

8.0

*IBM MQ Developing Applications
Reference*



Not

Bu bilgileri ve desteklediđi ürünü kullanmadan önce, [“Özel notlar” sayfa 1995](#) bölümündeki bilgileri okuyun.

Bu basım, yeni basımlarında tersi belirtilmediđi sürece, IBM® MQ sürüm 8 yayın düzeyi 0 ve sonraki tüm yayın düzeyleri ve deđişiklikler için geçerlidir.

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, 2023.

İçindekiler

Uygulama başvurusu geliştirilmesi.....	7
MQI uygulamaları başvurusu.....	7
Kod örnekleri.....	8
Değişmezler.....	61
MQI ' da kullanılan veri tipleri.....	236
İşlev çağrıları.....	605
Nesnelerin öznitelikleri.....	778
Dönüş kodları.....	847
MQI seçeneklerinin geçerliliğini denetleme kuralları.....	848
Kuyruğa alınan yayınlama/abone olma komutu iletileri.....	851
Makine kodlamaları.....	872
Rapor seçenekleri ve ileti işaretleri.....	875
Veri dönüştürme.....	879
MQRFH2 öğeleri olarak belirtilen özellikler.....	901
Kod sayfası dönüşümü.....	909
64 bit altyapılarda kodlama standartları.....	963
IBM i Application Programming Reference (ILE/RPG).....	967
Veri tipi açıklamaları.....	967
İşlev çağrıları.....	1213
Nesnelerin öznitelikleri.....	1331
Uygulamalar.....	1375
IBM i için dönüş kodları (ILE RPG).....	1388
IBM i için MQI seçeneklerinin geçerliliğini denetleme kuralları (ILE RPG).....	1389
Machine encodings on IBM i.....	1392
Rapor seçenekleri ve ileti işaretleri.....	1395
Veri dönüştürme.....	1398
Dönüştürme işlemi.....	1399
İşleme kuralları.....	1400
Rapor iletilerinin dönüştürülmesi.....	1403
MQDXP-Data-conversion çıkış değiştirgesi.....	1404
MQXCNCV-Karakterleri dönüştür.....	1409
MQCONVX-Veri dönüştürme çıkışı.....	1414
SOAP başvurusu.....	1417
amqwSOAPNETListener: IBM MQ SOAP dinleyicisi for .NET Framework 1 ya da 2.....	1418
amqswsdl: .NET Framework 1 ya da 2 hizmeti için WSDL oluşturur.....	1420
amqwclientconfig: SOAP için IBM MQ iletimi için Axis 1.4 Web hizmetleri istemcisi konuşlandırma tanımlayıcısı yaratın.....	1420
amqwdeployWMQService: web hizmeti yardımcı programını konuşlandır.....	1421
amqwRegisterdotNet: SOAP for SOAP to .NET(SOAP için IBM MQ iletimi).....	1428
Apache yazılım lisansı.....	1429
MQMD SOAP ayarları.....	1432
MQRFH2 SOAP ayarları.....	1438
runivt: SOAP kuruluşu doğrulama sınaması için IBM MQ iletimi.....	1439
SOAP için IBM MQ iletimi üzerinden Web hizmetleri güvenliğini sağlayın.....	1442
SimpleJavaListener: IBM MQ SOAP Listener for Axis 1.4.....	1445
IBM MQ SOAP dinleyicileri.....	1448
SOAP gönderen içinIBM MQ iletimi.....	1453
İşlemler.....	1454
Web hizmeti konuşlandırması için URI sözdizimi ve değiştirgeleri.....	1455
W3C SOAP over JMS URI for the IBM MQ Axis 2 client.....	1462
Desteklenen Web hizmetleri.....	1468
SOAP Web hizmeti istemcileri içinIBM MQ iletimi.....	1471

Kullanıcı çıkışları, API çıkışları ve kurulabilir hizmetler başvurusu.....	1473
MQIEP yapısı.....	1473
Veri dönüştürme çıkış başvurusu.....	1477
MQ_PUBLISH_EXIT-Yayınlama çıkışı.....	1481
Kanal-çıkış çağrılar ve veri yapıları.....	1489
Küme iş yükü çıkış çağrısı ve veri yapıları.....	1550
API çıkış başvurusu.....	1574
Kurulabilir hizmetler arabirimi başvuru bilgileri.....	1635
Kurulabilir hizmetler arabirimi başvuru bilgileri.....	1697
IBM MQ bridge for HTTP için başvuru malzemesi.....	1736
HTTP DELETE : IBM MQ bridge for HTTP komutu.....	1736
HTTP GET : IBM MQ bridge for HTTP komutu.....	1739
HTTP POST : IBM MQ bridge for HTTP komutu.....	1742
HTTP üstbilgileri.....	1746
HTTP dönüş kodları.....	1761
HTTP için WebSphere Köprüsü için ileti tipleri ve ileti eşlemeleri.....	1769
URI Biçimi.....	1771
IBM MQ .NET sınıfları ve arabirimleri.....	1771
MQAsyncStatus.NET sınıfı.....	1771
MQAuthenticationInformationRecord.NET sınıfı.....	1772
MQDestination.NET sınıfı.....	1773
MQEnvironment.NET sınıfı.....	1776
MQException.NET sınıfı.....	1778
MQGetMessageOptions.NET sınıfı.....	1779
MQManagedObject.NET sınıfı.....	1782
MQMessage.NET sınıfı.....	1784
MQProcess.NET sınıfı.....	1796
MQPropertyDescriptor.NET sınıfı.....	1798
MQPutMessageOptions.NET sınıfı.....	1800
MQQueue.NET sınıfı.....	1802
MQQueueManager.NET sınıfı.....	1810
MQSubscription.NET sınıfı.....	1822
MQTopic.NET sınıfı.....	1823
IMQObjectTrigger.NET arabirimi.....	1829
MQC.NET arabirimi.....	1830
.NET uygulamaları için karakter kümesi tanıtıcıları.....	1830
IBM MQ C++ sınıfları.....	1832
C++ ve MQI çapraz başvurusu.....	1834
ImqAuthenticationKayıt C++ sınıfı.....	1848
ImqBinary C++ sınıfı.....	1851
ImqCache C++ sınıfı.....	1852
ImqChannel C++ sınıfı.....	1855
ImqCICSBridgeHeader C++ sınıfı.....	1861
ImqDeadLetterHeader C++ sınıfı.....	1867
ImqDistributionListe C++ sınıfı.....	1869
ImqError C++ sınıfı.....	1871
ImqGetMessageOptions C++ sınıfı.....	1872
ImqHeader C++ sınıfı.....	1875
ImqIMSBridgeHeader C++ sınıfı.....	1877
ImqItem C++ sınıfı.....	1880
ImqMessage C++ sınıfı.....	1881
ImqMessageTracker C++ sınıfı.....	1888
ImqNamelist C++ sınıfı.....	1890
ImqObject C++ sınıfı.....	1892
ImqProcess C++ sınıfı.....	1897
ImqPutMessageOptions C++ sınıfı.....	1899
ImqQueue C++ sınıfı.....	1901
ImqQueueManager C++ sınıfı.....	1911

ImqReferenceHeader C++ sınıfı.....	1928
ImqString C++ sınıfı.....	1930
ImqTrigger C++ sınıfı.....	1935
ImqWorkÜstbilgi C++ sınıfı.....	1938
IBM MQ classes for JMS nesnelerinin özellikleri.....	1940
IBM MQ classes for JMS nesnelere ilişkin özellikler arasındaki bağımlılıklar.....	1944
UYGULAMA ADI.....	1945
ASYNCEXCEPTION.....	1946
BROKERCCDURSUBQ.....	1947
BROKERCCSUQ.....	1947
BROKERCONQ.....	1948
BROKERDURSUQ.....	1948
BROKERPUBQ.....	1949
BROKERPUBQMGR.....	1949
BROKERQMGR.....	1950
BROKERSUBQ.....	1950
BROKERVER.....	1950
CCDTURL.....	1951
CCSID.....	1951
Kanal.....	1952
Temizle.....	1952
CLEANUPINT.....	1953
CONNECTIONNAMELIST.....	1953
İSTEMCILERİYÜZLER.....	1954
CLIENTRECONNECTTIMEOUT.....	1955
CLIENTID.....	1955
ÇALIŞTIRMA.....	1955
KARMAŞIK.....	1956
MSG.....	1956
HAYIR.....	1957
CONNTAG.....	1958
AÇIKLAMA.....	1958
YÖN YÖNTEMCİ.....	1959
Kodlama.....	1959
Son kullanma tarihi.....	1960
FAILIFQUIESCE.....	1961
Anasistem adı.....	1961
YEREL ADRES.....	1962
MAPNAMESTILI.....	1962
MAXBUFFSIZE.....	1963
MDREAD.....	1963
MDWRITE.....	1964
MDMSGCTX.....	1964
MSGBATCHSZ.....	1965
MSGONT.....	1965
MSGRECALOG.....	1966
MSGSEÇEK.....	1967
ÇOK YA.....	1967
EN İYİ BİLDİRİM.....	1968
KALDIRIM.....	1968
Kalıcılık.....	1969
GİRİŞ TARİHİ.....	1969
PORT.....	1970
Öncelik.....	1970
SÜREÇ.....	1971
SAĞLAMA SÜ.....	1971
PROXYHOSTNAME.....	1974
TANITIM.....	1974

PUBKACT.....	1975
PUTASINCALLI.....	1975
YÖNETİM.....	1976
kuyruk.....	1976
OKUNABİLENİZİN.....	1977
READAHEADCLOSEPOLICY.....	1977
RECEIVECCSTnt.....	1978
ALINMASI.....	1978
RECEIVEI.....	1979
YENİDEN çıkış.....	1979
YENİDEN KALDIRMA.....	1980
REPLYTOSTYLE.....	1980
SıFIRA.....	1981
SEÇİLEN.....	1981
GÜNCELLEMİYİ.....	1982
SENDCHECKCOUNT.....	1982
SENDEXIT.....	1983
SENDEXITINIT.....	1983
SENABRALD.....	1984
SPARSESUBS.....	1984
SSLCIPHERSUIT.....	1985
SLSCRL.....	1985
SSLFIPSREQUIRE.....	1986
SLAYICI ADI.....	1986
SSLRESTCOUNT.....	1987
DEYİM YENİDEN.....	1987
ALT DÜZEN.....	1988
SYNCPOINTALLGETS.....	1988
HEDEF İSTEMCİ.....	1989
TARGCLIENTEŞLEŞTİRME.....	1989
TEMPMODEL.....	1990
TEMPQPREFIX.....	1990
TEMPTOPICFIX.....	1991
Konu.....	1991
Aktarım.....	1991
JOKER BİÇİMİ.....	1992
ENCODING özelliği.....	1993
JMS nesnelere ilişkin SSL özellikleri.....	1993

Özel notlar.....1995

Programlama arabirimi bilgileri.....	1996
Ticari Markalar.....	1996

Uygulama başvurusu geliştirilmesi

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

- [“MQI uygulamaları başvurusu” sayfa 7](#)
- [“Kullanıcı çıkışları, API çıkışları ve kurulabilir hizmetler başvurusu” sayfa 1473](#)
-  [“IBM i Application Programming Reference \(ILE/RPG\)” sayfa 967](#)
- [“SOAP başvurusu” sayfa 1417](#)
- [“IBM MQ bridge for HTTP için başvuru malzemesi” sayfa 1736](#)
- [“IBM MQ .NET sınıfları ve arabirimleri” sayfa 1771](#)
- [“IBM MQ C++ sınıfları” sayfa 1832](#)
- [IBM MQ classes for Java™ kitaplıkları](#)
- [IBM MQ classes for JMS](#)

İlgili bilgiler

[Uygulamaların geliştirilmesi](#)

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 236](#)
- [“İşlev çağrıları” sayfa 605](#)
- [“Nesnelerin öznitelikleri” sayfa 778](#)
- [“Dönüş kodları” sayfa 847](#)
- [“MQI seçeneklerinin geçerliliğini denetleme kuralları” sayfa 848](#)
- [“Makine kodlamaları” sayfa 872](#)
- [“Rapor seçenekleri ve ileti işaretleri” sayfa 875](#)
- [“Veri dönüştürme” sayfa 879](#)
- [“MQRFH2 öğeleri olarak belirtilen özellikler” sayfa 901](#)
- [“Kod sayfası dönüşümü” sayfa 909](#)

İlgili kavramlar

[“Kullanıcı çıkışları, API çıkışları ve kurulabilir hizmetler başvurusu” sayfa 1473](#)

Kullanıcı çıkışlarınızı, API çıkışlarınızı ve kurulabilir hizmet uygulamalarınızı geliştirmenize yardımcı olmak için bu bölümde sağlanan bağlantıları kullanın.

İlgili başvurular

[“SOAP başvurusu” sayfa 1417](#)

Alfabetik olarak düzenlenmiş SOAP başvuru bilgileri için IBM MQ iletimi.

[“IBM MQ bridge for HTTP için başvuru malzemesi” sayfa 1736](#)

Alfabetik olarak düzenlenmiş, IBM MQ bridge for HTTP için başvuru konuları

[“IBM MQ .NET sınıfları ve arabirimleri” sayfa 1771](#)

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

[“IBM MQ C++ sınıfları” sayfa 1832](#)

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

İlgili bilgiler

Uygulamaların geliştirilmesi

Java kitaplıkları için IBM MQ sınıfları

[../com.ibm.mq.javadoc.doc/WMQJMSClasses/index.html](http://com.ibm.mq.javadoc.doc/WMQJMSClasses/index.html)

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 yöneticiyle bağlantı kuruyor

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 /* Object descriptor */
MQLONG OpenOptions; /* Options control MQOPEN */

/*-----*/
/* Initialize the Object Descriptor (MQOD) */
/* control block. (The remaining fields */
/* are already initialized.) */
/*-----*/
strcpy( ObjDesc.ObjectName,
        SYSTEM_REPLY_MODEL,
        MQ_Q_NAME_LENGTH );
strcpy( 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 */
    /*-----*/
    MQErrorHandling( "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 };
    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))
    {
        sprintf(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. If the 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 \(z/OS\)\)](#)  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 MsgBufLen; /* 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 PutBufLen = 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, '\0', MQ_Q_NAME_LENGTH);
memset(MsgDesc.ReplyToQMGr, '\0', 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;
PutBuffLen = sizeof(PutBuffer);
MQPUT1(Hconn,
        &ObjDesc,
        &MsgDesc,
        &PutMsgOpts,
        PutBuffLen,
        &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 ; /* the MQGET call      */
    :                                     /* 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 MsgBuffLen; /* 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;
MsgBuffLen = 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 = MQRC_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,
      SelectorsTable,
      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 Ayarlanması

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 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 set call:     */
    /*      - Set SelectorsTable to the attributes to be  */
    /*      set                                             */
    /*      - Set IntAttrsTable to the required status    */
    /*      - All other variables are already set         */
    /*                                                    */
    /*                                                    */
    SelectorsTable[0] = MQIA_INHIBIT_GET;
    SelectorsTable[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,
      SelectorsTable,
      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 işleminin nasıl yayınlanmasını ve MQSTAT ile durum bilgilerinin nasıl alınacağını gösterir.

Bu alma, çağıran MQSTAT örnek uygulamasından alınır (program amqsapt0) IBM MQ for Windows sistemleriyle birlikte sağlanır. Diğer platformlardaki örnek uygulamaların adları ve yerleri için bkz. Örnek yordamsal programlar (z/OS dışındaki 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, 2022. 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:                                                */
/*      (1) The name of the target queue                         */
/*      optional:                                                */
/*      (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 yöneticiyle bağlantı kuruyor

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 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 \(z/OS dışındaki platformlar\)](#).

```

:
*
* 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 = MQRC-NOT-AUTHORIZED
*          MOVE M01-MESSAGE-3 TO M00-MESSAGE
*
*      WHEN OTHER
*          MOVE 'MQOPEN'          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 yöneticiyle 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-SYNCPPOINT +
*                                   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-SYNCPPOINT +
                        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 MQOD-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-SYNCPPOINT +
*                                         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.
*
*   CMQ0DV defines the object descriptor (MQOD).
*
01 MQM-OBJECT-DESCRIPTOR.
   COPY CMQ0DV.
*
*   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 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 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 , (0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0),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,CALLLST)
*
*      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
*
*      CALLLST CALL , (0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0),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,CALLLST)
*
*       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  ,(0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0),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 alınmıyor.

```

:
*          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,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
*      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  ,(0,0,0,0,0,0,0,0,0,0,0,0),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  WGM0_AREA,MQGM0_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
           WGM0,                   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  WGM0_AREA,MQGM0_AREA  INITIALIZE WORKING MQGMO  
      L    R5,=AL4(MQGM0_WAIT)  
      A    R5,=AL4(MQGM0_ACCEPT_TRUNCATED_MSG)  
      ST   R5,WGM0_OPTIONS  
      MVC  WGM0_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  
            WGM0,          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.  
*  
      LA   R5,MQRC_NO_MSG_AVAILABLE  IT MUST BE AN ERROR.  
                                          IS IT DUE TO AN EMPTY  
                                          QUEUE?  
      C    R5,REASON  
      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  
                                          MESSAGE?  
      C    R5,REASON  
      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
*
CALLLIST 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)
*
BE GET IS A MESSAGE AVAILABLE?
YES, SO GO AND GET IT
*
CLC SIG_ECB+1(3),=AL3(MQEC_WAIT_INTERVAL_EXPIRED)
*
BE NOMSG HAVE WE WAITED LONG ENOUGH?
YES, SO SAY NO MSG AVAILABLE
B BADCALL IF IT'S ANYTHING ELSE
GO TO ERROR ROUTINE.
*
*   DOWORK
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

```

```

*
COMPCODE DS F
REASON   DS F
BUFFLEN  DS F
DATALEN  DS F
OPTIONS  DS F
HCONN    DS F
HOBJ     DS F
SIG_ECB  DS F

```

```

*
BUFFER    DS CL80
BUFFER_LEN EQU *-BUFFER
*
WMD        CMQMDA DSECT=NO,LIST=NO
WGMO       CMQGMOA DSECT=NO,LIST=NO
*
CALLLIST   CALL , (0,0,0,0,0,0,0,0,0,0,0,0),VL,MF=L
*
:
:
END

```

Bir kuyruğun özniteliklerinin sorulmasına ve ayarlanmasını geciktirme

Bu örnek, bir kuyruğun özniteliklerini sorgulamak ve bir kuyruğun özniteliklerini değiştirmek için MQSET çağrısını kullanmak için MQINQ çağrısının nasıl kullanılacağını gösterir.

This extract is taken from the Queue Attributes sample application (program CSQ4CAC1) supplied with IBM MQ for z/OS.

```

:
DFHEISTG DSECT
:
OBJDESC   CMQODA LIST=YES   Working object descriptor
*
SELECTORCOUNT DS F       Number of selectors
INTATTRCOUNT DS F       Number of integer attributes
CHARATTRLENGTH DS F       char attributes length
CHARATTRS     DS C       Area for char attributes
*
OPTIONS DS F       Command options
HCONN   DS F       Handle of connection
HOBJ    DS F       Handle of object
COMPCODE DS F       Completion code
REASON  DS F       Reason code
SELECTOR DS 2F      Array of selectors
INTATTRS DS 2F      Array of integer attributes
:
OBJECT   DS CL(MQ_Q_NAME_LENGTH) Name of queue
:
CALLLIST CALL , (0,0,0,0,0,0,0,0,0,0,0,0),VL,MF=L
*****
*               PROGRAM EXECUTION STARTS HERE          *
:
CSQ4CAC1 DFHEIENT CODEREG=(R3),DATAREG=(R13)
:
*       Initialize the variables for the set 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 q attribute selector
      ST   R0,SELECTOR+0       Place in field
      LA   R0,MQIA_INHIBIT_PUT Load q attribute selector
      ST   R0,SELECTOR+4       Place in field
*
UPDTEST DS 0H
      CLC ACTION,CINHIB      Are we inhibiting?
      BE   UPDINHBT         Yes branch to section
*
      CLC ACTION,CALLOW     Are we allowing?
      BE   UPDALLOW        Yes branch to section
*

```



