEVAM EDERSE, AKTARMA, çıkışlar tarafından yapılan değişiklikleri kullanarak devam eder. Genel sonuç CANCEL\_TRANSFER ise, hedef aktarma uç çıkışları çağrılır, sonra kaynak aktarma uç çıkışları çağrılır. Son olarak aktarma iptal edildi. Aktarma günlüğündeki tamamlanma durumu "iptal edildi".

Bir kaynak ya da hedef çıkışa ilişkin bilgileri zincirde ya da yürütme sırasında aşağıdaki çıkışlara geçirmesi gerekiyorsa, aktarma meta verileri güncellenerek gerçekleştirilmelidir. Aktarım meta verilerinin kullanımı, çıkış somutlaması özeldir. Örneğin, bir çıkış, dönüş sonucunu CANCEL\_TRANSFER olarak ayarlarsa ve aktarma işleminin iptal edildiği aşağıdaki çıkışlarla iletişim kurması gerekiyorsa, diğer çıkışlar tarafından anlaşılacak şekilde bir veri aktarma meta veri değeri ayarlanarak gerçekleştirilmelidir.

## Örnek

```
sourceTransferStartExitClasses=com.ibm.wmqfte.test.MFTTestSourceTransferStartExit
sourceTransferEndExitClasses=com.ibm.wmqfte.test.MFTTestSourceTransferEndExit
destinationTransferStartExitClasses=com.ibm.wmqfte.test.MFTTestDestinationTransferStartExit
destinationTransferEndExitClasses=com.ibm.wmqfte.test.MFTTestDestinationTransferEndExit
exitClassPath=C:/mycomp/mqft/exits/encryptFileExit.jar;C:/mycomp/mqft/exits/fileFilter.jar
```

## İlgili kavramlar

[Kullanıcı çıkışlarıyla MFT ' in uyarlanması](#)

[“MFT kullanıcı çıkışları için meta veri” sayfa 2093](#)

There are three different types of metadata that can be supplied to user exit routines for Managed File Transfer: environment, transfer, and file metadata. Bu meta veri, Java anahtar-değer çiftleri eşlemleri olarak sunulur.

[“MFT kullanıcı çıkışları içinJava arabirimleri” sayfa 2102](#)

Kullanıcı çıkış yordamları için Java arabirimlerine ilişkin başvuru bilgileri için bu bölümdeki konuları kullanın.

## İlgili başvurular

[“MFT kaynak izleyicisi kullanıcı çıkışları” sayfa 2096](#)

Kaynak izleme programı kullanıcı çıkışları, bir izleme programının tetikleme koşulu yerine getirildiğinde, ilişkili görev başlatılmadan önce çalıştırılacak özel kodu yapılandırmanıza olanak tanır.

[MFT özelliklerindeki ortam değişkenleri](#)

[The MFT agent.properties file](#)

## MFT kullanıcı çıkışları içinJava arabirimleri

Kullanıcı çıkış yordamları için Java arabirimlerine ilişkin başvuru bilgileri için bu bölümdeki konuları kullanın.

## ***CDCredentialExit.java arabirimi***

### **CDCredentialExit.java**

```
/*
 * Licensed Materials - Property of IBM
 *
 * "Restricted Materials of IBM"
 *
 * 5724-H72
 *
 * © Copyright IBM Corp. 2011, 2024. All Rights Reserved.
 *
 * US Government Users Restricted Rights - Use, duplication or
 * disclosure restricted by GSA ADP Schedule Contract with
 * IBM Corp.
 */
package com.ibm.wmqfte.exitroutine.api;

import java.util.Map;

/**
 * An interface that is implemented by classes that are invoked as part of
 * user exit routine processing. This interface defines methods that are
 * invoked by a Connect:Direct bridge agent to map the IBM MQ user ID of the transfer to credentials
 * that are used to access the Connect:Direct node.
 * There will be one instance of each implementation class per Connect:Direct bridge agent. The methods
 * can be called from different threads so the methods must be synchronized.
 */
public interface CDCredentialExit {

    /**
     * Invoked once when a Connect:Direct bridge agent is started. It is intended to initialize
     * any resources that are required by the exit
     *
     * @param bridgeProperties
     *     The values of properties defined for the Connect:Direct bridge.
     *     These values can only be read, they cannot be updated by
     *     the implementation.
     *
     * @return true if the initialisation is successful and false if unsuccessful
     *     If false is returned from an exit the Connect:Direct bridge agent does not
     *     start.
     */
    public boolean initialize(final Map<String, String> bridgeProperties);

    /**
     * Invoked once per transfer to map the IBM MQ user ID in the transfer message to the
     * credentials to be used to access the Connect:Direct node.
     *
     * @param mqUserId The IBM MQ user ID from which to map to the credentials to be used
     *     to access the Connect:Direct node
     * @param snode The name of the Connect:Direct SNODE specified as the cdNode in the
     *     file path. This is used to map the correct user ID and password for the
     *     SNODE.
     * @return A credential exit result object that contains the result of the map and
     *     the credentials to use to access the Connect:Direct node
     */
    public CDCredentialExitResult mapMQUserId(final String mqUserId, final String snode);

    /**
     * Invoked once when a Connect:Direct bridge agent is shutdown. This method releases
     * any resources that were allocated by the exit
     *
     * @param bridgeProperties
     *     The values of properties defined for the Connect:Direct bridge.
     *     These values can only be read, they cannot be updated by
     *     the implementation.
     *
     * @return
     */
    public void shutdown(final Map<String, String> bridgeProperties); }
}
```

## ***CredentialExitResult.java arabirimi***

## CredentialExitResult.java

```
/*
 * Licensed Materials - Property of IBM
 *
 * "Restricted Materials of IBM"
 *
 * 5724-H72
 *
 * © Copyright IBM Corp. 2008, 2024. All Rights Reserved.
 *
 * US Government Users Restricted Rights - Use, duplication or
 * disclosure restricted by GSA ADP Schedule Contract with
 * IBM Corp.
 */

package com.ibm.wmqfte.exitroutine.api;

/**
 * The result of invoking a Credential mapMQUserId exit method. It is composed of a result
 * code, which determines whether the mapping of the user id was successful, and an optional
 * Credentials object if the mapping is successful.
 */
public class CredentialExitResult {

    private final CredentialExitResultCode resultCode;
    private final Credentials credentials;

    /**
     * Constructor. Creates a credential exit result object with a specified result
     * code and optionally credentials.
     *
     * @param resultCode
     *     The result code to associate with the exit result being created.
     *
     * @param credentials
     *     The credentials to associate with the exit result being created.
     *     A value of <code>null</code> can be specified to indicate no
     *     credentials. If the resultCode is USER_SUCCESSFULLY_MAPPED the
     *     credentials must be set to a non-null value,
     */
    public CredentialExitResult(CredentialExitResultCode resultCode, Credentials credentials) {
        this.resultCode = resultCode;
        this.credentials = credentials;
    }

    /**
     * Returns the result code associated with this credential exit result
     *
     * @return
     *     the result code associated with this exit result.
     */
    public CredentialExitResultCode getResultCode() {
        return resultCode;
    }

    /**
     * Returns the credentials associated with this credential exit result
     *
     * @return
     *     the explanation associated with this credential exit result.
     */
    public Credentials getCredentials() {
        return credentials;
    }
}
```

### İlgili görevler

[MFT ürününü kullanıcı çıkışlarıyla özelleştirme](#)

### İlgili başvurular

[“SourceTransferStartExit.java arabirimi” sayfa 2131](#)

[“DestinationTransferStartExit.java arabirimi” sayfa 2106](#)

[“DestinationTransferEndExit.java arabirimi” sayfa 2105](#)

[“MonitorExit.java arabirimi” sayfa 2124](#)

## ***DestinationTransferEndExit.java arabirimi***

### **DestinationTransferEndExit.java**

```
/*
 * Licensed Materials - Property of IBM
 *
 * "Restricted Materials of IBM"
 *
 * 5724-H72
 *
 * © Copyright IBM Corp. 2008, 2024. All Rights Reserved.
 *
 * US Government Users Restricted Rights - Use, duplication or
 * disclosure restricted by GSA ADP Schedule Contract with
 * IBM Corp.
 */
package com.ibm.wmqfte.exitpoint.api;

/**
 * An interface that is implemented by classes that want to be invoked as part of
 * user exit routine processing. This interface defines a method that will be
 * invoked immediately after completing a transfer on the agent acting as the
 * destination of the transfer.
 */
public interface DestinationTransferEndExit {

    /**
     * Invoked immediately after the completion of a transfer on the agent acting as
     * the destination of the transfer.
     *
     * @param transferExitResult
     *     a result object reflecting whether or not the transfer completed
     *     successfully.
     *
     * @param sourceAgentName
     *     the name of the agent acting as the source of the transfer.
     *
     * @param destinationAgentName
     *     the name of the agent acting as the destination of the
     *     transfer. This is the name of the agent that the
     *     implementation of this method will be invoked from.
     *
     * @param environmentMetaData
     *     meta data about the environment in which the implementation
     *     of this method is running. This information can only be read,
     *     it cannot be updated by the implementation. The constants
     *     defined in <code>EnvironmentMetaDataConstants</code> class can
     *     be used to access the data held by this map.
     *
     * @param transferMetaData
     *     meta data to associate with the transfer. The information can
     *     only be read, it cannot be updated by the implementation. This
     *     map may also contain keys with IBM reserved names. These
     *     entries are defined in the <code>TransferMetaDataConstants</code>
     *     class and have special semantics.
     *
     * @param fileResults
     *     a list of file transfer result objects that describe the source
     *     file name, destination file name and result of each file transfer
     *     operation attempted.
     *
     * @return
     *     an optional description to enter into the log message describing
     *     transfer completion. A value of <code>null</code> can be used
     *     when no description is required.
     */
    String onDestinationTransferEnd(TransferExitResult transferExitResult,
        String sourceAgentName,
        String destinationAgentName,
        Map<String, String>environmentMetaData,
        Map<String, String>transferMetaData,
        List<FileTransferResult>fileResults);
}
```

## İlgili görevler

MFT ürününü kullanıcı çıktılarıyla özelleştirme

## İlgili başvurular

[“SourceTransferStartExit.java arabirimi” sayfa 2131](#)

[“SourceTransferEndExit.java arabirimi” sayfa 2129](#)

[“DestinationTransferStartExit.java arabirimi” sayfa 2106](#)

[“MonitorExit.java arabirimi” sayfa 2124](#)

[“ProtocolBridgeCredentialExit.java arabirimi” sayfa 2125](#)

## ***DestinationTransferStartExit.java arabirimi***

### **DestinationTransferStartExit.java**

```
/*
 * Licensed Materials - Property of IBM
 *
 * "Restricted Materials of IBM"
 *
 * 5724-H72
 *
 * © Copyright IBM Corp. 2008, 2024. All Rights Reserved.
 *
 * US Government Users Restricted Rights - Use, duplication or
 * disclosure restricted by GSA ADP Schedule Contract with
 * IBM Corp.
 */
package com.ibm.wmqfte.exitpoint.api;

/**
 * An interface that is implemented by classes that want to be invoked as part of
 * user exit routine processing. This interface defines a method that will be
 * invoked immediately prior to starting a transfer on the agent acting as the
 * destination of the transfer.
 */
public interface DestinationTransferStartExit {

    /**
     * Invoked immediately prior to starting a transfer on the agent acting as
     * the destination of the transfer.
     *
     * @param sourceAgentName
     *         the name of the agent acting as the source of the transfer.
     *
     * @param destinationAgentName
     *         the name of the agent acting as the destination of the
     *         transfer. This is the name of the agent that the
     *         implementation of this method will be invoked from.
     *
     * @param environmentMetaData
     *         meta data about the environment in which the implementation
     *         of this method is running. This information can only be read,
     *         it cannot be updated by the implementation. The constants
     *         defined in EnvironmentMetaDataConstants class can
     *         be used to access the data held by this map.
     *
     * @param transferMetaData
     *         meta data to associate with the transfer. The information can
     *         only be read, it cannot be updated by the implementation. This
     *         map may also contain keys with IBM reserved names. These
     *         entries are defined in the TransferMetaDataConstants
     *         class and have special semantics.
     *
     * @param fileSpecs
     *         a list of file specifications that govern the file data to
     *         transfer. The implementation of this method can modify the
     *         entries in this list and the changes will be reflected in the
     *         files transferred. However, new entries may not be added and
     *         existing entries may not be removed.
     *
     * @return
     *         a transfer exit result object which is used to determine if the
     *         transfer should proceed, or be cancelled.
     */
}
```

```
TransferExitResult onDestinationTransferStart(String sourceAgentName,
                                             String destinationAgentName,
                                             Map<String, String> environmentMetaData,
                                             Map<String, String> transferMetaData,
                                             List<Reference<String>> fileSpecs);
```

## İlgili görevler

[MFT ürününü kullanıcı çıktılarıyla özelleştirme](#)

## İlgili başvurular

[“SourceTransferStartExit.java arabirimi” sayfa 2131](#)

[“SourceTransferEndExit.java arabirimi” sayfa 2129](#)

[“DestinationTransferEndExit.java arabirimi” sayfa 2105](#)

[“MonitorExit.java arabirimi” sayfa 2124](#)

[“ProtocolBridgeCredentialExit.java arabirimi” sayfa 2125](#)

## ***FileTransferResult.java arabirimi***

### **FileTransferResult.java**

```
/*
 * Licensed Materials - Property of IBM
 *
 * "Restricted Materials of IBM"
 *
 * 5724-H72
 *
 * © Copyright IBM Corp. 2008, 2024. All Rights Reserved.
 *
 * US Government Users Restricted Rights - Use, duplication or
 * disclosure restricted by GSA ADP Schedule Contract with
 * IBM Corp.
 */

package com.ibm.wmqfte.exitroutine.api;

/**
 * Result information about a file transfer.
 */
public interface FileTransferResult {

    /** An enumeration for the <code>getCorrelatorType()</code> method. */
    public enum CorrelationInformationType {
        /** No correlation information is available for this result */
        NONE,
        /**
         * The correlation information relates to work done in
         * IBM Sterling File Gateway.
         */
        SFG
    }

    /**
     * Returns the source file specification, from which the file was transferred.
     *
     * @return the source file specification, from which the file was
     * transferred.
     */
    String getSourceFileSpecification();

    /**
     * Returns the destination file specification, to which the file was transferred.
     *
     * @return the destination file specification, to which the file was
     * transferred. A value of <code>null</code> may be returned
     * if the transfer did not complete successfully.
     */
    String getDestinationFileSpecification();

    /**
     * Returns the result of the file transfer operation.
     *
     * @return the result of the file transfer operation.
     */
}
```

```

    */
    FileExitResult getExitResult();

    /**
     * @return an enumerated value that identifies the product to which this correlating
     *         information relates.
     */
    CorrelationInformationType getCorrelatorType();

    /**
     * @return the first string component of the correlating identifier that relates
     *         this transfer result to work done in another product. A value of null
     *         may be returned either because the other product does not utilize a
     *         string based correlation information or because there is no correlation
     *         information.
     */
    String getString1Correlator();

    /**
     * @return the first long component of the correlating identifier that relates
     *         this transfer result to work done in another product. A value of zero
     *         is returned when there is no correlation information or the other
     *         product does not utilize long based correlation information or because
     *         the value really is zero!
     */
    long getLong1Correlator();
}

```

## İlgili görevler

[MFT ürününü kullanıcı çıkışlarıyla özelleştirme](#)

## İlgili başvurular

[“SourceTransferStartExit.java arabirimi” sayfa 2131](#)

[“DestinationTransferStartExit.java arabirimi” sayfa 2106](#)

[“DestinationTransferEndExit.java arabirimi” sayfa 2105](#)

[“MonitorExit.java arabirimi” sayfa 2124](#)

[“ProtocolBridgeCredentialExit.java arabirimi” sayfa 2125](#)

## ***IOExit.java arabirimi***

### **IOExit.java**

```

/*
 * Licensed Materials - Property of IBM
 *
 * "Restricted Materials of IBM"
 *
 * 5724-H72
 *
 * © Copyright IBM Corp. 2011, 2024. All Rights Reserved.
 *
 * US Government Users Restricted Rights - Use, duplication or
 * disclosure restricted by GSA ADP Schedule Contract with
 * IBM Corp.
 */
package com.ibm.wmqfte.exitroutine.api;

import java.io.IOException;
import java.util.Map;

import com.ibm.wmqfte.exitroutine.api.IOExitRecordResourcePath.RecordFormat;

/**
 * An interface that is implemented by classes that you want to be invoked as
 * part of user exit routine processing. This interface defines methods that
 * will be invoked during transfers to perform the underlying file system I/O
 * work for WMQFTE transfers.
 * <p>
 * The {@link #initialize(Map)} method will be called once when the exit is
 * first installed. The WMQFTE agent properties are passed to this method, thus
 * enabling the exit to understand its environment.
 * <p>
 * The {@link #isSupported(String)} method will be invoked during WMQFTE

```



```

* transfers to determine whether the user exit should be used. If the
* {@link #isSupported(String)} method returns a value of {@code true}, the
* {@link #newPath(String)} method will be invoked for the paths specified for
* the transfer request. The returned {@link IOExitPath} instance from a
* {@link #newPath(String)} method invocation will then be used by the WMQFTE
* transfer to obtain information about the resource and to transfer data to or
* from the resource.
* <p>
* To obtain transfer context for an I/O exit, a {@link SourceTransferStartExit}
* or {@link DestinationTransferStartExit} as appropriate, should be installed
* to enable information to be seen by this exit. The
* {@link SourceTransferStartExit} or {@link DestinationTransferStartExit} are
* passed the transfer's environment, metadata, and a list of file
* specifications for the transfer. The paths for the file specifications are
* the paths passed to the I/O exit's {@link #newPath(String)} method.
* <p>
* Note also that the {@link #isSupported(String)} and {@link #newPath(String)}
* methods might be called at other times by a WMQFTE agent and not just during
* transfers. For example, at transfer setup time the I/O system is queried to
* resolve the full resource paths for transfer.
*/
public interface IOExit {

    /**
     * Invoked once when the I/O exit is first required for use. It is intended
     * to initialize any resources that are required by the exit.
     *
     * @param agentProperties
     *     The values of properties defined for the WMQFTE agent. These
     *     values can only be read, they cannot be updated by the
     *     implementation.
     * @return {@code true} if the initialization is successful and {@code
     *         false} if unsuccessful. If {@code false} is returned from an
     *         exit, the exit will not be used.
     */
    boolean initialize(final Map<String, String> agentProperties);

    /**
     * Indicates whether this I/O user exit supports the specified path.
     * <p>
     * This method is used by WMQFTE to determine whether the I/O user exit
     * should be used within a transfer. If no I/O user exit returns true for
     * this method, the default WMQFTE file I/O function will be used.
     *
     * @param path
     *     The path to the required I/O resource.
     * @return {@code true} if the specified path is supported by the I/O exit,
     *         {@code false} otherwise
     */
    boolean isSupported(String path);

    /**
     * Obtains a new {@link IOExitPath} instance for the specified I/O resource
     * path.
     * <p>
     * This method will be invoked by WMQFTE only if the
     * {@link #isSupported(String)} method has been called for the path and
     * returned {@code true}.
     *
     * @param path
     *     The path to the required I/O resource.
     * @return A {@link IOExitPath} instance for the specified path.
     * @throws IOException
     *     If the path cannot be created for any reason.
     */
    IOExitPath newPath(String path) throws IOException;

    /**
     * Obtains a new {@link IOExitPath} instance for the specified I/O resource
     * path and passes record format and length information required by the
     * WMQFTE transfer.
     * <p>
     * Typically this method will be called for the following cases:
     * <ul>
     * <li>A path where a call to {@link #newPath(String)} has previously
     * returned a {@link IOExitRecordResourcePath} instance and WMQFTE is
     * re-establishing a new {@link IOExitPath} instance for the path, from an
     * internally-serialized state. The passed recordFormat and recordLength
     * will be the same as those for the original
     * {@link IOExitRecordResourcePath} instance.</li>
     * <li>A transfer destination path where the source of the transfer is
     * record oriented. The passed recordFormat and recordLength will be the

```

```

* same as those for the source.</li>
* </ul>
* The implementation can act on the record format and length information as
* deemed appropriate. For example, for a destination agent if the
* destination does not already exist and the source of the transfer is
* record oriented, the passed recordFormat and recordLength information
* could be used to create an appropriate record-oriented destination path.
* If the destination path already exists, the passed recordFormat and
* recordLength information could be used to perform a compatibility check
* and throw an {@link IOException} if the path is not compatible. A
* compatibility check could ensure that a record oriented path's record
* format is the same as the passed record format or that the record length
* is greater or equal to the passed record length.
* <p>
* This method will be invoked by WMQFTE only if the
* {@link #isSupported(String)} method has been called for the path and
* returned {@code true}.
*
* @param path
*         The path to the required I/O resource.
* @param recordFormat
*         The advised record format.
* @param recordLength
*         The advised record length.
* @return A {@link IOExitPath} instance for the specified path.
* @throws IOException
*         If the path cannot be created for any reason. For example,
*         the passed record format or length is incompatible with the
*         path's actual record format or length.
*/
IOExitPath newPath(String path, RecordFormat recordFormat, int recordLength)
    throws IOException;

```

## İlgili görevler

[MFT aktarım G/Ç kullanıcı çıkışlarının kullanılması](#)

[MFT ürününü kullanıcı çıkışlarıyla özelleştirme](#)