```

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 alınmıyor.

```
%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 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);
:

/*****
/* 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 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 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 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);
:
/*****
/* 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 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 BUFLLEN           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 */
*****/
BUFLLEN = 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,
            BUFLLEN,
            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 = 'I';
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 alınmıyor

Bu örnek, bir iletiyi kuyruktan kaldırmak için MQGET 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 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 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 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;
  WHEN (MQCC_FAILED) DO;
  :
  CALL ERROR_ROUTINE;
  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 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 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 sağlar. Bu üstbilgi dosyaları tabloda özetlenir:

Çizelge 1. C üstbilgi dosyaları-çağrı prototipleri, veri tipleri, dönüş kodları, sabitler ve yapılar					
Dosya adı	Tanım	IBM i	UNIX ve Linux® sistemleri	Windows	z/OS
Çağrı prototipleri, veri tipleri, dönüş kodları, sabitler ve yapılar					
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	
Anahtar: C= Dosyalar sağlandı					

COBOL COPY dosyaları

Çeşitli COPY dosyaları, MQI kullanan COBOL uygulama programlarını yazmanıza yardımcı olmak için sağlanmıştır. Bu dosyalar tabloda özetlenmiştir:

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

Dosya adı	Tanım	IBM i	UNIX sistemleri	Windows	z/OS
Dönüş kodları ve değişmezleri					
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
Yapılar					
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
CMQCHRVX	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	UNIX sistemleri	Windows	z/OS
CMQDXPx	MQDXP-Veri dönüştürme çıkış değıştirgeleri	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-SSL configuratooon seçenekleri		V L	V L	
CMQTMx	MQTM-Tetikleme iletisi	V L		V L	V L
CMQTMcx	MQTMc-Tetikleyici iletisi karakteri	V L	V L		
CMQTMc2x	MQTMc2 -Tetikleyici iletisi 2 karakter	V L	V L	V L	V L
CMQWIHx	MQWIH-İş bilgisi üstbilgisi	V L	V L	V L	V L
CMQXQHx	MQXQH-İletim kuyruğu üstbilgisi	V L	V L	V L	V L
Anahtar: - Sağlanan başlangıç değğerlerine sahip dosyalar, x = V - Başlangıç değğerleri sağlanmadan dosyalar, x = L					

PL/I içğğme dosyaları

Aşağıdaki INCLUDE dosyaları PL/I programlama dili için sağlanır. Bu dosyalar yalnızca z/OS üzerinde kullanılabilir.

Çizelge 3. PL/I, kütük-veri tiplerini, dönüş kodlarını, değışmezleri ve yapıları içğğir

Dosya adı	Tanım	IBM i	UNIX sistemleri	Windows	z/OS
Veri tipleri, dönüş kodları, sabitler ve yapılar					

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

Dosya adı	Tanım	IBM i	UNIX sistemleri	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					

RPG kopyalama dosyaları

RPG programlama dili için aşağıdaki KOPY kütükleri sağlanmıştır. Bu dosyalar yalnızca IBM üzerinde kullanılabilir.

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

Dosya adı	Tanım	IBM i	UNIX sistemleri	Windows	z/OS
Dönüş kodları ve değişmezleri					
CMQX	MQI tanımlamaları	G R			
CMQCFX	PCF tanımlamaları	G			
CMQPSI	Yayınlama/Abone Olma tanımları	G			
CMQXX	Kanal ve çıkış tanımları	G R			
Yapılar					
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	UNIX sistemleri	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	MQCSP-Güvenlik değiştirgeleri	G H			
CMQCXPx	MQCXP-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			
CMQPXPX	MQPXP-Yayınlama/Abone Olma yöneltme 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			
CMQTMc2x	MQTMc2 -Tetikleyici iletisi 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	UNIX sistemleri	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					

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 ve aşağıdaki çizelgede özetlenir:

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

Dosya adı	Tanım	IBM i	UNIX ve Linux sistemleri	Windows	z/OS
Çağrı bildirimleri, veri tipleri, dönüş kodları, sabitler ve yapılar					
CMQCB	MQI tanımlamaları			B	
CMQBB	MQAI tanımlamaları			B	
CMQCFB	PCF tanımlamaları			B	
CMQXB	Kanal ve çıkış tanımları			B	
Anahtar: B= Dosya sağlandı					

z/OS Assembler COPY dosyaları

Çeşitli COPY dosyaları, MQI kullanan z/OS Assembler uygulama programlarını yazmanıza yardımcı olmak için sağlanmıştır. Bu dosyalar tabloda özetlenmiştir:

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

Dosya adı	Tanım	IBM i	UNIX sistemleri	Windows	z/OS
Veri tipleri, dönüş kodları ve sabitler					
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
Yapılar					
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	UNIX sistemleri	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	UNIX sistemleri	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
CMQTMCM2A	MQTMC2 -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
CMQWXPMA	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
CMQXMA	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)

Ad	Ondalık değ�r	Onaltılı değ�r
MQ_ABEND_CODE_LENGTH	4	X'00000004'
MQ_ACCOUNTING_TOKEN_LENGTH	32	X'00000020'
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_UZUNLUĐU	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_CICS_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'

Ad	Ondalık değ�r	Onaltılı değ�r
MQ_DATE_LENGTH	12	X'0000000C'
MQ_DISTINGUISH_NAME_LENGTH	1024	X'00000400'
MQ_DNS_GROUP_NAME_LENGTH	18	X'00000012'
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'

Ad	Ondalık değ�r	Onaltılı değ�r
MQ_NAMELIST_DESC_LENGTH	64	X'00000040'
MQ_NAMELIST_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'
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_CRYPT0_HARDWARE_LENGTH	256	X'00000100'
MQ_SSL_HANDSHAKKE_STAGE_LENGTH	32	X'00000020'

Ad	Ondalık değ�r	Onaltılı değ�r
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_LEGTH	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'
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_LEGTH	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)

Ad	Ondalık değer	Onaltılı değer
MQ_ARCHIVE_PFX_LENNGTH	36	X'00000024'
MQ_ARCHIVE_UNIT_LENGTH	8	X'00000008'
MQ_ASID_LENCH	4	X'00000004'
MQ_AUTH_PROFILE_NAME_LENGTH	48	X'00000030'
MQ_CF_LEID_LENNGTH	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_LENCH	8	X'00000008'
V8.0.0.2 V8.0.0.2 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_LENCH	24	X'00000018'
MQ_LOG_PATH_LENGTH	1024	X'00000400'
MQ_LRSN_LENGTH	12	X'0000000C'
MQ_ORIGIN_NAME_LENGTH	8	X'00000008'
MQ_PSB_NAME_LENCH	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_LEGTH	4	X'00000004'
MQ_UOW_ID_LENGTH	256	X'00000100'
MQ_USER_DATA_LENGTH	10240	X'00002800'
MQ_VOLSER ' i NGUTH	6	X'00000006'

MQACH_* (API çıkış zinciri alanı üstbilgi yapısı)

Ad	Yapı
MQACH_STRUCT_ID	"ACH"
MQACH_STRUC_ID_ARRAY	'A','C','H',' '

Not: " " simgesi tek bir boş karakteri temsil eder.

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)	
MQACH_CURRENT_LENGTH	(value differs by platform or version)	

MQACT_* (Hesap Belirteci)

Ad	Değer
MQACT_NONE	X'00...00' (32 boş değer)
MQACT_NONE_DIZISI	'\0','\0',... (32 boş değer)

MQACT_* (Komut biçimi İşlem Seçenekleri)

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)

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)

Ad	Onaltılı değer
MQACTT_UNKNOWN	X'00'
MQACTT_CICS_LUOW_ID	X'01'

Ad	Onaltılı değer
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

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

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)

Ad	Yapı
MQAIR_STRUCT_ID	"AIR↵"
MQAIR_STRUC_ID_ARRAY	'A','I','R','↵'

Not: ↵ simgesi tek bir boş karakteri temsil eder.

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)

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)

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'
MQAS_DEVREDİŞİ	7	X'00000007'

MQAT_ * (Koyma Uygulaması Tipleri)

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'

Ad	Ondalık değ�r	Onaltılı değ�r
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_K��PR��S��	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'
MQAT_BATCH	32	X'00000020'
MQAT_RRS_BATCH	33	X'00000021'
MQAT_SIB	34	X'00000022'
MQAT_DEFAULT	(value differs by platform or version)	
MQAT_USER_FIRST	65536	X'00010000'
MQAT_USER_SON	999999999	X'3B9AC9FF'

MQAUTH_ * (Komut bi imi Yetki Değ rleri)

Ad	Ondalık değ�r	Onaltılı değ�r
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'

Ad	Ondalık değ�r	Onaltılı değ�r
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_�ZGE�MI�	21	X'00000015'
MQAUTH_SYSTEM	22	X'00000016'

MQAUTHOPT_ * (Komut bi imi Yetki Se enekleri)

Ad	Ondalık değ�r	Onaltılı değ�r
MQAUTHOPT_K�M�LATIF	256	X'00000100'
MQAUTHOPT_ENTITY_EXPLICIT	1	X'00000001'
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  ıkı  ba lamı yapısı)

Ad	Yapı
MQAXC_STRUCT_ID	"AXC�"
MQAXC_STRUC_ID_ARRAY	'A', 'X', 'C', '�'

Not:   simgesi tek bir bo  karakteri temsil eder.

Ad	Ondalık değ�r	Onaltılı değ�r
MQAXC_VERSION_1	1	X'00000001'
MQAXC_CURRENT_VERSION	1	X'00000001'

MQAXP_ * (API çıkış değıştirgesi yapısı)

Ad	Yapı
MQAXP_STRUCT_ID	"AXP¬"
MQAXP_STRUC_ID_ARRAY	'A', 'X', 'P', '¬'

Not: ¬ simgesi tek bir boş karakteri temsil eder.

Ad	Ondalık değęer	Onaltılı değęer
MQAXP_VERSION_1	1	X'00000001'
MQAXP_VERSION_2	2	X'00000002'
MQAXP_CURRENT_VERSION	2	X'00000002'

MQBA_ * (Bayt Öznitelik Seçicileri)

Ad	Ondalık değęer	Onaltılı değęer
MQBA_FIRST	6001	X'00001771'
MQBA_SON	8000	X'00001F40'

MQBACF_ * (Komut biçimi Byte Değıştirgesi Tipleri)

Ad	Ondalık değęer	Onaltılı değęer
MQBACF_FIRST	7001	X'00001B59'
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)

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

MQBMHO_* (Arabellekten ileti işleme seçenekleri ve yapısı)

Arabellek-ileti tanıtıcısı seçenekleri yapısı

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

Not: ¬ simgesi tek bir boş karakteri temsil eder.

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

Arabellek-İleti Tanıtıcısı Seçenekleri

Ad	Ondalık değer	Onaltılı değer
MQBMHO_YOK	0	X'00000000'
MQBMHO_DELETE_PROPERTIES	1	X'00000001'

MQBND_* (Varsayılan Bağ Tanımları)

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

Ad	Yapı
MQBO_STRUC_ID	"BO¬¬"
MQBO_STRUC_ID_ARRAY	'B','O','¬','¬'

Not: ¬ simgesi tek bir boş karakteri temsil eder.

Ad	Ondalık değer	Onaltılı değer
MQBO_VERSION_1	1	X'00000001'
MQBO_CURRENT_VERSION	1	X'00000001'

Başlangıç Seçenekleri

Ad	Ondalık değer	Onaltılı değer
MQBO_NONE	0	X'00000000'

MQBT_ * (Komut biçimi Köprü Tipleri)

Ad	Ondalık değer	Onaltılı değer
MQBT_OTMA	1	X'00000001'

MQCA_ * (Karakter Özniteliği Seçicileri)

Ad	Ondalık değer	Onaltılı değer
MQCA_ADMIN_TOPIC_NAME	2105	X'00000839'
MQCA_ALTERATION_DATE	2027	X'000007EB'
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'

Ad	Ondalık değ�r	Onaltılı değ�r
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'
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'

Ad	Ondalık değ�r	Onaltılı değ�r
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_CRYPT0_HARDWARE	2051	X'00000803'
MQCA_SSL_KEY_LIBRARY	2069	X'00000815'
MQCA_SSL_KEY_MEMBER	2070	X'00000816'
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'

Ad	Ondalık değer	Onaltılı değer
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)

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'
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'

Ad	Ondalık değ�r	Onaltılı değ�r
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'
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'

Ad	Ondalık değ�r	Onaltılı değ�r
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'
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'

Ad	Ondalık değ�r	Onaltılı değ�r
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_�YESİ	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_ADİ	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'
MQCACF_ETKİNY_DESC	3134	X'00000C3E'
MQCACF_APPL_IDENTITY_DATA	3135	X'00000C3F'

Ad	Ondalık değ�r	Onaltılı değ�r
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_ADİ	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 ADİ	3163	X'00000C5B'
MQCACF_TO_ALT ADİ	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_LAST_KULLANİLDİ	3172	X'00000C64'

MQCACH_* (Komut biçimi Karakter Kanalı Parametresi Tipleri)

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_ADİ	3504	X'00000DB0'
MQCACH_XMIT_Q_ADİ	3505	X'00000DB1'
MQCACH_CONNECTION_NAME	3506	X'00000DB2'
MQCACH_MCA_ADİ	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_PAROLASI	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'

Ad	Ondalık değ�r	Onaltılı değ�r
MQCACH_SSL_HANDSHARE_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'
MQCACH_SSL_CERT_ISSUER_NAME	3550	X'00000DDE'
MQCACH_LU_NAME	3551	X'00000DDF'
MQCACH_IP_ADDRESS	3552	X'00000DE0'
MQCACH_TCP_ADı	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  stbilgisi ADS a ıklayıcıları)

Ad	Ondalık değ�r	Onaltılı değ�r
MQCADSD_NONE	0	X'00000000'
MQCADSD_SEND	1	X'00000001'
MQCADSD_RECV	16	X'00000010'
MQCADSD_MSGSAYı	256	X'00000100'

MQCAFTY_* (Baęlantı İlişkisi Deęerleri)

Ad	Ondalık değ�r	Onaltılı değ�r
MQCAFTY_NONE	0	X'00000000'
MQCAFTY_PREFERDI	1	X'00000001'

MQCAMO_* (Komut bięimi Karakter İzleme Parametresi Tipleri)

Ad	Ondalık değ�r	Onaltılı değ�r
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'

Ad	Ondalık değ�r	Onaltılı değ�r
M�CAMO_DISC_TIME	2706	X'00000A92'
M�CAMO_END_DATE	2707	X'00000A93'
M�CAMO_END_TIME	2708	X'00000A94'
M�CAMO_OPEN_DATE	2709	X'00000A95'
M�CAMO_OPEN_TIME	2710	X'00000A96'
M�CAMO_START_DATE	2711	X'00000A97'
M�CAMO_START_TIME	2712	X'00000A98'
M�CAMO_LAST_SED	2712	X'00000A98'

M CBC_ * (M CBC deėiřmezleri yapısı)

Ad	Yapı
M�CBC_STRUC_ID	"CBC�"
M�CBC_STRUC_ID_ARRAY	'C','B','C','�'

Not:   simgesi tek bir boř karakteri temsil eder.

Ad	Ondalık değ�r	Onaltılı değ�r
M�CBC_VERSION_1	1	X'00000001'
M�CBC_CURRENT_VERSION	1	X'00000001'

M CBCF_ * (M CBC deėiřmezleri iřaretleri)

Ad	Ondalık değ�r	Onaltılı değ�r
M�CBCF_NONE	0	X'00000000'
M�CBCF_READA_BUFFER_BOř	1	X'00000001'

M CBCT_ * (M CBC deėiřmezleri Callback tipi)

Ad	Ondalık değ�r	Onaltılı değ�r
M�CBCT_START_CALL	1	X'00000001'
M�CBCT_STOP_CALL	2	X'00000002'
M�CBCT_REGISTER_CALL	3	X'00000003'
M�CBCT_DEREGISTIER_CALL	4	X'00000004'
M�CBCT_EVENT_CALL	5	X'00000005'
M�CBCT_MSG_REMOVED	6	X'00000006'
M�CBCT_MSG_NOT_REMOVED	7	X'00000007'

M CBD_ * (M CBD deėiřmezleri yapısı)

Ad	Yapı
M�CBD_STRUC_ID	"CBD�"

Ad	Yapı
MQCBDD_STRUC_ID_ARRAY	'C' , 'B' , 'D' , '↵'

Not: ↵ simgesi tek bir boş karakteri temsil eder.

Ad	Ondalık değer	Onaltılı değer
MQCBDD_VERSION_1	1	X'00000001'
MQCBDD_CURRENT_VERSION	1	X'00000001'

MQCBDO_* (MQCBDD değişmezleri Geri Çağırma Seçenekleri)

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_DEREGISTER_CALL	512	X'00000200'
MQCBDO_FAIL_IF_QUIESCING	8192	X'00002000'

MQCBO_* (mqCreateBag için Oluştur-Bag Seçenekleri)

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_* (MQCBDD değişmezleri bu, Callback İşlevinin tipidir)

Ad	Ondalık değer	Onaltılı değer
MQCBT_MESSAGE_CONSUMER	1	X'00000001'
MQCBT_EVENT_HANDLER	2	X'00000002'

MQCC_ * (Tamamlanma Kodları)

Ad	Ondalık değer	Onaltılı değer
MQCC_OK	0	X'00000000'
MQCC_UYARI	1	X'00000001'
MQCC_FAILED	2	X'00000002'
MQCC_BILINMIYOR	-1	X'FFFFFFFF'

MQCCSI_ * (Kodlanmış Karakter Kümesi Tanıtıcıları)

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'FFFFFFFFE'
MQCCSI_EMBEDDED	-1	X'FFFFFFFF'
MQCCSI_APPL	-3	X'FFFFFFFD'

MQCCT_ * (CICS bilgi üstbilgisi Conversasyonel Görev Seçenekleri)

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

MQCD_ * (Kanal tanımlama yapısı)

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_XX_ENCODE_CASE_ONE current_version	11	X'0000000B'
MQCD_LENGTH_4	(value differs by platform or version)	

Ad	Ondalık değ�r	Onaltılı değ�r
MQCD_LENGTH_5	(value differs by platform or version)	
MQCD_LENGTH_6	(value differs by platform or version)	
MQCD_LENGTH_7	(value differs by platform or version)	
MQCD_LENGTH_8	(value differs by platform or version)	
MQCD_LENGTH_9	(value differs by platform or version)	
MQCD_LENGTH_10	(value differs by platform or version)	
MQCD_LENGTH_11	(value differs by platform or version)	
MQCD_CURRENT_LENGTH	(value differs by platform or version)	

MQCDC_ * (Kanal Veri D n          )

Ad	Ondalık değ�r	Onaltılı değ�r
MQCDC_SENDER_CONVERSION	1	X'00000001'
MQCDC_NO_SENDER_CONVERSION	0	X'00000000'

MQCERT_ * (Sertifika Ge  erlilik 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)

Ad	Ondalık değ�r	Onaltılı değ�r
MQCF_NONE	0	X'00000000'
MQCF_DIST_LISTS	1	X'00000001'

MQCFAC_* (CICS bilgi üstbilgisi Olanağı)

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 değiştirgesi yapısı)

Ad	Ondalık değer	Onaltılı değer
MQCFBF_STRUC_LENGTH_FIXED	20	X'00000014'

MQCFBS_* (Komut biçimi byte dizgisi değiştirgesi yapısı)

Ad	Ondalık değer	Onaltılı değer
MQCFBS_STRUC_LENGTH_FIXED	16	X'00000010'

MQCFC_* (Komut biçimi üstbilgi denetim seçenekleri)

Ad	Ondalık değer	Onaltılı değer
MQCFC_SON	1	X'00000001'
MQCFC_NOT_SON	0	X'00000000'

MQCFGR_* (Komut biçimi grup değiştirgesi yapısı)

Ad	Ondalık değer	Onaltılı değer
MQCFGR_STRUC_LENGTH	16	X'00000010'

MQCFH_* (Komut biçimi üstbilgi yapısı)

Ad	Ondalık değer	Onaltılı değer
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 değiştirgesi yapısı)

Ad	Ondalık değer	Onaltılı değer
MQCFIFI_STRUCT_LENGTH	20	X'00000014'

MQCFIL_* (Komut biçimi tamsayı listesi değiştirge yapısı)

Ad	Ondalık değer	Onaltılı değer
MQCFIL_STRUCT_LENGTH_FIXED	16	X'00000010'

MQCFIL64_* (Komut biçimi 64-bit tamsayı listesi değiştirge yapısı)

Ad	Ondalık değer	Onaltılı değer
MQCFIL64_STRUC_LENGTH_FIXED	16	X'00000010'

MQCFIN_* (Komut biçimi tamsayı değiştirgesi yapısı)

Ad	Ondalık değer	Onaltılı değer
MQCFIN_STRUCT_LENGTH	16	X'00000010'

MQCFIN64_* (Komut biçimi 64-bit tamsayı değiştirgesi yapısı)

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

Ad	Ondalık değer	Onaltılı değer
MQCFO_REFRESH_REPOSITORY_YES	1	X'00000001'
MQCFO_REFRESH_REPOSITORY_NO	0	X'00000000'

Komut Biçimi Kaldırma Kuyrukları Seçenekleri

Ad	Ondalık değer	Onaltılı değer
MQCFO_REMOVE_QUEUES_YES	1	X'00000001'
MQCFO_REMOVE_QUEUES_NO	0	X'00000000'

MQCFOP_* (Komut biçimi Süzgeç İşleçleri)

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'
MQCFOP_EXCLUDES	13	X'0000000D'
MQCFOP_CONTAINS_GEN	26	X'0000001A'

Ad	Ondalık değ�r	Onaltılı değ�r
M�CFOP_EXCLUDES_GEN	29	X'0000001D'

M CFR_* (CF Kurtarılabilirliđi)

Ad	Ondalık değ�r	Onaltılı değ�r
M�CFR_YES	1	X'00000001'
M�CFR_NO	0	X'00000000'

M CFSF_* (Komut biđimi dizgi s zgeci deđiřtirge yapısı)

Ad	Ondalık değ�r	Onaltılı değ�r
M�CFSF_STRUC_LENGTH_FIXED	24	X'00000018'

M CFSL_* (Komut biđimi dizgi listesi deđiřtirge yapısı)

Ad	Ondalık değ�r	Onaltılı değ�r
M�CFSL_STRUC_LENGTH_FIXED	24	X'00000018'

M CFST_* (Komut biđimi dizgi deđiřtirgesi yapısı)

Ad	Ondalık değ�r	Onaltılı değ�r
M�CFST_STRUC_LENGTH_FIXED	20	X'00000014'

M CFSTATUS_* (Komut biđimi CF Durumu)

Ad	Ondalık değ�r	Onaltılı değ�r
M�CFSTATUS_NOT_FOUND	0	X'00000000'
M�CFSTATUS_ETK�N	1	X'00000001'
M�CFSTATUS_IN_RECOVER	2	X'00000002'
M�CFSTATUS_IN_BACKUP	3	X'00000003'
M�CFSTATUS_FAILED	4	X'00000004'
M�CFSTATUS_NONE	5	X'00000005'
M�CFSTATUS_UNKNOWN	6	X'00000006'
M�CFSTATUS_ADMIN_TAMAMLANMAMIř	20	X'00000014'
M�CFSTATUS_NEVER_KULLANILDI	21	X'00000015'
M�CFSTATUS_NO_BACKUP	22	X'00000016'
M�CFSTATUS_NOT_FAILED	23	X'00000017'
M�CFSTATUS_NOT_RECOVERABLE	24	X'00000018'
M�CFSTATUS_XES_ERROR	25	X'00000019'

MQCFT_* (Structure Structure Of Structure)

Ad	Ondalık değ�r	Onaltılı değ�r
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'

MQCFTYPE_* (Komut biçimi CF tipleri)

Ad	Ondalık değ�r	Onaltılı değ�r
MQCFTYPE_APPL	0	X'00000000'
MQCFTYPE_ADMIN	1	X'00000001'

MQCFUNC_* (CICS bilgi  stbilgisi i levleri)

Ad	Yapı
MQCFUNC_MQCONN	"CONN"
MQCFUNC_MQGET	"GET~"
MQCFUNC_MQINQ	"INQ~"

Ad	Yapı
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 Al)

Ad	Ondalık değer	Onaltılı değer
MQCGWI_VARSAYILAN	-2	X'FFFFFFFFE'

MQCHAD_ * (Kanal Otomatik Tanımlaması)

Ad	Ondalık değer	Onaltılı değer
MQCHAD_ENGELLI	0	X'00000000'
MQCHAD_ENABLED	1	X'00000001'

MQCHNiti_ * (Komut biçimi Kuşku Durumu Durumu)

Ad	Ondalık değer	Onaltılı değer
MQCHIDS_NOT_INDOUBT	0	X'00000000'
MQCHGINDOUBT	1	X'00000001'

MQCHLD_ * (Komut biçimi Kanal Atımları)

Ad	Ondalık değer	Onaltılı değer
MQCHLD_ALL	-1	X'FFFFFFFF'
MQCHLD_VARSAYILAN	1	X'00000001'
MQCHLD_SHARED	2	X'00000002'
MQCHLD_PRIVATE	4	X'00000004'
MQCHLD_FIXSHARED	5	X'00000005'

MQCHS_* (Komut biçimi Kanal Durumu)

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)

Ad	Ondalık değer	Onaltılı değer
MQCHSH_RESTART_NO	0	X'00000000'
MQCHSH_RESTART_YES	1	X'00000001'

MQCHSR_* (Komut biçimi Kanal Durdurma Seçenekleri)

Ad	Ondalık değer	Onaltılı değer
MQCHSR_STOP_NOT_REQUESTED	0	X'00000000'
MQCHSR_STOP_REQUESTED	1	X'00000001'

MQCHSSTATE_* (Komut biçimi Kanal Alt Devletleri)

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'
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'

Ad	Ondalık değ�r	Onaltılı değ�r
MQCHSSTATE_IN_MREXIT	1100	X'0000044C'
MQCHSSTATE_IN_CHADEXIT	1200	X'000004B0'
MQCHSSTATE_NET_BAĐLANTI	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)

Ad	Ondalık değ�r	Onaltılı değ�r
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 biĐimi Kanal  izelgesi Tipleri)

Ad	Ondalık değ�r	Onaltılı değ�r
MQCHTAB_Q_MGR	1	X'00000001'
MQCHTAB_CLNTCONN	2	X'00000002'

MQCI_* (İlinti Tanıtıcısı)

Ad	DeĐer
MQCI_NONE	X'00...00' (24 boş değ�r)
MQCI_NONE_DIZISI	'\0', '\0', ... (24 boş değ�r)
MQCI_NEW_SESSION	X'414D5121...'
MQCI_NEW_SESSION_ARRAY	'\x41', '\x4D', '\51', '\x21', ...

MQCIH_ * (CICS bilgi üstbilgisi yapısı ve işaretleri)

CICS bilgi üstbilgisi yapısı

Ad	Yapı
MQCIH_STRUC_ID	"CIH¬"
MQCIH_STRUC_ID_ARRAY	'C','I','H','¬'

Not: ¬ simgesi tek bir boş karakteri temsil eder.

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

Ad	Ondalık değer	Onaltılı değer
MQCIH_NONE	0	X'00000000'
MQCIH_PASS_EXPECTATION	1	X'00000001'
MQCIH_UNlimited_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)

Ad	Ondalık değer	Onaltılı değer
MQCLCT_STATIC	0	X'00000000'
MQCLCT_DYNAMIC	1	X'00000001'

MQCLRS_ * (Komut biçimi Konu Dizgisi Kapsamını Temizle)

Ad	Ondalık değer	Onaltılı değer
MQCLRS_LOCAL	1	X'00000001'
MQCLRS_GLOBAL	2	X'00000002'

MQCLRT_ * (Komut biçimi Konu Dizgisi Tipini Temizle)

Ad	Ondalık değer	Onaltılı değer
MQCLRT_ALIKONUR	1	X'00000001'

MQCLT_ * (CICS bilgi üstbilgisi Bağlantı Tipleri)

Ad	Ondalık değer	Onaltılı değer
MQCLT_PROGRAM	1	X'00000001'
MQCLT_TRANSACTION	2	X'00000002'

MQCLWL_ * (Küme İş Yüğü)

Ad	Ondalık değer	Onaltılı değer
MQCLWL_USEQ_LOCAL	0	X'00000000'
MQCLWL_USEQ_ANY	1	X'00000001'
MQCLWL_USEQ_AS_Q_MGR	-3	X'FFFFFFFFD'

MQCLXQ_ * (Küme iletim kuyruğu tipi)

MQCLXQ_*, DEFCLXQ kuyruk yöneticisi öznitelide ayarlayabileceğiniz değerlerdir. DEFCLXQ özniteliği, ileti almak için varsayılan olarak küme gönderen kanalları tarafından hangi iletim kuyruğunun seçildiğini, iletileri küme alıcılı kanallara göndermesini denetler.

Ad	Ondalık değer	Onaltılı değer
MQCLXQ_SCTQ	0	X'00000000'
MQCLXQ_CHANNEL	1	X'00000001'

İlgili başvurular

[“DefClusterXmitQueueTipi \(MQUBE\)” sayfa 793](#)

DefClusterXmitQueueType özniteliği, ileti almak için varsayılan olarak küme gönderen kanalları tarafından hangi iletim kuyruğunun seçildiğini, iletileri küme alıcılı kanallara göndermesini sağlar.

[“MQINQ-Nesne özniteliklerinin sorgulması” sayfa 686](#)

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

İlgili bilgiler

[Kuyruk Yöneticisini Değiştir](#)

[Sorgu Kuyruğu Yöneticisi](#)

[Sorgu Kuyruğu Yöneticisi \(Yanıt\)](#)

MQCMD_ * (Komut Kodları)

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'

Ad	Ondalık değ�r	Onaltılı değ�r
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'
MQCMD_PING_Q_MGR	40	X'00000028'
MQCMD_INQUIRE_Q_STATUS	41	X'00000029'

Ad	Ondalık değ�r	Onaltılı değ�r
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'

Ad	Ondalık değ�r	Onaltılı değ�r
MQCMD_CHANGE_LISTENER	93	X'0000005D'
MQCMD_COPY_DINLEYICISI	94	X'0000005E'
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_STRUCTION_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'

Ad	Ondalık değ�r	Onaltılı değ�r
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'
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'

Ad	Ondalık değ�r	Onaltılı değ�r
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'
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 bi imi Komut Bilgileri Değ rleri)

Ad	Ondalık değ�r	Onaltılı değ�r
MQCMDI_CMDSCOPE_ACCEPT	1	X'00000001'
MQCMDI_CMDSCOPE_GENERAT�R	2	X'00000002'
MQCMDI_CMDSCOPE_COMPLETED	3	X'00000003'
MQCMDI_QSG_DISPD_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'

Ad	Ondalık değ�r	Onaltılı değ�r
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 D zeyleri)

Ad	Değ�r
MQCMDL_LEVEL_600	600
MQCMDL_LEVEL_700	700
MQCMDL_LEVEL_701	701
MQCMDL_LEVEL_710	710
MQCMDL_LEVEL_711	711
MQCMDL_LEVEL_750	750
MQCMDL_LEVEL_800	800
> V8.0.0.2 > V8.0.0.2 MQCMDL_LEVEL_801	801
> V8.0.0.3 > V8.0.0.3 MQCMDL_LEVEL_802	802

Notlar:

1. MQCMDL_LEVEL_801 applies only to UNIX platformları and IBM i, when you install IBM MQ 8.0.0, D zeltme Paketi 2.
2. **> V8.0.0.3** MQCMDL_LEVEL_802 applies only to Windows, UNIX platformları, and IBM i, when you install IBM MQ S r m 8.0.0, D zeltme Paketi 3

MQCMHO_* (İleti i leme se enekleri ve yapısı yarat)

İleti tanıtıcısı se enekleri yapısını yarat

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

Not: ↵ simgesi tek bir bo  karakteri temsil eder.

Ad	Ondalık değ�r	Onaltılı değ�r