## ***IOExitChannel.java arabirimi***

### **IOExitChannel.java**

```

/*
 * Licensed Materials - Property of IBM
 *
 * "Restricted Materials of IBM"
 *
 * 5724-H72
 *
 * □ Copyright IBM Corp. 2011, 2024. All Rights Reserved.
 *
 * US Government Users Restricted Rights - Use, duplication or
 * disclosure restricted by GSA ADP Schedule Contract with
 * IBM Corp.
 */
package com.ibm.wmqfte.exitroutine.api;

import java.io.IOException;
import java.nio.ByteBuffer;

/**
 * Represents a channel that enables data to be read from or written to an
 * {@link IOExitResourcePath} resource.
 */
public interface IOExitChannel {

    /**
     * Obtains the data size for the associated {@link IOExitResourcePath} in
     * bytes.
     *
     * @return The data size in bytes.
     * @throws IOException
     *         If a problem occurs while attempting obtain the size.
     */
    long size() throws IOException;

}

```

```

* Closes the channel, flushing any buffered write data to the resource and
* releasing any locks.
*
* @throws RecoverableIOException
*         If a recoverable problem occurs while closing the resource.
*         This means that WMQFTE can attempt to recover the transfer.
* @throws IOException
*         If some other I/O problem occurs. For example, the channel might
*         already be closed.
*/
void close() throws RecoverableIOException, IOException;

/**
* Reads data from this channel into the given buffer, starting at this
* channel's current position, and updates the current position by the
* amount of data read.
* <p>
* Data is copied into the buffer starting at its current position and up to
* its limit. On return, the buffer's position is updated to reflect the
* number of bytes read.
*
* @param buffer
*         The buffer that the data is to be copied into.
* @return The number of bytes read, which might be zero, or -1 if the end of
*         data has been reached.
* @throws RecoverableIOException
*         If a recoverable problem occurs while reading the data. For a
*         WMQFTE transfer this means that it will attempt to recover.
* @throws IOException
*         If some other I/O problem occurs. For a WMQFTE transfer this
*         means that it will be failed.
*/
int read(ByteBuffer buffer) throws RecoverableIOException, IOException;

/**
* Writes data to this channel from the given buffer, starting at this
* channel's current position, and updates the current position by the
* amount of data written. The channel's resource is grown to accommodate
* the data, if necessary.
* <p>
* Data is copied from the buffer starting at its current position and up to
* its limit. On return, the buffer's position is updated to reflect the
* number of bytes written.
*
* @param buffer
*         The buffer containing the data to be written.
* @return The number of bytes written, which might be zero.
* @throws RecoverableIOException
*         If a recoverable problem occurs while writing the data. For a
*         WMQFTE transfer this means that it will attempt to recover.
* @throws IOException
*         If some other I/O problem occurs. For a WMQFTE transfer this
*         means that it will be failed.
*/
int write(ByteBuffer buffer) throws RecoverableIOException, IOException;

/**
* Forces any updates to this channel's resource to be written to its
* storage device.
* <p>
* This method is required to force changes to both the resource's content
* and any associated metadata to be written to storage.
*
* @throws RecoverableIOException
*         If a recoverable problem occurs while performing the force.
*         For a WMQFTE transfer this means that it will attempt to
*         recover.
* @throws IOException
*         If some other I/O problem occurs. For a WMQFTE transfer this
*         means that it will be failed.
*/
void force() throws RecoverableIOException, IOException;

/**
* Attempts to lock the entire resource associated with the channel for
* shared or exclusive access.
* <p>
* The intention is for this method not to block if the lock is currently
* unavailable.
*
* @param shared
*         {@code true} if a shared lock is required, {@code false} if an

```

```

*         exclusive lock is required.
* @return A {@link IOExitLock} instance representing the newly acquired
*         lock or null if the lock cannot be obtained.
* @throws IOException
*         If a problem occurs while attempting to acquire the lock.
*/
IOExitLock tryLock(boolean shared) throws IOException;
}

```

## İlgili görevler

[MFT aktarım G/Ç kullanıcı çıkışlarının kullanılması](#)

[MFT ürününü kullanıcı çıkışlarıyla özelleştirme](#)

## *IOExitLock.java arabirimi*

### IOExitLock.java

```

/*
 * Licensed Materials - Property of IBM
 *
 * "Restricted Materials of IBM"
 *
 * 5724-H72
 *
 * © Copyright IBM Corp. 2011, 2024. All Rights Reserved.
 *
 * US Government Users Restricted Rights - Use, duplication or
 * disclosure restricted by GSA ADP Schedule Contract with
 * IBM Corp.
 */
package com.ibm.wmqfte.exitroutine.api;

import java.io.IOException;

/**
 * Represents a lock on a resource for either shared or exclusive access.
 * {@link IOExitLock} instances are returned from
 * {@link IOExitChannel#tryLock(boolean)} calls and WMQFTE will request the
 * release of the lock at the appropriate time during a transfer. Additionally, when
 * a {@link IOExitChannel#close()} method is called it will be the
 * responsibility of the channel to release any associated locks.
 */
public interface IOExitLock {

    /**
     * Releases the lock.
     * <p>
     * After this method has been successfully called the lock is to be deemed as invalid.
     *
     * @throws IOException
     *         If the channel associated with the lock is not open or
     *         another problem occurs while attempting to release the lock.
     */
    void release() throws IOException;

    /**
     * Indicates whether this lock is valid.
     * <p>
     * A lock is considered valid until its @ {@link #release()} method is
     * called or the associated {@link IOExitChannel} is closed.
     *
     * @return {@code true} if this lock is valid, {@code false} otherwise.
     */
    boolean isValid();

    /**
     * @return {@code true} if this lock is for shared access, {@code false} if
     *         this lock is for exclusive access.
     */
    boolean isShared();
}

```

## İlgili görevler

[MFT aktarım G/Ç kullanıcı çıkışlarının kullanılması](#)

## ***IOExitPath.java arabirimi***

### **IOExitPath.java**

```
/*
 * Licensed Materials - Property of IBM
 *
 * "Restricted Materials of IBM"
 *
 * 5724-H72
 *
 * □ Copyright IBM Corp. 2011, 2024. All Rights Reserved.
 *
 * US Government Users Restricted Rights - Use, duplication or
 * disclosure restricted by GSA ADP Schedule Contract with
 * IBM Corp.
 */
package com.ibm.wmqfte.exitroutine.api;

/**
 * Represents an abstract path that can be inspected and queried by WMQFTE for
 * transfer purposes.
 * <p>
 * There are two types of path supported:
 * <ul>
 * <li>{@link IOExitResourcePath} - Represents a path that denotes a data
 * resource. For example, a file, directory, or group of database records.</li>
 * <li>{@link IOExitWildcardPath} - Represents a wildcard path that can be
 * expanded to multiple {@link IOExitResourcePath} instances.</li>
 * </ul>
 */
public abstract interface IOExitPath {

    /**
     * Obtains the abstract path as a {@link String}.
     *
     * @return The abstract path as a {@link String}.
     */
    String getPath();

    /**
     * Obtains the name portion of this abstract path as a {@link String}.
     * <p>
     * For example, a UNIX-style file system implementation evaluates the
     * path {@code /home/fteuser/file1.txt} as having a name of {@code
     * file1.txt}.
     *
     * @return the name portion of this abstract path as a {@link String}.
     */
    String getName();

    /**
     * Obtains the parent path for this abstract path as a {@link String}.
     * <p>
     * For example, a UNIX-style file system implementation evaluates the
     * path {@code /home/fteuser/file1.txt} as having a parent path of {@code
     * /home/fteuser}.
     *
     * @return The parent portion of the path as a {@link String}.
     */
    String getParent();

    /**
     * Obtains the abstract paths that match this abstract path.
     * <p>
     * If this abstract path denotes a directory resource, a list of paths
     * for all resources within the directory are returned.
     * <p>
     * If this abstract path denotes a wildcard, a list of all paths
     * matching the wildcard are returned.
     * <p>
     * Otherwise null is returned, because this abstract path probably denotes a
     * single file resource.
     *
     * @return An array of {@link IOExitResourcePath}s that
     * match this path, or null if this method is not applicable.
     */
}
```

```
*/
IOExitResourcePath[] listPaths();
}
```

## İlgili görevler

[MFT aktarım G/Ç kullanıcı çıkışlarının kullanılması](#)

[MFT ürününü kullanıcı çıkışlarıyla özelleştirme](#)

## ***IOExitProperties.java arabirimi***

### **IOExitProperties.java**

```
/*
 * Licensed Materials - Property of IBM
 *
 * "Restricted Materials of IBM"
 *
 * 5724-H72
 *
 * © Copyright IBM Corp. 2011, 2024. All Rights Reserved.
 *
 * US Government Users Restricted Rights - Use, duplication or
 * disclosure restricted by GSA ADP Schedule Contract with
 * IBM Corp.
 */
package com.ibm.wmqfte.exitroutine.api;

/**
 * Properties that determine how WMQFTE treats an {@link IOExitPath} for certain
 * aspects of I/O. For example, whether to use intermediate files.
 */
public class IOExitProperties {

    private boolean rereadSourceOnRestart = true;
    private boolean rechecksumSourceOnRestart = true;
    private boolean rechecksumDestinationOnRestart = true;
    private boolean useIntermediateFileAtDestination = true;
    private boolean requiresSingleThreadedChannelIO = false;

    /**
     * Determines whether the I/O exit implementation expects the resource to be
     * re-read from the start if a transfer is restarted.
     *
     * @return {@code true} if, on restart, the I/O exit expects the source
     * resource to be opened at the beginning and re-read from the
     * beginning (the {@link IOExitPath#openForRead(long)} method is
     * always invoked with 0L as an argument). {@code false} if, on
     * restart, the I/O exit expects the source to be opened at the
     * offset that the source agent intends to start reading from (the
     * {@link IOExitPath#openForRead(long)} method can be invoked with a
     * non-zero value as its argument).
     */
    public boolean getRereadSourceOnRestart() {
        return rereadSourceOnRestart;
    }

    /**
     * Sets the value to determine whether the I/O exit implementation expects
     * the resource to be re-read from the beginning if a transfer is restarted.
     * <p>
     * The default is {@code true}. The I/O exit should call this method when
     * required to change this value.
     *
     * @param rereadSourceOnRestart
     *        {@code true} if, on restart, the I/O exit expects the source
     *        resource to be opened at the beginning and re-read from the
     *        beginning (the {@link IOExitPath#openForRead(long)} method
     *        is always invoked with 0L as an argument). {@code false}
     *        if, on restart, the I/O exit expects the source to be opened
     *        at the offset that the source agent intends to start reading
     *        from (the {@link IOExitPath#openForRead(long)} method can be
     *        invoked with a non-zero value as its argument).
     */
    public void setRereadSourceOnRestart(boolean rereadSourceOnRestart) {
        this.rereadSourceOnRestart = rereadSourceOnRestart;
    }
}
```

```

/**
 * Determines whether the I/O exit implementation requires the source
 * resource to be re-checksummed if the transfer is restarted.
 * Re-checksumming takes place only if the
 * {@link #getRereadSourceOnRestart()} method returns {@code true}.
 *
 * @return {@code true} if, on restart, the I/O exit expects the already-
 * transferred portion of the source to be re-checksummed for
 * inconsistencies. Use this option in environments
 * where the source could be changed during a restart. {@code
 * false} if, on restart, the I/O exit does not require the
 * already-transferred portion of the source to be re-checksummed.
 */
public boolean getRechecksumSourceOnRestart() {
    return rechecksumSourceOnRestart;
}

/**
 * Sets the value to determine whether the I/O exit implementation requires
 * the source resource to be re-checksummed if the transfer is restarted.
 * Re-checksumming takes place only if the
 * {@link #getRereadSourceOnRestart()} method returns {@code true}.
 * <p>
 * The default is {@code true}. The I/O exit should call this method when
 * required to change this value.
 *
 * @param rechecksumSourceOnRestart
 *     {@code true} if, on restart, the I/O exit expects the already
 *     transferred portion of the source to be re-checksummed
 *     for inconsistencies. Use this option in environments
 *     where the source could be changed during a restart.
 *     {@code false} if, on restart, the I/O exit does not
 *     require the already-transferred portion of the source to be
 *     re-checksummed.
 */
public void setRechecksumSourceOnRestart(boolean rechecksumSourceOnRestart) {
    this.rechecksumSourceOnRestart = rechecksumSourceOnRestart;
}

/**
 * Determines whether the I/O exit implementation requires the destination
 * resource to be re-checksummed if the transfer is restarted.
 *
 * @return {@code true} if, on restart, the I/O exit expects the already
 * transferred portion of the destination to be re-checksummed to
 * check for inconsistencies. This option should be used in
 * environments where the destination could have been changed while
 * a restart is occurring. {@code false} if, on restart, the I/O exit
 * does not require the already transferred portion of the
 * destination to be re-checksummed.
 */
public boolean getRechecksumDestinationOnRestart() {
    return rechecksumDestinationOnRestart;
}

/**
 * Sets the value to determine whether the I/O exit implementation requires
 * the destination resource to be re-checksummed if the transfer is
 * restarted.
 * <p>
 * The default is {@code true}. The I/O exit should call this method when
 * required to change this value.
 *
 * @param rechecksumDestinationOnRestart
 *     {@code true} if, on restart, the I/O exit expects the already-
 *     transferred portion of the destination to be re-checksummed
 *     for inconsistencies. Use this option in environments
 *     where the destination could have been changed during a
 *     restart. {@code false} if, on restart, the I/O exit does not
 *     require the already-transferred portion of the destination
 *     to be re-checksummed.
 */
public void setRechecksumDestinationOnRestart(
    boolean rechecksumDestinationOnRestart) {
    this.rechecksumDestinationOnRestart = rechecksumDestinationOnRestart;
}

/**
 * Determines whether the I/O exit implementation requires the use of an
 * intermediate file when writing the data at the destination. The
 * intermediate file mechanism is typically used to prevent an incomplete

```

```

* destination resource from being processed.
*
* @return {@code true} if data should be written to an intermediate file at
*         the destination and then renamed (to the requested destination
*         path name as specified in the transfer request) after the transfer is
*         complete. {@code false} if data should be written directly to the
*         requested destination path name without the use of an
*         intermediate file.
*/
public boolean getUseIntermediateFileAtDestination() {
    return useIntermediateFileAtDestination;
}

/**
 * Sets the value to determine whether the I/O exit implementation requires
 * the use of an intermediate file when writing the data at the destination.
 * The intermediate file mechanism is typically used to prevent an
 * incomplete destination resource from being processed.
 *
 * <p>
 * The default is {@code true}. The I/O exit should call this method when
 * required to change this value.
 *
 * @param useIntermediateFileAtDestination
 *        {@code true} if data should be written to an intermediate file
 *        at the destination and then renamed (to the requested
 *        destination path name as specified in the transfer request) after
 *        the transfer is complete. {@code false} if data should be written
 *        directly to the requested destination path name without the
 *        use of an intermediate file
 */
public void setUseIntermediateFileAtDestination(
    boolean useIntermediateFileAtDestination) {
    this.useIntermediateFileAtDestination = useIntermediateFileAtDestination;
}

/**
 * Determines whether the I/O exit implementation requires
 * {@link IOExitChannel} instances to be accessed by a single thread only.
 *
 * @return {@code true} if {@link IOExitChannel} instances are to be
 *         accessed by a single thread only.
 */
public boolean requiresSingleThreadedChannelIO() {
    return requiresSingleThreadedChannelIO;
}

/**
 * Sets the value to determine whether the I/O exit implementation requires
 * channel operations for a particular instance to be accessed by a
 * single thread only.
 *
 * <p>
 * For certain I/O implementations it is necessary that resource path
 * operations such as open, read, write, and close are invoked only from a
 * single execution {@link Thread}. When set {@code true}, WMQFTE ensures
 * that the following are invoked on a single thread:
 *
 * <ul>
 * <li>{@link IOExitResourcePath#openForRead(long)} method and all methods of
 * the returned {@link IOExitChannel} instance.</li>
 * <li>{@link IOExitResourcePath#openForWrite(boolean)} method and all
 * methods of the returned {@link IOExitChannel} instance.</li>
 * </ul>
 *
 * <p>
 * This has a slight performance impact, hence enable single-threaded channel
 * I/O only when absolutely necessary.
 *
 * <p>
 * The default is {@code false}. The I/O exit should call this method when
 * required to change this value.
 *
 * @param requiresSingleThreadedChannelIO
 *        {@code true} if {@link IOExitChannel} instances are to be
 *        accessed by a single thread only.
 */
public void setRequiresSingleThreadedChannelIO(boolean requiresSingleThreadedChannelIO) {
    this.requiresSingleThreadedChannelIO = requiresSingleThreadedChannelIO;
}
}

```

## İlgili görevler

MFT aktarım G/Ç kullanıcı çıktıların kullanılması



## ***IOExitRecordChannel.java arabirimi***

### **IOExitRecordChannel.java**

```
/*
 * Licensed Materials - Property of IBM
 *
 * "Restricted Materials of IBM"
 *
 * 5724-H72
 *
 * © Copyright IBM Corp. 2011, 2024. All Rights Reserved.
 *
 * US Government Users Restricted Rights - Use, duplication or
 * disclosure restricted by GSA ADP Schedule Contract with
 * IBM Corp.
 */
package com.ibm.wmqfte.exitroutine.api;

import java.io.IOException;
import java.nio.ByteBuffer;

/**
 * Represents a channel that enables records of data to be read from or written
 * to an {@link IOExitRecordResourcePath} resource.
 * <p>
 * This is an extension of the {@link IOExitChannel} interface such that the
 * {@link #read(java.nio.ByteBuffer)} and {@link #write(java.nio.ByteBuffer)}
 * methods are expected to deal in whole records of data only. That is, the
 * {@link java.nio.ByteBuffer} returned from the read method and passed to the
 * write method is assumed to contain one or more complete records.
 */
public interface IOExitRecordChannel extends IOExitChannel {

    /**
     * Reads records from this channel into the given buffer, starting at this
     * channel's current position, and updates the current position by the
     * amount of data read.
     * <p>
     * Record data is copied into the buffer starting at its current position
     * and up to its limit. On return, the buffer's position is updated to
     * reflect the number of bytes read.
     * <p>
     * Only whole records are copied into the buffer.
     * <p>
     * For a fixed-record-format resource, this might be multiple records. The
     * amount of data in the return buffer does not necessarily need to be a
     * multiple of the record length, but the last record is still to be treated
     * as a complete record and padded as required by the caller.
     * <p>
     * For a variable-format resource, this is a single whole record of a size
     * corresponding to the amount of return data or multiple whole records with
     * all except the last being treated as records of maximum size.
     *
     * @param buffer
     *         The buffer that the record data is to be copied into.
     * @return The number of bytes read, which might be zero, or -1 if the end of
     *         data has been reached.
     * @throws RecoverableIOException
     *         If a recoverable problem occurs while reading the data. For a
     *         WMQFTE transfer this means that it will attempt to recover.
     * @throws IOException
     *         If some other I/O problem occurs, for example, if the passed
     *         buffer is insufficient to contain at least one complete
     *         record). For a WMQFTE transfer this means that it will be
     *         failed.
     */
    int read(ByteBuffer buffer) throws RecoverableIOException, IOException;

    /**
     * Writes records to this channel from the given buffer, starting at this
     * channel's current position, and updates the current position by the
     * amount of data written. The channel's resource is grown to accommodate
     * the data, if necessary.
     * <p>
     * Record data is copied from the buffer starting at its current position

```

```

* and up to its limit. On return, the buffer's position is updated to
* reflect the number of bytes written.
* <p>
* The buffer is expected to contain only whole records.
* <p>
* For a fixed-record-format resource, this might be multiple records and if
* there is insufficient data in the buffer for a complete record, the
* record is to be padded as required to complete the record.
* <p>
* For a variable-record format resource the buffer is normally expected to
* contain a single record of length corresponding to the amount of data
* within the buffer. However, if the amount of data within the buffer
* exceeds the maximum record length, the implementation can either:
* <ol>
* <li>throw an {@link IOException} indicating that it cannot handle the
* situation.</li>
* <li>Consume a record's worth of data from the buffer, leaving the remaining
* data within the buffer.</li>
* <li>Consume all the buffer data and just write what it can to the current
* record. This effectively truncates the data.</li>
* <li>Consume all the buffer data and write to multiple records.</li>
* </ol>
*
* @param buffer
*         The buffer containing the data to be written.
* @return The number of bytes written, which might be zero.
* @throws RecoverableIOException
*         If a recoverable problem occurs while writing the data. For a
*         WMQFTE transfer this means that it will attempt to recover.
* @throws IOException
*         If some other I/O problem occurs. For a WMQFTE transfer this
*         means that it will be failed.
*/
int write(ByteBuffer buffer) throws RecoverableIOException, IOException;
}

```

## İlgili görevler

[MFT aktarım G/Ç kullanıcı çıkışlarının kullanılması](#)

[MFT ürününü kullanıcı çıkışlarıyla özelleştirme](#)

## IOExitRecordResourcePath.java arabirimi

### IOExitRecordResourcePath.java

```

/*
 * Licensed Materials - Property of IBM
 *
 * "Restricted Materials of IBM"
 *
 * 5724-H72
 *
 * © Copyright IBM Corp. 2011, 2024. All Rights Reserved.
 *
 * US Government Users Restricted Rights - Use, duplication or
 * disclosure restricted by GSA ADP Schedule Contract with
 * IBM Corp.
 */
package com.ibm.wmqfte.exitroutine.api;

import java.io.IOException;

/**
 * Represents a path that denotes a record-oriented data resource (for example,
 * a z/OS data set). It allows the data to be located, the record format to be
 * understood, and {@link IOExitRecordChannel} instances to be created for read
 * or write operations.
 */
public interface IOExitRecordResourcePath extends IOExitResourcePath {

    /**
     * Record formats for record-oriented resources.
     */
    public enum RecordFormat {
        FIXED, VARIABLE
    }
}

```

```

/**
 * Obtains the record length for records that are maintained by the resource
 * denoted by this abstract path.
 * <p>
 * For a resource with fixed-length records, the data for each record read
 * and written is assumed to be this length.
 * <p>
 * For a resource with variable-length records, this is the maximum length
 * for a record's data.
 * <p>
 * This method should return a value greater than zero, otherwise it can
 * result in the failure of a WMQFTE transfer that involves this abstract
 * path.
 *
 * @return The record length, in bytes, for records maintained by the
 *         resource.
 */
int getRecordLength();

/**
 * Obtains record format, as a {@link RecordFormat} instance, for records
 * that are maintained by the resource denoted by this abstract path.
 *
 * @return A {@link RecordFormat} instance for the record format for records
 *         that are maintained by the resource denoted by this abstract
 *         path.
 */
RecordFormat getRecordFormat();

/**
 * Opens a {@link IOExitRecordChannel} instance for reading data from the
 * resource denoted by this abstract path. The current data byte position
 * for the resource is expected to be the passed position value, such that
 * when {@link IOExitRecordChannel#read(java.nio.ByteBuffer)} is called,
 * data starting from that position is read.
 * <p>
 * Note that the data byte read position will be on a record boundary.
 *
 * @param position
 *         The required data byte read position.
 * @return A new {@link IOExitRecordChannel} instance allowing data to be
 *         read from the resource denoted by this abstract path.
 * @throws RecoverableIOException
 *         If a recoverable problem occurs while attempting to open the
 *         resource for reading. This means that WMQFTE can attempt to
 *         recover the transfer.
 * @throws IOException
 *         If some other I/O problem occurs.
 */
IOExitRecordChannel openForRead(long position)
    throws RecoverableIOException, IOException;

/**
 * Opens a {@link IOExitRecordChannel} instance for writing data to the
 * resource denoted by this abstract path. Writing of data, using the
 * {@link IOExitRecordChannel#write(java.nio.ByteBuffer)} method, starts at
 * either the beginning of the resource or end of the current data for the
 * resource, depending on the specified append parameter.
 *
 * @param append
 *         When {@code true} indicates that data written to the resource
 *         should be appended to the end of the current data. When
 *         {@code false} indicates that writing of data is to start at
 *         the beginning of the resource; any existing data is lost.
 * @return A new {@link IOExitRecordChannel} instance allowing data to be
 *         written to the resource denoted by this abstract path.
 * @throws RecoverableIOException
 *         If a recoverable problem occurs while attempting to open the
 *         resource for writing. This means that WMQFTE can attempt to
 *         recover the transfer.
 * @throws IOException
 *         If some other I/O problem occurs.
 */
IOExitRecordChannel openForWrite(boolean append)
    throws RecoverableIOException, IOException;
}

```

## İlgili görevler

MFT aktarım G/Ç kullanıcı çıkışlarının kullanılması

## ***IOExitResourcePath.java arabirimi***

### **IOExitResourcePath.java**

```
/*
 * Licensed Materials - Property of IBM
 *
 * "Restricted Materials of IBM"
 *
 * 5724-H72
 *
 * © Copyright IBM Corp. 2011, 2024. All Rights Reserved.
 *
 * US Government Users Restricted Rights - Use, duplication or
 * disclosure restricted by GSA ADP Schedule Contract with
 * IBM Corp.
 */
package com.ibm.wmqfte.exitroutine.api;

import java.io.IOException;

/**
 * Represents a path that denotes a data resource (for example, a file,
 * directory, or group of database records). It allows the data to be located
 * and {@link IOExitChannel} instances to be created for read or write
 * operations.
 * <p>
 * There are two types of data resources as follows:
 * <ul>
 * <li>Directory - a container for other data resources. The
 * {@link #isDirectory()} method returns {@code true} for these.</li>
 * <li>File - a data container. This allows data to be read from or written to
 * it. The {@link #isFile()} method returns {@code true} for these.</li>
 * </ul>
 */
public interface IOExitResourcePath extends IOExitPath {

    /**
     * Creates a new {@link IOExitResourcePath} instance for a child path of the
     * resource denoted by this abstract path.
     * <p>
     * For example, with a UNIX-style path, {@code
     * IOExitResourcePath("/home/fteuser/test").newPath("subtest")} could be
     * equivalent to: {@code IOExitResourcePath("/home/fteuser/test/subtest")}
     *
     * @param child
     *         The child path name.
     * @return A new {@link IOExitResourcePath} instance that represents a child
     *         of this path.
     */
    IOExitResourcePath newPath(final String child);

    /**
     * Creates the directory path for the resource denoted by this abstract
     * path, including any necessary but nonexistent parent directories. If the
     * directory path already exists, this method has no effect.
     * <p>
     * If this operation fails, it might have succeeded in creating some of the
     * necessary parent directories.
     *
     * @throws IOException
     *         If the directory path cannot be fully created, when it does
     *         not already exist.
     */
    void makePath() throws IOException;

    /**
     * Obtains the canonical path of the abstract path as a {@link String}.
     * <p>
     * A canonical path is defined as being absolute and unique. For example,
     * the path can be represented as UNIX-style relative path: {@code
     * test/file.txt} but the absolute and unique canonical path representation
     * is: {@code /home/fteuser/test/file.txt}
     *
     * @return The canonical path as a {@link String}.
     * @throws IOException
     */
}
```

```

*           If the canonical path cannot be determined for any reason.
*/
String getCanonicalPath() throws IOException;

/**
 * Tests if this abstract path is an absolute path.
 * <p>
 * For example, a UNIX-style path, {@code /home/ftuser/test} is an absolute
 * path, whereas {@code ftuser/test} is not.
 *
 * @return {@code true} if this abstract path is an absolute path, {@code
 *         false} otherwise.
 */
boolean isAbsolute();

/**
 * Tests if the resource denoted by this abstract path exists.
 *
 * @return {@code true} if the resource denoted by this abstract path
 *         exists, {@code false} otherwise.
 * @throws IOException
 *         If the existence of the resource cannot be determined for any
 *         reason.
 */
boolean exists() throws IOException;

/**
 * Tests whether the calling application can read the resource denoted by
 * this abstract path.
 *
 * @return {@code true} if the resource for this path exists and can be
 *         read, {@code false} otherwise.
 * @throws IOException
 *         If a problem occurs while attempting to determine if the
 *         resource can be read.
 */
boolean canRead() throws IOException;

/**
 * Tests whether the calling application can modify the resource denoted by
 * this abstract path.
 *
 * @return {@code true} if the resource for this path exists and can be
 *         modified, {@code false} otherwise.
 * @throws IOException
 *         If a problem occurs while attempting to determine if the
 *         resource can be modified.
 */
boolean canWrite() throws IOException;

/**
 * Tests whether the specified user is permitted to read the resource
 * denoted by this abstract path.
 * <p>
 * When WMQFTE invokes this method, the user identifier is the MQMD user
 * identifier for the requesting transfer.
 *
 * @param userId
 *         User identifier to test for access.
 * @return {@code true} if the resource for this abstract path exists and is
 *         permitted to be read by the specified user, {@code false}
 *         otherwise.
 * @throws IOException
 *         If a problem occurs while attempting to determine if the user
 *         is permitted to read the resource.
 */
boolean readPermitted(String userId) throws IOException;

/**
 * Tests whether the specified user is permitted to modify the resource
 * denoted by this abstract path.
 * <p>
 * When WMQFTE invokes this method, the user identifier is the MQMD user
 * identifier for the requesting transfer.
 *
 * @param userId
 *         User identifier to test for access.
 * @return {@code true} if the resource for this abstract path exists and is
 *         permitted to be modified by the specified user, {@code false}
 *         otherwise.
 * @throws IOException
 *         If a problem occurs while attempting to determine if the user

```

```

*           is permitted to modify the resource.
*/
boolean writePermitted(String userId) throws IOException;

/**
 * Tests if the resource denoted by this abstract path is a directory-type
 * resource.
 *
 * @return {@code true} if the resource denoted by this abstract path is a
 *         directory type resource, {@code false} otherwise.
 */
boolean isDirectory();

/**
 * Creates the resource denoted by this abstract path, if it does not
 * already exist.
 *
 * @return {@code true} if the resource does not exist and was successfully
 *         created, {@code false} if the resource already existed.
 * @throws RecoverableIOException
 *         If a recoverable problem occurs while attempting to create
 *         the resource. This means that WMQFTE can attempt to recover
 *         the transfer.
 * @throws IOException
 *         If some other I/O problem occurs.
 */
boolean createNewPath() throws RecoverableIOException, IOException;

/**
 * Tests if the resource denoted by this abstract path is a file-type
 * resource.
 *
 * @return {@code true} if the resource denoted by this abstract path is a
 *         file type resource, {@code false} otherwise.
 */
boolean isFile();

/**
 * Obtains the last modified time for the resource denoted by this abstract
 * path.
 * <p>
 * This time is measured in milliseconds since the epoch (00:00:00 GMT,
 * January 1, 1970).
 *
 * @return The last modified time for the resource denoted by this abstract
 *         path, or a value of 0L if the resource does not exist or a
 *         problem occurs.
 */
long lastModified();

/**
 * Deletes the resource denoted by this abstract path.
 * <p>
 * If the resource is a directory, it must be empty for the delete to work.
 *
 * @throws IOException
 *         If the delete of the resource fails for any reason.
 */
void delete() throws IOException;

/**
 * Renames the resource denoted by this abstract path to the specified
 * destination abstract path.
 * <p>
 * The rename should still be successful if the resource for the specified
 * destination abstract path already exists and it is possible to replace
 * it.
 *
 * @param destination
 *         The new abstract path for the resource denoted by this
 *         abstract path.
 * @throws IOException
 *         If the rename of the resource fails for any reason.
 */
void renameTo(IOExitResourcePath destination) throws IOException;

/**
 * Creates a new path to use for writing to a temporary resource that did
 * not previously exist.
 * <p>
 * The implementation can choose the abstract path name for the temporary
 * resource. However, for clarity and problem diagnosis, the abstract path

```

```

* name for the temporary resource should be based on this abstract path
* name with the specified suffix appended and additional characters to make
* the path unique (for example, sequence numbers), as required.
* <p>
* When WMQFTE transfers data to a destination it normally attempts to first
* write to a temporary resource then on transfer completion renames the
* temporary resource to the required destination. This method is called by
* WMQFTE to create a new temporary resource path. The returned path should
* be new and the resource should not previously exist.
*
* @param suffix
*     Recommended suffix to use for the generated temporary path.
*
* @return A new {@link IOExitResourcePath} instance for the temporary
*     resource path, that did not previously exist.
* @throws RecoverableIOException
*     If a recoverable problem occurs whilst attempting to create
*     the temporary resource. This means that WMQFTE can attempt to
*     recover the transfer.
* @throws IOException
*     If some other I/O problem occurs.
*/
IOExitResourcePath createTempPath(String suffix)
    throws RecoverableIOException, IOException;

/**
* Opens a {@link IOExitChannel} instance for reading data from the resource
* denoted by this abstract path. The current data byte position for the
* resource is expected to be the passed position value, such that when
* {@link IOExitChannel#read(java.nio.ByteBuffer)} is called, data starting
* from that position is read.
*
* @param position
*     The required data byte read position.
* @return A new {@link IOExitChannel} instance allowing data to be read
*     from the resource denoted by this abstract path.
* @throws RecoverableIOException
*     If a recoverable problem occurs while attempting to open the
*     resource for reading. This means that WMQFTE can attempt to
*     recover the transfer.
* @throws IOException
*     If some other I/O problem occurs.
*/
IOExitChannel openForRead(long position) throws RecoverableIOException,
    IOException;

/**
* Opens a {@link IOExitChannel} instance for writing data to the resource
* denoted by this abstract path. Writing of data, using the
* {@link IOExitChannel#write(java.nio.ByteBuffer)} method, starts at either
* the beginning of the resource or end of the current data for the
* resource, depending on the specified append parameter.
*
* @param append
*     When {@code true} indicates that data written to the resource
*     should be appended to the end of the current data. When
*     {@code false} indicates that writing of data is to start at
*     the beginning of the resource; any existing data is lost.
* @return A new {@link IOExitChannel} instance allowing data to be written
*     to the resource denoted by this abstract path.
* @throws RecoverableIOException
*     If a recoverable problem occurs whilst attempting to open the
*     resource for writing. This means that WMQFTE can attempt to
*     recover the transfer.
* @throws IOException
*     If some other I/O problem occurs.
*/
IOExitChannel openForWrite(boolean append) throws RecoverableIOException,
    IOException;

/**
* Tests if the resource denoted by this abstract path is in use by another
* application. Typically, this is because another application has a lock on
* the resource either for shared or exclusive access.
*
* @return {@code true} if resource denoted by this abstract path is in use
*     by another application, {@code false} otherwise.
*/
boolean inUse();

/**
* Obtains a {@link IOExitProperties} instance for properties associated

```

```

* with the resource denoted by this abstract path.
* <p>
* WMQFTE will read these properties to govern how a transfer behaves when
* interacting with the resource.
*
* @return A {@link IOExitProperties} instance for properties associated
*         with the resource denoted by this abstract path.
*/
IOExitProperties.getProperties();
}

```

### İlgili görevler

[MFT aktarım G/Ç kullanıcı çıkışlarının kullanılması](#)

[MFT ürününü kullanıcı çıkışlarıyla özelleştirme](#)

### ***IOExitWildcardPath.java arabirimi***

#### **IOExitWildcardPath.java**

```

/*
 * Licensed Materials - Property of IBM
 *
 * "Restricted Materials of IBM"
 *
 * 5724-H72
 *
 * Copyright IBM Corp. 2011, 2024. All Rights Reserved.
 *
 * US Government Users Restricted Rights - Use, duplication or
 * disclosure restricted by GSA ADP Schedule Contract with
 * IBM Corp.
 */
package com.ibm.wmqfte.exitroutine.api;

/**
 * Represents a path that denotes a wildcard. This can be used to match multiple
 * resource paths.
 */
public interface IOExitWildcardPath extends IOExitPath {

```

### İlgili görevler

[MFT aktarım G/Ç kullanıcı çıkışlarının kullanılması](#)

[MFT ürününü kullanıcı çıkışlarıyla özelleştirme](#)

### ***MonitorExit.java arabirimi***

#### **MonitorExit.java**

```

/*
 * Licensed Materials - Property of IBM
 *
 * "Restricted Materials of IBM"
 *
 * 5724-H72
 *
 * Copyright IBM Corp. 2009, 2024. All Rights Reserved.
 *
 * US Government Users Restricted Rights - Use, duplication or
 * disclosure restricted by GSA ADP Schedule Contract with
 * IBM Corp.
 */
package com.ibm.wmqfte.exitroutine.api;

import java.util.Map;

/**
 * An interface that is implemented by classes that want to be invoked as part of
 * user exit routine processing. This interface defines a method that will be
 * invoked immediately prior to starting a task as the result of a monitor trigger
 */

```



```

public interface MonitorExit {

    /**
     * Invoked immediately prior to starting a task as the result of a monitor
     * trigger.
     *
     * @param environmentMetaData
     *      meta data about the environment in which the implementation
     *      of this method is running. This information can only be read,
     *      it cannot be updated by the implementation. The constant
     *      defined in <code>EnvironmentMetaDataConstants</code> class can
     *      be used to access the data held by this map.
     *
     * @param monitorMetaData
     *      meta data to associate with the monitor. The meta data passed
     *      to this method can be altered, and the changes will be
     *      reflected in subsequent exit routine invocations. This map
     *      also contains keys with IBM reserved names. These entries are
     *      defined in the <code>MonitorMetaDataConstants</code> class and
     *      have special semantics. The the values of the IBM reserved names
     *      cannot be modified by the exit
     *
     * @param taskDetails
     *      An XML String representing the task to be executed as a result of
     *      the monitor triggering. This XML string may be modified by the
     *      exit
     *
     * @return
     *      a monitor exit result object which is used to determine if the
     *      task should proceed, or be cancelled.
     */
    MonitorExitResult onMonitor(Map<String, String> environmentMetaData,
                               Map<String, String> monitorMetaData,
                               Reference<String> taskDetails);
}

```

### İlgili görevler

MFT kaynaklarının izlenmesi

MFT ürününü kullanıcı çıkışlarıyla özelleştirme

### İlgili başvurular

[“SourceTransferStartExit.java arabirimi” sayfa 2131](#)

[“SourceTransferEndExit.java arabirimi” sayfa 2129](#)

[“DestinationTransferStartExit.java arabirimi” sayfa 2106](#)

[“DestinationTransferEndExit.java arabirimi” sayfa 2105](#)

[“ProtocolBridgeCredentialExit.java arabirimi” sayfa 2125](#)

### ***ProtocolBridgeCredentialExit.java arabirimi***

#### **ProtocolBridgeCredentialExit.java**

```

/**
 * Licensed Materials - Property of IBM
 *
 * "Restricted Materials of IBM"
 *
 * 5724-H72
 *
 * © Copyright IBM Corp. 2008, 2024. All Rights Reserved.
 *
 * US Government Users Restricted Rights - Use, duplication or
 * disclosure restricted by GSA ADP Schedule Contract with
 * IBM Corp.
 */
package com.ibm.wmqfte.exitroutine.api;

import java.util.Map;

/**
 * An interface that is implemented by classes that are to be invoked as part of
 * user exit routine processing. This interface defines methods that will
 * be invoked by a protocol bridge agent to map the MQ user ID of the transfer to credentials
 * that are to be used to access the protocol server.

```

```

* There will be one instance of each implementation class per protocol bridge agent. The methods
* can be called from different threads so the methods must be synchronized.
*/
public interface ProtocolBridgeCredentialExit {

    /**
     * Invoked once when a protocol bridge agent is started. It is intended to initialize
     * any resources that are required by the exit
     *
     * @param bridgeProperties
     *       The values of properties defined for the protocol bridge.
     *       These values can only be read, they cannot be updated by
     *       the implementation.
     *
     * @return true if the initialization is successful and false if unsuccessful
     *         If false is returned from an exit the protocol bridge agent will not
     *         start
     */
    public boolean initialize(final Map<String> bridgeProperties);

    /**
     * Invoked once for each transfer to map the MQ user ID in the transfer message to the
     * credentials to be used to access the protocol server
     *
     * @param mqUserId The MQ user ID from which to map to the credentials to be used
     *                access the protocol server
     * @return A credential exit result object that contains the result of the map and
     *         the credentials to use to access the protocol server
     */
    public CredentialExitResult mapMQUserId(final String mqUserId);

    /**
     * Invoked once when a protocol bridge agent is shutdown. It is intended to release
     * any resources that were allocated by the exit
     *
     * @param bridgeProperties
     *       The values of properties defined for the protocol bridge.
     *       These values can only be read, they cannot be updated by
     *       the implementation.
     *
     * @return
     */
    public void shutdown(final Map<String> bridgeProperties);
}

```

## İlgili görevler

[MFT ürününü kullanıcı çıkışlarıyla özelleştirme](#)

[Çıkış sınıflarını kullanarak bir dosya sunucusuna ilişkin kimlik bilgilerini eşleme](#)

## ***ProtocolBridgeCredentialExit2.java arabirimi***

### **ProtocolBridgeCredentialExit2.java**

```

/*
 * Licensed Materials - Property of IBM
 *
 * "Restricted Materials of IBM"
 *
 * 5724-H72
 *
 * © Copyright IBM Corp. 2011, 2024. All Rights Reserved.
 *
 * US Government Users Restricted Rights - Use, duplication or
 * disclosure restricted by GSA ADP Schedule Contract with
 * IBM Corp.
 */
package com.ibm.wmqfte.exitroutine.api;

/**
 * An interface that is implemented by classes that are invoked as part of user
 * exit routine processing. This interface defines methods that are invoked by a

```

```

* protocol bridge agent to map the MQ user ID of the transfer to credentials
* used to access a specified protocol bridge server. There will be one instance
* of each implementation class for each protocol bridge agent. The methods can
* be called from different threads so the methods must be synchronized.
*/
public interface ProtocolBridgeCredentialExit2 extends
    ProtocolBridgeCredentialExit {

    /**
     * Invoked once for each transfer to map the MQ user ID in the transfer
     * message to the credentials used to access a specified protocol server.
     *
     * @param endPoint
     *     Information that describes the protocol server to be accessed.
     * @param mqUserId
     *     The MQ user ID from which to map the credentials used to
     *     access the protocol server.
     * @return A {@link CredentialExitResult} instance that contains the result
     *     of the map and the credentials to use to access the protocol
     *     server.
     */
    public CredentialExitResult mapMQUserId(
        final ProtocolServerEndPoint endPoint, final String mqUserId);
}

```

## İlgili görevler

MFT ürününü kullanıcı çıkışlarıyla özelleştirme

[Çıkış sınıflarını kullanarak bir dosya sunucusuna ilişkin kimlik bilgilerini eşleme](#)

## ***ProtocolBridgePropertiesExit2.java arabirimi***

### **ProtocolBridgePropertiesExit2.java**

```

/*
 * Licensed Materials - Property of IBM
 *
 * "Restricted Materials of IBM"
 *
 * 5724-H72
 *
 * © Copyright IBM Corp. 2011, 2024. All Rights Reserved.
 *
 * US Government Users Restricted Rights - Use, duplication or
 * disclosure restricted by GSA ADP Schedule Contract with
 * IBM Corp.
 */
package com.ibm.wmqfte.exitroutine.api;

import java.util.Map;
import java.util.Properties;

/**
 * An interface that is implemented by classes that are to be invoked as part of
 * user exit routine processing. This interface defines methods that will be
 * invoked by a protocol bridge agent to look up properties for protocol servers
 * that are referenced in transfers.
 * <p>
 * There will be one instance of each implementation class for each protocol
 * bridge agent. The methods can be called from different threads so the methods
 * must be synchronised.
 */
public interface ProtocolBridgePropertiesExit2 {

    /**
     * Invoked once when a protocol bridge agent is started. It is intended to
     * initialize any resources that are required by the exit.
     *
     * @param bridgeProperties
     *     The values of properties defined for the protocol bridge.
     *     These values can only be read, they cannot be updated by the
     *     implementation.
     * @return {@code true} if the initialization is successful and {@code
     *     false} if unsuccessful. If {@code false} is returned from an exit
     *     the protocol bridge agent will not start.
     */
    public boolean initialize(final Map<String, String> bridgeProperties);
}