MQCMHO_VERSION_1	1	X'00000001'
MQCMHO_CURRENT_VERSION	1	X'00000001'

İleti Tanıtıcısı Yaratma Seçenekleri

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

Ad	Yapı
MQCNO_STRUC_ID	"CNO↯"
MQCNO_STRUC_ID_ARRAY	'C','N','O','↯'

Not: ↯ simgesi tek bir boş karakteri temsil eder.

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

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'

Ad	Ondalık değ�r	Onaltılı değ�r
M�CNO_ACCOUNTING_M�I_ENABLED	4096	X'00001000'
M�CNO_ACCOUNTING_M�I_DISABLLANT	8192	X'00002000'
M�CNO_ACCOUNTING_Q_ENABLED	16384	X'00004000'
M�CNO_ACCOUNTING_Q_DISABET	32768	X'00008000'
M�CNO_NO_CONV_SHARING	65536	X'00010000'
M�CNO_ALL_CONVS_SHARE	262144	X'00040000'
M�CNO_CD_FOR_OUTPUT_ONLY	524288	X'00080000'
M�CNO_USE_CD_SE�IMI	1048576	X'00100000'
M�CNO_RECONNECT	16777216	X'01000000'
M�CNO_RECONNECT_AS_DEF	0	X'00000000'
M�CNO_RECONNECT_DISABLE	33554432	X'02000000'
M�CNO_RECONNECT_Q_MGR	67108864	X'04000000'
M�CNO_ACTIVITY_TRACE_ENABLED	134217728	X'08000000'
M�CNO_ACTIVITY_TRACE_DISABLDI	268435456	X'10000000'
M�CNO_NONE	0	X'00000000'

M CO_ * (Kapatma Se enekleri)

Ad	Ondalık değ�r	Onaltılı değ�r
M�CO_IMMEDIATE	0	X'00000000'
M�CO_NONE	0	X'00000000'
M�CO_DELETE	1	X'00000001'
M�CO_DELETE_PURGE	2	X'00000002'
M�CO_KEEP_SUB	4	X'00000004'
M�CO_REMOVE_SUB	8	X'00000008'
M�CO_QUIESCE	32	X'00000020'

M CIDL_ * (CICS bilgi  stbilgisi  ıkıř Veri Uzunluęu)

Ad	Ondalık değ�r	Onaltılı değ�r
M�CIDL_AS_INPUT	-1	X'FFFFFFFF'

M COMPRESS_ * (Kanal Sıkıřtırma)

Ad	Ondalık değ�r	Onaltılı değ�r
M�COMPRESS_NOT_VAR	-1	X'FFFFFFFF'
M�COMPRESS_NONE	0	X'00000000'
M�COMPRESS_RLE	1	X'00000001'
M�COMPRESS_ZLIBFAST	2	X'00000002'

Ad	Ondalık değ�r	Onaltılı değ�r
MQCOMPRESS_ZLIBHIGH	4	X'00000004'
MQCOMPRESS_SYSTEM	8	X'00000008'
MQCOMPRESS_ANY	268435455	X'0FFFFFFF'

MQCONNID_ * (Bağlantı Tanıtıcısı)

Ad	Değ�r
MQCONNID_NONE	X'00...00' (24 boş değ�r)
MQCONNID_NONE_DIZISI	'\0', '\0', ... (24 boş değ�r)

MQCOPY_ * (Özellik Kopyası Seçenekleri)

Ad	Ondalık değ�r	Onaltılı değ�r
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_ * (K me Kuyruęu Tipleri)

Ad	Ondalık değ�r	Onaltılı değ�r
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  stbilgisi D n ş Kodları)

Ad	Ondalık değ�r	Onaltılı değ�r
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'

Ad	Ondalık değeri	Onaltılı değeri
MQCRC_BRIDGE_TIMEOUT	8	X'00000008'
MQCRC_TRANSID_NOT_VAR	9	X'00000009'

MQCS_ * (MQCBC değışmezleri tüketici durumu)

Ad	Ondalık değeri	Onaltılı değeri
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)

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 değıştirgeleri yapısı ve kimlik denetimi tipleri)

Bağlantı güvenliği parametreleri yapısı

Ad	Yapı
MQCSP_STRUC_ID	"CSP¬"
MQCSP_STRUC_ID_ARRAY	'C','S','P','¬'

Not: ¬ simgesi tek bir boş karakteri temsil eder.

Ad	Ondalık değeri	Onaltılı değeri
MQCSP_VERSION_1	1	X'00000001'
MQCSP_XX_ENCODE_CASE_ONE current_version	1	X'00000001'

Bağlantı güvenliği parametreleri Kimlik Doğrulama Tipleri

Ad	Ondalık değer	Onaltılı değer
MQCSP_AUTH_NONE	0	X'00000000'
MQCSP_AUTH_USER_ID_AND_PWD	1	X'00000001'

MQCSRV_* (Komut Sunucusu Seçenekleri)

Ad	Ondalık değer	Onaltılı değer
MQCSRV_CONVERT_NO	0	X'00000000'
MQCSRV_CONVERT_YES	1	X'00000001'
MQCSRV_DLQ_NO	0	X'00000000'
MQCSRV_DLQ_YES	1	X'00000001'

MQCT_* (Kuyruk Yöneticisi Bağlantı Etiketi)

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)

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

Ad	Yapı
MQCTLO_STRUC_ID	"CTLO"
MQCTLO_STRUC_ID_ARRAY	'C', 'T', 'L', 'O'

Not: ↯ simgesi tek bir boş karakteri temsil eder.

Ad	Ondalık değer	Onaltılı değer
MQCTLO_VERSION_1	1	X'00000001'
MQCTLO_CURRENT_VERSION	1	X'00000001'

MQCTL Seçenekleri Tüketici Denetimi Seçenekleri

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)

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'
MQCUOWC_SON	272	X'00000110'
MQCUOWC_COMMIT	256	X'00000100'
MQCUOWC_BACKOUT	4352	X'00001100'

MQCXP_ * (Kanal çıkış değiştirgesi yapısı)

Ad	Yapı
MQCXP_STRUC_ID	"CXP¬"
MQCXP_STRUC_ID_ARRAY	'C','X','P','¬'

Not: ¬ simgesi tek bir boş karakteri temsil eder.

Ad	Ondalık değer	Onaltılı değer
MQCXP_VERSION_1	1	X'00000001'
MQCXP_VERSION_2	2	X'00000002'
MQCXP_VERSION_3	3	X'00000003'
MQCXP_VERSION_4	4	X'00000004'
MQCXP_VERSION_5	5	X'00000005'
MQCXP_VERSION_6	6	X'00000006'
MQCXP_VERSION_7	7	X'00000007'
MQCXP_VERSION_8	8	X'00000008'
MQCXP_VERSION_9	9	X'00000009'
 MQCXP_XX_ENCODE_CASE_ONE current_version	8	X'00000008'
 MQCXP_XX_ENCODE_CASE_ONE current_version	9	X'00000009'

MQDC_* (Hedef Sınıf)

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

MQDCC_* (Dönüştürme Seçenekleri, Maskeler ve Üreticiler)

Dönüştürme Seçenekleri

Ad	Ondalık değer	Onaltılı değer
MQDC_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)	
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)	
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

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)

Ad	Ondalık değer	Onaltılı değer
MQDELO_NONE	0	X'00000000'
MQDELO_LOCAL	4	X'00000004'

MQDH_* (Dağıtım üstbilgisi yapısı)

Ad	Yapı
MQDH_STRUC_ID	"DH↵"
MQDH_STRUC_ID_ARRAY	'D','H','↵','↵'

Not: ↵ simgesi tek bir boş karakteri temsil eder.

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)

Ad	Ondalık değer	Onaltılı değer
MQDHF_NEW_MSG_IDS	1	X'00000001'
MQDHF_NONE	0	X'00000000'

MQDISCONNECT_* (Komut biçimi Bağlantı Kesme Tipleri)

Ad	Ondalık değer	Onaltılı değer
MQDISCONNECT_NORMAL	0	X'00000000'
MQDISCONNECT_IMPLICIT	1	X'00000001'
MQDISCONNECT_Q_MGR	2	X'00000002'

MQDL_* (Dağıtım Listeleri)

Ad	Ondalık değer	Onaltılı değer
MQDL_DESTEKLENEN	1	X'00000001'
MQDL_NOT_SUPPORTANT	0	X'00000000'

MQDLH_* (Ölü harf üstbilgi yapısı)

Ad	Yapı
MQDLH_STRUC_ID	"DLH↵"
MQDLH_STRUC_ID_ARRAY	'D','L','H','↵'

Not: ↵ simgesi tek bir boş karakteri temsil eder.

Ad	Ondalık değer	Onaltılı değer
MQDLH_VERSION_1	1	X'00000001'
MQDLH_CURRENT_VERSION	1	X'00000001'

MQDLV_ * (Persistent/kalıcı Olmayan İleti Teslimi)

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

Ad	Yapı
MQDMHO_STRUC_ID	"DMHO"
MQDMHO_STRUC_ID_ARRAY	'D','M','H','O'

Not: ↯ simgesi tek bir boş karakteri temsil eder.

Ad	Ondalık değer	Onaltılı değer
MQDMHO_VERSION_1	1	X'00000001'
MQDMHO_CURRENT_VERSION	1	X'00000001'

İleti Tanıtıcısı Seçeneklerini Sil

Ad	Ondalık değer	Onaltılı değer
MQDMHO_NONE	0	X'00000000'

MQDMMPO_ * (İleti özelliği seçeneklerini ve yapısını sil)

İleti özelliği seçenekleri yapısını sil

Ad	Yapı
MQDMMPO_STRUCT_ID	"DMPO"
MQDMMPO_STRUC_ID_ARRAY	'D','M','P','O'

Not: ↯ simgesi tek bir boş karakteri temsil eder.

Ad	Ondalık değer	Onaltılı değer
MQDMMPO_VERSION_1	1	X'00000001'
MQDMMPO_CURRENT_VERSION	1	X'00000001'

İleti Özelliği Seçeneklerini Sil

Ad	Ondalık değer	Onaltılı değer
MQDMMPO_DEL_FIRST	0	X'00000000'
MQDMMPO_DEL_PROP_UNDER_CURSOR	1	X'00000001'

Ad	Ondalık değ�r	Onaltılı değ�r
MQDMPO_NONE	0	X'00000000'

MQDNSWLM_ * (DNS WLM)

Ad	Ondalık değ�r	Onaltılı değ�r
MQDNSWLM_NO	0	X'00000000'
MQDNSWLM_YES	1	X'00000001'

MQDT_ * (Hedef Tipleri)

Ad	Ondalık değ�r	Onaltılı değ�r
MQDT_APPL	1	X'00000001'
MQDT_BROKER	2	X'00000002'

MQDXP_ * (D n şt rme  ıkı ı parametre yapısı)

Ad	Yapı
MQDXP_STRUC_ID	"DXP�"
MQDXP_STRUC_ID_ARRAY	'D', 'X', 'P', '�'

Not:   simgesi tek bir bo  karakteri temsil eder.

Ad	Ondalık değ�r	Onaltılı değ�r
MQDXP_VERSION_1	1	X'00000001'
MQDXP_VERSION_2	2	X'00000002'
MQDXP_CURRENT_VERSION	2	X'00000002'

MQEC_ * (  aret Değ rleri)

Ad	Ondalık değ�r	Onaltılı değ�r
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_ * (S re bitimi)

Ad	Ondalık değ�r	Onaltılı değ�r
MQE_UNSiNSiZ	-1	X'FFFFFFFF'

MQENC_* (Kodlama)

MQENC_* (Kodlama)

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 '
	SPARC üzerindeSolaris	273	X'00000111 '
	UNIX sistemleri	273	X'00000111 '
	Windows	546	X'00000222 '
	Mikro Focus COBOL on Windows	17	X'00000011 '
	z/OS	785	X'00000311 '

MQENC_* (Kodlama Mister)

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)

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)

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)

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

Ad	Ondalık değ�r	Onaltılı değ�r
MQENC_FLOAT_IEEEE_TERS	512	X'00000200'
MQENC_FLOAT_S390	768	X'00000300'
MQENC_FLOAT_TNS	1024	X'00000400'

MQEPH_* (Embedded komut bi imi  stbilgi yapısı ve i aretleri)

G m l  komut bi imi  stbilgi yapısı

Ad	Yapı
MQEPH_STRUCT_ID	"EPH�"
MQEF_STRUC_ID_ARRAY	'E', 'P', 'H', '�'

Not:   simgesi tek bir bo  karakteri temsil eder.

Ad	Ondalık değ�r	Onaltılı değ�r
MQEF_STRUCT_LENGTH_FIXED	68	X'00000044'
MQEPH_VERSION_1	1	X'00000001'
MQEPH_CURRENT_VERSION	1	X'00000001'

G m l  komut bi imi  stbilgi i aretleri

Ad	Ondalık değ�r	Onaltılı değ�r
MQEPH_NONE	0	X'00000000'
MQEF_CCSID_EMBEDDED	1	X'00000001'

MQET_* (Komut bi imi  ıkı  Tipleri)

Ad	Ondalık değ�r	Onaltılı değ�r
MQET_MQSC	1	X'00000001'

MQEVO_* (Komut bi imi Olay Originleri)

Ad	Ondalık değ�r	Onaltılı değ�r
MQEVO_DI�ER	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'

MQEVR_* (Komut biçimi Olay Kaydı)

Ad	Ondalık değer	Onaltılı değer
MQEVR_DISABLE	0	X'00000000'
MQEVRENABLED	1	X'00000001'
MQEVR_EXCEPTION	2	X'00000002'
MQEVR_NO_DISPLAY	3	X'00000003'

MQEXPI_* (Süre Bitimi Tarama Aralığı)

Ad	Ondalık değer	Onaltılı değer
MQEXPI_OFF	0	X'00000000'

MQFB_* (Geribildirim 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'
MQFB_MAX_ACTIVITIES	282	X'0000011A'
MQFB_NOT_ILETILDI	283	X'0000011B'
MQFB_NOT_TESLIM	284	X'0000011C'

Ad	Ondalık değ�r	Onaltılı değ�r
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_YETKILI	402	X'00000192'
MQFB_CICS_BRIDGE_FAILURE	403	X'00000193'
MQFB_CICS_CORREL_ID_ERROR	404	X'00000194'
MQFB_CICS_CCSID_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)

Ad	Ondalık değer	Onaltılı değer
MQFC_YES	1	X'00000001'
MQFC_NO	0	X'00000000'

MQFMT_ * (Biçimler)

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	"MQHRF~~~"
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','~','~'

Ad	Değer
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','¬'
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.

MQGA_* (Grup Özniteliği Seçicileri)

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

MQGACF_* (Komut biçimi Grup Parametresi Tipleri)

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'
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'

Ad	Ondalık değ�r	Onaltılı değ�r
MQGACF_CHL_STATISTICS_DATA	8012	X'00001F4C'
MQGACF_LAST_KULLANILDI	8012	X'00001F4C'

MQGI_* (Grup Tanıtıcısı)

Ad	Değ�r
MQGI_NONE	X'00...00' (24 boş değ�r)
MQGI_NONE_DIZISI	'\0', '\0', ... (24 boş değ�r)

MQGMO_* (İleti se eneklerini ve yapısını al)

İleti se enekleri yapısını al

Ad	Yapı
MQGMO_STRUC_ID	"GMO¬"
MQGMO_STRUC_ID_ARRAY	'G', 'M', 'O', '¬'

Not: ¬ simgesi tek bir boş karakteri temsil eder.

Ad	Ondalık değ�r	Onaltılı değ�r
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 Se eneklerini Al

Ad	Ondalık değ�r	Onaltılı değ�r
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'
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'

Ad	Ondalık değ�r	Onaltılı değ�r
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_�ZELLİKLERİ	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)

Ad	Değ�r
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)

Ad	Ondalık değ�r	Onaltılı değ�r
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)

Ad	Ondalık değer	Onaltılı değer
MQHB_UNUSABLE_HBAG	-1	X'FFFFFFFF'
MQHB_NONE	-2	X'FFFFFFFE'

MQHC_ * (Bağlantı Elleri)

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)

Ad	Ondalık değer	Onaltılı değer
MQHM_UNUSABLE_HMSG	-1	X'FFFFFFFF'
MQHM_NONE	0	X'00000000'

MQHO_ * (Nesne Tanıtıcısı)

Ad	Ondalık değer	Onaltılı değer
MQHO_UNUSABLE_HOBJ	-1	X'FFFFFFFF'
MQHO_NONE	0	X'00000000'

MQHSTATE_ * (Komut biçimi tanıtıcı durumları)

Ad	Ondalık değer	Onaltılı değer
MQHSTATE_DEVREDİŞİ	0	X'00000000'
MQHSTATE_ETKİN	1	X'00000001'

MQIA_ * (Tamsayı Özniteliği Seçicileri)

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'

Ad	Ondalık değ�r	Onaltılı değ�r
MQIA_ADOPTNEWMCA_TYPE	103	X'00000067'
MQIA_ADOPTNEWMCA_INTERVAL	104	X'00000068'
MQIA_APPL_TYPE	1	X'00000001'
MQIA_ADOPT_CONTEXT	260	X'00000104'
MQIA_AMQP_CAPABILITY	265	X'00000109'
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'
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'

Ad	Ondalık değ�r	Onaltılı değ�r
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'
MQIA_INHIBIT_GET	9	X'00000009'
MQIA_INHIBIT_PUB	181	X'000000B5'
MQIA_INHIBIT_PUT	10	X'0000000A'

Ad	Ondalık değ�r	Onaltılı değ�r
MQIA_INHIBIT_SUB	182	X'000000B6'
MQIA_INTRA_GROUP_queuing	64	X'00000040'
MQIA_IP_ADDRESS_VERSION	93	X'0000005D'
MQIA_LAST	2000	X'000007D0'
MQIA_LAST_USED	266	X'0000010A'
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'

Ad	Ondalık değ�r	Onaltılı değ�r
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'

Ad	Ondalık değ�r	Onaltılı değ�r
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'
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'

Ad	Ondalık değ�r	Onaltılı değ�r
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'
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 bi imi Tamsayı Deėi tirgesi Tipleri)

Ad	Ondalık değ�r	Onaltılı değ�r
MQIACF_FIRST	1001	X'000003E9'

Ad	Ondalık değ�r	Onaltılı değ�r
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'
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_SE�ENEKLER	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'

Ad	Ondalık değeri	Onaltılı değeri
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 SEÇENEKLERİ	1082	X'0000043A'
MQIACF_CLUSTER_INFO	1083	X'0000043B'
MQIACF_Q_MGR_DEFINITION_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'
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'

Ad	Ondalık değeri	Onaltılı değeri
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'
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_ADı	1147	X'0000047B'
MQIACF_Q_MGR_NUMARASI	1148	X'0000047C'
MQIACF_Q_MGR_STATUS	1149	X'0000047D'

Ad	Ondalık değeri	Onaltılı değeri
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_KÜMESİ	1169	X'00000491'
MQIACF_USAGE_BUFFER_POOL	1170	X'00000492'
MQIACF_MOVE_COUNT	1171	X'00000493'
MQIACF_EXPIRY_Q_COUNT	1172	X'00000494'
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'

Ad	Ondalık değ�r	Onaltılı değ�r
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_G�REVLERİ	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'
MQIACF_SYSP_QUIESCE_INTERVAL	1212	X'000004BC'
MQIACF_SYSP_TIMESTAMP	1213	X'000004BD'
MQIACF_SYSP_UNIT_ADRESİ	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'

Ad	Ondalık değeri	Onaltılı değeri
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_ETKİNLİĞİ	1236	X'000004D4'
MQIACF_DISCONTINUITY_COUNT	1237	X'000004D5'
MQIACF_ROUTE_BIRIKIM	1238	X'000004D6'
MQIACF_ROUTE_DELİÇE	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_GERİBİLDİRİM	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'
MQIACF_PERSISTENCE	1252	X'000004E4'
MQIACF_PRIORITY	1253	X'000004E5'
MQIACF_REASON_CODE	1254	X'000004E6'
MQIACF_REPORT	1255	X'000004E7'

Ad	Ondalık değeri	Onaltılı değeri
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'

Ad	Ondalık değeri	Onaltılı değeri
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'

Ad	Ondalık değ�r	Onaltılı değ�r
MQIACF_SMDS_EXPANDST	1376	X'00000560'
MQIACF_STRUC_LENGTH	1377	X'00000561'
MQIACF_ITEM_SAYISI	1378	X'00000562'
MQIACF_EXPIRY_TIME	1379	X'00000563'
MQIACF_CONNECT_TIME	1380	X'00000564'
MQIACF_DISCONNECT_TIME	1381	X'00000565'
MQIACF_HSUB	1382	X'00000566'
MQIACF_SUBRQ_SE�ENEKLER	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'

Ad	Ondalık değ�r	Onaltılı değ�r
➤ V8.0.0.4 ➤ V8.0.0.4 MQIACF_AUTH_REC_TYPE	1412	X'00000584'
MQIACF_LAST_KULLANILDI	1412	X'00000584'

MQIACH_* (Komut bi imi Tamsayı Kanal Tipleri)

Ad	Ondalık değ�r	Onaltılı değ�r
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'
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'

Ad	Ondalık değ�r	Onaltılı değ�r
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'
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'

Ad	Ondalık değ�r	Onaltılı değ�r
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'
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'

Ad	Ondalık değ�r	Onaltılı değ�r
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_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'
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_LAST_USED	1645	X'0000066D'

MQIAMO_* (Komut biçimi Tamsayı İzleme Parametresi Tipleri)

Ad	Ondalık değ�r	Onaltılı değ�r
MQIAMO_FIRST	701	X'000002BD'

Ad	Ondalık değ�r	Onaltılı değ�r
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'
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_A�AR	733	X'000002DD'
MQIAMO_PUT1S	734	X'000002DE'
MQIAMO_TAKS	735	X'000002DF'
MQIAMO_PUT_MAX_BYTES	736	X'000002E0'

Ad	Ondalık değ�r	Onaltılı değ�r
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_CLOSSES_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'
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'

Ad	Ondalık değ�r	Onaltılı değ�r
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_G�NDERILDI	790	X'00000316'
MQIAMO_BYTES_G�NDERILDI	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_G�NDERILDI	810	X'0000032A'
MQIAMO_TOTAL_MSGS_G�NDERILDI	811	X'0000032B'
MQIAMO_TOTAL_BYTES_G�NDERILDI	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'
MQIAMO_MSGS_RCVD	817	X'00000331'

Ad	Ondalık değ�r	Onaltılı değ�r
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)

Ad	Ondalık değ�r	Onaltılı değ�r
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)

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'
MQIAASY_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'FFFFF830'

MQIAUT_* (IMS bilgi üstbilgisi kimliği doğrulayıcı)

Ad	Değer
MQIAUT_NONE	"~~~~~"
MQIAUT_NONE_DIZISI	'~','~','~','~','~','~','~','~','~'

Not: ~ simgesi tek bir boş karakteri temsil eder.

MQIAV_* (Tamsayı Özniteliği Değerleri)

Ad	Ondalık değer	Onaltılı değer
MQIAV_NOT_GEÇERLI	-1	X'FFFFFFFF'
MQIAV_UNDEFINED	-2	X'FFFFFFFE'

MQICM_* (IMS bilgi üstbilgisi Kesinleştir Kipleri)

Ad	Değer
MQICM_COMMIT_THEN_SEND	'0'
MQICM_SEND_THEN_COMMIT	'1'

MQIDO_* (Komut biçimi Kuşku Seçenekleri Seçenekleri)

Ad	Ondalık değer	Onaltılı değer
MQIDO_COMMIT	1	X'00000001'
MQIDO_BACKUT	2	X'00000002'

MQIEP_ * (Arabirim giriş noktaları)

Bağlantı güvenliği parametreleri yapısı

Ad	Yapı
MQIEP_STRUC_ID	"IEP¬"
MQIEP_STRUC_ID_ARRAY	'I','E','P','¬'

Not: ¬ simgesi tek bir boş karakteri temsil eder.

Ad	Ondalık değer	Onaltılı değer
MQIEP_VERSION_1	1	X'00000001'
MQDXP_CURRENT_VERSION	1	X'00000001'

MQIGQ_ * (Intra-Group queuing)

Ad	Ondalık değer	Onaltılı değer
MQIGQ_DISABLE	0	X'00000000'
MQIGQ_ENABLED	1	X'00000001'

MQIGQPA_ * (Intra-Group queuing put Authority)

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

Ad	Yapı
MQIIH_STRUC_ID	"IIH¬"
MQIIH_STRUC_ID_ARRAY	'I','I','H','¬'

Not: ¬ simgesi tek bir boş karakteri temsil eder.

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

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

Ad	Yapı
MQIMPO_STRUC_ID	"IMPO"
MQIMPO_STRUC_ID_ARRAY	'I', 'M', 'P', 'O'

Not: ↯ simgesi tek bir boş karakteri temsil eder.

Ad	Ondalık değer	Onaltılı değer
MQIMPO_VERSION_1	1	X'00000001'
MQIMPO_CURRENT_VERSION	1	X'00000001'

Sorgu İleti Özelliği Seçenekleri

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)

Ad	Ondalık değer	Onaltılı değer
MQINBD_Q_MGR	0	X'00000000'
MQINBD_GROUP	3	X'00000003'

MQIND_* (Özel Dizin Değerleri)

Ad	Ondalık değer	Onaltılı değer
MQIND_NONE	-1	X'FFFFFFFF'
MQIND_ALL	-2	X'FFFFFFFE'

MQIPADDR_* (IP Adresi Sürümleri)

Ad	Ondalık değer	Onaltılı değer
MQIPADDR_IPv4	0	X'00000000'
MQIPADDR_IPv6	1	X'00000001'

MQISS_* (IMS bilgi üstbilgisi Güvenlik Kapsamları)

Ad	Değer
MQISS_CHECK	'C'
MQISS_FULL	'F'

MQIT_* (Dizin Tipleri)

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)

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)

Ad	Değer
MQITII_NONE	X'00...00' (16 boş değer)

Ad	Değer
MQITII_NONE_DIZISI	'\0', '\0', ... (16 boş değer)

MQITS_ * (IMS bilgi üstbilgisi hareket durumları)

Ad	Değer
MQITS_IN_MUHABBET	'C'
MQITS_NOT_IN_SOHBET	'¬'
MQITS_ARCHITECTED	'A'

Not: ¬ simgesi tek bir boş karakteri temsil eder.

MQKAI_ * (KeepAlive Aralığı)

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

MQMASTER_ * (Ana denetim)

Ad	Ondalık değer	Onaltılı değer
MQMASTER_NO	0	X'00000000'
MQMASTER_EVET	1	X'00000001'

MQMCAS_ * (Komut biçimi İleti Kanalı Aracısı Durumu)

Ad	Ondalık değer	Onaltılı değer
MQMCAS_DURDU	0	X'00000000'
MQMCAS_RUNNING	3	X'00000003'

MQMCAT_ * (MCA Tipleri)

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

Ad	Ondalık değer	Onaltılı değer
MQMCD_KLASÖRÜ	1	X'00000001'

Yayınlama/Abone Olma Seçenekleri Etiket Etiketleri adları

Ad	Değer
MQMCD_MSG_DOMAIN	"Msd"

Ad	Değer
MQMCD_MSG_KÜMESİ	"Set"
MQMCD_MSG_TYPE	"Type"
MQMCD_MSG_BİÇİMİ	"Fmt"

Yayınlama/Abone Olma Seçenekleri Etiketi XML etiketi adları

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 Etiketi değerleri

Ad	Değer
MQMCD_DOMAIN_YOK	"none"
MQMCD_DOMAIN_NEON	"neon"
MQMCD_DOMAIN_MRM	"mim"
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)

Ad	Yapı
MQMD_STRUC_ID	"MD↯"
MQMD_STRUC_ID_ARRAY	'M', 'D', '↯', '↯'

Not: ↯ simgesi tek bir boş karakteri temsil eder.

Ad	Ondalık değer	Onaltılı değer
MQMD_VERSION_1	1	X'00000001'
MQMD_VERSION_2	2	X'00000002'

Ad	Ondalık değeri	Onaltılı değeri
MQMD_CURRENT_VERSION	2	X'00000002'

MQMDE_* (İleti tanımlayıcı uzantısı yapısı)

Ad	Yapı
MQMDE_STRUC_ID	"MDE↵"
MQMDE_STRUC_ID_ARRAY	'M','D','E','↵'

Not: ↵ simgesi tek bir boş karakteri temsil eder.

Ad	Ondalık değeri	Onaltılı değeri
MQMDE_VERSION_2	2	X'00000002'
MQMDE_CURRENT_Sürüm	2	X'00000002'
MQMDE_LENGTH_2	72	X'00000048'

MQMDEF_* (İleti tanımlayıcısı uzantı işaretleri)

Ad	Ondalık değeri	Onaltılı değeri
MQMDEF_NONE	0	X'00000000'

MQMDS_* (İleti Teslim Sırası)

Ad	Ondalık değeri	Onaltılı değeri
MQMDS_PRIORITY	0	X'00000000'
MQMDS_FIFO	1	X'00000001'

MQMF_* (İleti İşaretleri)

Ad	Ondalık değeri	Onaltılı değeri
MQMF_SEGMENTATION_ENGELLENDİ	0	X'00000000'
MQMF_SEGMENTATION_ALLOW	1	X'00000001'
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

Ad	Yapı
MQMHBO_STRUCT_ID	"MHBO"

Ad	Yapı
MQMHBO_STRUC_ID_ARRAY	'M', 'H', 'B', 'O'

Not: ↪ simgesi tek bir boş karakteri temsil eder.

Ad	Ondalık değer	Onaltılı değer
MQMHBO_VERSION_1	1	X'00000001'
MQMHBO_CURRENT_VERSION	1	X'00000001'

İleti Tanıtıcısı-Arabellek Seçenekleri

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)

Ad	Değer
MQMI_NONE	X'00...00' (24 boş değer)
MQMI_NONE_DIZISI	'\0', '\0', ... (24 boş değer)

MQMMBI_ * (İleti İşareti-Göz At Aralığı)

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

MQMO_ * (Eşleme Seçenekleri)

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)

Ad	Ondalık değer	Onaltılı değer
MQMODE_FORCE	0	X'00000000'
MQMODE_QUIESCE	1	X'00000001'
MQMODE_TERMINATE	2	X'00000002'

MQMON_ * (İzleme 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)

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'
MQMT_MQE_FIELDS	113	X'00000071'
MQMT_SYSTEM_SON	65535	X'0000FFFF'
MQMT_APPL_FIRST	65536	X'00010000'
MQMT_APPL_SON	999999999	X'3B9AC9FF'

MQMTOK_ * (İleti Simgesi)

Ad	Değer
MQMTOK_NONE	X'00...00' (16 boş değer)
MQMTOKT_NONE_DIZISI	'\0','\0',... (16 boş değer)

Ad	Ondalık değer	Onaltılı değer
MQMTOK_NONE	X'00...00'	(16 nulls)
MQMTOKT_NONE_DIZISI	'\0','\0',...	(16 nulls)

MQNC_ * (Ad Sayısı)

Ad	Ondalık değer	Onaltılı değer
MQNC_MAX_NAMELIST_NAME_COUNT	256	X'00000100'

MQNPM_ * (Kalıcı Olmayan İleti Sınıfı)

Ad	Ondalık değer	Onaltılı değer
MQNPM_CLASS_NORMAL	0	X'00000000'
MQNPM_CLASS_YÜKSEK	10	X'0000000A'

MQNPMS_ * (NonPersistent-İleti Hızları)

Ad	Ondalık değer	Onaltılı değer
MQNPMS_NORMAL	1	X'00000001'
MQNPMS_FAST	2	X'00000002'

MQNT_ * (Ad Listesi Tipleri)

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'
MQNT_ALL	1001	X'000003E9'

MQNVS_ * (Ad/Değer Dizgisi için adlar)

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)

Ad	Ondalık değer	Onaltılı değer
MQOA_FIRST	1	X'00000001'
MQOA_SON	9000	X'00002328'

MQOD_ * (Nesne tanımlayıcı yapısı)

Ad	Yapı
MQOD_STRUC_ID	"OD↵↵"
MQOD_STRUC_ID_ARRAY	'0','D','↵','↵'

Not: ↯ simgesi tek bir boş karakteri temsil eder.

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)	

MQOLI_ * (Nesne Eşgörünümlü Tanıtıcısı)

Ad	Değer
MQOL_NONE	X'00...00' (24 boş değer)
MQOII_NONE_DIZISI	'\0','\0',... (24 boş değer)

MQOL_ * (Özgün Uzunluk)

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

MQOM_ * (Sorgu Grubuna ilişkin Kullanım Dışı Db2 İletileri seçenekleri)

Ad	Ondalık değer	Onaltılı değer
MQOM_NO	0	X'00000000'
MQOM_YES	1	X'00000001'

MQOO_ * (Açma Seçenekleri)

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'

Ad	Ondalık değ�r	Onaltılı değ�r
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)

Ad	Ondalık değ�r	Onaltılı değ�r
MQOO_RESOLVE_NAMES	65536	X'00010000'
MQOO_RESOLVE_LOCAL_Q	262144	X'00040000'

MQOP_* (MQCTL ve MQCB i in i lem kodları)

MQCTL 'ye ili kin i lem kodları

Ad	Ondalık değ�r	Onaltılı değ�r
MQOP_START	1	X'00000001'
MQOP_START_SABE	2	X'00000002'
MQOP_STOP	4	X'00000004'

MQCB i in i lem kodları

Ad	Ondalık değ�r	Onaltılı değ�r
MQOP_KAYDOL	256	X'00000100'
MQOP_DEREGISTER	512	X'00000200'

MQCTL ve MQCB 'ye ili kin i lem kodları

Ad	Ondalık değ�r	Onaltılı değ�r
MQOP_SUSPEND	65536	X'00010000'
MQOP_RESUME	131072	X'00020000'

MQOPEN_ * (MQOPEN_PRIV yapısıyla ilgili değerler)

Ad	Ondalık değer	Onaltılı değer
MQOPEN_PRIV_VERSION_1	1	X'00000001'
MQOPEN_PRIV_CURRENT_VERSION	1	X'00000001'

MQOPER_ * (Etkinlik İşlemleri)

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'
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

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'

Ad	Ondalık değ�r	Onaltılı değ�r
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'

Geniřletilmiř Nesne Tipleri

Ad	Ondalık değ�r	Onaltılı değ�r
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'
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)

Ad	Ondalık değ�r	Onaltılı değ�r
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