```

```

    /**
     * Invoked when the Protocol Bridge needs to access the protocol bridge credentials XML file.
     *
     * @return a {@link String} object giving the location of the ProtocolBridgeCredentials.xml
     */
    public String getCredentialLocation ();

    /**
     * Obtains a set of properties for the specified protocol server name.
     * <p>
     * The returned {@link Properties} must contain entries with key names
     * corresponding to the constants defined in
     * {@link ProtocolServerPropertyConstants} and in particular must include an
     * entry for all appropriate constants described as required.
     *
     * @param protocolServerName
     *       The name of the protocol server whose properties are to be
     *       returned. If a null or a blank value is specified, properties
     *       for the default protocol server are to be returned.
     * @return The {@link Properties} for the specified protocol server, or null
     *         if the server cannot be found.
     */
    public Properties getProtocolServerProperties(
        final String protocolServerName);

    /**
     * Invoked once when a protocol bridge agent is shut down. It is intended to
     * release any resources that were allocated by the exit.
     *
     * @param bridgeProperties
     *       The values of properties defined for the protocol bridge.
     *       These values can only be read, they cannot be updated by the
     *       implementation.
     */
    public void shutdown(final Map<String, String> bridgeProperties);
}

```

## İlgili görevler

[ProtocolBridgePropertiesExit: Protokol dosya sunucusu özellikleri arıyor](#)

[MFT ürününü kullanıcı çıkışlarıyla özelleştirme](#)

[Çıkış sınıflarını kullanarak bir dosya sunucusuna ilişkin kimlik bilgilerini eşleme](#)

## ***SourceFileExitFileSpecification.java sınıfı***

### **SourceFileExitFileSpecification.java**

```

/*
 * Licensed Materials - Property of IBM
 *
 * "Restricted Materials of IBM"
 *
 * 5724-H72
 *
 * © Copyright IBM Corp. 2012, 2024. All Rights Reserved.
 *
 * US Government Users Restricted Rights - Use, duplication or
 * disclosure restricted by GSA ADP Schedule Contract with
 * IBM Corp.
 */
package com.ibm.wmqfte.exitroutine.api;

import java.util.Map;

/**
 * A specification of the file names to use for a file transfer, as evaluated by the
 * agent acting as the source of the transfer.
 */
public final class SourceFileExitFileSpecification {

    private final String sourceFileSpecification;
    private final String destinationFileSpecification;
    private final Map<String, String> sourceFileMetaData;
    private final Map<String, String> destinationFileMetaData;
}

```

```

/**
 * Constructor. Creates a source file exit file specification.
 *
 * @param sourceFileSpecification
 *         the source file specification to associate with the source file
 *         exit file specification.
 *
 * @param destinationFileSpecification
 *         the destination file specification to associate with the
 *         source file exit file specification.
 *
 * @param sourceFileMetaData
 *         the source file meta data.
 *
 * @param destinationFileMetaData
 *         the destination file meta data .
 */
public SourceFileExitFileSpecification(final String sourceFileSpecification,
                                       final String destinationFileSpecification,
                                       final Map<String, String> sourceFileMetaData,
                                       final Map<String, String> destinationFileMetaData) {
    this.sourceFileSpecification = sourceFileSpecification;
    this.destinationFileSpecification = destinationFileSpecification;
    this.sourceFileMetaData = sourceFileMetaData;
    this.destinationFileMetaData = destinationFileMetaData;
}

/**
 * Returns the destination file specification.
 *
 * @return the destination file specification. This represents the location,
 *         on the agent acting as the destination for the transfer, where the
 *         file should be written. Exit routines installed into the agent
 *         acting as the destination for the transfer may override this value.
 */
public String getDestination() {
    return destinationFileSpecification;
}

/**
 * Returns the source file specification.
 *
 * @return the source file specification. This represents the location where
 *         the file data will be read from.
 */
public String getSource() {
    return sourceFileSpecification;
}

/**
 * Returns the file meta data that relates to the source file specification.
 *
 * @return the file meta data that relates to the source file specification.
 */
public Map<String, String> getSourceFileMetaData() {
    return sourceFileMetaData;
}

/**
 * Returns the file meta data that relates to the destination file specification.
 *
 * @return the file meta data that relates to the destination file specification.
 */
public Map<String, String> getDestinationFileMetaData() {
    return destinationFileMetaData;
}
}

```

## İlgili kavramlar

[“MFT kullanıcı çıkışları için meta veri” sayfa 2093](#)

There are three different types of metadata that can be supplied to user exit routines for Managed File Transfer: environment, transfer, and file metadata. Bu meta veri, Java anahtar-değer çiftleri eşlemleri olarak sunulur.

## ***SourceTransferEndExit.java arabirimi***

## SourceTransferEndExit.java

```
/*
 * Licensed Materials - Property of IBM
 *
 * "Restricted Materials of IBM"
 *
 * 5724-H72
 *
 * © Copyright IBM Corp. 2008, 2024. All Rights Reserved.
 *
 * US Government Users Restricted Rights - Use, duplication or
 * disclosure restricted by GSA ADP Schedule Contract with
 * IBM Corp.
 */
package com.ibm.wmqfte.exitpoint.api;

/**
 * An interface that is implemented by classes that want to be invoked as part of
 * user exit routine processing. This interface defines a method that will be
 * invoked immediately after completing a transfer on the agent acting as the
 * source of the transfer.
 */
public interface SourceTransferEndExit {

    /**
     * Invoked immediately after the completion of a transfer on the agent acting as
     * the source of the transfer.
     *
     * @param transferExitResult
     *        a result object reflecting whether or not the transfer completed
     *        successfully.
     *
     * @param sourceAgentName
     *        the name of the agent acting as the source of the transfer.
     *        This is the name of the agent that the implementation of this
     *        method will be invoked from.
     *
     * @param destinationAgentName
     *        the name of the agent acting as the destination of the
     *        transfer.
     *
     * @param environmentMetaData
     *        meta data about the environment in which the implementation
     *        of this method is running. This information can only be read,
     *        it cannot be updated by the implementation. The constants
     *        defined in EnvironmentMetaDataConstants class can
     *        be used to access the data held by this map.
     *
     * @param transferMetaData
     *        meta data to associate with the transfer. The information can
     *        only be read, it cannot be updated by the implementation. This
     *        map may also contain keys with IBM reserved names. These
     *        entries are defined in the TransferMetaDataConstants
     *        class and have special semantics.
     *
     * @param fileResults
     *        a list of file transfer result objects that describe the source
     *        file name, destination file name and result of each file transfer
     *        operation attempted.
     *
     * @return
     *        an optional description to enter into the log message describing
     *        transfer completion. A value of null can be used
     *        when no description is required.
     */
    String onSourceTransferEnd(TransferExitResult transferExitResult,
                              String sourceAgentName,
                              String destinationAgentName,
                              Map<String, String>environmentMetaData,
                              Map<String, String>transferMetaData,
                              List<FileTransferResult>fileResults);

}

```

### İlgili görevler

[MFT ürününü kullanıcı çıkışlarıyla özelleştirme](#)

## İlgili başvurular

[“SourceTransferStartExit.java arabirimi” sayfa 2131](#)

[“DestinationTransferStartExit.java arabirimi” sayfa 2106](#)

[“DestinationTransferEndExit.java arabirimi” sayfa 2105](#)

[“MonitorExit.java arabirimi” sayfa 2124](#)

[“ProtocolBridgeCredentialExit.java arabirimi” sayfa 2125](#)

## **SourceTransferStartExit.java arabirimi**

### **SourceTransferStartExit.java**

```
/*
 * Licensed Materials - Property of IBM
 *
 * "Restricted Materials of IBM"
 *
 * 5724-H72
 *
 * © Copyright IBM Corp. 2008, 2024. All Rights Reserved.
 *
 * US Government Users Restricted Rights - Use, duplication or
 * disclosure restricted by GSA ADP Schedule Contract with
 * IBM Corp.
 */
package com.ibm.wmqfte.exitpoint.api;

import java.util.List;
import java.util.Map;

/**
 * An interface that is implemented by classes that want to be invoked as part of
 * user exit routine processing. This interface defines a method that will be
 * invoked immediately prior to starting a transfer on the agent acting as the
 * source of the transfer.
 */
public interface SourceTransferStartExit {

    /**
     * Invoked immediately prior to starting a transfer on the agent acting as
     * the source of the transfer.
     *
     * @param sourceAgentName
     *     the name of the agent acting as the source of the transfer.
     *     This is the name of the agent that the implementation of this
     *     method will be invoked from.
     *
     * @param destinationAgentName
     *     the name of the agent acting as the destination of the
     *     transfer.
     *
     * @param environmentMetaData
     *     meta data about the environment in which the implementation
     *     of this method is running. This information can only be read,
     *     it cannot be updated by the implementation. The constants
     *     defined in EnvironmentMetaDataConstants class can
     *     be used to access the data held by this map.
     *
     * @param transferMetaData
     *     meta data to associate with the transfer. The meta data passed
     *     to this method can be altered, and the changes to will be
     *     reflected in subsequent exit routine invocations. This map may
     *     also contain keys with IBM reserved names. These entries are
     *     defined in the TransferMetaDataConstants class and
     *     have special semantics.
     *
     * @param fileSpecs
     *     a list of file specifications that govern the file data to
     *     transfer. The implementation of this method can add entries,
     *     remove entries, or modify entries in this list and the changes
     *     will be reflected in the files transferred.
     *
     * @return
     *     a transfer exit result object which is used to determine if the
     *     transfer should proceed, or be cancelled.
     */
}
```

```

*/
TransferExitResult onSourceTransferStart(String sourceAgentName,
String destinationAgentName,
Map<String, String> environmentMetaData,
Map<String, String>transferMetaData,
List<SourceFileExitFileSpecification>fileSpecs);
}

```

## İlgili görevler

MFT ürününü kullanıcı çıkışlarıyla özelleştirme

## İlgili başvurular

[“SourceFileExitFileSpecification.java sınıfı” sayfa 2128](#)

[“SourceTransferEndExit.java arabirimi” sayfa 2129](#)

[“DestinationTransferStartExit.java arabirimi” sayfa 2106](#)

[“DestinationTransferEndExit.java arabirimi” sayfa 2105](#)

[“MonitorExit.java arabirimi” sayfa 2124](#)

[“ProtocolBridgeCredentialExit.java arabirimi” sayfa 2125](#)

## ***TransferExitResult.java arabirimi***

### **TransferExitResult.java**

```

/*
 * Licensed Materials - Property of IBM
 *
 * "Restricted Materials of IBM"
 *
 * 5724-H72
 *
 * © Copyright IBM Corp. 2008, 2024. All Rights Reserved.
 *
 * US Government Users Restricted Rights - Use, duplication or
 * disclosure restricted by GSA ADP Schedule Contract with
 * IBM Corp.
 */

package com.ibm.wmqfte.exitroutine.api;

/**
 * The result of invoking a transfer exit routine. It is composed of a result
 * code, which determines if the transfer should proceed, and an optional explanatory
 * message. The explanation, if present, is entered into the log message.
 */
public class TransferExitResult {

    private final TransferExitResultCode resultCode;
    private final String explanation;

    /**
     * For convenience, a static "proceed" result with no associated explanation
     * message.
     */
    public static final TransferExitResult PROCEED_RESULT =
        new TransferExitResult(TransferExitResultCode.PROCEED, null);

    /**
     * Constructor. Creates a transfer exit result object with a specified result
     * code and explanation.
     *
     * @param resultCode
     *     The result code to associate with the exit result being created.
     *
     * @param explanation
     *     The explanation to associate with the exit result being created.
     *     A value of <code>null</code> can be specified to indicate no
     *     explanation.
     */
    public TransferExitResult(TransferExitResultCode resultCode, String explanation) {
        this.resultCode = resultCode;
        this.explanation = explanation;
    }
}

```



```

/**
 * Returns the explanation associated with this transfer exit result.
 *
 * @return the explanation associated with this exit result.
 */
public String getExplanation() {
    return explanation;
}

/**
 * Returns the result code associated with this transfer exit result.
 *
 * @return the result code associated with this exit result.
 */
public TransferExitResultCode getResultCode() {
    return resultCode;
}
}

```

### İlgili görevler

[MFT ürününü kullanıcı çıkışlarıyla özelleştirme](#)

### İlgili başvurular

[“SourceTransferStartExit.java arabirimi” sayfa 2131](#)

[“DestinationTransferStartExit.java arabirimi” sayfa 2106](#)

[“DestinationTransferEndExit.java arabirimi” sayfa 2105](#)

[“MonitorExit.java arabirimi” sayfa 2124](#)

[“ProtocolBridgeCredentialExit.java arabirimi” sayfa 2125](#)

## MFT Agent komut kuyruğuna yerleştirebileceğiniz iletiler için ileti biçimleri

Bu XML şemaları, aracının bir işlem gerçekleştirmesini istemek için aracı komut kuyruğuna yerleştirilebilir iletiler için biçimleri tanımlar. XML iletisi, komut satırı komutları ya da bir uygulama kullanılarak aracı komut kuyruğuna yerleştirilebilir.

- [Dosya aktarma isteği ileti biçimi](#)
- [MFT izleme isteği ileti biçimleri](#)
- [Ping MFT aracı istek iletisi biçimi](#)
- [MFT aracı yanıt iletisi biçimi](#)

## Messaging REST API başvurusu

messaging REST API ile ilgili başvuru bilgileri.

messaging REST API kullanımıyla ilgili daha fazla bilgi için bkz. [REST API ile ileti alışverişi](#).

### REST API Kaynak

Bu konu derlemi, messaging REST API kaynaklarının her biri için başvuru bilgileri sağlar.

messaging REST API kullanımıyla ilgili daha fazla bilgi için bkz. [REST API ile ileti alışverişi](#).

### /message/qmgr/{qmgrName}/queue/{queueName}/message

The messaging REST API allows messages to be put to a queue, **V9.2.0** ya da göz atılacak iletiler or destructively got from a queue, using the /messaging/qmgr/{qmgrName}/queue/{queueName}/message resource.

## POST

İletileri belirtilen kuyruk yöneticisine belirtilen kuyruğa koymak için, /messaging/qmgr/{qmgrName}/queue/{queueName}/message kaynağıyla HTTP POST yöntemini kullanabilirsiniz.

HTTP isteği gövde içeren bir IBM MQ iletisini belirtilen kuyruk yöneticisine ve kuyruğuna koyar. Kuyruk yöneticisi, mqweb sunucusu ile aynı makinede olmalıdır. Yöntem yalnızca metin tabanlı HTTP istek gövdeleri destekler. İletiler, MQSTR ya da JMS TextMessage tarafından biçimlendirilmiş ileteler olarak gönderilir ve yürürlükteki kullanıcı bağlamı kullanılarak yazılır.

**V 9.2.5** REST API V3 , kullanıcı tanımlı ileti özelliklerini belirleme ve ileti önceliğini içerme yeteneğini ekler. Ibm-mq-md-priority ve ibm-mq-usr istek üstbilgileri yalnızca REST API V3 ile kullanılabilir. Ibm-mq-md-correlationId istek üstbilgisi, REST API V3' te farklı bir biçime sahiptir. Üstbilgi, uygulamaya özgü bir tanıttıcı olabilir ya da kodlanmış bir dize, ID: önekini gerektiriyorsa. POST isteğiniz kullanıcı tanımlı ileteler ya da uygulamaya özgü ilinti tanıttıcısı içeriyorsa, ileti bir JMS TextMessage olarak biçimlendirilir.

- “Kaynak URL Adresi” sayfa 2134
- “İstek üstbilgileri” sayfa 2135
- “İstek gövde biçimi” sayfa 2137
- “Güvenlik Gereksinimleri” sayfa 2137
- “Yanıt durumu kodları” sayfa 2137
- “Yanıt üstbilgileri” sayfa 2138
- “Yanıt gövdesi biçimi” sayfa 2139
- “Örnekler” sayfa 2139

## Kaynak URL Adresi

`https://host:port/ibmmq/rest/v1/messaging/qmgr/{qmgrName}/queue/{queueName}/message`

`https://host:port/ibmmq/rest/v2/messaging/qmgr/{qmgrName}/queue/{queueName}/message`

**V 9.2.5** `https://host:port/ibmmq/rest/v3/messaging/qmgr/{qmgrName}/queue/{queueName}/message`

### qmgrName

İleti alışverişi için bağlantı kurmak üzere kuyruk yöneticisinin adını belirtir. Kuyruk yöneticisi, mqweb sunucusu ile aynı makinede olmalıdır.

Kuyruk yöneticisi adı büyük ve küçük harfe duyarlıdır.

Kuyruk yöneticisi adı eğik çizgi, nokta ya da yüzde işareti içeriyorsa, bu karakterlerin kodlanmış URL adresi olması gerekir:

- Eğik çizgi %2F olarak kodlanmalıdır.
- Nokta, %2E şeklinde kodlanmalıdır.
- Yüzde imi, %25 olarak kodlanmalıdır.

### queueName

İletinin gönderileceği kuyruğun adını belirtir.

Kuyruk, belirtilen kuyruk yöneticisine yerel, uzak ya da bir diğer ad olarak tanımlanmalıdır; kümelenmiş bir kuyruğa da gönderme gösterebilir.

Kuyruk adı büyük ve küçük harfe duyarlıdır.

Kuyruk adı bir eğik çizgi ya da yüzde işareti içeriyorsa, bu karakterlerin kodlanmış URL adresi olması gerekir:

- Sağa eğik çizgi, /, %2F olarak kodlanmalıdır.
- Yüzde işareti, %, %25 olarak kodlanmalıdır.

HTTP bağlantılarını etkinleştirdiğinizde HTTPS yerine HTTP ' yi kullanabilirsiniz. HTTP ' nin etkinleştirilmesiyle ilgili daha fazla bilgi için [HTTP ve HTTPS kapılarının yapılandırılması](#) başlıklı konuya bakın.

## İstek üstbilgileri

İstekle birlikte aşağıdaki üstbilgiler gönderilmelidir:

### Yetkilendirme

Temel kimlik doğrulaması kullanıyorsanız bu üstbilgi gönderilmelidir. Daha fazla bilgi için bkz. [REST API ile HTTP temel kimlik doğrulamasını kullanma](#).

### İçerik Tipi

Bu üstbilgi, aşağıdaki değerlerden biriyle gönderilmelidir:

- text/plain; charset=utf-8
- text/html; charset=utf-8
- text/xml; charset=utf-8
- application/json; charset=utf-8
- application/xml; charset=utf-8

**Not:** If *karakter kümesi* is omitted from the Context-Type header, UTF-8 is assumed.

### ibm-mq-rest-csrf-token

Bu üstbilgi ayarlanmalıdır, ancak değer boş olmak da dahil olmak üzere herhangi bir şey olabilir.

İsteğe bağlı olarak aşağıdaki üstbilgiler istekle birlikte gönderilebilir:

### Kabul Et-Dil

Bu üstbilgi, yanıt iletisi gövdesinde döndürülen kural dışı durumlar ya da hata iletileri için gerekli dili belirtir.

### REST API V1 REST API V2 ibm-mq-md-correlationId

Bu üstbilgi, yaratılan iletinin ilinti tanıtıcısını ayarlar. Üstbilgi, 24 baytı gösteren, 48 karakterden oluşan onaltılı kodlanmış bir dizgi olarak belirtilmelidir. Değeri "ID: " ile önlemeyin; REST API 'si bu dizgiyi otomatik olarak ekler.

Örneğin:

```
ibm-mq-md-correlationId: 414d5120514d4144455620202020202067d8bf5923582e02
```

### V 9.2.5 REST API V3 ibm-mq-md-correlationId

Bu üstbilgi, yaratılan iletinin ilinti tanıtıcısını ayarlar. İlinti tanıtıcısı aşağıdaki biçimlerden birini alabilir:

- A 48 character hexadecimal encoded string, representing 24 bytes, prefixed with the string "ID: ".  
Örneğin:

```
ibm-mq-md-correlationId: ID:414d5120514d4144455620202020202067d8bf5923582e02
```

- Uygulamaya özgü bir değer. Değer, uygulamaya özgü bir dizgidir:

```
ibm-mq-md-correlationId: My-Custom-CorrelId
```

Bu ilinti tanıtıcısı biçimini belirtirseniz, ileti hedefi WMQ\_CLIENT\_JMS\_COMPLIANT olarak hedeflenmektedir ve bu nedenle bir MQRFH2 üstbilgisi içerir.

### ibm-mq-md-süre bitimi

Bu üstbilgi, oluşturulan ileti için süre bitimi süresini ayarlar. Bir iletinin süre bitimi, ileti kuyruktan geldiği zamandan başlar. Sonuç ağı gecikme süresi yok sayılır. Üstbilgi, aşağıdaki değerlerden biri olarak belirtilmelidir:

**SINIRSIZ**

İletinin süresi dolmaz.

Bu değer, varsayılan değerdir.

**Tamsayı değeri**

İleti süre bitiminden önce milisaniye.

0-99999999900 aralığıyla sınırlanmıştır.

**ibm-mq-md-persistence**

Bu üstbilgi, oluşturulan iletiye ilişkin kalıcılığı ayarlar. Üstbilgi, aşağıdaki değerlerden biri olarak belirtilmelidir:

**nonPersistent**

İleti, sistem arızalarını ya da kuyruk yöneticisini yeniden başlatma işlemini sürdürmez.

Bu değer, varsayılan değerdir.

**Kalıcı**

İleti, sistem arızalarını ya da kuyruk yöneticisini yeniden başlatma işlemini sürdürür.

**V 9.2.5 REST API V3 ibm-mq-md-priority**

Bu üstbilgi, oluşturulan iletinin önceliğini ayarlar. Üstbilgi, aşağıdaki değerlerden biri olarak belirtilmelidir:

**asDestination**

İleti, temeldeki IBM MQ kuyruk nesnesinin DEFPRTY özniteliğinde belirtilen önceliği kullanır.