Ad	Yapı
MQPD_STRUC_ID	"PD¬¬"
MQPD_STRUC_ID_ARRAY	'P','D','¬','¬'

Not: ¬ simgesi tek bir boş karakteri temsil eder.

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

Ad	Ondalık değer	Onaltılı değer
MQPD_NONE	0	X'00000000'

Özellik Desteği Seçenekleri

Ad	Ondalık değer	Onaltılı değer
MQPD_SUPPORT_OPTIONAL	1	X'00000001'
MQPD_SUPPORT_REQUIRY	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

Ad	Ondalık değer	Onaltılı değer
MQPD_NO_CONTEXT	0	X'00000000'
MQPD_USER_CONTEXT	1	X'00000001'

MQPER_* (Kalıcılık 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)

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'
MQPL_WINDOWS_NT	11	X'0000000B'
MQPL_VMS	12	X'0000000C'
MQPL_NSK	13	X'0000000D'
MQPL_NSS	13	X'0000000D'
MQPL_OPEN_TP1	15	X'0000000F'
MQPL_VM	18	X'00000012'
MQPL_TPF	23	X'00000017'
MQPL_VSE	27	X'0000001B'
 V8.0.0.2  V8.0.0.2 MQP_APPLIANCE	28	X'0000001C'
MQP_NATIVE	1	X'00000001'

MQPMO_* (Yayınlama maskesi için ileti seçenekleri ve yapısı kullanın)

İleti seçenekleri yapısını koy

Ad	Yapı
MQPMO_STRUC_ID	"PMO¬"
MQPMO_STRUC_ID_ARRAY	'P','M','O','¬'

Not: ¬ simgesi tek bir boş karakteri temsil eder.

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)	

İleti Koyma Seçenekleri

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'
MQPMO_ALTERNATE_USER_AUTHORITY	4096	X'00001000'
MQPMO_FAIL_IF QUIESCING	8192	X'00002000'
MQPMO_NO_BAĞLAMI	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

Ad	Ondalık değer	Onaltılı değer
MQPMO_PUB_OPTIONS_MASK	2097152	X'00200000'

MQPMRF_ * (İleti Kaydı Alanları Koy)

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'

Ad	Ondalık değ�r	Onaltılı değ�r
MQPMRF_ACCOUNTING_TOKEN	16	X'00000010'
MQPMRF_NONE	0	X'00000000'

MQPO_ * (Komut bi imi Temizleme Se enekleri)

Ad	Ondalık değ�r	Onaltılı değ�r
MQPO_YES	1	X'00000001'
MQPO_NO	0	X'00000000'

MQPRI_ * ( ncelik)

Ad	Ondalık değ�r	Onaltılı değ�r
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  zelli i Denetim De erleri ve Maksimum  zellik Uzunlu u)

Kuyruk ve Kanal  zellik Denetimi De erleri

Ad	Ondalık değ�r	Onaltılı değ�r
MQPROP_UYUMLULUK	0	X'00000000'
MQPROP_NONE	1	X'00000001'
MQPROP_ALL	2	X'00000002'
MQPROP_FORCE_MQRFH2	3	X'00000003'

 zellik Uzunlu u  st Sınırı

Ad	Ondalık değ�r	Onaltılı değ�r
MQPROP_UNRESTRICTED_LENGTH	-1	X'FFFFFFFF'

MQPRT_ * (Yanıt De erlerini Koyma)

Ad	Ondalık değ�r	Onaltılı değ�r
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

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'
MQPS_STATUS_REFRED	6	X'00000006'

Etiketleri dize olarak Yayınla/Abone Ol

MQPS_COMMAND	"MQPSCommand"
MQPS_COMP_CODE	"MQPSCompCode"
MQPS_COREL_ID	"MQPSCorelId"
MQPS_DELEE	"MQPSDe10pts"
MQPS_ERROR_ID	"MQPSErrorId"
MQPS_ERROR_POS	"MQPSErrorPos"
MQPS_INTEGER_DATA	"MQPSIntData"
MQPS_PARAMETER_ID	"MQPSParmId"
MQPS_YAYINLAMA SEÇENEKLERİ	"MQSPubOpts"
MQPS_PUBLISH_TIMESTAMP	"MQSPubTime"
MQPS_Q_MGR_NAME	"MQPSQMgrName"
MQPS_Q_ADı	"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	"¬MQPSDelOpts¬"
MQPS_ERROR_ID_B	"¬MQPSErrorId¬"
MQPS_ERROR_POS_B	"¬MQPSErrorPos¬"
MQPS_INTEGER_DATA_B	"¬MQPSIntData¬"
MQPS_PARAMETER_ID_B	"¬MQPSParmId¬"
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 Etiketi 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"

Yayınlama/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_ISTEĞİ	"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_SÖZNİZ_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 Etiketi Yayınlama/Abone Olma Komut Klasörü (psc) Etiketleri)

Ad	Ondalık değer	Onaltılı değer
MQPSC_FOLDER_VERSION	1	X'00000001'

MQPSC_ * (Yayınlama/Abone Olma Seçenekleri Etiket Etiketi 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 Etiketi XML etiketi 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 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 Değerleri dizgi 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

Ad	Ondalık değer	Onaltılı değer
MQPSCR_FOLDER_VERSION	1	X'00000001'

Yayınlama/Abone Olma Seçenekleri Etiket Etiketi adları

MQPSCR_COMPLETION	"Completion"
MQPSCR_RESPONSE	"Response"
MQPSCR_REASON	"Reason"

Yayınlama/Abone Olma Seçenekleri Etiketi 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 Etiketi değerleri

MQPSCR_OK	"ok"
MQPSCR_UYARI	"warning"
MQPSCR_HATASI	"error"

MQPSM_* (Pub/Alt Kipi)

Ad	Ondalık değer	Onaltılı değer
MQPSM_DISABLE	0	X'00000000'
MQPSM_COMPAT	1	X'00000001'
MQPSMENABLED	2	X'00000002'

MQPSPROP_* (Pub/Alt İleti Özellikleri)

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)

Ad	Ondalık değer	Onaltılı değer
MQPSST_ALL	0	X'00000000'
MQPSST_LOCAL	1	X'00000001'
MQPSST_PARENT	2	X'00000002'
MQPSST_COCUK	3	X'00000003'

MQPUBO_* (Yayınlama/Abone Olma Yayın Seçenekleri)

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'

MQPXP_* (Yayınlama/abone olma yöneltme çıkış değiştirgesi yapısı)

Ad	Yapı
MQPXP_STRUC_ID	"PXP¬"
MQPXP_STRUC_ID_ARRAY	'P','X','P','¬'

Not: ¬ simgesi tek bir boş karakteri temsil eder.

Ad	Ondalık değer	Onaltılı değer
MQPXP_VERSION_1	1	X'00000001'
MQPXP_XX_ENCODE_CASE_ONE current_version	1	X'00000001'

MQQA_* (Kuyruk öznitelikleri)

Alma Değerlerini Engelle

Ad	Ondalık değer	Onaltılı değer
MQQA_GET_INHIBITED	1	X'00000001'

Ad	Ondalık değ�r	Onaltılı değ�r
MQQA_GET_ALLOWD	0	X'00000000'

Koyma Değ rlerini Engelle

Ad	Ondalık değ�r	Onaltılı değ�r
MQQA_PUT_INHIBITED	1	X'00000001'
MQQA_PUT_ALLOWD	0	X'00000000'

Kuyruk Paylaşılabilirliğı

Ad	Ondalık değ�r	Onaltılı değ�r
MQQA_SHAREABLE	1	X'00000001'
MQQA_NOT_SHAREABLE	0	X'00000000'

Geri D n ş Sertleşmesi

Ad	Ondalık değ�r	Onaltılı değ�r
MQQA_BACKUT_HARDENED	1	X'00000001'
MQQA_BACKUT_NOT_HARDENED	0	X'00000000'

MQQDT_ * (Kuyruk Tanımı Tipleri)

Ad	Ondalık değ�r	Onaltılı değ�r
MQQDT_�NCEDEN 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)

Ad	Ondalık değ�r	Onaltılı değ�r
MQQF_LOCAL_Q	1	X'00000001'
MQQF_CLWL_USEQ_ANY	64	X'00000040'
MQQF_CLWL_USEQ_LOCAL	128	X'00000080'

MQQMDT_ * (Komut biçimi Kuyruk Y neticisi Tanımlaması Tipleri)

Ad	Ondalık değ�r	Onaltılı değ�r
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)

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 Olanağı)

Ad	Ondalık değer	Onaltılı değer
MQQMFAT_IMS_BRIDGE	1	X' 00000001 '
MQQMFAC_DB2	2	X' 00000002 '

MQQMSTA_ * (Komut biçimi Kuyruk Yöneticisi Durumu)

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)

Ad	Ondalık değer	Onaltılı değer
MQQMT_NORMAL	0	X' 00000000 '
MQQMT_REPOSITORY	1	X' 00000001 '

MQQO_ * (Komut biçimi Quiesce Seçenekleri)

Ad	Ondalık değer	Onaltılı değer
MQQO_YES	1	X' 00000001 '
MQQO_NO	0	X' 00000000 '

MQQSGD_ * (Kuyruk Paylaşımı Grubu Atımları)

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 '
MQQSGD_PRIVATE	4	X' 00000004 '
MQQSGD_LIVE	6	X' 00000006 '

MQQSGS_* (QSG Komutu Durumu)

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)

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)

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)

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)

Ad	Ondalık değer	Onaltılı değer
MQQSUM_YES	1	X'00000001'
MQQSUM_NO	0	X'00000000'

MQQT_ * (Kuyruk Tipleri ve Geniřletilmiş Kuyruk Tipleri)

Kuyruk Tipleri

Ad	Ondalık değęer	Onaltılı değęer
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'

Geniřletilmiş Kuyruk Tipleri

Ad	Ondalık değęer	Onaltılı değęer
MQQT_ALL	1001	X'000003E9'

MQRC_ * (Neden Kodları)

Ad	Ondalık değęer	Onaltılı değęer
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'
MQRC_HCONN_ERROR	2018	X'000007E2'
MQRC_HOBJ_ERROR	2019	X'000007E3'

Ad	Ondalık değ�r	Onaltılı değ�r
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_AVAIABLE	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'

Ad	Ondalık değ�r	Onaltılı değ�r
MQRC_Q_MGR_NAME_ERROR	2058	X'0000080A'
MQRC_Q_MGR_NOT_VAR	2059	X'0000080B'
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'

Ad	Ondalık değeri	Onaltılı değeri
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'
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'

Ad	Ondalık değ�r	Onaltılı değ�r
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'
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'

Ad	Ondalık değeri	Onaltılı değeri
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'
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'

Ad	Ondalık değ�r	Onaltılı değ�r
MQRC_RAS_�ZELI_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_CCIDS	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'
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_S�r�m�	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'

Ad	Ondalık değeri	Onaltılı değeri
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'
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_IPTAL	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'

Ad	Ondalık değ�r	Onaltılı değ�r
MQRC_SELECTOR_NOT_UNIQUE	2305	X'00000901'
MQRC_INDEX_NOT_PRESENT	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'
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'

Ad	Ondalık değeri	Onaltılı değeri
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_DÜZEYİ	2366	X'0000093E'
MQRC_CONFIG_CREATE_OBJECT	2367	X'0000093F'
MQRC_CONFIG_CHANGE_OBJECT	2368	X'00000940'
MQRC_CONFIG_DELET_OBJECT	2369	X'00000941'
MQRC_CONFIG_REFRESH_OBJECT	2370	X'00000942'
MQRC_CHANNEL_SSL_ERROR	2371	X'00000943'
MQRC_AğLAN_KURULUŞ_NOT_	2372	X'00000944'
MQRC_CF_STRUC_FAILED	2373	X'00000945'
MQRC_API_EXIT_ERROR	2374	X'00000946'

Ad	Ondalık değ�r	Onaltılı değ�r
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_AVAIABLE	2379	X'0000094B'
MQRC_SCO_ERROR	2380	X'0000094C'
MQRC_KEY_REPOSITORY_ERROR	2381	X'0000094D'
MQRC_CRYPTO_HARDWARE_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'
MQC_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_FESHEDILDI	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'
MQRC_COMMAND_MQSC	2412	X'0000096C'

Ad	Ondalık değeri	Onaltılı değeri
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_ALTYAPI	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'

Ad	Ondalık değ�r	Onaltılı değ�r
MQRC_HMSG_ERROR	2460	X'0000099C'
MQRC_CMHO_ERROR	2461	X'0000099D'
MQRC_DMHO_ERROR	2462	X'0000099E'
MQRC_SMPO_ERROR	2463	X'0000099F'
MQRC_IMPO_ERROR	2464	X'000009A0'
MQRC_�ZELLİĐİN_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_�ZELLİĐİ TYPE_	2473	X'000009A9'
MQRC_�ZELLİK_TOO_B�Y�K	2478	X'000009AE'
MQRC_PUT_NOT_ALICILI	2479	X'000009AF'
MQRC_ALIAS_TARGTYPE_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_�ZELLİĐİ	2490	X'000009BA'
MQRC_PROP_NAME_NOT_CONVERTED	2492	X'000009BC'
MQRC_GET_ENABLED ışıĐı	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_ETK�N	2500	X'000009C4'
MQRC_MHBO_ERROR	2501	X'000009C5'

Ad	Ondalık değ�r	Onaltılı değ�r
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'
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'

Ad	Ondalık değ�r	Onaltılı değ�r
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'
MQRC_CURSOR_NOT_VALID	6105	X'000017D9'
MQRCENCODING_ERROR	6106	X'000017DA'
MQRC_STRUC_ID_ERROR	6107	X'000017DB'
MQRC_NULL_POINTER	6108	X'000017DC'
MQRC_NO_CONNECTION_REFERENCE	6109	X'000017DD'
MQRC_NO_BUFFER	6110	X'000017DE'
MQRC_BINARY_DATA_LENGTH_ERROR	6111	X'000017DF'
MQRC_BUFFER_NOT_AUTOMAKRED	6112	X'000017E0'
MQRC_INSUFFICIENT_BUFFER	6113	X'000017E1'
MQRC_INSUFFICIENT_DATA	6114	X'000017E2'
MQRC_DATA_TRUNCATED	6115	X'000017E3'
MQRC_ZERO_LENGTH	6116	X'000017E4'
MQRC_NEGATIVE_LENGTH	6117	X'000017E5'
MQRC_NEGATIVE_OFFSET	6118	X'000017E6'
MQRC_INCONSTENT_BI�IMI	6119	X'000017E7'
MQRC_INCONSISTENT_OBJECT_STATE	6120	X'000017E8'
MQRC_CONTEXT_OBJECT_NOT_VALID	6121	X'000017E9'
MQRC_CONTEXT_OPEN_ERROR	6122	X'000017EA'
MQRC_STRUC_LENGTH_ERROR	6123	X'000017EB'
MQRC_NOT_CONNECTED	6124	X'000017EC'
MQRC_NOT_A�IK	6125	X'000017ED'
MQRC_DISTRIBUTION_LIST_EMPTY	6126	X'000017EE'
MQRC_INCONSISTENT_OPEN_OPTIONS	6127	X'000017EF'
MQRC_WRONG_S�R�M	6128	X'000017F0'

Ad	Ondalık değ�r	Onaltılı değ�r
MQRRCF_REFERENCE_ERROR	6129	X'000017F1'

MQRCCF_* (Komut biçimi üstbilgi neden kodları)

Programcı yanıtına ilişkin ek bilgi için [PCF neden kodları](#) başlıklı konuya bakın.

Ad	Ondalık değ�r	Onaltılı değ�r
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'
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'

Ad	Ondalık değ�r	Onaltılı değ�r
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'
MQRCCF_MSG_KESILDI	3048	X'00000BE8'
MQRCCF_CCSIDI	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'

Ad	Ondalık değ�r	Onaltılı değ�r
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_CCSID_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'
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'

Ad	Ondalık değ�r	Onaltılı değ�r
MQRCCF_ALREADY_JOINED	3157	X'00000C55'
MQRCCF_OBJECT_IN_USE	3160	X'00000C58'
MQRCCF_UNKNOWN_FILE_ADı	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_ETK�N	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'
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_HATASı	3215	X'00000C8F'
MQRCCF_NO_CHANNEL_INITIATOR	3217	X'00000C91'
MQRCCF_CHANNEL_INITIATOR_ERROR	3218	X'00000C92'

Ad	Ondalık değeri	Onaltılı değeri
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_LISTENER_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_LENERR	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'
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'

Ad	Ondalık değ�r	Onaltılı değ�r
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_LENGTH_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'
MQRCCF_CHLAUTH_ALREADY_EXISTS	3337	X'00000D09'

Ad	Ondalık değeri	Onaltılı değeri
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_OBJECT_ALREADY_EXISTS	4001	X'00000FA1'
MQRCCF_OBJECT_HATA_TIP1	4002	X'00000FA2'
MQRCCF_LIKE_OBJECT_HATA_TIP1	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_TIP1	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_STORSAYı	4018	X'00000FB2'
MQRCCF_NO_COMMS_YöNETME	4019	X'00000FB3'

Ad	Ondalık değ�r	Onaltılı değ�r
MQRCCF_LISTENER_NOT_STARTED	4020	X'00000FB4'
MQRCCF_BIND_FAILED	4024	X'00000FB8'
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'

Ad	Ondalık değ�r	Onaltılı değ�r
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_ETK�N	4064	X'00000FE0'
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'

Ad	Ondalık değ�r	Onaltılı değ�r
MQRCCF_RETAINED_NOT_SUPPORSED	4095	X'00000FFF'

MQRCN_ * (İstemci Yeniden Baėlan Deėişmezleri)

Ad	Ondalık değ�r	Onaltılı değ�r
MQRCN_NO	0	X'00000000'
MQRCN_YES	1	X'00000001'
MQRCN_Q_MGR	2	X'00000002'
MQRCN_DISABLED	3	X'00000003'

MQRCVTIME_ * (Alma Zamanařımı Tipleri)

Ad	Ondalık değ�r	Onaltılı değ�r
MQRCVTIME_�ARP	0	X'00000000'
MQRCVTIME_ADD	1	X'00000001'
MQRCVTIME_EQUAL	2	X'00000002'

MQREADA_ * (İleriye Okuma Deėerleri)

Ad	Ondalık değ�r	Onaltılı değ�r
MQREADA_NO	0	X'00000000'
MQREADA_YES	1	X'00000001'
MQREADA_DEVRE Dıřı	2	X'00000002'
MQREADA_ENGELLEYICI	3	X'00000003'
MQREADA_BACKLOG	4	X'00000004'

MQRECORDING_ * (Kayıt Se enekleri)

Ad	Ondalık değ�r	Onaltılı değ�r
MQRECORDING_DEVRE Dıřı	0	X'00000000'
MQRECORDING_Q	1	X'00000001'
MQRECORDING_MSG	2	X'00000002'

MQREGO_ * (Yayınlama/Abone Olma Kayıt Se enekleri)

Ad	Ondalık değ�r	Onaltılı değ�r
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'

Ad	Ondalık değ�r	Onaltılı değ�r
MQREGO_YAYINISI_YALNIZ_ISTEĐI	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'
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 bi imleme  stbilgisi yapısı ve i aretleri)

Kurallar ve bi imlendirme  stbilgisi yapısı

Ad	Yapı
MQRFH_STRUC_ID	"RFH¬"
MQRFH_STRUC_ID_ARRAY	'R','F','H','¬'

Not: ¬ simgesi tek bir bo  karakteri temsil eder.

Ad	Ondalık değ�r	Onaltılı değ�r
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 bi imlendirme  stbilgisi i aretleri

Ad	Ondalık değ�r	Onaltılı değ�r
MQRFH_NONE	0	X'00000000'
MQRFH_NO_FLAGS	0	X'00000000'

MQRFH2_* (Yayınlama/Abone Olma Seçenekleri Etiketi RFH2 En Üst Düzey Klasör Etiketleri)

Ad	Ondalık değer	Onaltılı değer
MQRFH2_NAME_VALUE_VERSION	1	X'00000001'

MQRFH2_* (Yayınlama/Abone Olma Seçenekleri Etiket Etiketi 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 Etiketi 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)

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

MQRMH_* (Başvuru iletisi üstbilgi yapısı)

Ad	Yapı
MQRMH_STRUC_ID	"RMH↯"
MQRMH_STRUC_ID_ARRAY	'R','M','H','↯'

Not: ↯ simgesi tek bir boş karakteri temsil eder.

Ad	Ondalık değer	Onaltılı değer
MQRMH_VERSION_1	1	X'00000001'
MQRMH_CURRENT_VERSION	1	X'00000001'

MQRMHF_* (Başvuru iletisi üstbilgi işaretleri)

Ad	Ondalık değer	Onaltılı değer
MQRMHF_SON	1	X'00000001'

Ad	Ondalık değ�r	Onaltılı değ�r
MQRMHF_NOT_SON	0	X'00000000'

MQRO_* (Rapor Seenekleri)

Ad	Ondalık değ�r	Onaltılı değ�r
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 Seenekleri Mister)

Ad	Ondalık değ�r	Onaltılı değ�r
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_ACTIVITIES)

Ad	Ondalık değer	Onaltılı değer
MQROUTE_UNLIMITED_ACTIVITIES	0	X'00000000'

Trace-route Detail (MQIACF_ROUTE_DETAIL)

Ad	Ondalık değer	Onaltılı değer
MQROUTE_AYRINTISI	2	X'00000002'
MQROUTE_DETAIL_MEDIA	8	X'00000008'
MQROUTE_AYRINTI_YÜKSEK	32	X'00000020'

Trace-route Forwarding (MQIACF_ROUTE_FORWARDING)

Ad	Ondalık değer	Onaltılı değer
MQROUTE_FORWARD_ALL	256	X'00000100'
MQROUTE_FORWARD_IF_SUPPORTED	512	X'00000200'
MQROUTE_FORWARD_REJ_UNSUP_MASK	-65536	X'FFFF0000'

İzleme Rotası Teslimi (MQIACF_ROUTE_DELIVERY)

Ad	Ondalık değer	Onaltılı değer
MQROUTE_DELIVER_YES	4096	X'00001000'
MQROUTE_DELIVER_NO	8192	X'00002000'
MQROUTE_DELIVER_REJ_UNSUP_MASK	-65536	X'FFFF0000'

Trace-Route Birikimi (MQIACF_ROUTE_BIRIKIM)

Ad	Ondalık değer	Onaltılı değer
MQROUTE_ACCUMULATE_NONE	65539	X'00010003'
MQROUT_ACCUMULATE_IN_MSG	65540	X'00010004'
MQROUTE_ACCUMULATE_AND_REPLY	65541	X'00010005'

MQRP_ * (Komut biçimi Değiştirme Seçenekleri)

Ad	Ondalık değer	Onaltılı değer
MQRP_YES	1	X'00000001'
MQRP_NO	0	X'00000000'

MQRQ_ * (Komut biçimi Neden Niteleyicileri)

Ad	Ondalık değer	Onaltılı değer
MQRQ_CONN_NOT_YETKILI	1	X'00000001'

Ad	Ondalık değ�r	Onaltılı değ�r
MQRQ_OPEN_NOT_YETKILI	2	X'00000002'
MQRQ_CLOSE_NOT_YETKILI	3	X'00000003'
MQRQ_CMD_NOT_YETKILI	4	X'00000004'
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'

MQRT_ * (Komut biçimi Yenileme Tipleri)

Ad	Ondalık değ�r	Onaltılı değ�r
MQRT_YAPILANIŞI	1	X'00000001'
MQRT_EXPIRY	2	X'00000002'
MQRT_NSPROC	3	X'00000003'
MQRT_PROXYSUB	4	X'00000004'

MQRU_ * (Yalnızca İstek)

Ad	Ondalık değer	Onaltılı değer
MQRU_PUBLISH_ON_REQUEST	1	X'00000001'
MQRU_YAYINLAYICI_	2	X'00000002'

MQSCA_ * (SSL İstemci Doğrulaması)

Ad	Ondalık değer	Onaltılı değer
MQSCA_REQUIREM	0	X'00000000'
MQSCA_OPTIONAL	1	X'00000001'

MQSCO_ * (SSL yapılandırma seçenekleri)

SSL yapılandırma seçenekleri yapısı

Ad	Yapı
MQSCO_STRUC_ID	"SCO"
MQSCO_STRUC_ID_ARRAY	'S','C','O',' '

Not: ' ' simgesi tek bir boş karakteri temsil eder.

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.

SSL yapılandırma seçenekleri Anahtarı İlk Duruma Getirme Sayısı

Ad	Ondalık değer	Onaltılı değer
MQSCO_RESET_COUNT_VARSAYILAN	0	X'00000000'

Komut Biçimi Kuyruk Tanımı Kapsamı

Ad	Ondalık değer	Onaltılı değer
MQSCO_Q_MGR	1	X'00000001'
MQSCO_CEL	2	X'00000002'

MQSCOPE_ * (Yayınlama kapsamı)

Ad	Ondalık değer	Onaltılı değer
MQSCOPE_ALL	0	X'00000000'

Ad	Ondalık değ�r	Onaltılı değ�r
MQSCOPE_AS_PARENT	1	X'00000001'
MQSCOPE_QMGR	4	X'00000004'

MQSCYC_* (G venlik Senaryosu)

Ad	Ondalık değ�r	Onaltılı değ�r
MQSCYC_UPPER	0	X'00000000'
MQSCYC_MIXED	1	X'00000001'

MQSD_* (Nesne tanımlayıcı yapısı)

Ad	Yapı
MQSD_STRUC_ID	"SD¬¬"
MQSD_STRUC_ID_ARRAY	'S','D','¬','¬'

Not: ¬ simgesi tek bir boş karakteri temsil eder.

Ad	Ondalık değ�r	Onaltılı değ�r
MQSD_VERSION_1	1	X'00000001'
MQSD_CURRENT_VERSION	1	X'00000001'

MQSECITEM_* (Komut biçimi G venlik   geleri)

Ad	Ondalık değ�r	Onaltılı değ�r
MQSECEĞİN_T�m�	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'

MQSECSW_* (Komut biçimi Güvenlik Anahtarları ve Anahtar Durumları)

Komut biçimi Güvenlik Anahtarları

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_RESOURCES	11	X'0000000B'
MQSECSW_Q_MGR	15	X'0000000F'
MQSECSW_QSG	16	X'00000010'

Komut biçimi Güvenlik Anahtarı Durumları

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)

Ad	Ondalık değer	Onaltılı değer
MQSECTYPE_AUTHSERV	1	X'00000001'
MQSECTYPE_SSL	2	X'00000002'
MQSECTYPE_CLASSICS	3	X'00000003'

MQSEG_* (Kesimlere ayırma)

Ad	Değer
MQSEG_INHIBITED	'¬'
MQSEG_ALLOWD	'A'

Not: ¬ simgesi tek bir boş karakteri temsil eder.

MQSEL_ * (Özel Seçici 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)

Ad	Ondalık değer	Onaltılı değer
MQSELTYPE_NONE	0	X'00000000'
MQSELTYPE_STANDARD	1	X'00000001'
MQSELTYPE_EXTENDED	2	X'00000002'

MQSID_ * (Güvenlik Tanıtıcısı)

Ad	Değer
MQSID_NONE	X'00...00' (40 boş değer)
MQSID_NONE_DIZISI	'\0', '\0', ... (40 boş değer)

MQSIDT_ * (Güvenlik Tanıtıcısı Tipleri)

Ad	Onaltılı değer
MQSIDT_NONE	X'00'
MQSIDT_NT_SECURITY_ID	X'01'
MQSIDT_WAS_SECURITY_ID	X'02'

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

Ad	Yapı
MQSMPO_STRUC_ID	"SMPO"
MQSMPO_STRUC_ID_ARRAY	'S', 'M', 'P', 'O'

Not: ↵ simgesi tek bir boş karakteri temsil eder.

Ad	Ondalık değ�r	Onaltılı değ�r
MQSMPO_VERSION_1	1	X'00000001'
MQSMPO_CURRENT_VERSION	1	X'00000001'

İleti Özelliđi Seçeneklerini Ayarla

Ad	Ondalık değ�r	Onaltılı değ�r
MQSMPO_SET_FIRST	0	X'00000000'
MQSMPPPO_SET_PROP_UNDER_CURSOR	1	X'00000001'
MQSMPPPO_SET_PROP_AFTER_CURSOR	2	X'00000002'
MQSMPO_APPEND_�ZELLİđİ	4	X'00000004'
MQSMPPPO_SET_PROP_BEFORE_CURSOR	8	X'00000008'
MQSMPO_NONE	0	X'00000000'

MQSO_ * (Abone Olma Seçenekleri)

Ad	Ondalık değ�r	Onaltılı değ�r
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'
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)

Ad	Ondalık değer	Onaltılı değer
MQSP_AVALABILIR	1	X'00000001'
MQSP_NOT_VAR	0	X'00000000'

MQSQQM_ * (Paylaşılan Kuyruk Kuyruğu Yöneticisi Adı)

Ad	Ondalık değer	Onaltılı değer
MQSQQM_USE	0	X'00000000'
MQSQQM_IGNORE	1	X'00000001'

MQSR_ * (İşlem)

Ad	Ondalık değer	Onaltılı değer
MQSR_ACTION_YAYINI	1	X'00000001'

MQSRO_ * (Abonelik isteği seçenekleri yapısı)

Ad	Yapı
MQSRO_STRUC_ID	"SRO¬"
MQSRO_STRUC_ID_ARRAY	'S','R','O','¬'

Not: ¬ simgesi tek bir boş karakteri temsil eder.

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)

Ad	Yapı
MQSS_NOT_A_KESIMI	'¬'
MQSS_SEGMENT	'S'
MQSS_LAST_SEGMENT	'L'

Not: ¬ simgesi tek bir boş karakteri temsil eder.

MQSSL_ * (SSL FIPS Gereksinimleri)

Ad	Ondalık değer	Onaltılı değer
MQSSL_FIPS_NO	0	X'00000000'
MQSSL_FIPS_YES	1	X'00000001'

MQSTAT_ * (Stat Seçenekleri)

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)

Ad	Yapı
MQSTS_STRUC_ID	"STAT"
MQSTS_STRUC_ID_ARRAY	'S','T','A','T'

Not: ↵ simgesi tek bir boş karakteri temsil eder.

Ad	Ondalık değer	Onaltılı değer
MQSTS_VERSION_1	1	X'00000001'
MQSTS_CURRENT_VERSION	1	X'00000001'

MQSUB_ * (Durabilen abonelikler)

Sürekli abonelikler

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

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)

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)

Ad	Ondalık değer	Onaltılı değer
MQSUS_YES	1	X'00000001'
MQSUS_NO	0	X'00000000'

MQSVC_ * (Hizmet)

Hizmet tipleri

Ad	Ondalık değer	Onaltılı değer
MQSVC_TYPE_COMMAND	0	X'00000000'
MQSVC_TYPE_SERVER	1	X'00000001'

Hizmet Denetimleri

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

Ad	Ondalık değer	Onaltılı değer
MQSVC_STATUS_DURDU	0	X'00000000'
MQSVC_STATUS_STARTING	1	X'00000001'
MQSVC_STATUS_RUNING	2	X'00000002'
MQSVC_STATUS_DURMA	3	X'00000003'
MQSVC_STATUS_RETRING	4	X'00000004'

MQSYNCPOINT_ * (Pub/Sub geçişine ilişkin komut biçimi Syncpoint değerleri)

Ad	Ondalık değer	Onaltılı değer
MQSYNCPOINT_YES	0	X'00000000'
MQSYNCPOINT_IFPER	1	X'00000001'

MQSYSP_ * (Komut biçimi Sistem Parametre 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'

Ad	Ondalık değ�r	Onaltılı değ�r
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'
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  znitelikleri)

jokerler

Ad	Ondalık değ�r	Onaltılı değ�r
MQTA_BLOCK	1	X'00000001'
MQTA_PASPASS	2	X'00000002'

 zin Verilen Abonelik

Ad	Ondalık değ�r	Onaltılı değ�r
MQTA_SUB_AS_PARENT	0	X'00000000'
MQTA_SUB_INHIBITED	1	X'00000001'
MQTA_SUB_ALLOWD	2	X'00000002'

Yetkili Sunucu Alt Yayma

Ad	Ondalık değ�r	Onaltılı değ�r
MQTA_PROXY_SUB_FORCE	1	X'00000001'
MQTA_PROXY_SUB_FIRSTUSE	2	X'00000002'

Yayına  zin Verilen

Ad	Ondalık değ�r	Onaltılı değ�r
MQTA_PUB_AS_PARENT	0	X'00000000'

Ad	Ondalık değ�r	Onaltılı değ�r
MQTA_PUB_INHIBITED	1	X'00000001'
MQTA_PUB_ALLOWD	2	X'00000002'

MQTC_* (Tetikleyici Denetimleri)

Ad	Ondalık değ�r	Onaltılı değ�r
MQTC_OFF	0	X'00000000'
MQTC_ON	1	X'00000001'

MQTCPKEEP_* (TCP Keepalive)

Ad	Ondalık değ�r	Onaltılı değ�r
MQTCPKEEP_NO	0	X'00000000'
MQTCPKEEP_YES	1	X'00000001'

MQTCPSTACK_* (TCP YıĖın Tipleri)

Ad	Ondalık değ�r	Onaltılı değ�r
MQTCPSTACK_SATIR	0	X'00000000'
MQTCPSTACK_MULTIPLE	1	X'00000001'

MQTIME_* (Komut bi imi zaman birimleri)

Ad	Ondalık değ�r	Onaltılı değ�r
MQTIME_UNIT_MINS	0	X'00000000'
MQTIME_UNIT_SECS	1	X'00000001'

MQTM_* (İleti yapısını tetikle)

Ad	Yapı
MQTM_STRUC_ID	"TM↵"
MQTM_STRUC_ID_ARRAY	'T','M','↵','↵'

Not: ↵ simgesi tek bir boş karakteri temsil eder.

Ad	Ondalık değ�r	Onaltılı değ�r
MQTM_VERSION_1	1	X'00000001'
MQTM_CURRENT_VERSION	1	X'00000001'

MQTMC_* (Tetikleyici ileti karakter bi imi yapısı)

Ad	Yapı
MQTMC_STRUC_ID	"TMC↵"
MQTMC_STRUC_ID_ARRAY	'T','M','C','↵'

Ad	Yapı
MQTMC_VERSION_1	"-1"
MQTMC_VERSION_2	"-2"
MQTMC_CURRENT_VERSION	"-2"
MQTMC_VERSION_1_ARRAY	'-', '-', '-', '1'
MQTMC_VERSION_2_ARRAY	'-', '-', '-', '2'
MQTMC_CURRENT_VERSION_ARRAY	'-', '-', '-', '2'

MQTOPT_* (Konu Tipi)

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)

Ad	Ondalık değer	Onaltılı değer
MQTRAXSTR_NO	0	X'00000000'
MQTRAXSTR_YES	1	X'00000001'

MQTScope_* (Abonelik Kapsamı)

Ad	Ondalık değer	Onaltılı değer
MQTScope_QMGR	1	X'00000001'
MQTScope_ALL	2	X'00000002'

MQTT_* (Tetikleme Tipleri)

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)

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'

Ad	Ondalık değ�r	Onaltılı değ�r
MPIOT_RRS	2	X'00000002'
MQUOWT_IMS	3	X'00000003'