**Tamsayı değeri**

Gerçek önceliği 0-9 aralığında bir tamsayı olarak belirtin.

Örneğin:

```
ibm-mq-md-priority: asDestination
```

**ibm-mq-md-replyTo**

Bu üstbilgi, oluşturulan iletiye ilişkin yanıtlama hedefini ayarlar. Üstbilginin biçimi, yanıt kuyruğu ve isteğe bağlı bir kuyruk yöneticisi sağlamak için standart gösterimini kullanır: `replyQueue[@replyQmgr]`

Örneğin:

```
ibm-mq-md-replyTo: myReplyQueue@myReplyQMGR
```

**V 9.2.5 REST API V3 ibm-mq-usr**

İstek ileti kullanıcı tanımlı özellikleri ayarlayın. İletide birden çok özellik ayarlanabilir. Tek bir `ibm-mq-usr` istek üstbilgisinde birden çok virgülle ayrılmış özellik belirleyebilir ya da `ibm-mq-usr` istek üstbilgisinin iki ya da daha fazla örneğini kullanabilirsiniz.

Örneğin:

```
ibm-mq-usr: myIPprop;5;short
ibm-mq-usr: mySProp;"hi";string
ibm-mq-usr: myBProp;true;boolean
ibm-mq-usr: myA;5;byte,myB;-10;integer
```

Özelliklerde aşağıdaki sözdizimi bulunur:

```
ibm-mq-usr: property_name; user_value; user_type
```

**özellik\_adi**

Belirlenmekte olan kullanıcı özelliğinin adı. Bu geçerli bir JMS özelliği adı olmalıdır.

**kullanıcı\_değeri**

Özelliğin değeri.

## kullanıcı\_türü

Özelliğın tipi:

- boolean (true/false, MQBOOL)
- byte (8 bit tamsayı, MQINT8)
- short (16 bitlik tamsayı, MQINT16)
- integer (32 bitlik tamsayı, MQINT32)
- long (64 bitlik tamsayı, MQINT64)
- float (32 bit gerçek, MQFLOAT32)
- double (64 bit gerçek, MQFLOAT64)
- string (tırnak içine alınmış dizgi)

**Not:** POST için ileti önceliğı, REST API V1 ve REST API 'si V2 , varsayılan olarak REST API 'si V3 için asDestination olarak ayarlanır.için her zaman 4 olur.

## İstek gövde biçimi

İstek gövdesi metin olmalı ve UTF-8 kodlaması kullanılmalıdır. Belirli bir metin yapısı gerekli değildir. İstek gövdesi metnini içeren MQSTR biçimlendirilmiş bir ileti oluşturulur ve belirtilen kuyruğa konmasını sağlar.

**V 9.2.5** **REST API V3** REST API 'si V3 kullanıcı tanımlı özellikleri ya da uygulamaya özgü ilinti tanıtıcısı özellikleri kullanılırsa, istek gövdesi metnini içeren bir JMS TextMessage biçimli ileti yaratılır ve belirtilen kuyruğa konursa.

Ek bilgi için bkz. [örnekler](#).

## Güvenlik Gereksinimleri

Çağırın, mqweb sunucusunda kimliğı doğrulanmalıdır. MQWebAdmin ve MQWebAdminRO rolleri, messaging REST API için geçerli değildir. REST API güvenlik güvenliğıyle ilgili ek bilgi için [IBM MQ Console and REST API security](#) başlıklı konuya bakın.

Kimliğı doğrulandıktan sonra, kullanıcı hem messaging REST API hem de administrative REST API' ı kullanabilen bir mqweb sunucusuna doğrulanır.

Çağırana ilişkin güvenlik birincil kullanıcısının, iletileri belirtilen kuyruğa koyma yeteneğı verilmelidir:

- Kaynak URL 'sinin *{queueName}* kısmı tarafından belirtilen kuyruk, PUT etkinleştirilmiş olmalıdır.

**MQ Appliance** **ALW** Kaynak URL adresinin *{queueName}* kısmı tarafından belirtilen kuyruk için, çağırın kullanıcının güvenlik birincil kullanıcısı için +PUT yetkisi verilmelidir.

- **z/OS** Kaynak URL adresinin *{queueName}* kısmı tarafından belirtilen kuyruk için, çağırın kullanıcının güvenlik birincil kullanıcısı için UPDATE erişimi verilmelidir.

**ALW** AIX, Linux, and Windows' da, **setmqaut** komutunu kullanarak IBM MQ kaynaklarını kullanmak için güvenlik birincil kullanıcıları için yetki verebilirsiniz. Daha fazla bilgi için bkz. [setmqaut](#) (yetki verilmesi ya da iptal etmek).

**z/OS** z/OS işletim üzerinde bkz. [z/OS üzerinde güvenliğın ayarlanması](#).

If you use Advanced Message Security (AMS) with the messaging REST API, note that all messages are encrypted by using the context of the mqweb server, not the context of the user that posts the message.

## Yanıt durumu kodları

### 201

İleti yaratıldı ve başarıyla gönderildi.

#### 400

Geçersiz veri sağlandı.

Örneğin, geçersiz bir istek üstbilgisi değeri belirtildi.

#### 401

Kimlik doğrulanmadı.

Çağırın, mqweb sunucusunda kimlik doğrulanmalıdır ve MQWebAdmin, MQWebAdminROya da MQWebUser rollerinden birinin ya da daha fazlasının üyesi olmalıdır. `ibm-mq-rest-csrf-token` üstbilgisi de belirtilmelidir. Daha fazla bilgi için bkz. "[Güvenlik Gereksinimleri](#)" sayfa 2137.

#### 403

Yetkili değil.

Çağırın, mqweb sunucusuna doğrulanır ve geçerli bir birincil kullanıcıyla ilişkilendirilir. Ancak, birincil kullanıcının tümüne erişim ya da gerekli IBM MQ kaynaklarının bir alt kümesine erişimi yok ya da MQWebUser rolünde yok. Gerekli erişim hakkında daha fazla bilgi için bkz. "[Güvenlik Gereksinimleri](#)" sayfa 2137.

#### 404

Kuyruk yok.

#### 405

Kuyruk, PUT engelleyicidir.

#### 415

İleti üstbilgisi ya da gövdesi desteklenmeyen bir ortam tipidir.

Örneğin, Content-Type üstbilgisi desteklenmeyen bir ortam tipine ayarlıdır.

#### 500

IBM MQ' dan sunucu sorunu ya da hata kodu.

#### 502

İleti alışverişi sağlayıcısı gerekli işlevi desteklemediğinden, yürürlükteki güvenlik birincil kullanıcısı iletiyi gönderemiyor. Örneğin, mqweb sunucusu sınıf yolu geçersizse.

#### 503

Kuyruk yöneticisi çalışmıyor.

## Yanıt üstbilgileri

Yanıtın aşağıdaki üstbilgileri döndürülür:

### İçerik-Dil

Herhangi bir hata ya da kural dışı durum olayında yanıt iletilisinin dil tanıtıcısını belirtir. Herhangi bir hata ya da kural dışı durum koşuluna ilişkin gerekli dili belirtmek üzere Accept-Language istek üstbilgisiyle birlikte kullanılır. İstenen dil desteklenmiyorsa, mqweb sunucusu varsayılan değeri kullanılır.

### İçerik Uzunluğu

HTTP yanıt gövdesinin uzunluğunu, içerik olmadığı da belirtir. Başarı üzerine, değer sıfır olur.

### İçerik Tipi

Yanıt gövdesinin tipini belirtir. Başarı üzerine, değer `text/plain;charset=utf-8` olur. Herhangi bir hata ya da kural dışı durum durumunda, değer `application/json;charset=utf-8` olur.

### REST API V1 > REST API V2 **ibm-mq-md-messageId**

IBM MQ tarafından bu iletiye ayrılmış olan ileti tanıtıcısını belirtir. `ibm-mq-md-correlationId` istek üstbilgisinde olduğu gibi, 24 baytı gösteren, 48 karakterden oluşan onaltılı kodlanmış bir dizgi olarak gösterilir.

Örneğin:

```
ibm-mq-md-messageId: 414d5120514d4144455620202020202067d8ce5923582f07
```

## V 9.2.5 REST API V3 **ibm-mq-md-messageId**

IBM MQ tarafından bu iletiye ayrılmış olan ileti tanıtıcısını belirtir. Like the `ibm-mq-md-correlationId` request header, it is represented as a 48 character hexadecimal encoded string, representing 24 bytes, prefixed by the string ID:.

Örneğin:

```
ibm-mq-md-messageId: ID:414d5120514d4144455620202020202067d8ce5923582f07
```

## Yanıt gövdesi biçimi

İleti başarıyla gönderilirse, yanıt gövdesi boş olur. Bir hata ortaya çıkarsa, yanıt gövdesi bir hata iletisi içerir. Daha fazla bilgi için bkz. [REST API hata işleme](#).

## Örnekler

Aşağıdaki örneklerde v2 kaynak URL 'si kullanılır. IBM MQ Daha önceki bir IBM MQ 9.1.5 sürümünü kullanıyorsanız, bunun yerine v1 kaynak URL adresini kullanmanız gerekir. Yani, kaynak URL 'sinde, örnek URL 'nin v2adresini kullandığı v1 yerine kullanın.

The following example logs in a user called `mquser` with the password `mquser`. cURL' de, oturum açma isteği aşağıdaki Windows örneğine benzebilir. LTPA simgesi, -c işareti kullanılarak `cookiejar.txt` dosyasında depolanır:

```
curl -k "https://localhost:9443/ibmmq/rest/v2/login" -X POST
-H "Content-Type: application/json" --data "{\"username\":\"mquser\", \"password\":\"mquser\"}"
-c c:\cookiejar.txt
```

Kullanıcı oturum açtıktan sonra, başka istekleri doğrulamak için LTPA simgesi ve `ibm-mq-rest-csrf-token` HTTP üstbilgisi kullanılır. The `ibm-mq-rest-csrf-token` `token_value` can be any value, including blank.

- The following Windows cURL example sends a message to queue Q1 on queue manager QM1, using default options. İleti, "*Merhaba Dünya!*"metnini içeriyor:

```
curl -k "https://localhost:9443/ibmmq/rest/v2/messaging/qmgr/QM1/queue/Q1/message"
-X POST -b c:\cookiejar.txt -H "ibm-mq-rest-csrf-token: token_value"
-H "Content-Type: text/plain;charset=utf-8" --data "Hello World!"
```

- The following Windows cURL example sends a persistent message to queue Q1 on queue manager QM1, with an expiry of 2 minutes. İleti, "*Merhaba Dünya!*"metnini içeriyor:

```
curl -k "https://localhost:9443/ibmmq/rest/v2/messaging/qmgr/QM1/queue/Q1/message"
-X POST -b c:\cookiejar.txt -H "ibm-mq-rest-csrf-token: token_value"
-H "Content-Type: text/plain;charset=utf-8" -H "ibm-mq-md-persistence: persistent"
-H "ibm-mq-md-expiry: 120000" --data "Hello World!"
```

- Aşağıdaki Windows cURL örneği, süre bitimi ve tanımlanmış ilinti tanıtıcısı olmadan, QM1kuyruk yöneticisiyle ilgili Q1 kuyruğuna kalıcı olmayan bir ileti gönderir. İleti, "*Merhaba Dünya!*"metnini içeriyor:

```
curl -k "https://localhost:9443/ibmmq/rest/v2/messaging/qmgr/QM1/queue/Q1/message"
-X POST -b c:\cookiejar.txt -H "ibm-mq-rest-csrf-token: token_value"
-H "Content-Type: text/plain;charset=utf-8" -H "ibm-mq-md-persistence: nonPersistent"
-H "ibm-mq-md-expiry: unlimited" -H "ibm-mq-md-correlationId:
414d5120514d41444556202020202067d8b
f5923582e02" --data "Hello World!"
```

## V 9.2.0 **GET**

İlişkili kuyruk yöneticisinden ve kuyruğundan iletilere göz atmak için, `/messaging/qmgr/{qmgrName}/queue/{queueName}/message` kaynağıyla HTTP GET yöntemini kullanabilirsiniz.

Belirlenen kuyruk yöneticisinden ve kuyruktan ilk kullanılabilir iletiyi göz attır. Kuyruk yöneticisi, mqweb sunucusu ile aynı makinede olmalıdır. İleti gövdesi HTTP yanıt gövdesinde döndürülür. İleti, MQSTR ya da JMS TextMessage biçiminde olmalıdır ve yürürlükteki kullanıcı bağlamı kullanılarak alınır.

Kuyruğun üzerinde tüm iletiler bırakılır ve uygun durum kodu, uygunsuz iletiler için çağırıcıya döndürülür. Örneğin, MQSTR ya da JMS TextMessage biçiminde olmayan bir ileti.

**V 9.2.5** REST API V3 , kullanıcı tanımlı ileti özelliklerini belirleme ve ileti önceliğini iletilerle içermeye yeteneğini ekler. Ibm-mq-md-priority ve ibm-mq-usr yanıt üstbilgileri yalnızca REST API 'si V3 ile kullanılabilir. Ibm-mq-md-correlationId istek üstbilgisi, REST API V3' te farklı bir biçime sahiptir. Üstbilgi, uygulamaya özgü bir tanıttıcı olabilir ya da kodlanmış bir dize, ID: önekini koruyabilir. ibm-mq-md-messageId yanıt üstbilgisi ve sorgu parametresi, REST API V3' te farklı bir biçime sahiptir ve ID: önekini korur.

- [“Kaynak URL Adresi” sayfa 2140](#)
- [“İsteğe bağlı sorgu değıştirmeleri” sayfa 2141](#)
- [“İstek üstbilgileri” sayfa 2142](#)
- [“İstek gövde biçimi” sayfa 2142](#)
- [“Güvenlik Gereksinimleri” sayfa 2142](#)
- [“Yanıt durumu kodları” sayfa 2142](#)
- [“Yanıt üstbilgileri” sayfa 2143](#)
- [“Yanıt gövdesi biçimi” sayfa 2145](#)
- [“Örnekler” sayfa 2145](#)

## Kaynak URL Adresi

`https://host:port/ibmmq/rest/v1/messaging/qmgr/{qmgrName}/queue/{queueName}/message`

`https://host:port/ibmmq/rest/v2/messaging/qmgr/{qmgrName}/queue/{queueName}/message`

**V 9.2.5** `https://host:port/ibmmq/rest/v3/messaging/qmgr/{qmgrName}/queue/{queueName}/message`

### qmgrName

İleti alışverişi için bağlantı kurmak üzere kuyruk yöneticisinin adını belirtir. Kuyruk yöneticisi, mqweb sunucusu ile aynı makinede olmalıdır.

Kuyruk yöneticisi adı büyük ve küçük harfe duyarlıdır.

Kuyruk yöneticisi adı eğik çizgi, nokta ya da yüzde işareti içeriyorsa, bu karakterlerin kodlanmış URL adresi olması gerekir:

- Eğik çizgi (/), %2F biçiminde kodlanmalıdır.
- Yüzde işareti (%), %25 olarak kodlanmalıdır.

### queueName

İletin gönderileceği kuyruğun adını belirtir.

Kuyruk, yerel bir kuyruğa işaret eden yerel ya da diğer ad olarak tanımlanmalıdır.

Kuyruk adı büyük ve küçük harfe duyarlıdır.

Kuyruk adı bir eğik çizgi ya da yüzde işareti içeriyorsa, bu karakterlerin kodlanmış URL adresi olması gerekir:

- Sağa eğik çizgi, /, %2F olarak kodlanmalıdır.
- Yüzde işareti, %, %25 olarak kodlanmalıdır.



HTTP bağlantılarını etkinleştirdiğinizde HTTPS yerine HTTP ' yi kullanabilirsiniz. HTTP ' nin etkinleştirilmesiyle ilgili daha fazla bilgi için [HTTP ve HTTPS kapılarının yapılandırılması](#) başlıklı konuya bakın.

## İsteğe bağlı sorgu değiştirgeleri

### REST API V1 > REST API V2 **correlationId=hexValue**

HTTP yönteminin, ilgili ilinti tanıtıcısıyla sonraki iletiyi döndüreceğini belirtir.

#### hexValue

Sorgu parametresi, 24 baytı gösteren, onaltılı kodlanmış 48 karakterden oluşan bir dizilim olarak belirtilmelidir.

Örneğin:

```
../message?correlationId=414d5120514d4144455620202020202067d8bf5923582e02
```

### V 9.2.5 REST API V3 **correlationId= Tanıtıcı:hexValue ya da correlationId=application\_specific\_value**

HTTP yönteminin, ilgili ilinti tanıtıcısıyla sonraki iletiyi döndüreceğini belirtir.

#### hexValue

The query parameter must be specified as a 48 character hexadecimal encoded string, representing 24 bytes, and preceded by the string "ID: ".

Örneğin:

```
../message?correlationId=ID:414d5120514d4144455620202020202067d8bf5923582e02
```

#### uygulama\_belirtim\_değeri

Sorgu parametresi, uygulamaya özgü bir dizgi olarak belirtilebilir.

Örneğin:

```
../message?correlationId=My-Custom-CorrelId
```

### REST API V1 > REST API V2 **messageId=hexValue**

HTTP yönteminin, ilgili ileti tanıtıcısı ile sonraki iletiyi döndüreceğini belirler.

#### hexValue

Sorgu parametresi, 24 baytı gösteren, onaltılı kodlanmış 48 karakterden oluşan bir dizilim olarak belirtilmelidir.

Örneğin:

```
../message?messageId=414d5120514d4144455620202020202067d8ce5923582f07
```

### V 9.2.5 REST API V3 **messageId= Tanıtıcı:hexValue**

HTTP yönteminin, ilgili ileti tanıtıcısı ile sonraki iletiyi döndüreceğini belirler.

#### hexValue

The query parameter must be specified as a 48 character hexadecimal encoded string, representing 24 bytes, and preceded by the string "ID: ".

Örneğin:

```
../message?messageId=ID:414d5120514d4144455620202020202067d8ce5923582f07
```

## İstek üstbilgileri

İstekle birlikte aşağıdaki üstbilgiler gönderilmelidir:

### Yetkilendirme

Temel kimlik doğrulaması kullanıyorsanız bu üstbilgi gönderilmelidir. Daha fazla bilgi için bkz. [REST API ile HTTP temel kimlik doğrulamasını kullanma](#).

### ibm-mq-rest-csrf-token

Bu üstbilgi ayarlanmalıdır, ancak değer boş olmak da dahil olmak üzere herhangi bir şey olabilir.

İsteğe bağlı olarak aşağıdaki üstbilgiler istekle birlikte gönderilebilir:

### Kabul Et-Karakter Kümesi

Bu üstbilgi, yanıt için hangi karakter kümesinin kabul edilebilir olduğunu göstermek için kullanılabilir. Belirtilirse, bu üstbilgi UTF-8 olarak ayarlanmalıdır.

### Kabul Et-Dil

Bu üstbilgi, yanıt iletisi gövdesinde döndürülen kural dışı durumlar ya da hata iletileri için gerekli dili belirtir.

## İstek gövde biçimi

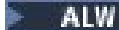


Yok.


## Güvenlik Gereksinimleri


Çağırın, mqweb sunucusunda kimliği doğrulanmalıdır. MQWebAdmin ve MQWebAdminRO rolleri, messaging REST API için geçerli değildir. REST API güvenlik güvenliğiyle ilgili ek bilgi için [IBM MQ Console and REST API security](#) başlıklı konuya bakın.

Kimliği doğrulandıktan sonra, kullanıcı hem messaging REST API hem de administrative REST API' i kullanabilen bir mqweb sunucusuna doğrulanır.

Çağırının güvenlik birincil kullanıcıasına, belirtilen kuyruktaki iletilere göz atma yeteneği verilmelidir:

- Kaynak URL adresinin *{queueName}* kısmı tarafından belirtilen kuyruk BROWSE etkinleştirilmiş olmalıdır.
-   Kaynak URL adresinin *{queueName}* kısmı tarafından belirtilen kuyruk için, çağırının güvenlik birincil kullanıcıasına +GET, +INQ ve +BROWSE yetkisi verilmelidir.
-  Kaynak URL adresinin *{queueName}* kısmı tarafından belirtilen kuyruk için UPDATE, çağırının güvenlik birincil kullanıcıasına erişim verilmelidir.

 AIX, Linux, and Windows' da, **setmqaut** komutunu kullanarak IBM MQ kaynaklarını kullanmak için güvenlik birincil kullanıcıları için yetki verebilirsiniz. Daha fazla bilgi için bkz. [setmqaut](#) (yetki verilmesi ya da iptal etmek).

 z/OS işletim üzerinde bkz. [z/OS üzerinde güvenliğin ayarlanması](#).

## Yanıt durumu kodları

### 200

İleti başarıyla alındı.

### 204

Kullanılabilir bir ileti yok.

### 400

Geçersiz veri sağlandı.

Örneğin, geçersiz bir sorgu değiştirgesi değeri belirtildi.

### 401

Kimlik doğrulanmadı.

Çağırın, mqweb sunucusunda kimlik doğrulanmalıdır ve MQWebAdmin, MQWebAdminROya da MQWebUser rollerinden birinin ya da daha fazlasının üyesi olmalıdır. `ibm-mq-rest-csrf-token` üstbilgisi de belirtilmelidir. Daha fazla bilgi için bkz. "[Güvenlik Gereksinimleri](#)" sayfa 2142.

#### 403

Yetkili değil.

Çağırın, mqweb sunucusuna doğrulanır ve geçerli bir birincil kullanıcıyla ilişkilendirilir. Ancak, birincil kullanıcının tümüne erişim ya da gerekli IBM MQ kaynaklarının bir alt kümesine erişimi yok ya da MQWebUser rolünde yok. Gerekli erişim hakkında daha fazla bilgi için bkz. "[Güvenlik Gereksinimleri](#)" sayfa 2142.