MQUOWT_XA	4	X'00000004'

MQUS_* (Kuyruk Kullanımları)

Ad	Ondalık değ�r	Onaltılı değ�r
MQUS_NORMAL	0	X'00000000'
MQUS_ILETIMI	1	X'00000001'

MQUSAGE_* (Komut bi imi Sayfa K mesi Kullanım Değ rleri ve Veri K mesi Kullanım Değ rleri)

Komut Bi imi Sayfa K mesi Kullanım Değ rleri

Ad	Ondalık değ�r	Onaltılı değ�r
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 bi imi Veri K mesi Kullanım Değ rleri

Ad	Ondalık değ�r	Onaltılı değ�r
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_* (Değ r Uzunluęu)

Ad	Ondalık değ�r	Onaltılı değ�r
MVQL_NULL_TERMINATED	-1	X'FFFFFFFF'
MQVL_EMPTY_STRING	0	X'00000000'

MQVU_* (Değ şken Kullanıcı Kimlięi)

Ad	Ondalık değ�r	Onaltılı değ�r
MQVU_FIXED_USER	1	X'00000001'

Ad	Ondalık değ�r	Onaltılı değ�r
MVQU_ANY_USER	2	X'00000002'

MQWDR_* (K me iř y k   ıkıř hedefi kayıt yapısı)

Ad	Yapı
MQWDR_STRUC_ID	"WDR�"
MQWDR_STRUC_ID_ARRAY	'W','D','R','�'

Not:   simgesi tek bir boř karakteri temsil eder.

Ad	Ondalık değ�r	Onaltılı değ�r
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 Aralıęı)

Ad	Ondalık değ�r	Onaltılı değ�r
MQWI_UNSI�MA	-1	X'FFFFFFFF'

MQWIH_* (İř y k  bilgisi  stbilgi yapısı ve iřaretleri)

İř y k  bilgileri  stbilgi yapısı

Ad	Yapı
MQWIH_STRUC_ID	"WIH�"
MQWIH_STRUC_ID_ARRAY	'W','I','H','�'

Not:   simgesi tek bir boř karakteri temsil eder.

Ad	Ondalık değ�r	Onaltılı değ�r
MQWIH_VERSION_1	1	X'00000001'
MQWIH_CURRENT_VERSION	1	X'00000001'
MQWIH_LENGTH_1	120	X'00000078'
MQWIH_CURRENT_LENGTH	120	X'00000078'

İř y k  bilgileri  stbilgi iřaretleri

Ad	Ondalık değ�r	Onaltılı değ�r
MQWIH_NONE	0	X'00000000'

MQWQR_ * (Küme iş yükü çıkış kuyruğu kayıt yapısı)

Ad	Yapı
MQWQR_STRUC_ID	"WQR¬"
MQWQR_STRUC_ID_ARRAY	'W', 'Q', 'R', '¬'

Not: ¬ simgesi tek bir boş karakteri temsil eder.

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)

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)

Ad	Yapı
MQWXP_STRUC_ID	"WXP¬"
MQWXP_STRUC_ID_ARRAY	'W', 'X', 'P', '¬'

Not: ¬ simgesi tek bir boş karakteri temsil eder.

Ad	Ondalık değer	Onaltılı değer
MQWXP_VERSION_1	1	X'00000001'
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)

Ad	Ondalık değer	Onaltılı değer
MQWXP_PUT_BY_CLUSTER_CHL	2	X'00000002'

İlgili başvurular

“MQWXP içindeki alanlar-Küme iş yükü çıkış değiştirgesi yapısı” sayfa 1557

MQWXP -Cluster iş yükü çıkış değiştirgesi yapısındaki alanların tanımı

MQXACT_ * (API Caller Tipleri)

Ad	Ondalık değer	Onaltılı değer
MQXACT_EXTERNAL	1	X'00000001'
MQXACT_INTERNAL	2	X'00000002'

MQXC_ * (Çıkış Komutları)

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)

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'
MQXCC_SUPRESS_EXIT	-5	X'FFFFFFFB'
MQXCC_CLOSE_CHANNEL	-6	X'FFFFFFFA'
MQXCC_REQUEST_ACK	-7	X'FFFFFFF9'
MQXCC_FAILED	-8	X'FFFFFFF8'

MQXDR_* (Çıkış Yanıtı)

Ad	Ondalık değer	Onaltılı değer
MQXDR_OK	0	X'00000000'
MQXDR_CONVERSION_FAILED	1	X'00000001'

MQXE_* (Ortamlar)

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

Ad	Yapı
MQXEPO_STRUCT_ID	"XEPO"
MQXEPO_STRUC_ID_ARRAY	'X','E','P','O'

Not: ↵ simgesi tek bir boş karakteri temsil eder.

Ad	Ondalık değer	Onaltılı değer
MQXEPO_VERSION_1	1	X'00000001'
MQXEPO_CURRENT_VERSION	1	X'00000001'

Çıkış Seçenekleri

Ad	Ondalık değer	Onaltılı değer
MQXEPO_NONE	0	X'00000000'

MQXF_* (API İşlev Tanıtıcıları)

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'

Ad	Ondalık değ�r	Onaltılı değ�r
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_CMtT	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_XARLOLARI	32	X'00000020'
MQXF_XASTART	33	X'00000021'
MQXF_AXREG	34	X'00000022'
MQXF_AXUNREG	35	X'00000023'

MQXP_ * (API geçiř ıkıř parametresi yapısı)

Ad	Yapı
MQXP_STRUC_ID	"XP¬¬"
MQXP_STRUC_ID_ARRAY	'X','P','¬','¬'

Not: ¬ simgesi tek bir boş karakteri temsil eder.

Ad	Ondalık değ�r	Onaltılı değ�r
MQXP_VERSION_1	1	X'00000001'

MQXPDA_* (Sorun Saptama Alanı)

Ad	Değer
MQXPDA_NONE	X'00...00' (48 boş değer)
MQXPDA_NONE_DIZISI	'\0', '\0', ... (48 boş değer)

MQXPT_* (İletim Tipleri)

Ad	Ondalık değer	Onaltılı değer
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 kuyruğu üstbilgi yapısı)

Ad	Yapı
MQXQH_STRUC_ID	"XQH-"
MQXQH_STRUC_ID_ARRAY	'X', 'Q', 'H', '-'

Not: - simgesi tek bir boş karakteri temsil eder.

Ad	Ondalık değer	Onaltılı değer
MQXQH_VERSION_1	1	X'00000001'
MQXQH_CURRENT_VERSION	1	X'00000001'

MQXR_* (Exit Reasons)

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'

Ad	Ondalık değ�r	Onaltılı değ�r
MQXR_RETRY	17	X'00000011'
MQXR_AUTO_CLUSSDR	18	X'00000012'
MQXR_AUTO_RECEIVER	19	X'00000013'
MQXR_CLWL_ASİ	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_PARMS	29	X'0000001D'

MQXR2_* (Çıkış Yanıt 2)

Ad	Ondalık değ�r	Onaltılı değ�r
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)

Ad	Ondalık değ�r	Onaltılı değ�r
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'

Ad	Ondalık değ�r	Onaltılı değ�r
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 Değ rinden  ık)

Ad	Değ�r
MQXUA_NONE	X'00...00' (16 boş değ�r)
MQXUA_NONE_DIZISI	'\0', '\0', ... (16 boş değ�r)

MQXWD_* ( ıkış bekleme tanımlayıcı yapısı)

Ad	Yapı
MQXWD_STRUC_ID	"XWD�"
MQXWD_STRUC_ID_ARRAY	'X', 'W', 'D', '�'

Not:   simgesi tek bir boş karakteri temsil eder.

Ad	Ondalık değ�r	Onaltılı değ�r
MQXWD_VERSION_1	1	X'00000001'

MQZAC_* (Uygulama bağlamı yapısı)

Ad	Yapı
MQZAC_STRUC_ID	"ZAC�"
MQZAC_STRUC_ID_ARRAY	'Z', 'A', 'C', '�'

Not:   simgesi tek bir boş karakteri temsil eder.

Ad	Ondalık değ�r	Onaltılı değ�r
MQZAC_VERSION_1	1	X'00000001'
MQZAC_CURRENT_VERSION	1	X'00000001'

MQZAD_* (Yetki verileri yapısı)

Ad	Yapı
MQZAD_STRUC_ID	"ZAD�"
MQZAD_STRUC_ID_ARRAY	'Z', 'A', 'D', '�'

Not:   simgesi tek bir boş karakteri temsil eder.

Ad	Ondalık değ�r	Onaltılı değ�r
MQZAD_VERSION_1	1	X'00000001'
MQZAD_VERSION_2	2	X'00000002'
MQZAD_CURRENT_VERSION	2	X'00000002'

MQZAET_* (Kurulabilir Hizmetler Varlık Tipleri)

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)

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'
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)

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)

Ad	Ondalık değer	Onaltılı değer
MQZAT_INITIAL_CONTEXT	0	X'00000000'
MQZAT_CHANGE_CONTEXT	1	X'00000001'

MQZCI_ * (Kurulabilir Hizmetler Devamı Göstergesi)

Ad	Ondalık değer	Onaltılı değer
MQZCI_VARSAYILAN	0	X'00000000'
MQZCI_CONTINY	0	X'00000000'
MQZCI_STOP	1	X'00000001'

MQZED_ * (Varlık veri yapısı)

Ad	Yapı
MQZED_STRUC_ID	"ZED¬"
MQZED_STRUC_ID_ARRAY	'Z','E','D','¬'

Not: ¬ simgesi tek bir boş karakteri temsil eder.

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)

Ad	Yapı
MQZFP_STRUC_ID	"ZFP¬"
MQZFP_STRUC_ID_ARRAY	'Z','F','P','¬'

Not: ¬ simgesi tek bir boş karakteri temsil eder.

Ad	Ondalık değ�r	Onaltılı değ�r
MQZFP_VERSION_1	1	X'00000001'
MQZFP_CURRENT_VERSION	1	X'00000001'

MQZIC_* (Kimlik baėlamı yapısı)

Ad	Yapı
MQZIC_STRUC_ID	"ZIC¬"
MQZIC_STRUC_ID_ARRAY	'Z','I','C','¬'

Not: ¬ simgesi tek bir boş karakteri temsil eder.

Ad	Ondalık değ�r	Onaltılı değ�r
MQZIC_VERSION_1	1	X'00000001'
MQZIC_CURRENT_VERSION	1	X'00000001'

MQZID_* (hizmetlere ilişkin iřlev tanıtıcıları)

T m hizmetler i in ortak olan iřlev tanıtıcıları

Ad	Ondalık değ�r	Onaltılı değ�r
MQZID_INIT	0	X'00000000'
MQZID_TERM	1	X'00000001'

Yetki hizmeti i in iřlev tanıtıcıları

Ad	Ondalık değ�r	Onaltılı değ�r
MQZID_INIT_AUTHORITY	0	X'00000000'
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_A�IK_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

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

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)

Ad	Ondalık değer	Onaltılı değer
MQZI_BIRINCIL	0	X'00000000'
MQZIO_IKINCIL	1	X'00000001'

MQZNS_ * (Ad Hizmet Arabirimi Sürümü)

Ad	Ondalık değer	Onaltılı değer
MQZNS_VERSION_1	1	X'00000001'

MQZSE_ * (Kurulabilir Hizmetler Başlangıcı-Sıralama Göstergesi)

Ad	Ondalık değer	Onaltılı değer
MQZSE_START	1	X'00000001'
MQZSE_DEVAM	0	X'00000000'

MQZSL_ * (Kurulabilir Hizmetler Seçici Göstergesi)

Ad	Ondalık değer	Onaltılı değer
MQZSL_NOT_RETURNED	0	X'00000000'
MQZSL_RETURNED	1	X'00000001'

MQZTO_ * (Kurulabilir Hizmetler Sonlandırma Seçenekleri)

Ad	Ondalık değer	Onaltılı değer
MQZTO_BIRINCIL	0	X'00000000'
MQZTO_IKINCIL	1	X'00000001'

MQZUS_ * (Kullanıcı Kimliği Hizmet Arabirimi Sürümü)

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 'da kullanılan veri tiplerinin tanıtılması

Bu bölümde, MQI ' da kullanılan veri tipleri tanıtılır ve desteklenen programlama dillerinde kullanmanız için size bazı yönergeler verilir.

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 türü	Tanım
MQBOOL	Boole	MQBOOL veri tipi bir boole değeri temsil eder. 0 değeri false değerini gösterir. Başka bir değer, true değerini gösterir. MQH00L, MQHOT veri tipi için aynı şekilde hizalanmalıdır.

Temel veri tipi adı	Veri türü	Tanım
MQBYTE	Byte	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. Farklı karakter kümeleri ya da kodlamalar kullanan kuyruk yöneticileri arasında MQBYTE verileri gönderildiğinde, MQBYTE verileri hiçbir şekilde <i>dönüştürülemez</i>. MQMD yapısındaki <i>MsgId</i> ve <i>CorrelId</i> alanları şöyle olur. 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. 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.

Temel veri tipi adı	Veri türü	Tanım
MQBYTEN	n bayta ilişkin dizgi	Her MQBYTEN veri tipi n byte dizgisini gösterir; burada n, şu değerlerden herhangi birini alabilir: 8, 16, 24, 32, 40 ya da 128. Her bayt, MQBYTE veri tipi tarafından tanımlanır. Özel hizalama gerekmez. Bayt dizilimindeki veriler, dizilimin tanımlı uzunluğundan daha kısaysa, dizgiyi doldurmak için veriler boş değerlerle doldurulmalıdır. Kuyruk yöneticisi uygulamaya bayt dizgileri döndürdüğünde (örneğin, MQGET çağrısında), kuyruk yöneticisi altlıkları, dizginin tanımlı uzunluğuna kadar boş değere sahip olur. Bayt dizgi alanlarının uzunluklarını tanımlamak için, adlandırılan sabitler kullanılabilir. Bunlar “Değişmezler” sayfa 61’inde listelenir.
MQCHAR	Karakter	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. 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 “MQGMO-Get-message seçenekleri” sayfa 358 ' ta bu seçeneğin tanımına bakın.

Temel veri tipi adı	Veri türü	Tanım
MQCHARN	n karakter dizgisi	Her MQCHARn veri tipi bir n karakteri dizesini gösterir; burada n , 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. 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. 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 ve “Değişmezler” sayfa 61’inde yer alan, adlandırılmış sabitler kullanılabilir.
MQFLOAT32	32 bitlik kayan noktalı sayı	MQFLOAT32 veri tipi, standart IEEE kayar noktalı biçim kullanılarak gösterilen, 32 bitlik bir kayan noktalı sayıdır. Bir MQFLOAT32 , 4 baytlık bir sınır üzerinde hizalanmalıdır. The use of MQFLOAT32 in C on z/OS requires the use of the FLOAT(IEEE) compiler flag. COBOL ' de MQFLOAT32 kullanımı, IEEE biçimindeki kayan noktalı sayıları destekleyen derleyiciler ile sınırlandırılır. Bu, FLOAT (YEREL) derleyici işaretinin kullanılmasını gerektirebilir.

Temel veri tipi adı	Veri türü	Tanım
MQFLOAT64	64 bitlik kayan noktalı sayı	MQFLOAT64 veri tipi, standart IEEE kayar noktalı biçim kullanılarak gösterilen, 64 bitlik kayan noktalı bir sayıdır. Bir MQFLOAT64 , 8 baytlık bir sınır üzerinde hizalanmalıdır. The use of MQFLOAT64 in C on z/OS requires the use of the FLOAT(IEEE) compiler flag. COBOL ' de MQFLOAT64 kullanımı, IEEE formatındaki kayan noktalı sayıları destekleyen derleyiciyle sınırlıdır. Bu, FLOAT (YEREL) derleyici işaretinin kullanılmasını gerektirebilir.
MQHCONFIG	Yapılandırma tutamacı	MQHCONFIG veri tipi, belirli bir kurulabilir hizmet için yapılandırılmakta olan bir yapılanış tanıtıcısını (yani, bileşen) gösterir. Bir yapılandırma tanıtıcısı, doğal sınırı üzerinde hizalanmalıdır. 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.
MQHCONN	Bağlantı tanıtıcı	MQHCONN veri tipi bir bağlantı tanıtıcısını, yani belirli bir kuyruk yöneticisiyle olan bağlantıyı gösterir. Bir bağlantı tanıtıcısı, 4 baytlık bir sınır üzerinde hizalanmalıdır. 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.

Temel veri tipi adı	Veri türü	Tanım
MQHMSG	İleti tanıtıcısı	MQHMSG veri tipi, bir iletiye erişim veren bir ileti tanıtıcısını gösterir. Bir ileti tanıtıcısı, 8 byte 'lık bir sınırdan hizalanmalıdır. 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.
MQHOBJS	Nesne tanıtıcısı	MQHOBJS veri tipi, bir nesneye erişim veren bir nesne tanıtıcısını gösterir. Bir nesne tanıtıcısı, 4 baytlık bir sınır üzerinde hizalanmalıdır. 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.
MQINT8	8 bitlik işaretli 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.
MQINT16	16 bitlik işaretli 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.
MQINT32	32 bitlik işaretli tamsayı	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. MQLONG' un tanımına bakın.

Temel veri tipi adı	Veri türü	Tanım
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 işaretsiz 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. MQULONG tanımına bakın.
MQUINT64	64 bit işaretsiz 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 sınırlandırılır. 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 sınırlandırılır. Bir MQULONG, 4 baytlık bir sınır üzerinde hizalanmalıdır.
PMQACH	İşaretçi	MQACH tipinde bir veri yapısına ilişkin gösterge
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

Temel veri tipi adı	Veri türü	Tanım
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	İmlecin 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
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

Temel veri tipi adı	Veri türü	Tanım
PMQFLOAT32	İşaretçi	MQFLOAT32tipinde bir veri yapısına ilişkin gösterge
PMQFLOAT64	İşaretçi	MQFLOAT64tipinde bir veri yapısına ilişkin gösterge
PMQFUNC	İşaretçi	İşleve ilişkin gösterge
PMQGMO	İşaretçi	MQGMO 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
PMQHOBJ	İşaretçi	MQHOBJ 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	İmlecin veri tipi MQUBE
PMQMD	İşaretçi	İmlecin 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
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

Temel veri tipi adı	Veri türü	Tanım
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	MQRFH2tipinde 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	MQTM2tipinde bir veri yapısına işaret eden gösterge
PMQUINT8	İşaretçi	MQUINT8veri tipine işaretçi
PMQUINT16	İşaretçi	MQUINT16veri tipine işaretçi
PMQUINT32	İşaretçi	MQUINT32veri tipine işaret eden gösterge
PMQUINT64	İşaretçi	MQUINT64veri tipine işaret eden gösterge
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

Veri türü	Gösterim
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>
MQCHAR28	<pre>typedef MQCHAR MQCHAR28[28];</pre>
MQCHAR32	<pre>typedef MQCHAR MQCHAR32[32];</pre>
MQCHAR48	<pre>typedef MQCHAR MQCHAR48[48];</pre>
MQCHAR64	<pre>typedef MQCHAR MQCHAR64[64];</pre>
MQCHAR128	<pre>typedef MQCHAR MQCHAR128[128];</pre>

Veri türü	Gösterim
MQCHAR256	<pre>typedef MQCHAR MQCHAR256[256];</pre>
MQFLOAT32	<pre>typedef float MQFLOAT32;</pre>
MQFLOAT64	<pre>typedef double MQFLOAT64;</pre>
MQHCONFIG	<pre>typedef void MQPOINTER MQHCONFIG;</pre>
MQHCONN	<pre>typedef MQLONG MQHCONN;</pre>
MQHOBJ	<pre>typedef MQLONG MQHOBJ;</pre>
MQINT8	<pre>typedef signed char MQINT8;</pre>
MQINT16	<pre>typedef short MQINT16;</pre>
MQINT64	64 bit UNIX sistemleri üzerinde: <pre>typedef long;</pre> 32 bit AIX, Solaris ve HP-UX üzerinde: <pre>typedef int64_t;</pre> IBM i, Linux ve z/OS üzerinde: <pre>typedef long long;</pre> Açık Windows: <pre>typedef _int64;</pre>
MQLONG	Açık IBM i: <pre>typedef long MQLONG;</pre> diğer platformlar: <pre>if defined(MQ_64_BIT) typedef int MQLONG; else typedef long MQLONG;</pre>

Veri türü	Gösterim
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>
MQUINT64	64 bit UNIX sistemleri üzerinde: <pre>typedef unsigned long;</pre> 32 bit AIX, Solaris ve HP-UX üzerinde <pre>typedef uint64_t;</pre> IBM i, Linux ve z/OS üzerinde <pre>typedef unsigned long long;</pre> Açık Windows: <pre>typedef unsigned _int64;</pre>
MQULONG	Açık IBM i: <pre>typedef unsigned long MQULONG;</pre> diğer platformlar: <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>

Veri türü	Gösterim
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>
PMQBYTE32	<pre>typedef MQBYTE32[32] MQPOINTER PMQBYTE32[32];</pre>
PMQBYTE40	<pre>typedef MQBYTE40[40] MQPOINTER PMQBYTE40[40];</pre>
PMQBYTE128	<pre>typedef MQBYTE128[128] MQPOINTER PMQBYTE128[128];</pre>
PMQCHAR	<pre>typedef MQCHAR MQPOINTER PMQCHAR;</pre>
PMQCHAR4	<pre>typedef MQCHAR4[4] MQPOINTER PMQCHAR4[4];</pre>
PMQCHAR8	<pre>typedef MQCHAR8[8] MQPOINTER PMQCHAR8[8];</pre>
PMQCHAR12	<pre>typedef MQCHAR12[12] MQPOINTER PMQCHAR12[12];</pre>
PMQCHAR20	<pre>typedef MQCHAR20[20] MQPOINTER PMQCHAR20[20];</pre>
PMQCHAR28	<pre>typedef MQCHAR28[28] MQPOINTER PMQCHAR28[28];</pre>
PMQCHAR32	<pre>typedef MQCHAR32[32] MQPOINTER PMQCHAR32[32];</pre>
PMQCHAR48	<pre>typedef MQCHAR48[48] MQPOINTER PMQCHAR48[48];</pre>
PMQCHAR64	<pre>typedef MQCHAR64[64] MQPOINTER PMQCHAR64[64];</pre>
PMQCHAR128	<pre>typedef MQCHAR128[128] MQPOINTER PMQCHAR128[128];</pre>
PMQCHAR256	<pre>typedef MQCHAR256[256] MQPOINTER PMQCHAR256[256];</pre>

Veri türü	Gösterim
PMQCHAR264	<code>typedef MQCHAR264[264] MQPOINTER PMQCHAR264[264];</code>
PMQCIH	<code>typedef MQCIH MQPOINTER PMQCIH;</code>
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>
PMQHOBj	<code>typedef MQHOBj MQPOINTER PMQHOBj;</code>
PMQIL	<code>typedef MQIIH MQPOINTER PMQIIH;</code>
PMQINT8	<code>typedef MQINT8 MQPOINTER PMQINT8;</code>
PMQINT16	<code>typedef MQINT16 MQPOINTER PMQINT16;</code>
PMQUZA	<code>typedef MQLONG MQPOINTER PMQLONG;</code>
PMQMD	<code>typedef MQMD MQPOINTER PMQMD;</code>
PMQMD1	<code>typedef MQMD1[1] MQPOINTER PMQMD1[1];</code>

Veri türü	Gösterim
PMQMDE	<code>typedef MQMDE MQPOINTER PMQMDE;</code>
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 MQTMC2[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>

Veri türü	Gösterim
PPMQCHAR	<code>typedef PMQCHAR MQPOINTER PPMQCHAR;</code>
PPMQCNO	<code>typedef PMQCNO MQPOINTER PPMQCNO;</code>
PPMQGMO	<code>typedef PMQGMO MQPOINTER PPMQGMO;</code>
PPMQHCONN	<code>typedef PMQHCONN MQPOINTER PPMQHCONN;</code>
PPMQHOBJ	<code>typedef PMQHOBJ MQPOINTER PPMQHOBJ;</code>
PPMQLONG	<code>typedef PMQLONG MQPOINTER PPMQLONG;</code>
PPMQMD	<code>typedef PMQMD MQPOINTER PPMQMD;</code>
PPMQOD	<code>typedef PMQOD MQPOINTER PPMQOD;</code>
PPMQPMO	<code>typedef PMQPMO MQPOINTER PPMQPMO;</code>
PPMQULONG	<code>typedef PMQULONG MQPOINTER PPMQULONG;</code>
PPMQVOID	<code>typedef PMQVOID MQPOINTER PPMQVOID;</code>
Burada defined (MQ_64_BIT) , 64 bitlik bir platform anlamına gelir.	

MQPOINTER makro değişkeninin tanımı için [“Veri tipleri” sayfa 263](#) konusuna bakın.

COBOL bildirimleri

Veri türü	Gösterim
MQBOOL	<code>PIC S9(9) BINARY</code>
MQBYTE	<code>PIC X</code>
MQBYTE8	<code>PIC X(8)</code>

Veri türü	Gösterim
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)
MQFLOAT32	USAGE COMP-1
MQFLOAT64	USAGE COMP-2

Veri türü	Gösterim
MQHCONN	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.

Veri türü	Gösterim
MQBOOL	fixed bin(31)
MQBYTE	char(1)
MQBYTE8	char(8)
MQBYTE16	char(16)

Veri türü	Gösterim
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)
MQCHAR128	char(128)
MQCHAR256	char(256)
MQFLOAT32	binary float(21) ieee
MQFLOAT64	binary float(52) ieee
MQHCONN	fixed bin(31)

Veri türü	Gösterim
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.

Veri türü	Gösterim
MQBOOL	DS F
MQBYTE	DS XL1
MQBYTE8	DS XL8
MQBYTE16	DS XL16
MQBYTE24	DS XL24

Veri türü	Gösterim
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
MQCHAR128	DS CL128
MQCHAR256	DS CL256
MQFLOAT32	DS EB
MQFLOAT64	DS DB
MQHCONN	DS F
MQHOBJ	DS F

Veri türü	Gösterim
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-giriş

Bu bölümde, MQI 'da kullanılan yapı veri tipleri ele alınmıştır. Yapı veri tipleri, sonraki bölümlerde anlatılır.

Özet

Aşağıdaki çizelgeler, MQI ' da kullanılan yapı veri tiplerini özetlemektedir.

Çizelge 7. MQI çağrılarında kullanılan yapı veri tipleri (ya da çıkış işlevleri):		
Yapı	Tanım	Kullanıldığı yerler
MQACH	API çıkış zinciri üstbilgisi	
MQAIR	Kimlik doğrulama bilgileri kaydı	MQCONN
MQAXC	API çıkış bağlamı	
MQAXP	API çıkış parametresi	
MQBMHO	Arabellek-ileti tanıtıcısı seçenekleri	MQBUFMH
MQBO	Başlangıç seçenekleri	MQBEGIN
MQCBD	Geri Çağırma	MQCB
MQCBO	Paket oluşturma seçenekleri	mqCreateÇanta
MQCHARV	Değişken uzunluklu dizgi	MQINQMP

Çizelge 7. MQI çağrılarında kullanılan yapı veri tipleri (ya da çıkış işlevleri): (devamı var)

Yapı	Tanım	Kullanıldığı yerler
MQCNO	Bağlanma seçenekleri	MQCONN
MQCSP	Güvenlik değiştiricileri	MQCONN
MQCTLO	Geri Arama Seçenekleri	MQCTL
MQDMPO	İleti özelliği seçeneklerini sil	MQDLTMP
MQGMO	Get-message seçenekleri	MQGet
MQIMPO	Sorgu iletisi özellik seçenekleri	MQINQMP
Mqmd	İleti tanımlayıcısı	MQBUFMH , MQMHBUF , MQCB , MQGET , MQPUT , MQPUT1
MQMHBO	İleti tanıtıcısı-arabellek seçenekleri	MQMHBUF
MQOD	Nesne tanımlayıcısı	MQOPEN , MQPUT1
MQOR	Nesne kaydı	MQOPEN , MQPUT1
MQPD	Özellik tanımlayıcısı	MQSETMP
MQPMO	Put-message seçenekleri	MQPUT , MQPUT1
MQPMR	Put-message kaydı	MQPUT , MQPUT1
MQRR	Yanıt kaydı	MQOPEN , MQPUT , MQPUT1
MQSCO	SSL yapılandırma seçenekleri	MQCONN
MQSD	Abonelik tanımlayıcısı	MQSUB
MQSMPO	İleti özelliği seçeneğini belirle	MQSETMP
MQSRO	Abonelik isteği seçenekleri	MQSUBRQ
MQSTS	Durum raporlama yapısı	MQSTAT

Çizelge 8. İleti verilerinde kullanılan yapı veri tipleri:

Yapı	Tanım
MQCIH	CICS bilgi üstbilgisi
MQCFH	PCF üstbilgisi
MQEPH	Gömülü PCF üstbilgisi
MQDH	Dağıtım üstbilgisi
MQDLH	Ölü harf (teslim edilmemiş ileti) üstbilgisi
MQIIH	IMS bilgi üstbilgisi
MQMDE	İleti tanımlayıcı uzantısı
MQRFH	Kurallar ve biçimlendirme üstbilgisi
MQRFH2	Kurallar ve biçimlendirme üstbilgisi 2
MQRMH	Başvuru iletisi üstbilgisi
MQTM	Tetikleme iletisi

Çizelge 8. İleti verilerinde kullanılan yapı veri tipleri: (devamı var)

Yapı	Tanım
<u>MQTMC2</u>	Tetikleme iletisi (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 çağrıları ile birlikte “Veri dönüştürme” sayfa 879’ünde açıklanmaktadır.

Yapı veri tiplerine ilişkin kurallar

Programlama dilleri, yapılara ilişkin destek düzeylerine göre değişir ve belirli kurallar ve kurallar, her bir programlama dilinde tutarlı bir şekilde MQI yapılarının eşlenmesi için benimsenir:

1. Yapıların doğal sınırları üzerinde hizalanması gerekir.
 - Birçok MQI yapısı için 4 byte 'lık hizalama gerekir.
 - IBM üzerinde, işaretçileri içeren yapılar 16 baytlık hizalama gerektirir; bunlar şunlardır: MQCNO, MQOD, MQPMO.
2. Bir yapıdaki her alan, doğal sınırları üzerinde hizalanmalıdır.
 - MQlong 'a denk gelen veri türlerine sahip alanlar 4 baytlık sınırlar üzerinde hizalanmalıdır.
 - MQPTR 'ye denk gelen veri tiplerine sahip alanlar, IBM üzerindeki 16 baytlık sınırlar ve diğer ortamlardaki 4 baytlık sınırlar üzerinde hizalanmalıdır.
 - Diğer alanlar 1 baytlık sınırlar ile hizalanmış olur.
3. Bir yapının uzunluğu, sınır hizalamasının bir katı olmalıdır.
 - Birçok MQI yapısı, 4 baytlık katları olan uzunluklara sahiptir.
 - IBM üzerinde, işaretçileri içeren yapılar 16 bayttan oluşan katları olan uzunluklara sahiptir.
4. Yukarıda belirtilen kurallara uygunluğu sağlamak için gereken yerlerde doldurma byte 'ları ya da alanları eklenmelidir.

Tanımlarda kullanılan kurallar

Her bir yapı veri tipine ilişkin açıklama şunları içerir:

- Yapının amacına ve kullanımına genel bakış
- Yapıdaki alanların açıklamalarına, programlama dilinden bağımsız bir biçimde
- Desteklenen programlama dillerinin her birinde yapının nasıl bildirilmiş olduğuna ilişkin örnekler

Her yapı veri tipinin tanımı aşağıdaki bölümleri içerir:

Yapı adı

Yapının adı, ardından yapıdaki alanların bir özeti izler.

Genel Bakış

Yapının amacına ve kullanımına ilişkin kısa bir açıklama.

Alanlar

Alanların tanımları. Her alan için, alanın adı, parantez içindeki temel veri tipi () ile izlenilir. Metinde, alan adları italik yazı tipi kullanılarak gösterilir; örneğin, *Version*.

Alanın alabileceği herhangi bir değer listesiyle birlikte alanın amacına ilişkin bir açıklama da vardır. Değişmezlerin adları büyük harfli olarak gösterilir; örneğin, MQGMO_STRUC_ID. * karakteri kullanılarak aynı öneke sahip bir sabitler kümesi gösterilir; örneğin: MQIA_.*.

Alanların tanımlarında, aşağıdaki terimler kullanılır:

Giriş

Bir arama yaparken alana bilgi sağladınız.

output

Kuyruk yöneticisi, çağrı tamamlandığında ya da başarısız olduğunda alana bilgi döndürür.

giriş/çıkış

Bir arama yaparken alana bilgi sağlarmısınız ve kuyruk yöneticisi, arama tamamlanınca ya da başarısız olduğunda bilgileri değiştirir.

Başlangıçtaki değerler

MQI ile birlikte verilen veri tanımı dosyalarındaki her alana ilişkin ilk değerleri gösteren bir çizelge.

C bildirimi

C ' deki yapının tipik açıklaması.

COBOL bildirimi

COBOL ' de tipik bir yapı bildirimi.

PL/I bildirimi

PL/I. ' daki yapının tipik açıklaması.

System/390 çevirici bildirimi

System/390 çevirici dilindeki yapıdaki tipik bildirim.

Visual Basic bildirimi

Visual Basic 'te yapılan yapıyla ilgili tipik bildirim.

C programlama

Bu bölümde, C programlama dilindeki MQI ' yi kullanmanıza yardımcı olacak bilgiler yer alır.

Üstbilgi dosyaları

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

Bu üstbilgi dosyaları [Çizelge 9 sayfa 262](#) içinde özetlenir.

Çizelge 9. C üstbilgi dosyaları	
Dosya	İçindekiler
CMQC	Ana MQI için işlev prototipleri, veri tipleri ve adlandırılmış değişmezler
CMQXC	Veri dönüştürme çıkışı için işlev prototipleri, veri tipleri ve adlandırılmış sabitler
CMQEC	Ana MQI, veri dönüştürme çıkışı ve Arabirim Giriş Noktaları yapısı (CMQEC, CMQXC ve CMQC içerir) için işlev prototipleri, veri tipleri ve adlandırılmış değişmezler
CMQSTRC	MQI sabit tanımlarını metin eşdeğerine dönüştüren işlevler.  Uyarı:  z/OS için geçerli değildir.

Uygulamaların taşınabilirliğini artırmak için, `#include` ön işlemcisi yönergesinde üstbilgi dosyasının adını küçük harfli olarak kodlayın:

```
#include "cmqec.h"
```

İşlevler

Bir işlevi her çağırdığınızda, adresle geçirilen tüm parametreleri belirtmenize gerek yoktur.

- *Yalnızca giriş* olan ve MQHCONN, MQHOBJ ya da MQHOBE değerine göre değer içeren parametreleri geçirin.
- Diğer tüm parametreleri adrese göre geçirin.

Belirli bir parametrenin gerekli olmadığı durumlarda, parametre verilerinin adresi yerine, işlev çağırımında parametre olarak boş değerli bir gösterge kullanın. Bunun olası olduğu parametreler, çağrı açıklamalarında tanımlanır.

İşleve değer olarak bir parametre döndürülmez; C terminolojisinde bu, tüm işlevlerin void döndürdüğü anlamına gelir.

İşleve ilişkin öznitelikler MQENTRY makro değişkeniyle tanımlanır; bu makro değişkeninin değeri ortama bağlıdır.

Tanımlanmamış veri tipine sahip parametreler

MQGET, MQPUT ve MQPUT1 işlevlerindeki *Buffer* değiştirgesi tanımlanmamış bir veri tipi içeriyor. Bu parametre, uygulamanın ileti verilerini göndermek ve almak için kullanılır.

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. Gerçek işlev değiştirgesini bir gösterge-boşluk olarak bildirin ve işlev çağırısındaki parametre olarak herhangi bir veri sırasının adresini belirtin.

Veri tipleri

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 P harfiyle önekli olan temel ya da yapı veri tipinin adıdır. MQPOINTER makro değişkenini kullanarak işaretçinin 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 dizgileri işleme

İkili verilerin dizilimlerini MQBYTEn veri tiplerinden biri olarak bildirebilirsiniz.

Bu tipteki alanları kopyaladığınızda, karşılaştırdığınızda ya da ayarladığınızda, C işlevlerini memcpy , memcpy ya 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));
```