#### 404

Kuyruk yok.

#### 500

IBM MQ' dan sunucu sorunu ya da hata kodu.

#### 501

HTTP yanıtı oluşturulamadı.

Örneğin, alınan iletinin tipi yanlış ya da doğru tip var, ancak gövde işlenemedi.

#### 502

İleti alışverişi sağlayıcısı gereken işlevi desteklemediği için, yürürlükteki güvenlik birincil kullanıcısı iletiyi alamaz. Örneğin, mqweb sunucusu sınıf yolu geçersizse.

#### 503

Kuyruk yöneticisi çalışmıyor.

## Yanıt üstbilgileri

Yanıtın aşağıdaki üstbilgileri döndürülür:

### İçerik-Dil

Herhangi bir hata ya da kural dışı durum olayında yanıt iletisinin dil tanıtıcısını belirtir. Herhangi bir hata ya da kural dışı durum koşuluna ilişkin gerekli dili belirtmek üzere Accept - Language istek üstbilgisiyle birlikte kullanılır. İstenen dil desteklenmiyorsa, mqweb sunucusu varsayılan değeri kullanılır.

### İçerik Uzunluğu

HTTP yanıt gövdesinin uzunluğunu, içerik olmadığında da belirtir. Değer, ileti verilerinin uzunluk (byte) değerini içerir.

### İçerik Tipi

Alınan iletinin yanıt gövdesinde döndürülen içeriğin tipini belirtir. Başarı üzerine, değer `text/plain; charset=utf-8` olur. Herhangi bir hata ya da kural dışı durum durumunda, değer `application/json; charset=utf-8` olur.

### REST API V1 REST API V2 **ibm-mq-md-correlationId**

Alınan iletinin ilintilendirme tanıtıcısını belirtir. Alınan ileti geçerli bir ilinti tanıtıcısı içerirse, üstbilgi döndürülür. Bu, 24 baytı gösteren, 48 karakterden oluşan onaltılı kodlanmış bir dizgi olarak gösterilir.

Örneğin:

```
ibm-mq-md-correlationId: 414d5120514d4144455620202020202067d8bf5923582e02
```

### V 9.2.5 REST API V3 **ibm-mq-md-correlationId**

Alınan iletinin ilintilendirme tanıtıcısını belirtir. Alınan ileti geçerli bir ilinti tanıtıcısı içerirse, üstbilgi döndürülür. İlinti tanıtıcısı aşağıdaki biçimlerden birini alabilir:

- A 48 character hexadecimal encoded string, representing 24 bytes, prefixed with the string "ID: ".  
Örneğin:

```
ibm-mq-md-correlationId: ID:414d5120514d4144455620202020202067d8bf5923582e02
```

- Uygulamaya özgü bir değer. Değer, uygulamaya özgü bir dizgidir:

```
ibm-mq-md-correlationId: My-Custom-CorrelId
```

### **ibm-mq-md-süre bitimi**

Alınan iletinin kalan süre bitimi süresini belirtir. Üstbilgi, aşağıdaki değerlerden biri olabilir:

#### **sınırsız**

İletinin süresi dolmaz.

#### **Tamsayı değeri**

İleti süre bitiminden önce kalan milisaniye sayısı.

### **REST API V1 REST API V2 ibm-mq-md-messageId**

IBM MQ tarafından bu iletiye ayrılmış olan ileti tanıtıcısını belirtir. `ibm-mq-md-correlationId` üstbilgisinde olduğu gibi, 24 baytı gösteren, 48 karakterden oluşan onaltılı kodlanmış bir dizgi olarak gösterilir.

Örneğin:

```
ibm-mq-md-messageId: 414d5120514d414445562020202020202067d8ce5923582f07
```

### **V 9.2.5 REST API V3 ibm-mq-md-messageId**

IBM MQ tarafından bu iletiye ayrılmış olan ileti tanıtıcısını belirtir. Like the `ibm-mq-md-correlationId` header, it is represented as a 48 character hexadecimal encoded string, representing 24 bytes, prefixed with the string "ID: "

Örneğin:

```
ibm-mq-md-messageId: ID:414d5120514d414445562020202020202067d8ce5923582f07
```

### **ibm-mq-md-persistence**

Alınan iletinin kalıcılığını belirtir. Üstbilgi, aşağıdaki değerlerden biri olabilir:

#### **nonPersistent**

İleti, sistem arızalarını ya da kuyruk yöneticisini yeniden başlatma işlemini sürdürmez.

#### **Kalıcı**

İleti, sistem arızalarını ya da kuyruk yöneticisini yeniden başlatma işlemini sürdürür.

### **V 9.2.5 REST API V3 ibm-mq-md-priority**

İleti önceliğinin ayarını döndürür. Örneğin:

```
ibm-mq-md-priority: 3
```

### **ibm-mq-md-replyTo**

Alınan iletiye ilişkin yanıtlama hedefini belirtir. Üstbilginin biçimi, yanıtlama kuyruğu ve kuyruk yöneticisinin standart gösterimini ( `replyQueue@replyQmgr`) kullanır.

Örneğin:

```
ibm-mq-md-replyTo: myReplyQueue@myReplyQMGR
```

### **V 9.2.5 REST API V3 ibm-mq-usr**

İleti kullanıcı tanımlı özellikleri döndürür. İletide birden çok özellik ayarlanabilir; bu durumda, `ibm-mq-usr` yanıt üstbilgisinin iki ya da daha çok ayrı eşgörunümü olacaktır.

Örneğin:

```
ibm-mq-usr: myIPprop;5;short
ibm-mq-usr: mySProp;"hi";string
ibm-mq-usr: myBProp>true;boolean
```

Özelliklerde aşağıdaki sözdizimi bulunur:

```
ibm-mq-usr: property_name; user_value; user_type
```

#### **özellik\_adi**

Belirlenmekte olan kullanıcı özelliğinin adı. Bu geçerli bir JMS özelliği adı olmalıdır.

#### **kullanıcı\_değeri**

Özelliğin değeri.

#### **kullanıcı\_türü**

Özelliğin tipi:

- `boolean` (true/false, MQBOOL)
- `byte` (8 bit tamsayı, MQINT8)
- `short` (16 bitlik tamsayı, MQINT16)
- `integer` (32 bitlik tamsayı, MQINT32)
- `long` (64 bitlik tamsayı, MQINT64)
- `float` (32 bit gerçek, MQFLOAT32)
- `double` (64 bit gerçek, MQFLOAT64)
- `string` (tırnak içine alınmış dizgi)

## **Yanıt gövdesi biçimi**

İşlem başarılı olduğunda, yanıt gövdesi alınan iletiden ileti gövini içerir. Bir hata ortaya çıkarsa, yanıt gövdesi bir JSON biçimli hata iletisi içerir. Her iki yanıt da UTF-8 kodlanmıştır. Daha fazla bilgi için bkz. [REST API hata işleme](#).

Bir ileti alınırken yalnızca IBM MQ MQSTR ya da JMS `TextMessage` biçimlenmiş iletilerin desteklendiğine dikkat edin.

GET engelleyici olarak işaretlenen bir kuyruğa göz atılırken içerik döndürülmez.

Göz atılmakta olan kuyruk, yinelenen ileti tanıtıcıları içeren iletiler içeriyorsa, ileti tanıtıcısında süzgeç uygulanırken ilk ileti döndürülür.

## **Örnekler**

Aşağıdaki örneklerde v2 kaynak URL 'si kullanılır. IBM MQ Daha önceki bir IBM MQ 9.1.5 sürümünü kullanıyorsanız, bunun yerine v1 kaynak URL adresini kullanmanız gerekir. Yani, kaynak URL 'sinde, örnek URL 'nin v2adresini kullandığı v1 yerine kullanın.

The following example logs in a user called `mquser` with the password `mquser`. cURL' de, oturum açma isteği aşağıdaki Windows örneğine benzebilir. LTPA simgesi, `-c` işareti kullanılarak `cookiejar.txt` dosyasında depolanır:

```
curl -k "https://localhost:9443/ibmmq/rest/v2/login" -X POST
-H "Content-Type: application/json" --data "{\"username\":\"mquser\", \"password\":\"mquser\"}"
-c c:\cookiejar.txt
```

Kullanıcı oturum açtıktan sonra, başka istekleri doğrulamak için LTPA simgesi ve `ibm-mq-rest-csrf-token` HTTP üstbilgisi kullanılır. The `ibm-mq-rest-csrf-token` `token_value` can be any value, including blank.

- The following Windows cURL example browses the next available message from queue Q1 on queue manager QM1, using default options:

```
curl -k "https://localhost:9443/ibmmq/rest/v2/messaging/qmgr/QM1/queue/Q1/message"
-X GET -b c:\cookiejar.txt -H "ibm-mq-rest-csrf-token: token-value"
-H "Accept: text/plain"
```



**V 9.2.5** `https://host:port/ibmmq/rest/v3/messaging/qmgr/{qmgrName}/queue/{queueName}/message`

### qmgrName

İleti alışverişi için bağlantı kurmak üzere kuyruk yöneticisinin adını belirtir. Kuyruk yöneticisi, mqweb sunucusu ile aynı makinede olmalıdır.

Kuyruk yöneticisi adı büyük ve küçük harfe duyarlıdır.

Kuyruk yöneticisi adı eğik çizgi, nokta ya da yüzde işareti içeriyorsa, bu karakterlerin kodlanmış URL adresi olması gerekir:

- Eğik çizgi (/), %2Fbiçiminde kodlanmalıdır.
- Yüzde işareti (%), %25olarak kodlanmalıdır.

### queueName

Sonraki iletinin gönderileceği kuyruğun adını belirler.

Kuyruk, yerel ya da yerel bir kuyruğu gösteren bir diğer ad olarak tanımlanmalıdır.

Kuyruk adı büyük ve küçük harfe duyarlıdır.

Kuyruk adı bir eğik çizgi ya da yüzde işareti içeriyorsa, bu karakterlerin kodlanmış URL adresi olması gerekir:

- Sağa eğik çizgi,/ , %2Folarak kodlanmalıdır.
- Yüzde işareti,% , %25olarak kodlanmalıdır.

HTTP bağlantılarını etkinleştirdiğinizde HTTPS yerine HTTP ' yi kullanabilirsiniz. HTTP ' nin etkinleştirilmesiyle ilgili daha fazla bilgi için [HTTP ve HTTPS kapılarının yapılandırılması](#) başlıklı konuya bakın.

## İsteğe bağlı sorgu değişirgeleri

**REST API V1** **REST API V2** `correlationId=hexValue`

HTTP yönteminin, ilgili ilinti tanıtıcısıyla sonraki iletiyi döndüreceğini belirtir.

### hexValue

Sorgu parametresi, 24 baytı gösteren, onaltılı kodlanmış 48 karakterden oluşan bir dizilim olarak belirtilmelidir.

Örneğin:

```
../message?correlationId=414d5120514d4144455620202020202067d8bf5923582e02
```

**V 9.2.5** **REST API V3** `correlationId= Tanıtıcı:hexValue ya da`

`correlationId=application_specific_value`

HTTP yönteminin, ilgili ilinti tanıtıcısıyla sonraki iletiyi döndüreceğini belirtir.

### hexValue

The query parameter must be specified as a 48 character hexadecimal encoded string, representing 24 bytes, and preceded by the string "ID: ".

Örneğin:

```
../message?correlationId=ID:414d5120514d4144455620202020202067d8bf5923582e02
```

### uygulama\_belirtim\_değeri

Sorgu parametresi, uygulamaya özgü bir dizgi olarak belirtilebilir.

Örneğin:

```
../message?correlationId=My-Custom-CorrelId
```

## REST API V1 REST API V2 **messageId=hexValue**

HTTP yönteminin, ilgili ileti tanıtıcısı ile sonraki iletiyi döndüreceğini belirler.

### **hexValue**

Sorgu parametresi, 24 baytı gösteren, onaltılı kodlanmış 48 karakterden oluşan bir dizilim olarak belirtilmelidir.

Örneğin:

```
../message?messageId=414d5120514d4144455620202020202067d8ce5923582f07
```

## V 9.2.5 REST API V3 **messageId= Tanıtıcı:hexValue**

HTTP yönteminin, ilgili ileti tanıtıcısı ile sonraki iletiyi döndüreceğini belirler.

### **hexValue**

The query parameter must be specified as a 48 character hexadecimal encoded string, representing 24 bytes, and preceded by the string "ID: ".

Örneğin:

```
../message?messageId=ID:414d5120514d4144455620202020202067d8ce5923582f07
```

## **wait=integerValue**

HTTP yönteminin sonraki iletinin kullanılabilir olması için *integerValue* milisaniye bekleyeceğini belirtir.

### **integerValue**

Sorgu parametresi, milisaniye süresini gösteren bir tamsayı değeri olarak belirtilmelidir. En yüksek değer 2147483647 'dir.

Örneğin:

```
../message?wait=120000
```

## **İstek üstbilgileri**

İstekle birlikte aşağıdaki üstbilgiler gönderilmelidir:

### **Yetkilendirme**

Temel kimlik doğrulaması kullanıyorsanız bu üstbilgi gönderilmelidir. Daha fazla bilgi için bkz. [REST API ile HTTP temel kimlik doğrulamasını kullanma](#).

### **ibm-mq-rest-csrf-token**

Bu üstbilgi ayarlanmalıdır, ancak değer boş olmak da dahil olmak üzere herhangi bir şey olabilir.

İsteğe bağlı olarak aşağıdaki üstbilgiler istekle birlikte gönderilebilir:

### **Kabul Et-Karakter Kümesi**

Bu üstbilgi, yanıt için hangi karakter kümesinin kabul edilebilir olduğunu göstermek için kullanılabilir. Belirtilirse, bu üstbilgi UTF-8 olarak ayarlanmalıdır.

### **Kabul Et-Dil**

Bu üstbilgi, yanıt ileti gövdesinde döndürülen kural dışı durumlar ya da hata iletileri için gerekli dili belirtir.

## **İstek gövde biçimi**

Yok.



## **Güvenlik Gereksinimleri**


Çağırın, mqweb sunucusunda kimliği doğrulanmalıdır. MQWebAdmin ve MQWebAdminRO rolleri, messaging REST API için geçerli değildir. REST API güvenlik güvenliğiyle ilgili ek bilgi için [IBM MQ Console and REST API security](#) başlıklı konuya bakın.




Kimliği doğrulandıktan sonra, kullanıcı hem messaging REST API hem de administrative REST API'ı kullanabilen bir mqweb sunucusuna doğrulanır.

Çağırana ilişkin güvenlik birincil kullanıcısının, belirtilen kuyruktan ileti alma yeteneği verilmelidir:

- Kaynak URL 'sinin *{queueName}* kısmı tarafından belirtilen kuyruk, GET etkinleştirilmiş olmalıdır.
-  For the queue that is specified by the *{queueName}* portion of the resource URL, +GET, +INQ, and +BROWSE authority must be granted to the security principal of the caller.
-  Kaynak URL adresinin *{queueName}* kısmı tarafından belirtilen kuyruk için UPDATE, arayan kişinin güvenlik birincil kullanıcısı için erişim izni verilmelidir.

 AIX, Linux, and Windows' da, **setmqaut** komutunu kullanarak IBM MQ kaynaklarını kullanmak için güvenlik birincil kullanıcıları için yetki verebilirsiniz. Daha fazla bilgi için bkz. **setmqaut** (yetki verilmesi ya da iptal etmek).

 z/OS işletim üzerinde bkz. [z/OS üzerinde güvenliğin ayarlanması](#).

## Yanıt durumu kodları

### 200

İleti başarıyla alındı.

### 204

Kullanılabilir bir ileti yok.

### 400

Geçersiz veri sağlandı.

Örneğin, geçersiz bir sorgu değiştirgesi değeri belirtildi.

### 401

Kimlik doğrulanmadı.

Çağırın, mqweb sunucusunda kimlik doğrulanmalıdır ve MQWebAdmin, MQWebAdminROya da MQWebUser rollerinden birinin ya da daha fazlasının üyesi olmalıdır. [ibm-mq-rest-csrf-token](#) üstbilgisi de belirtilmelidir. Daha fazla bilgi için bkz. “Güvenlik Gereksinimleri” sayfa 2148.

### 403

Yetkili değil.

Çağırın, mqweb sunucusuna doğrulanır ve geçerli bir birincil kullanıcıyla ilişkilendirilir. Ancak, birincil kullanıcının tümüne erişim ya da gerekli IBM MQ kaynaklarının bir alt kümesine erişimi yok ya da MQWebUser rolünde yok. Gerekli erişim hakkında daha fazla bilgi için bkz. [“Güvenlik Gereksinimleri” sayfa 2148](#).

### 404

Kuyruk yok.

### 405

Kuyruk, GET engelleyicidir.

### 500

IBM MQ' dan sunucu sorunu ya da hata kodu.

### 501

HTTP yanıtı oluşturulamadı.

Örneğin, alınan iletinin tipi yanlış ya da doğru tip var, ancak gövde işlenemedi.

### 502

İleti alışverişi sağlayıcısı gereken işlevi desteklemediği için, yürürlükteki güvenlik birincil kullanıcısı iletiyi alamaz. Örneğin, mqweb sunucusu sınıf yolu geçersizse.

### 503

Kuyruk yöneticisi çalışmıyor.

## Yanıt üstbilgileri

Yanıtın aşağıdaki üstbilgileri döndürülür:

### İçerik-Dil

Herhangi bir hata ya da kural dışı durum olayında yanıt iletilisinin dil tanıtıcısını belirtir. Herhangi bir hata ya da kural dışı durum koşuluna ilişkin gerekli dili belirtmek üzere Accept-Language istek üstbilgisiyle birlikte kullanılır. İstenen dil desteklenmiyorsa, mqweb sunucusu varsayılan değeri kullanılır.

### İçerik Uzunluğu

HTTP yanıt gövdesinin uzunluğunu, içerik olmadığında da belirtir. Değer, ileti verilerinin uzunluk (byte) değerini içerir.

### İçerik Tipi

Alınan iletilinin yanıt gövdesinde döndürülen içeriğin tipini belirtir. Başarı üzerine, değer text/plain; charset=utf-8 olur. Herhangi bir hata ya da kural dışı durum durumunda, değer application/json; charset=utf-8 olur.

#### REST API V1 REST API V2 **ibm-mq-md-correlationId**

Alınan iletilinin ilintilendirme tanıtıcısını belirtir. Alınan ileti geçerli bir ilinti tanıtıcısı içerirse, üstbilgi döndürülür. Bu, 24 baytı gösteren, 48 karakterden oluşan onaltılı kodlanmış bir dizgi olarak gösterilir. Örneğin:

```
ibm-mq-md-correlationId: 414d5120514d4144455620202020202067d8bf5923582e02
```

#### V 9.2.5 REST API V3 **ibm-mq-md-correlationId**

Alınan iletilinin ilintilendirme tanıtıcısını belirtir. Alınan ileti geçerli bir ilinti tanıtıcısı içerirse, üstbilgi döndürülür. İlinti tanıtıcısı aşağıdaki biçimlerden birini alabilir:

- A 48 character hexadecimal encoded string, representing 24 bytes, prefixed with the string "ID:". Örneğin:

```
ibm-mq-md-correlationId: ID:414d5120514d4144455620202020202067d8bf5923582e02
```

- Uygulamaya özgü bir değer. Değer, uygulamaya özgü bir dizgidir:

```
ibm-mq-md-correlationId: My-Custom-CorrelId
```

### ibm-mq-md-süre bitimi

Alınan iletilinin kalan süre bitimi süresini belirtir. Üstbilgi, aşağıdaki değerlerden biri olabilir:

#### sınırsız

İletinin süresi dolmaz.

#### Tamsayı değeri

İleti süre bitiminden önce kalan milisaniye sayısı.

#### REST API V1 REST API V2 **ibm-mq-md-messageId**

IBM MQ tarafından bu iletiye ayrılmış olan ileti tanıtıcısını belirtir. `ibm-mq-md-correlationId` üstbilgisinde olduğu gibi, 24 baytı gösteren, 48 karakterden oluşan onaltılı kodlanmış bir dizgi olarak gösterilir.

Örneğin:

```
ibm-mq-md-messageId: 414d5120514d4144455620202020202067d8ce5923582f07
```

#### V 9.2.5 REST API V3 **ibm-mq-md-messageId**

IBM MQ tarafından bu iletiye ayrılmış olan ileti tanıtıcısını belirtir. Like the `ibm-mq-md-correlationId` header, it is represented as a 48 character hexadecimal encoded string, representing 24 bytes, prefixed with the string "ID: "

Örneğin:

```
ibm-mq-md-messageId: ID:414d5120514d4144455620202020202067d8ce5923582f07
```

### **ibm-mq-md-persistence**

Alınan iletinin kalıcılığını belirtir. Üstbilgi, aşağıdaki değerlerden biri olabilir:

#### **nonPersistent**

İleti, sistem arızalarını ya da kuyruk yöneticisini yeniden başlatma işlemini sürdürmez.

#### **Kalıcı**

İleti, sistem arızalarını ya da kuyruk yöneticisini yeniden başlatma işlemini sürdürür.

### **V 9.2.5 RESTAPI V3 ibm-mq-md-priority**

İleti önceliğinin ayarını döndürür. Örneğin:

```
ibm-mq-md-priority: 3
```

### **ibm-mq-md-replyTo**

Alınan iletiyle ilişkin yanıtlama hedefini belirtir. Üstbilginin biçimi, yanıtlama kuyruğu ve kuyruk yöneticisinin standart gösterimini ( `replyQueue@replyQmgr`) kullanır.

Örneğin:

```
ibm-mq-md-replyTo: myReplyQueue@myReplyQmgr
```

### **V 9.2.5 RESTAPI V3 ibm-mq-usr**

İleti kullanıcı tanımlı özellikleri döndürür. İletide birden çok özellik ayarlanabilir; bu durumda, `ibm-mq-usr` yanıt üstbilgisinin iki ya da daha çok ayrı eşgörünümü olacaktır.

Örneğin:

```
ibm-mq-usr: myIPprop;5;short
ibm-mq-usr: mySProp;"hi";string
ibm-mq-usr: myBProp>true;boolean
```

Özelliklerde aşağıdaki sözdizimi bulunur:

```
ibm-mq-usr: property_name; user_value; user_type
```

#### **özellik\_adi**

Belirlenmekte olan kullanıcı özelliğinin adı. Bu geçerli bir JMS özelliği adı olmalıdır.

#### **kullanıcı\_değeri**

Özelliğin değeri.

#### **kullanıcı\_türü**

Özelliğin tipi:

- `boolean` (true/false, MQBOOL)
- `byte` (8 bit tamsayı, MQINT8)
- `short` (16 bitlik tamsayı, MQINT16)
- `integer` (32 bitlik tamsayı, MQINT32)
- `long` (64 bitlik tamsayı, MQINT64)
- `float` (32 bit gerçek, MQFLOAT32)
- `double` (64 bit gerçek, MQFLOAT64)
- `string` (tırnak içine alınmış dizgi)

## Yanıt gövdesi biçimi

İşlem başarılı olduğunda, yanıt gövdesi alınan iletiden ileti gövini içerir. Bir hata ortaya çıkarsa, yanıt gövdesi bir JSON biçimli hata iletisi içerir. Her iki yanıt da UTF-8 kodlanmıştır. Daha fazla bilgi için bkz. [REST API hata işleme](#).

Bir ileti alınırken yalnızca IBM MQ MQSTR ve JMS TextMessage biçimlendirilmiş ileteler desteklendiğini unutmayın. Daha sonra, tüm ileteler Sync-point altında alınır ve işlenmeyen ileteler kuyruğun üzerinde bırakılır. IBM MQ kuyruğu, bu zehirli ileteleri alternatif bir hedefe taşımak için yapılandırılabilir. Daha fazla bilgi için [JMS için IBM MQ sınıflarındaki zehirli iletelerin işlenmesibaşlıklı konuya](#) bakın.

## Örnekler

Aşağıdaki örneklerde v2 kaynak URL 'si kullanılır. IBM MQ Daha önceki bir IBM MQ 9.1.5 sürümünü kullanıyorsanız, bunun yerine v1 kaynak URL adresini kullanmanız gerekir. Yani, kaynak URL 'sinde, örnek URL 'nin v2adresini kullandığı v1 yerine kullanın.

The following example logs in a user called mquser with the password mquser. cURL' de, oturum açma isteği aşağıdaki Windows örneğine benzebilir. LTPA simgesi, -c işareti kullanılarak cookiejar.txt dosyasında depolanır:

```
curl -k "https://localhost:9443/ibmmq/rest/v2/login" -X POST
-H "Content-Type: application/json" --data "{\"username\":\"mquser\", \"password\":\"mquser\"}"
-c c:\cookiejar.txt
```

Kullanıcı oturum açtıktan sonra, başka istekleri doğrulamak için LTPA simgesi ve ibm-mq-rest-csrf-token HTTP üstbilgisi kullanılır. The ibm-mq-rest-csrf-token token\_value can be any value, including blank.

- The following Windows cURL example removes the next available message from queue Q1 on queue manager QM1, using default options:

```
curl -k "https://localhost:9443/ibmmq/rest/v2/messaging/qmgr/QM1/queue/Q1/message"
-X DELETE -b c:\cookiejar.txt -H "ibm-mq-rest-csrf-token: token-value"
-H "Accept: text/plain"
```

- The following Windows cURL example removes a message with a specific correlation ID, 00000000000000000000000000000000abcdabcd, from queue Q1 on queue manager QM1:

```
curl -k "https://localhost:9443/ibmmq/rest/v2/messaging/qmgr/QM1/queue/Q1/message?
correlationId=00000000000000000000000000000000abcdabcd"
-X DELETE -b c:\cookiejar.txt -H "ibm-mq-rest-csrf-token: token-value"
-H "Accept: text/plain"
```

- The following Windows cURL example removes a message with a specific correlation ID, 00000000000000000000000000000000abcdabcd, from queue Q1 on queue manager QM1, waiting for up to 30 seconds for the message to become available. Belirlenen ileti kuyruğa konmadan 30 saniye geçiyorsa, DELETE çağrısı ileti olmadan döner:

```
curl -k "https://localhost:9443/ibmmq/rest/v2/messaging/qmgr/QM1/queue/Q1/message?
correlationId=00000000000000000000000000000000abcdabcd&wait=30000"
-X DELETE -b c:\cookiejar.txt -H "ibm-mq-rest-csrf-token: token-value"
-H "Accept: text/plain"
```

**V 9.2.0** [/message/qmgr/{qmgrName}/queue/{queueName}/messagelist](#)

**V 9.2.0** Belirtilen kuyruk yöneticisinden belirtilen kuyruktan var olan iletelerin listesini almak için, /messaging/qmgr/{qmgrName}/queue/{queueName}/messagelist kaynağıyla HTTP GET yöntemini kullanabilirsiniz.

## V 9.2.0 GET

Belirtilen kuyruk yöneticisinden belirtilen kuyruktan var olan iletilerin listesini almak için, /messaging/qmgr/{qmgrName}/queue/{queueName}/messageList kaynağıyla HTTP GET yöntemini kullanabilirsiniz.

Belirtilen kuyruk yöneticisinden ve kuyruğundan gelen iletilerin özet listesini göz attır. Kuyruk yöneticisi, mqweb sunucusu ile aynı makinede olmalıdır. Özet verileri, HTTP yanıt gövdesinde JSON olarak biçimlendirilmiş bir dizi olarak döndürülür. Veriler, iletilerin bilgi yükünü içermez ve yürürlükteki kullanıcı bağlamı kullanılarak alınır. İlişkili kuyruktan hiçbir ileti kaldırılmadı.

GET işlemi engellenmiş bir kuyruktan kullanılabilir iletilerin listesini almak için bir istek yapılırsa, boş bir JSON dizisi döndürülür.

- “Kaynak URL'si” sayfa 2153
- “İsteğe bağlı sorgu değiştiricileri” sayfa 2154
- “İstek üstbilgileri” sayfa 2155
- “İstek gövde biçimi” sayfa 2155
- “Güvenlik Gereksinimleri” sayfa 2155
- “Yanıt durumu kodları” sayfa 2155
- “Yanıt üstbilgileri” sayfa 2156
- “Yanıt gövdesi biçimi” sayfa 2156
- “Örnekler” sayfa 2157

## Kaynak URL'si

`https://host:port/ibmmq/rest/v1/messaging/qmgr/{qmgrName}/queue/{queueName}/messageList`

`https://host:port/ibmmq/rest/v2/messaging/qmgr/{qmgrName}/queue/{queueName}/messageList`

V 9.2.5 `https://host:port/ibmmq/rest/v3/messaging/qmgr/{qmgrName}/queue/{queueName}/messageList`

## qmgrName

İleti alışverişi için bağlantı kurmak üzere kuyruk yöneticisinin adını belirtir. Kuyruk yöneticisi, mqweb sunucusu ile aynı makinede olmalıdır.

Kuyruk yöneticisi adı büyük ve küçük harfe duyarlıdır.

Kuyruk yöneticisi adı eğik çizgi, nokta ya da yüzde işareti içeriyorsa, bu karakterlerin kodlanmış URL adresi olması gerekir:

- Eğik çizgi (/), %2Fbiçiminde kodlanmalıdır.
- Yüzde işareti (%), %25olarak kodlanmalıdır.

## queueName

İletilerin göz atılacağı kuyruğun adını belirtir.

Kuyruk, yerel bir kuyruğa işaret eden yerel ya da diğer ad olarak tanımlanmalıdır.

Kuyruk adı büyük ve küçük harfe duyarlıdır.

Kuyruk adı bir eğik çizgi ya da yüzde işareti içeriyorsa, bu karakterlerin kodlanmış URL adresi olması gerekir:

- Sağa eğik çizgi,/, %2Folarak kodlanmalıdır.
- Yüzde işareti,%, %25olarak kodlanmalıdır.

HTTP bağlantılarını etkinleştirdiğinizde HTTPS yerine HTTP ' yi kullanabilirsiniz. HTTP ' nin etkinleştirilmesiyle ilgili daha fazla bilgi için [HTTP ve HTTPS kapılarının yapılandırılması](#) başlıklı konuya bakın.

## İsteğe bağlı sorgu değıştirgeleri

### REST API V2 correlationId=hexValue

HTTP yönteminin, ilgili ilinti tanıtıcısıyla sonraki iletiyi döndüreceğini belirtir.

#### hexValue

Sorgu parametresi, 24 baytı gösteren, onaltılı kodlanmış 48 karakterden oluşan bir dizilim olarak belirtilmelidir.

Örneğin:

```
../messagelist?correlationId=414d5120514d41444556202020202067d8bf5923582e02
```

### V 9.2.5 REST API V3 correlationId= Tanıtıcı:hexValue ya da correlationId=application\_specific\_value

HTTP yönteminin, karşılık gelen ilinti tanıtıcısına sahip iletilerin bir listesini döndüreceğini belirtir.

#### hexValue

The query parameter must be specified as a 48 character hexadecimal encoded string, representing 24 bytes, and preceded by the string "ID: ".

Örneğin:

```
../message?correlationId=ID:414d5120514d41444556202020202067d8bf5923582e02
```

#### uygulama\_belirtim\_değeri

Sorgu parametresi, uygulamaya özgü bir dizgi olarak belirtilebilir.

Örneğin:

```
../message?correlationId=My-Custom-CorrelId
```

### REST API V1 REST API V2 messageId=hexValue

HTTP yönteminin, ilgili ileti tanıtıcısı ile sonraki iletiyi döndüreceğini belirler.

#### hexValue

Sorgu parametresi, 24 baytı gösteren, onaltılı kodlanmış 48 karakterden oluşan bir dizilim olarak belirtilmelidir.

Örneğin:

```
../message?messageId=414d5120514d41444556202020202067d8ce5923582f07
```

### V 9.2.5 REST API V3 messageId= Tanıtıcı:hexValue

HTTP yönteminin, ilgili ileti tanıtıcısı ile sonraki iletiyi döndüreceğini belirler.

#### hexValue

The query parameter must be specified as a 48 character hexadecimal encoded string, representing 24 bytes, and preceded by the string "ID: ".

Örneğin:

```
../message?messageId=ID:414d5120514d41444556202020202067d8ce5923582f07
```

#### limit=integerValue

HTTP yöntemi yanıt gövdesinin *integerValue* JSON öğeleriyle sınırlanmış olduğunu belirtir.

#### integerValue

Sorgu parametresi, JSON yanıt gövdesinde yer alan öğe sayısı üst sınırını temsil eden bir tamsayı değeri olarak belirtilmelidir.

Varsayılan değeri 10 'tır ve en yüksek değeri 2147483647 'dir.

Örneğin:

```
../messagelist?limit=250
```

## İstek üstbilgileri

İstekle birlikte aşağıdaki üstbilgiler gönderilmelidir:

### Yetkilendirme

Temel kimlik doğrulaması kullanıyorsanız bu üstbilgi gönderilmelidir. Daha fazla bilgi için bkz. [REST API ile HTTP temel kimlik doğrulamasını kullanma](#).

### ibm-mq-rest-csrf-token

Bu üstbilgi ayarlanmalıdır, ancak değer boş olmak da dahil olmak üzere herhangi bir şey olabilir.

İsteğe bağlı olarak aşağıdaki üstbilgiler istekle birlikte gönderilebilir:

### Kabul Et-Karakter Kümesi

Bu üstbilgi, yanıt için hangi karakter kümesinin kabul edilebilir olduğunu göstermek için kullanılabilir. Belirtilirse, bu üstbilgi UTF-8 olarak ayarlanmalıdır.

### Kabul Et-Dil

Bu üstbilgi, yanıt iletisi gövdesinde döndürülen kural dışı durumlar ya da hata iletileri için gerekli dili belirtir.

## İstek gövde biçimi




Yok.


## Güvenlik Gereksinimleri


Çağırın, mqweb sunucusunda kimliği doğrulanmalıdır. MQWebAdmin ve MQWebAdminRO rolleri, messaging REST API için geçerli değildir. REST API güvenlik güvenliğiyle ilgili ek bilgi için [IBM MQ Console and REST API security](#) başlıklı konuya bakın.

Kimliği doğrulandıktan sonra, kullanıcı hem messaging REST API hem de administrative REST API' i kullanabilen bir mqweb sunucusuna doğrulanır.

Çağırının güvenlik birincil kullanıcıasına, belirtilen kuyruktaki iletilere göz atma yeteneği verilmelidir:

- Kaynak URL adresinin *{queueName}* kısmı tarafından belirtilen kuyruk BROWSE etkinleştirilmiş olmalıdır.
-   Kaynak URL adresinin *{queueName}* kısmı tarafından belirtilen kuyruk için, çağırının güvenlik birincil kullanıcıasına +GET, +INQ ve +BROWSE yetkisi verilmelidir.
-  Kaynak URL adresinin *{queueName}* kısmı tarafından belirtilen kuyruk için UPDATE, çağırının güvenlik birincil kullanıcıasına erişim verilmelidir.

 AIX, Linux, and Windows' da, **setmqaut** komutunu kullanarak IBM MQ kaynaklarını kullanmak için güvenlik birincil kullanıcıları için yetki verebilirsiniz. Daha fazla bilgi için bkz. [setmqaut](#) (yetki verilmesi ya da iptal etmek).

 z/OS işletim üzerinde bkz. [z/OS üzerinde güvenliğin ayarlanması](#).

## Yanıt durumu kodları

### 200

İleti listesi başarıyla alındı.

### 400

Geçersiz veri sağlandı.

Örneğin, geçersiz bir sorgu değiştirgesi değeri belirtildi.

#### 401

Kimlik doğrulanmadı.

Çağırın, mqweb sunucusunda kimlik doğrulanmalıdır ve MQWebAdmin, MQWebAdminROya da MQWebUser rollerinden birinin ya da daha fazlasının üyesi olmalıdır. `ibm-mq-rest-csrf-token` üstbilgisi de belirtilmelidir. Daha fazla bilgi için bkz [“Güvenlik Gereksinimleri” sayfa 2155](#).

#### 403

Yetkili değil.

Çağırın, mqweb sunucusuna doğrulanır ve geçerli bir birincil kullanıcıyla ilişkilendirilir. Ancak, birincil kullanıcının tümüne erişim ya da gerekli IBM MQ kaynaklarının bir alt kümesine erişimi yok ya da MQWebUser rolünde yok. Gerekli erişim hakkında daha fazla bilgi için bkz. [“Güvenlik Gereksinimleri” sayfa 2155](#).

#### 404

Kuyruk yok.

#### 500

IBM MQ' dan sunucu sorunu ya da hata kodu.

#### 501

HTTP yanıtı oluşturulamadı.

Örneğin, alınan iletinin tipi yanlış ya da doğru tip var, ancak gövde işlenemedi.

#### 502

İleti alışverişi sağlayıcısı gereken işlevi desteklemediği için, yürürlükteki güvenlik birincil kullanıcısı iletiyi alamaz. Örneğin, mqweb sunucusu sınıf yolu geçersizse.

#### 503

Kuyruk yöneticisi çalışmıyor.

## Yanıt üstbilgileri

### İçerik-Dil

Herhangi bir hata ya da kural dışı durum olayında yanıt iletisinin dil tanıtıcısını belirtir. Herhangi bir hata ya da kural dışı durum koşuluna ilişkin gerekli dili belirtmek üzere `Accept - Language` istek üstbilgisiyle birlikte kullanılır. İstenen dil desteklenmiyorsa, mqweb sunucusu varsayılan değeri kullanılır.

### İçerik Uzunluğu

HTTP yanıt gövdesinin uzunluğunu, içerik olmadığında da belirtir. Değer, ileti verilerinin bayt cinsinden uzunluğunu içerir.

### İçerik Tipi

Yanıt gövdesinin tipini belirtir. Değer `application/json; charset=utf-8` dir.

## Yanıt gövdesi biçimi

Başarı üzerine yanıt gövdesi, UTF-8 kodlamalı bir yanıttır. Yanıt, messagesadlı tek bir JSON dizisi içeren bir dış JSON nesnesi içeriyor. Dizideki her öğe, kuyrukta bir iletiyle ilgili bilgileri içeren bir JSON nesnesidir. Her öğe aşağıdaki öznitelikleri içerir:

### REST API V1 REST API V2 **correlationId**

İletin ilinti tanıtıcısını belirtir. İleti geçerli bir ilinti tanıtıcısı içerirse değer döndürülür.

### V 9.2.5 REST API V3 **correlationId**

İletin ilinti tanıtıcısını belirtir. İleti geçerli bir ilinti tanıtıcısı içerirse değer döndürülür. İlinti tanıtıcısının başında "ID:" dizgisi var ya da uygulamaya özgü bir değer olabilir.

### REST API V1 REST API V2 **messageId**

IBM MQ tarafından bu iletiye ayrılmış olan ileti tanıtıcısını belirtir. 48 karakterden oluşan onaltılı kodlanmış bir dizgi olarak gösterilir ve 24 baytı gösterir.





```
-X GET -b c:\cookiejar.txt -H "ibm-mq-rest-csrf-token: token-value"
-H "Accept: application/json"
```

## V 9.2.0 /message/qmgr/{qmgrName}/topic/{topicString}/message

İletileri, belirtilen kuyruk yöneticisiyle ilgili olarak belirtilen konuya yayınlamak için, /messaging/qmgr/{qmgrName}/topic/{topicString}/message kaynağıyla HTTP POST yöntemini kullanabilirsiniz.

## V 9.2.0 GÖNDER

İletileri, belirtilen kuyruk yöneticisiyle ilgili olarak belirtilen konuya yayınlamak için, /messaging/qmgr/{qmgrName}/topic/{topicString}/message kaynağıyla HTTP POST yöntemini kullanabilirsiniz.

HTTP isteği gövdesinde metin tabanlı bir iletiyi belirtilen kuyruk yöneticisine ve konuya yayınlar. Kuyruk yöneticisi, mqweb sunucusuyla aynı makinede olmalı ve yalnızca metin tabanlı iletiler desteklenmelidir. İletiler, yürürlükteki kullanıcı bağlamını kullanan MQSTR ya da JMS TextMessage biçimli iletiler olarak yayınlanır ve REST API V2 için 4 varsayılan ileti önceliğine ve REST API V3 için asDestination ileti önceliğine sahiptir.

V 9.2.5 REST API V3 , kullanıcı tanımlı ileti özelliklerini belirleme ve ileti önceliğini içerme yeteneğini ekler. Ibm-mq-md-priority ve ibm-mq-usr istek üstbilgileri yalnızca REST API V3 ile kullanılabilir. Ibm-mq-md-correlationId istek üstbilgisi, REST API V3' te farklı bir biçime sahiptir. Üstbilgi, uygulamaya özgü bir tanıttıcı olabilir ya da kodlanmış bir dize, ID: önekini gerektiriyorsa. POST isteğiniz kullanıcı tanımlı iletiler ya da uygulamaya özgü ilinti tanıttıcısı içeriyorsa, ileti bir JMS TextMessage olarak biçimlendirilir.

- “Kaynak URL'si” sayfa 2158
- “İstek üstbilgileri” sayfa 2159
- “İstek gövde biçimi” sayfa 2161
- “Güvenlik Gereksinimleri” sayfa 2161
- “Yanıt durumu kodları” sayfa 2161
- “Yanıt üstbilgileri” sayfa 2162
- “Yanıt gövdesi biçimi” sayfa 2162
- “Örnekler” sayfa 2162

## Kaynak URL'si

`https://host:port/ibmmq/rest/v2/messaging/qmgr/{qmgrName}/topic/{topicString}/message`

V 9.2.5 `https://host:port/ibmmq/rest/v3/messaging/qmgr/{qmgrName}/topic/{topicString}/message`

### qmgrName

İleti alışverişi için bağlantı kurmak üzere kuyruk yöneticisinin adını belirtir. Kuyruk yöneticisi, mqweb sunucusu ile aynı makinede olmalıdır.

Kuyruk yöneticisi adı büyük ve küçük harfe duyarlıdır.

Kuyruk yöneticisi adı eğik çizgi, nokta ya da yüzde işareti içeriyorsa, bu karakterlerin kodlanmış URL adresi olması gerekir:

- Eğik çizgi %2F olarak kodlanmalıdır.
- Nokta, %2E şeklinde kodlanmalıdır.
- Yüzde imi, %25 olarak kodlanmalıdır.

### topicString

İletinin yayınlayacağı konu dizesini belirtir.

Konu dizgisi büyük ve küçük harfe duyarlıdır. Konu dizgisi, eğik çizgi sınırlayıcısına göre ayrılmış birden çok konu düzeyi içerebilir.

Konu dizgisinde bir yüzde işareti, bir nokta ya da soru işareti bulunuyorsa, bu karakterlerin kodlanmış URL adresi olması gerekir:

- Yüzde imi, %25olarak kodlanmalıdır.
- Nokta, %2Eşeklinde kodlanmalıdır.
- Bir soru işareti %3Folarak kodlanmalıdır.

Konu dizgisi bir sağa eğik çizgiyle başlarsa ya da biterse, bu bir %2File kodlanmalıdır.

Örneğin, konu dizgisinde yayınlamak için:

- sport/football on queue manager MY.QMGR, you use the following URL:

```
https://localhost:9443/ibmmq/rest/v2/messaging/qmgr/MY%2EQMGR/topic/sport/football/message
```

- /sport/football on queue manager MY.QMGR, you use the following URL:

```
https://localhost:9443/ibmmq/rest/v2/messaging/qmgr/MY%2EQMGR/topic/%2Fsport/football/message
```

HTTP bağlantılarını etkinleştirdiğinizde HTTPS yerine HTTP ' yi kullanabilirsiniz. HTTP ' nin etkinleştirilmesiyle ilgili daha fazla bilgi için [HTTP ve HTTPS kapılarının yapılandırılması](#) başlıklı konuya bakın.