Bu işlevler, MQBYTEn veri tipleriyle bildirilen veriler için doğru çalışmadığından, strcpy , strcmp , strncpy ya da strncmp dizgi işlevlerini kullanmayın.

Karakter dizgillerini işleme

Kuyruk yöneticisi uygulamaya karakter verisi döndürdüğünde, kuyruk yöneticisi karakter verilerini her zaman, alanın tanımlı uzunluğuna doğru boşluklarla doldurur; kuyruk yöneticisi *bunu yapmaz* boş karakterle sonlandırılmış dizgileri döndürür.

Bu nedenle, bu tür dizgileri kopyalarken, karşılaştırırken ya da bitiştirirken, strcpy , strcmp ya da strncat dizgi işlevlerini kullanın.

Dizginin boş değerle sonlandırılmasını gerektiren dizgi işlevlerini kullanmayın (strcpy , strcmp , strcat). Ayrıca, dizginin uzunluğunu belirlemek için strlen işlevini kullanmayın; alanın uzunluğunu belirlemek için sizeof işlevi yerine kullanın.

Yapılara ilişkin ilk değerler

Üstbilgi kütükleri, bu yapılara ilişkin eşgörünümleri bildirirken MQ yapılarına ilişkin ilk değerleri sağlamak için kullanabileceğiniz çeşitli makro değişkenlerini tanımlar.

Bu makro değişkenlerinin adları MQxxx_default biçiminin adlarına sahiptir; burada MQxxx, yapının adını gösterir. Aşağıdaki şekilde kullanılırlar:

```
MQMD    MyMsgDesc = {MQMD_DEFAULT};
MQPMO    MyPutOpts = {MQPMO_DEFAULT};
```

Bazı karakter alanları için (örneğin, çoğu yapıda oluşan *StrucId* alanları ya da MQMD 'de ortaya çıkan *Format* alanı), MQI, geçerli olan belirli değerleri tanımlar. Geçerli değerlerin her biri için, *iki* makro değişkeni sağlanır:

- Bir makro değişkeni, değeri, örtük boş (null) boş eşleşme dışında, alanın tanımlı uzunluğunun tam olarak dışında bir uzunluğa sahip bir dizgi olarak tanımlar. Örneğin, MQMD 'deki *Format* alanı için aşağıdaki makro değişkeni sağlanır (– boş bir karakteri temsil eder):

```
#define MQFMT_STRING "MQSTR- - -"
```

Bu formu memcpy ve memcmp işlevleriyle birlikte kullanın.

- Diğer makro değişkeni, değeri bir karakter dizisi olarak tanımlar; bu makro değişkeninin adı, _ARRAY ile birlikte suffixed dizgi biçiminin adıdır. Örneğin:

```
#define MQFMT_STRING_ARRAY 'M','Q','S','T','R','-','-','-'
```

Bu formu, yapının bir eşgörünümünü, MQMD_XX_ENCODE_CASE_ONE default makro değişkeni tarafından sağlanan değerlerden farklı değerleri olan bir yönetim ortamı bildirirken kullanıma hazırlamak için kullanın. (Bu her zaman gerekli değildir; bazı ortamlarda her iki durumda da değer dizgi biçimini kullanabilirsiniz. Ancak, bu durum C++ programlama diliyle uyumluluk için gerekli olduğu için, bildirim için dizi formunu kullanabilirsiniz.)

Dinamik yapılar için ilk değerler

Bir yapının eşgörünümünün değişken sayısı gerektiğinde, eşgörünüm genellikle calloc ya da malloc işlevleri kullanılarak dinamik olarak elde edilen ana depolamada yaratılır. Bu tür yapılardaki alanları kullanıma hazırlamak için aşağıdaki tekniği göz önünde bulundurun:

1. Yapıyı kullanıma hazırlamak için uygun MQxxx_XX_ENCODE_CASE_ONE default makro değişkenini kullanarak yapının bir eşgörünümünü bildirin. Bu yönetim ortamı, diğer yönetim ortamlarına ilişkin model haline gelir:

```
MQMD Model = {MQMD_DEFAULT}; /* declare model instance */
```

The static or auto keywords can be coded on the declaration in order to give the model instance static or dynamic lifetime, as required.

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 örneğini dinamik eşgörünümüne kopyalamak için memcpy işlevini kullanın:

```
memcpy(Instance,&Model,sizeof(MQMD)); /* initialize dynamic instance */
```

C++ dilinde kullan

C++ programlama dili için, üstbilgi kütükleri yalnızca bir C++ derleyicisi kullanırken içerilen aşağıdaki ek deyimleri içerir:

```
#ifndef __cplusplus
```



```

extern "C" {
#endif

/* rest of header file */

#ifdef __cplusplus
}
#endif

```

Notasyonla ilgili kurallar

Bu bilgiler, işlevlerin ve bildirme parametrelerinin nasıl çağrılacağını gösterir.

Bazı durumlarda, değiştirgeler değişmez büyüklükte dizilerdir. Bunlar için, bir sayısal değişmezi göstermek için küçük harfli bir n kullanılır. Bu parametreye ilişkin bildirimi kodladığınızda, n değerini gereken sayısal değerle değiştirin.

COBOL

Bu bölümde, COBOL programlama dilinden MQI ' yi kullanmanıza yardımcı olacak bilgiler yer alır.

Kütüklerin KOPYASI

Çeşitli COPY dosyaları, MQI kullanan COBOL uygulama programlarını yazmanıza yardımcı olmak için sağlanmıştır. Adlandırılmış değişmezleri içeren iki dosya ve her bir yapı için iki dosya vardır.

Her bir yapı iki biçimde sağlanır: başlangıç değerlerine sahip bir form ve olmayan bir form:

- Bir COBOL programının ÇALIŞMA-DEPOLAMA bölümünde başlangıç değerleri olan yapıları kullanın; bunlar, V harfiyle (Değerler için) suffixed adlarıyla COPY dosyalarında yer alır.
- Bir COBOL programının LINKAGE SECTION ' de başlangıç değerleri olmadan yapıları kullanın; bunlar L harfiyle (Linkage için) suffixed adlarıyla COPY dosyalarında yer alır.

Kopya dosyaları Çizelge 10 sayfa 265'inde özetlenir. Listelenen tüm dosyalar, tüm ortamlarda kullanılabilir değil.

Çizelge 10. COBOL COPY dosyaları		
Dosya (başlangıç değerleri ile)	Dosya (başlangıç değerleri olmadan)	İçindekiler
CMQAIRV	CMQAIRL	Kimlik doğrulama bilgileri kaydı
CMQBOV	CMQYükleme Belgesi	Seçenekler yapısını başlat
CMQCIHV	CMQCIHL	CICS bilgi üstbilgisi yapısı
CMQCNV	CMQCNOL	Bağlantı seçenekleri yapısı
CMQDHSV	CMQDHL	Dağıtım üstbilgisi yapısı
CMQDLHV	CMQDLHL	Ölü harf üstbilgisi yapısı
CMQDXPV	CMQDXPL	Veri dönüştürme çıkış parametresi yapısı
CMQGMV	CMQGMOL	İleti seçenekleri yapısını al
CMQIIHV	CMQIHL	IMS bilgi üstbilgisi yapısı
CMQMDV	CMQMDL	İleti tanımlayıcı yapısı
CMQMDEV	CMQMDEL	İleti tanımlayıcı uzantısı yapısı
CMQMD1V	CMQMD1L	İleti tanımlayıcı yapısı sürüm 1
CMQODV	CMQODL	Nesne tanımlayıcı yapısı
CMQORV	CMQORL	Nesne kaydı yapısı

Çizelge 10. COBOL COPY dosyaları (devamı var)

Dosya (başlangıç değerleri ile)	Dosya (başlangıç değerleri olmadan)	İçindekiler
CMQPMOV	CMQPMOL	İleti seçenekleri yapısını koy
CMQRFHV	CMQRFHL	Kurallar ve biçimlendirme üstbilgisi yapısı
CMQRFH2V	CMQRFH2L	Kurallar ve biçimlendirme üstbilgisi yapısı sürüm 2
CMQRMHV	CMQRMHL	Başvuru iletisi üstbilgi yapısı
CMQRRV	CMQRRL	Yanıt kaydı yapısı
CMQSCOV	CMQSCOL	SSL yapılandırma seçenekleri
CMQTMV	CMQTML	İleti yapısını tetikle
CMQTMCV	CMQTMCL	İleti yapısını tetikle (karakter biçimi)
CMQTM2V	CMQTM2L	İleti yapısını tetikle (karakter biçimi) sürüm 2
CMQWIHV	CMQWIHL	İş bilgileri üstbilgi yapısı
CMQXQHV	CMQXQHL	İletim kuyruğu üstbilgi yapısı
CMQV	-	Ana MQI için adlandırılan sabitler
CMQXV	-	Veri dönüştürme çıkışı için adlandırılan sabitler
CMQMD2V	CMQMD2L	İleti tanımlayıcı yapısı sürüm 2

Yapılar

In the COPY file, each structure declaration begins with a level-10 item; this enables you to declare several instances of the structure by coding the level-01 declaration and then using the COPY statement to copy in the remainder of the structure declaration. Uygun örneğe başvurmak için, IN anahtar sözcüğünü kullanın:

```
* Declare two instances of MQMD
01 MY-MQMD.
   COPY CMQMDV.
01 MY-OTHER-MQMD.
   COPY CMQMDV.

*
* Set MSGTYPE field in MY-OTHER-MQMD
  MOVE MQMT-REQUEST TO MQMD-MSGTYPE IN MY-OTHER-MQMD.
```

Yapıları uygun sınırlarla hizalayın. COPY deyimini, level-01 ögesi olmayan bir ögenin ardından bir yapı dahil etmek için kullanırsanız, yapının level-01 ögesinin başlangıcındaki uygun görelî konumda başladığından emin olun. Çoğu MQI yapısı 4 baytlık hizalama gerektirir; bunun için kural dışı durumlar, IBM üzerinde 16 byte 'lık hizalama gerektiren MQCNO, MQOD ve MQPMO ' dur.

Bu bölümde, yapılardaki alanların adları örnek olmadan gösterilir. COBOL içinde, alan adlarının başında, yapının adı ve ardından bir tire işareti bulunur. Ancak, yapı adı sayısal bir sayıyla biterse, yapının özgün yapının ikinci ya da daha sonraki bir sürümünü olduğunu gösterir. Sayısal rakam, önekle, sayısal bir basamaktan çıkarılır. COBOL ' de alan adları büyük harfle gösterilir (gerekirse küçük harfli ya da büyük harf karışık olarak kullanılabilir). For example, the field *MsgType* described in “MQMD-İleti tanımlayıcısı” sayfa 404 becomes MQMD-MSGTYPE in COBOL.

V-sonek yapıları, tüm alanlar için başlangıç değerleriyle bildirilir; yalnızca, sağlanan başlangıç değerinden farklı bir değer istediğiniz alanları ayarlamamız gerekir.

Göstergeler

Bazı yapıların, MÇOD yapısı tarafından ele alınan MÇOR ve MÇRR kayıtları gibi, yapıyla ayrılabilir isteğe bağlı verileri ele alınması gerekir.

Bu isteğe bağlı verileri ele almak için yapılar, işaretçi veri tipiyle bildirilen alanlar içerir. Ancak COBOL, tüm ortamlardaki işaretçi veri tipini desteklemez. Bu yüzden, isteğe bağlı veriler, yapının başlangıcından elde edilen verilerin görelî konumunu içeren alanlar kullanılarak da adreslenebilir.

Ortamlar arasında bir uygulamayı kapı aralamak istiyorsanız, işaretçi veri tipinin tüm amaçlanan ortamlarda kullanılabilir olup olmadığını kesin olarak belirleyin. Bu durumda değilse, uygulamanın isteğe bağlı verileri işaretçi alanları yerine görelî konum alanlarını kullanarak ele vermeleri gerekir.

İşaretçilerin desteklenmediği ortamlarda, gösterge alanlarını uygun uzunluğun byte dizgileri olarak bildirerek, ilk değer tüm boş bayt dizilimi olarak kabul edilir. Görelî konum alanlarını kullanıyorsanız, bu ilk değeri değiştirmeyin.

Adlandırılmış Değişmezler

Bu bölümde, adın bir parçası olarak altçizgi karakteri (_) içeren değişmezlerin adları gösterilir. COBOL içinde, tire karakterini (-) alt çizgi karakterini kullanarak kullanın.

Karakter dizilimi değerleri olan değişmezler, dizilim sınırlayıcısı olarak tek tırnak işaretini (') kullanır. Bazı ortamlarda, derleyicinin çift tırnak işaretinin yerine dizgi sınırlayıcı olarak tek tırnak işaretini kabul etmesi için uygun bir derleyici seçeneği belirlemeniz gerekebilir.

Adlandırılmış sabitler, COPY dosyalarında level-10 öge olarak bildirilir. Sabitleri kullanmak için, level-01 ögesini açık bir şekilde bildirip, sabitlerin bildirimlerinde kopyalamak için COPY deyimini kullanın:

```
* Declare a structure to hold the constants
01 MY-MQ-CONSTANTS.
   COPY CMQV.
```

Önceki y "ntem, deşiymezlerin, g" nderme girmemiyorlarsa, programdaki saklama yeri kaplamalar saşlar Değişmezleri aynı çalıştırma birimi içinde birçok ayrı programa dahil etseniz, sabitlerin birden çok kopyası bulunur ve ana depolamayı gereksiz yere tüketir. Aşağıdaki tekniklerden birini kullanarak bu etkiyi önlein:

- Add the GLOBAL clause to the level-01 declaration:

```
* Declare a global structure to hold the constants
01 MY-MQ-CONSTANTS GLOBAL.
   COPY CMQV.
```

Bu durum, çalıştırma birimindeki yalnızca tek bir sabit disk takımı için saklama alanı ayırır. Ancak, sabitler, yalnızca level-01 bildirimini içeren program değil, çalıştırma birimi içindeki herhangi bir program tarafından başvurulabilir.

Not: GLOBAL yantümcesi tüm ortamlarda desteklenmez.

- Her bir programa el ile kopyalama, yalnızca o programın gönderme yaptığı değişmezleri içerir. Tüm değişmezleri programa kopyalamak için COPY deyimini kullanmayın.

Notasyonla ilgili kurallar

Bu kısımdaki ikinci konular, çağrılar ve bildirme parametrelerinin nasıl çağrılacağını gösterir. Bazı durumlarda, değiştirgeler çizelgeler ya da karakter dizilimleridir; bunlar değişmez. Bunlar için, bir sayısal değişmezi göstermek için küçük harfli bir n kullanılır. Bu parametreye ilişkin bildirimi kodladığınızda, n değerini gereken sayısal değerle değiştirin.

System/390 çevirici programlaması

Bu bölümde, System/390 Assembler programlama dilindeki MQI ' yi kullanmanıza yardımcı olacak bilgiler yer alır.

Makrolar

MQI kullanan çevirici uygulama programları yazmanıza yardımcı olması için çeşitli makrolar sağlamıştır.

Adlandırılmış sabitler için iki makro ve her bir yapı için bir makro vardır. Bu dosyalar [Çizelge 11 sayfa 268](#) içinde özetlenir.

Çizelge 11. Çevirici makrolar	
Dosya	İçindekiler
CMQA	Ana MQI için adlandırılan sabitler (equates)
CMQCIHA	CICS bilgi üstbilgisi yapısı
CMQCNOA	Bağlantı seçenekleri yapısı
CMQDLHA	Ölü harf üstbilgisi yapısı
CMQDXPA	Veri dönüştürme çıkış parametresi yapısı
CMQGMOA	İleti seçenekleri yapısını al
CMQIIHA	IMS bilgi üstbilgisi yapısı
CMQMDA	İleti tanımlayıcı yapısı
CMQMDEA	İleti tanımlayıcı uzantısı yapısı
CMQODA	Nesne tanımlayıcı yapısı
CMQPMOA	İleti seçenekleri yapısını koy
CMQRFHA	Kurallar ve biçimlendirme üstbilgisi yapısı
CMQRFH2A	Kurallar ve biçimlendirme üstbilgisi yapısı sürüm 2
CMQRMHA	Başvuru iletisi üstbilgi yapısı
CMQTMA	İleti yapısını tetikle
CMQTM2A	İleti yapısını tetikle (karakter biçimi) sürüm 2
CMQVERA	Yapı sürümü denetimi
CMQWIHA	İş bilgileri üstbilgi yapısı
MQXA	Veri dönüştürme çıkışı için adlandırılan sabitler
CMQXPA	API geçiş çıkış parametresi yapısı
CMQXQHA	İletim kuyruğu üstbilgi yapısı

Yapılar

Yapılar, makronun eylemlerini denetlemek için çeşitli parametrelere sahip makrolar tarafından oluşturulur. Bu değiştirgeler aşağıdaki kısımlarda açıklanmıştır.

Zaman zaman MQ yapılarının yeni sürümlerine tanıtılır. Yeni bir sürümdeki ek alanlar, daha önce 256 byte 'tan daha küçük bir yapının 256 byte 'tan daha büyük bir yapıya dönüşmesine neden olabilir. Bu yüzden, bir MQ yapısını kopyalamak ya da bir MQ yapısını boş değere ayarlamak, 256 byte 'tan büyük yapılarla doğru şekilde çalışmak için tasarlanmış bir çevirici yönergelerinden kaynaklanır. Diğer bir seçenek olarak, yapının belirli bir sürümünü bildirmek için DCLVER makro değiştirgesini ya da CMQVERA makrosunu VERSION parametresiyle birlikte kullanın.

Yapının adını belirtme

Bir yapının birden çok eşgörünümünü bildirmek için, makro örnekleri yapıdaki her alanın adını kullanıcı belirenebilir bir dizgiyle ve alt çizgiyle örnekler.

Kullanılan dizgi, makronun çağırılması sırasında belirtilen etikettir. Etiket belirtilmediyse, öneki oluşturmak için yapının adı kullanılır:

```
* Declare two object descriptors
```

CMQODA	,	Prefix used="MQOD_" (the default)
MY_MQOD	,	Prefix used="MY_MQOD_"

Bu kısımda gösterilen yapı bildirimleri varsayılan öneki kullanır.

Yapı formunun belirtilmesi

Yapı bildirimleri, makro tarafından DSECT parametresi tarafından denetlenen iki biçimden birinde oluşturulabilir:

DSECT = EVET

Bir çevirici DSECT yönergesi yeni bir veri bölümü başlatmak için kullanılır; yapı tanımlaması DSECT deyiminden hemen sonra gelir. Makro çağırımı üzerindeki etiket, veri bölümünün adı olarak kullanılır; etiket belirtilmediyse, yapının adı kullanılır.

DSECT = NO

Çevirici DC yönergeleri, yordamı, yordamda geçerli konumda tanımlamak için kullanılır. Alanlar, makro çağırımı ile ilgili parametrelerin kodlanması yoluyla belirlenebilecek değerlerle kullanıma hazırlandı. Makro çağırımı üzerinde hiçbir değer belirtilmediği alanlar varsayılan değerlerle başlatılır.

Belirtilen değer büyük harfli olmalıdır. DSECT parametresi belirtilmemişse, DSECT=NO varsayılan değer olarak kabul edilir.

Yapı sürümünü denetleme

Varsayılan olarak, makrolar her zaman her yapının en son sürümünü bildirir.

Yapıda *Version* alanı için bir değer belirtmek üzere VERSION makro parametresini kullanabilirsiniz; ancak bu parametre, *Version* alanının başlangıç değerini tanımlar ve yapının gerçekte bildirilmiş sürümünü denetmez. Bildirilmiş yapının sürümünü denetlemek için, DCLVER parametresini kullanın:

DCLVER=CURRENT

Bildirilen sürüm, yürürlükteki (en son) sürüm olarak kabul edilir.

DCLVER=BELIRTILDI

Bildirilen sürüm, VERSION parametresi tarafından belirtilen sürümdür. VERSION parametresini çıkarırsanız, varsayılan sürüm 1 'dir.

VERSION parametresini belirtirseniz, değer kendi kendine tanımlı bir sayısal değişmez ya da gereken sürüm için adlandırılmış değişmezdir (örneğin, MQCNO_VERSION_3). If you specify some other value, the structure is declared as if DCLVER=CURRENT had been specified, even if the value of VERSION resolves to a valid value.

Belirtilen değer büyük harfli olmalıdır. DCLVER parametresini çıkarırsanız, kullanılan değer MQDCLVER genel makro değişkeninden alınır. CMQVERA makrosunu kullanarak bu değişkeni ayarlayabilirsiniz.

Bir yapının başka bir yapının içine gömülü olduğunu bildiren

Bir yapıyı, başka bir yapının bileşeni olarak bildirmek için, NESTED parametresini kullanın:

NESTED=YES

Yapı bildirimi, başka bir yapı içinde iç içe yerleştirilir.

NESTED=NO

Yapı bildirimi başka bir yapı içinde iç içe geçmiş değil.

Belirtilen değer büyük harfli olmalıdır. NESTED parametresini çıkarırsanız, NESTED=NO varsayılan değer olarak kabul edilir.

Alanlar için ilk değerleri belirtme

Makro çağırımında bu alanın adını (örnek olmadan) kodlayarak, gereken değer yanında, bir alanın adını (örnek olmadan) kodlayarak bir alanı kullanıma hazırlamak için kullanılacak değeri belirtin.

Örneğin, *MsgType* alanına sahip bir ileti tanımlayıcısı yapısını MQMT_REQUEST ile ilk kullanıma hazırlanmış ve *ReplyToQ* alanı "MY_REPLY_TO_QUEUE" dizisiyle kullanıma hazırlanmış olarak bildirmek için, aşağıdaki bilgileri kullanın:

```
MY_MQMD  CMQMDA  MSGTYPE=MQMT_REQUEST,
          REPLYTOQ=MY_REPLY_TO_QUEUE
```

Makro çağrısında değer olarak adlandırılmış bir değişmez (equate) belirtirseniz, adı belirtilen değişmezi tanımlamak için CMQA makrosunu kullanın. Karakter dizilimi değerlerini tek tırnak işareti içine almayın.

Listelemeyi denetleme

Control the appearance of the structure declaration in the assembler listing using the LIST parameter:

LISTE = EVET

Yapı bildirimi, çevirici listelerinde görünür.

LISTE = HAYIR

Yapı bildirimi, çevirici listelemesinde görünmez.

Belirtilen değer büyük harfli olmalıdır. LIST parametresini çıkarırsanız, LIST=NO varsayılan değer olarak kabul edilir.

CMQVERA makrosu

Bu makro, yapı makroları üzerinde DCLVER parametresi için kullanılacak varsayılan değeri ayarlamanıza olanak sağlar. The value specified by CMQVERA is used by the structure macro only if you omit the DCLVER parameter from the invocation of the structure macro. Varsayılan değer, DCLVER parametresiyle CMQVERA makrosu kodlanarak ayarlanır:

DCLVER=CURRENT

Varsayılan sürüm, geçerli (en son sürüm) sürüme ayarlanır.

DCLVER=BELIRTILDI

Varsayılan sürüm, VERSION parametresi tarafından belirtilen sürüme ayarlıdır.

DCLVER değiştirgesini belirtmeniz gerekir ve değer büyük harfli olmalıdır. CMQVERA tarafından ayarlanan değer, CMQVERA 'nın sonraki çağrısına ya da düzeneğin sonuna kadar varsayılan değer olmaya devam eder. CMQVERA 'yı çıkarırsanız, varsayılan ayar DCLVER=CURRENT 'dir.

Notasyonla ilgili kurallar

Sonraki kısımlarda, çağrılarının nasıl çağrılacağı ve parametrelerin bildirilmesi gösterilir. Bazı durumlarda, parametreler, bir sayısal değişmezi göstermek için küçük harfli bir n kullanıldığında diziler ya da karakter dizgileridir. Bu parametreye ilişkin bildirimi kodladığınızda, n değerini gereken sayısal değerle değiştirin.

MQAIR-Kimlik doğrulama bilgileri kaydı

MQAIR yapısı, kimlik doğrulama bilgileri kaydını temsil eder.

Aşağıdaki çizelge yapıdaki alanları özetlemektedir.

Çizelge 12. MQAIR 'deki alanlar		
Alan	Tanım	Konu
StrucId	Yapı tanıtıcısı	StrucId
S\u00f0fcr\u00f0cm	Yapı sürüm numarası	S\u00f0fcr\u00f0cm
AuthInfoTipi	Kimlik doğrulama bilgileri tipi	AuthInfoTipi
AuthInfoConnName	LDAP CRL sunucusunun bağlantı adı	AuthInfoConnName
LDAPUserNamePtr	LDAP kullanıcı adının adresi	LDAPUserNamePtr

Çizelge 12. MQAIR ' deki alanlar (devamı var)		
Alan	Tanım	Konu
LDAPUserNameGörelİ Konumu	MQSCO ' nun başlangıcından LDAP kullanıcı adının görelİ konumu	<u>LDAPUserNameGörelİ Konumu</u>
LDAPUserNameUzunluđu	LDAP kullanıcı adının uzunluđu	<u>LDAPUserNameUzunluk</u>
LDAPPASSWORD	LDAP sunucusuna erişmek için parola	<u>LDAPPASSWORD</u>
Not: Sürüm , MQAIR_VERSION_2değerinden küçükse, geri kalan alanlar yoksayılır.		
OCSPResponderURL	OCSP yanıtlayıcıya ulaşılabilirdiđi URL adresi	<u>OCSPResponderURL</u>

MQAIR ' ye Genel Bakış

MQAIR yapısı, bir uygulamanın IBM MQ MQI client olarak çalışan bir uygulamanın istemci bağlantısı için kullanılacak kimlik doğrulayıcıya ilişkin bilgileri belirtmesini sağlar. Yapı, MQCONNX çağrısına ilişkin bir giriş deđiştir.

Kullanılabilirlik: AIX, HP-UX, Solaris, Linux ve Windows istemcileri.

Karakter kümesi ve kodlama: MQAIR içindeki veriler, yerel kuyruk yöneticisinin karakter kümesinde ve kodlamasında olmalıdır; bunlar **CodedCharSetId** kuyruk yöneticisi özniteliđi ve MQENC_NATIVE ile verilir.

MQAIR için alanlar

MQAIR yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

AuthInfoConnName (MQCHAR264)

Bu ad, LDAP sunucusunun çalışmakta olduđu anasistemin adı ya da ađ adresi. Bunu, parantez içine alınmış isteđe bađlı bir kapı numarası takip edebilir. Varsayılan kapı numarası 389'dur.

Deđer, alanın uzunluđundan kısaysa, deđerİ boş deđerli bir karakterle sonlandırın ya da alanın uzunluđuna göre boşluklarla doldurur. Deđer geçersiz ise, çağrı neden kodu MQRC_AUTH_INFO_CONN_NAME_ERROR neden ile başarısız olur.

Bu bir giriş alanıdır. Bu alanın uzunluđu MQ_AUTH_INFO_CONN_NAME_LENGTH tarafından verilir. Bu alanın ilk deđerİ, C içindeki boş deđerli dizidir ve diđer programlama dillerindeki boşluk karakteridir.

AuthInfoTip (MQUBE)

Bu, kayıta bulunan kimlik doğrulama bilgilerinin tipidir.

Deđer, aşağıdaki iki deđiştirgelerden biri olabilir:

MQAIT_CRL_LDAP

LDAP sunucusu kullanılarak sertifika iptal denetimi.

MQAIT_OCSP

OCSP kullanılarak sertifika iptal denetimi.

Deđer geçerli deđilse, çağrı neden kodu MQRC_AUTH_INFO_TYPE_ERROR neden ile başarısız olur.

Bu bir giriş alanıdır. Bu alanın ilk deđerİ MQAIT_CRL_LDAP ' dir.

LDAPPASSWORD (MQCHAR32)

Bu, LDAP CRL sunucusuna erişmek için gerekli olan paroladır. Deđer, alanın uzunluđundan kısaysa, deđerİ boş deđerli bir karakterle sonlandırın ya da alanın uzunluđuna göre boşluklarla doldurur.

LDAP sunucusu parola gerektirmiyorsa ya da LDAP kullanıcı adını atarsanız, *LDAPPASSWORD* boş deđerli ya da boş olmalıdır. LDAP kullanıcı adını atarsanız ve *LDAPPASSWORD* boş deđerli ya da boş deđilse, 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.

LDAPUserNameUzunluk (MQUZE)

Bu, *LDAPUserNamePtr* ya da *LDAPUserNameOffset* alanı tarafından adreslenen LDAP kullanıcı adının bayt cinsinden uzunluğudur. Değer, MQ_DISTINGUISH_NAME_LENGTH ile sıfır aralığında olmalıdır. Değer geçerli değilse, çağrı neden kodu MQRC_LDAP_USER_NAME_LENGTHH_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.

LDAPUserNameGörelili Konumu (MQUZE)

Bu, MQAIR yapısının başlangıcındaki LDAP kullanıcı adının bayt cinsinden görelili konudur.

Görelili konum pozitif ya da negatif olabilir. *LDAPUserNameLength* sıfırsa, alan yoksayıılır.

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.

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* sıfırsa, alan yoksayıılır.

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 görelili 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.

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 çağırısı 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.

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.

MQAIR için ilk değerler ve dil bildirimleri

Çizelge 13. MQAIR ' deki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
StrucId	MQAIR_STRUCT_ID	' AIR↵ '
S\u00fcr\u00fcmleri	MQAIR_VERSION_1	1
AuthInfoTipi	MQAIT_CRL_LDAP	1
AuthInfoConnName	Yok	Dizgi ya da boşluk boş değeri
LDAPUserNamePtr	Yok	Boş değeri gösterge ya da boş değeri byte