## İstek üstbilgileri

İstekle birlikte aşağıdaki üstbilgiler gönderilmelidir:

### Yetkilendirme

Temel kimlik doğrulaması kullanıyorsanız bu üstbilgi gönderilmelidir. Daha fazla bilgi için bkz. [REST API ile HTTP temel kimlik doğrulamasını kullanma](#).

### İçerik Tipi

Bu üstbilgi, aşağıdaki değerlerden biriyle gönderilmelidir:

- text/plain;charset=utf-8
- text/html;charset=utf-8
- text/xml;charset=utf-8
- application/json;charset=utf-8
- application/xml;charset=utf-8

**Not:** If *karakter kümesi* is omitted from the Context-Type header, UTF-8 is assumed.

### ibm-mq-rest-csrf-token

Bu üstbilgi ayarlanmalıdır, ancak değer boş olmak da dahil olmak üzere herhangi bir şey olabilir.

İsteğe bağlı olarak aşağıdaki üstbilgiler istekle birlikte gönderilebilir:

### Kabul Et-Dil

Bu üstbilgi, yanıt iletisi gövdesinde döndürülen kural dışı durumlar ya da hata iletileri için gerekli dili belirtir.

### ibm-mq-md-süre bitimi

Bu üstbilgi, oluşturulan ileti için süre bitimi süresini ayarlar. İleti süre bitimi, iletinin kuyruk yöneticisinde geldiği zamandan başlar. Sonuç ağı gecikme süresi yok sayılır. Üstbilgi, aşağıdaki değerlerden biri olarak belirtilmelidir:

#### sınırsız

İletinin süresi dolmaz.

Bu değer, varsayılan değerdir.

### **Tamsayı değeri**

İleti süre bitiminden önce milisaniye.

0-99999999900 aralığıyla sınırlanmıştır.

### **ibm-mq-md-persistence**

Bu üstbilgi, oluşturulan iletiye ilişkin kalıcılığı ayarlar. Üstbilgi, aşağıdaki değerlerden biri olarak belirtilmelidir:

#### **nonPersistent**

İleti, sistem arızalarını ya da kuyruk yöneticisini yeniden başlatma işlemini sürdürmez.

Bu değer, varsayılan değerdir.

#### **Kalıcı**

İleti, sistem arızalarını ya da kuyruk yöneticisini yeniden başlatma işlemini sürdürür.

### **V 9.2.5 RESTAPI V3 ibm-mq-md-priority**

Bu üstbilgi, oluşturulan iletinin önceliğini ayarlar. Üstbilgi, aşağıdaki değerlerden biri olarak belirtilmelidir:

#### **asDestination**

İleti, temeldeki IBM MQ kuyruk nesnesinin DEFPTY özniteliğinde belirtilen önceliği kullanır.

#### **Tamsayı değeri**

Gerçek önceliği 0-9 aralığında bir tamsayı olarak belirtin.

Örneğin:

```
ibm-mq-md-priority: asDestination
```

### **ibm-mq-md-replyTo**

Bu üstbilgi, oluşturulan iletiye ilişkin yanıtlanma hedefini ayarlar. Üstbilginin biçimi, yanıt kuyruğu ve isteğe bağlı bir kuyruk yöneticisi sağlamak için standart gösterimini kullanır: replyQueue[@replyQmgr]

Örneğin:

```
ibm-mq-md-replyTo: myReplyQueue@myReplyQMGr
```

### **V 9.2.5 RESTAPI V3 ibm-mq-usr**

İstek iletisi kullanıcı tanımlı özellikleri ayarlayın. İletide birden çok özellik ayarlanabilir. Tek bir ibm-mq-usr istek üstbilgisinde birden çok virgülle ayrılmış özellik belirleyebilir ya da ibm-mq-usr istek üstbilgisinin iki ya da daha fazla örneğini kullanabilirsiniz.

Örneğin:

```
ibm-mq-usr: myIPprop;5;short
ibm-mq-usr: mySProp;"hi";string
ibm-mq-usr: myBProp>true;boolean
ibm-mq-usr: myA;5;byte,myB;-10;integer
```

Özelliklerde aşağıdaki sözdizimi bulunur:

```
ibm-mq-usr: property_name; user_value; user_type
```

#### **özellik\_adi**

Belirlenmekte olan kullanıcı özelliğinin adı. Bu geçerli bir JMS özelliği adı olmalıdır.

#### **kullanıcı\_değeri**

Özelliğin değeri.

#### **kullanıcı\_türü**

Özelliğin tipi:

- boolean (true/false, MQBOOL)

- byte (8 bit tamsayı, MQINT8)
- short (16 bitlik tamsayı, MQINT16)
- integer (32 bitlik tamsayı, MQINT32)
- long (64 bitlik tamsayı, MQINT64)
- float (32 bit gerçek, MQFLOAT32)
- double (64 bit gerçek, MQFLOAT64)
- string (tırnak içine alınmış dizgi)

## İstek gövde biçimi

İstek gövdesi metin olmalı ve UTF-8 kodlaması kullanılmalıdır. Belirli bir metin yapısı gerekli değildir. İstek gövdesi metnini içeren MQSTR biçimlendirilmiş bir ileti oluşturulur ve belirtilen konuya yayınlanır.

**V 9.2.5** **REST API V3** REST API 'si V3 kullanıcı tanımlı özellikleri ya da uygulamaya özgü ilinti tanıtıcısı özellikleri kullanılırsa, istek gövdesi metnini içeren bir JMS TextMessage biçimli ileti yaratılır ve belirtilen kuyruğa konursa.

Ek bilgi için bkz. [örnekler](#).

## Güvenlik Gereksinimleri

Çağırıcı, mqweb sunucusunda kimliği doğrulanmalıdır. MQWebAdmin ve MQWebAdminRO rolleri, messaging REST API için geçerli değildir. REST API güvenlik güvenliğiyle ilgili ek bilgi için [IBM MQ Console and REST API security](#) başlıklı konuya bakın.

Kimliği doğrulandıktan sonra, kullanıcı hem messaging REST API hem de administrative REST API' i kullanabilen bir mqweb sunucusuna doğrulanır.

Çağırıcı güvenlik birincil kullanıcısının, belirtilen konuya ileti yayınlama yeteneği verilmelidir:

- Kaynak URL 'sinin *{topicString}* kısmı tarafından belirtilen konu PUBLISH etkinleştirilmiş olmalıdır.
- **MQ Appliance** **ALW** Kaynak URL adresinin *{topicString}* kısmı tarafından belirtilen konu için, çağırıcı kullanıcısının güvenlik birincil kullanıcısına +PUB yetkisi verilmelidir.
- **z/OS** Kaynak URL adresinin *{topicString}* kısmı tarafından belirtilen konu için, çağırıcı kullanıcısının güvenlik birincil kullanıcısına UPDATE erişimi verilmelidir.

**ALW** AIX, Linux, and Windows' da, **setmqaut** komutunu kullanarak IBM MQ kaynaklarını kullanmak için güvenlik birincil kullanıcıları için yetki verebilirsiniz. Daha fazla bilgi için bkz. [setmqaut](#) (yetki verilmesi ya da iptal etmek).

**z/OS** z/OS işletim üzerinde bkz. [z/OS üzerinde güvenliğin ayarlanması](#).

If you use Advanced Message Security (AMS) with the messaging REST API, note that all messages are encrypted by using the context of the mqweb server, not the context of the user that posts the message.

## Yanıt durumu kodları

### 201

İleti yaratıldı ve başarıyla yayınlandı.

### 400

Geçersiz veri sağlandı.

Örneğin, geçersiz bir istek üstbilgisi değeri belirtildi.

### 401

Kimlik doğrulanmadı.

Çağırın, mqweb sunucusunda kimlik doğrulanmalıdır ve MQWebAdmin, MQWebAdminROya da MQWebUser rollerinden birinin ya da daha fazlasının üyesi olmalıdır. `ibm-mq-rest-csrf-token` üstbilgisi de belirtilmelidir. Daha fazla bilgi için bkz [“Güvenlik Gereksinimleri” sayfa 2161](#).

#### 403

Yetkili değil.

Çağırın, mqweb sunucusuna doğrulanır ve geçerli bir birincil kullanıcıyla ilişkilendirilir. Ancak, birincil kullanıcının tümüne erişim ya da gerekli IBM MQ kaynaklarının bir alt kümesine erişimi yok ya da MQWebUser rolünde yok. Gerekli erişim hakkında daha fazla bilgi için bkz. [“Güvenlik Gereksinimleri” sayfa 2161](#).

#### 404

Kuyruk yöneticisi yok.

#### 405

Konu, YAYININ engellenmesini sağlar.

#### 415

İleti üstbilgisi ya da gövdesi desteklenmeyen bir ortam tipidir.

Örneğin, Content-Type üstbilgisi desteklenmeyen bir ortam tipine ayarlıdır.

#### 500

IBM MQ' dan sunucu sorunu ya da hata kodu.

#### 502

İleti alışverişi sağlayıcısı gereken işlevi desteklemediği için, yürürlükteki güvenlik birincil kullanıcısı iletiyi yayınlamaz. Örneğin, mqweb sunucusu sınıf yolu geçersizse.

#### 503

Kuyruk yöneticisi çalışmıyor.

## Yanıt üstbilgileri

Yanıtın aşağıdaki üstbilgileri döndürülür:

### İçerik-Dil

Herhangi bir hata ya da kural dışı durum olayında yanıt iletilisinin dil tanıtıcısını belirtir. Herhangi bir hata ya da kural dışı durum koşuluna ilişkin gerekli dili belirtmek üzere Accept-Language istek üstbilgisiyle birlikte kullanılır. İstenen dil desteklenmiyorsa, mqweb sunucusu varsayılan değeri kullanılır.

### İçerik Uzunluğu

HTTP yanıt gövdesinin uzunluğunu, içerik olmadığına da belirtir. Başarı üzerine, değer sıfır olur.

### İçerik Tipi

Yanıt gövdesinin tipini belirtir. Başarı üzerine, değer `text/plain;charset=utf-8` olur. Herhangi bir hata ya da kural dışı durum durumunda, değer `application/json;charset=utf-8` olur.

## Yanıt gövdesi biçimi

İleti başarıyla yayınlanırsa, yanıt gövdesi boş olur. Bir hata ortaya çıkarsa, yanıt gövdesi bir hata ileti içerir. Daha fazla bilgi için bkz. [REST API hata işleme](#).

## Örnekler

The following example logs in a user called mquser with the password mquser. cURL' de, oturum açma isteği aşağıdaki Windows örneğine benzebilir. LTPA simgesi, -c işareti kullanılarak `cookiejar.txt` dosyasında depolanır:

```
curl -k "https://localhost:9443/ibmmq/rest/v1/login" -X POST
-H "Content-Type: application/json" --data "{\"username\":\"mquser\", \"password\":\"mquser\"}"
-c c:\cookiejar.txt
```

Kullanıcı oturum açtıktan sonra, başka istekleri doğrulamak için LTPA simgesi ve `ibm-mq-rest-csrf-token` HTTP üstbilgisi kullanılır. The `ibm-mq-rest-csrf-token` `token_value` can be any value, including blank.

- The following Windows cURL example publishes a message to the topic string `myTopic` on queue manager `QM1`, using default options. İleti, "*Merhaba Dünya!*"metnini içeriyor:

```
curl -k "https://localhost:9443/ibmmq/rest/v2/messaging/qmgr/QM1/topic/myTopic/message"
-X POST -b c:\cookiejar.txt -H "ibm-mq-rest-csrf-token: token_value"
-H "Content-Type: text/plain;charset=utf-8" --data "Hello World!"
```

- The following Windows cURL example publishes a persistent message to the topic string `myTopic/thisTopic` on queue manager `QM1`, with an expiry of 2 minutes. İleti, "*Merhaba Dünya!*"metnini içeriyor:

```
curl -k "https://localhost:9443/ibmmq/rest/v2/messaging/qmgr/QM1/topic/myTopic%2FthisTopic/
message"
-X POST -b c:\cookiejar.txt -H "ibm-mq-rest-csrf-token: token_value"
-H "Content-Type: text/plain;charset=utf-8" -H "ibm-mq-md-persistence: persistent"
-H "ibm-mq-md-expiry: 120000" --data "Hello World!"
```





## Özel notlar

Bu belge, ABD'de kullanıma sunulan ürünler ve hizmetler için hazırlanmıştır.

IBM, bu belgede sözü edilen ürün, hizmet ya da özellikleri diğer ülkelerde kullanıma sunmayabilir. Bulduğunuz yerde kullanıma sunulan ürün ve hizmetleri yerel IBM müşteri temsilcisinden ya da çözüm ortağınızdan öğrenebilirsiniz. Bir IBM ürün, program ya da hizmetine gönderme yapılması, açık ya da örtük olarak yalnızca o IBM ürünü, programı ya da hizmetinin kullanılabilirliğini göstermez. Aynı işlevi gören ve IBM'in fikri mülkiyet haklarına zarar vermeyen herhangi bir ürün, program ya da hizmet de kullanılabilir. Ancak, IBM dışı ürün, program ya da hizmetlerle gerçekleştirilen işlemlerin değerlendirilmesi ve doğrulanması kullanıcının sorumluluğundadır.

IBM'in, bu belgedeki konularla ilgili patentleri ya da patent başvuruları olabilir. Bu belgenin size verilmiş olması, patentlerin izinsiz kullanım hakkının da verildiği anlamına gelmez. Lisansla ilgili sorularınızı aşağıdaki adrese yazabilirsiniz:

IBM Director of Licensing  
IBM Corporation  
North Castle Drive  
Armonk, NY 10504-1785  
U.S.A.

Çift byte (DBCS) bilgilerle ilgili lisans soruları için, ülkenizdeki IBM'in Fikri Haklar (Intellectual Property) bölümüyle bağlantı kurun ya da sorularınızı aşağıda adrese yazın:

Intellectual Property Licensing  
Legal and Intellectual Property Law  
IBM Japan, Ltd.  
19-21, Nihonbashi-Hakozakicho, Chuo-ku  
Tokyo 103-8510, Japonya

**Aşağıdaki paragraf, İngiltere ya da bu tür hükümlerin yerel yasalarla uyuşmadığı diğer ülkelerde geçerli değildir:** INTERNATIONAL BUSINESS MACHINES CORPORATION BU YAYINI, HAK İHLALİ YAPILMAYACAĞINA DAİR GARANTİLERLE TİCARİLİK VEYA BELİRLİ BİR AMACA UYGUNLUK İÇİN ZİMNİ GARANTİLER DE DAHİL OLMAK VE FAKS BUNLARLA SINIRLI OLMAMAK ÜZERE AÇIK YA DA ZİMNİ HİÇBİR GARANTİ VERMEKSİZİN "OLDUĞU GİBİ" ESASIYLA SAĞLAMAKTADIR. Bazı ülkeler bazı işlemlerde garantinin açık ya da örtük olarak reddedilmesine izin vermez; dolayısıyla, bu bildirim sizin için geçerli olmayabilir.

Bu yayın teknik yanlışlar ya da yazım hataları içerebilir. Buradaki bilgiler üzerinde düzenli olarak değişiklik yapılmaktadır; söz konusu değişiklikler sonraki basımlara yansıtılacaktır. IBM, önceden bildirimde bulunmaksızın, bu yayında açıklanan ürünler ve/ya da programlar üzerinde iyileştirmeler ve/ya da değişiklikler yapabilir.

Bu belgede IBM dışı Web sitelerine yapılan göndermeler kullanıcıya kolaylık sağlamak içindir ve bu Web sitelerinin onaylanması anlamına gelmez. Bu Web sitelerinin içerdiği malzeme, bu IBM ürününe ilişkin malzemenin bir parçası değildir ve bu tür Web sitelerinin kullanılmasının sorumluluğu size aittir.

IBM'e bilgi ilettiğinizde, IBM bu bilgileri size karşı hiçbir yükümlülük almaksızın uygun gördüğü yöntemlerle kullanabilir ya da dağıtabilir.

(i) Bağımsız olarak yaratılan programlarla, bu program da içinde olmak üzere diğer programlar arasında bilgi değiş tokuşuna ve (ii) değiş tokuş edilen bilginin karşılıklı kullanımına olanak sağlamak amacıyla bu program hakkında bilgi sahibi olmak isteyen lisans sahipleri şu adrese yazabilirler:

IBM Corporation  
Yazılım Birlikte Çalışabilirlik Koordinatörü, Bölüm 49XA  
3605 Highway 52 N

Rochester, MN 55901  
U.S.A.

Bu tür bilgiler, ilgili kayıt ve koşullar altında ve bazı durumlarda bedelli olarak edinilebilir.

Bu belgede açıklanan lisanslı program ve bu programla birlikte kullanılacak tüm lisanslı malzeme, IBM tarafından, IBM Müşteri Sözleşmesi, IBM Uluslararası Program Lisansı Sözleşmesi ya da eşdeğer herhangi bir sözleşmenin kayıt ve koşulları altında sağlanır.

Burada belirtilen performans verileri denetimli bir ortamda elde edilmiştir. Bu nedenle, başka işletim ortamlarında çok farklı sonuçlar alınabilir. Bazı ölçümler geliştirilme düzeyindeki sistemlerde yapılmıştır ve bu ölçümlerin genel kullanıma sunulan sistemlerde de aynı olacağı garanti edilemez. Ayrıca, bazı sonuçlar öngörü yöntemiyle elde edilmiş olabilir. Dolayısıyla, gerçek sonuçlar farklı olabilir. Bu belgenin kullanıcıları, kendi ortamları için geçerli verileri kendileri doğrulamalıdır.

IBM dışı ürünlerle ilgili bilgiler, bu ürünleri sağlayan firmalardan, bu firmaların yayın ve belgelerinden ve genel kullanıma açık diğer kaynaklardan alınmıştır. IBM bu ürünleri sınamamıştır ve IBM dışı ürünlerle ilgili performans doğruluğu, uyumluluk gibi iddiaları doğrulayamaz. IBM dışı ürünlerin yeteneklerine ilişkin sorular, bu ürünleri sağlayan firmalara yöneltilmelidir.

IBM'in gelecekteki yönelim ve kararlarına ilişkin tüm bildirimler değişebilir ve herhangi bir duyuruda bulunulmadan bunlardan vazgeçilebilir; bu yönelim ve kararlar yalnızca amaç ve hedefleri gösterir.

Bu belge, günlük iş ortamında kullanılan veri ve raporlara ilişkin örnekler içerir. Örneklerin olabildiğince açıklayıcı olması amacıyla kişi, şirket, marka ve ürün adları belirtilmiş olabilir. Bu adların tümü gerçek dışıdır ve gerçek iş ortamında kullanılan ad ve adreslerle olabilecek herhangi bir benzerlik tümüyle rastlantıdır.

#### YAYIN HAKKI LİSANSI:

Bu belge, çeşitli işletim platformlarında programlama tekniklerini gösteren, kaynak dilde yazılmış örnek uygulama programları içerir. Bu örnek programları, IBM'e herhangi bir ödemede bulunmadan, örnek programların yazıldığı işletim altyapısına ilişkin uygulama programlama arabirimiyle uyumlu uygulama programlarının geliştirilmesi, kullanılması, pazarlanması ya da dağıtılması amacıyla herhangi bir biçimde kopyalayabilir, değiştirebilir ve dağıtabilirsiniz. Bu örnekler her koşul altında tüm ayrıntılarıyla sınanmamıştır. Dolayısıyla, IBM bu programların güvenilirliği, bakım yapılabilirliği ya da işlevleri konusunda açık ya da örtük güvence veremez.

Bu bilgileri elektronik kopya olarak görüntülediyseniz, fotoğraflar ve renkli resimler görünmeyebilir.

## Programlama arabirimi bilgileri

Programlama arabirimi bilgileri (sağlandıysa), bu programla birlikte kullanılmak üzere uygulama yazılımları yaratmanıza yardımcı olmak üzere hazırlanmıştır.

Bu kitap, müşterinin WebSphere MQ hizmetlerini edinmek üzere program yazmasına olanak tanıyan, amaçlanan programlama arabirimlerine ilişkin bilgiler içerir.

Ancak, bu bilgiler tanılama, değiştirme ve ayarlama bilgilerini de içerebilir. Tanılama, değiştirme ve ayarlama bilgileri, uygulama yazılımlarınızda hata ayıklamanıza yardımcı olur.

**Önemli:** Bu tanılama, değiştirme ve ayarlama bilgilerini bir programlama arabirimi olarak kullanmayın; bu, değişiklik söz konusu olduğunda kullanılır.

## Ticari Markalar

IBM, IBM logosu, ibm.com, IBM Corporation 'ın dünya çapında birçok farklı hukuk düzeninde kayıtlı bulunan ticari markalarıdır. IBM ticari markalarının güncel bir listesini Web üzerinde "Telif hakkı ve ticari marka bilgileri" [www.ibm.com/legal/copytrade.shtml](http://www.ibm.com/legal/copytrade.shtml) adresinde bulabilirsiniz. Diğer ürün ve hizmet adları IBM'in veya diğer şirketlerin ticari markaları olabilir.

Microsoft ve Windows, Microsoft Corporation'ın ABD ve/veya diğer ülkelerdeki ticari markalarıdır.

UNIX, The Open Group şirketinin ABD ve diğer ülkelerdeki tescilli ticari markasıdır.

Linux, Linus Torvalds'ın ABD ve/ya da diđer ÷lkelerdeki tescilli ticari markasıdır.

Bu ÷r÷n, Eclipse Project (<https://www.eclipse.org/>) tarafından geliřtirilen yazılımları ierir.

Java ve Java tabanlı t÷m markalar ve logolar, Oracle firmasının ve/ya da iřtiraklerinin markaları ya da tescilli markalarıdır.







Parça numarası:

(1P) P/N: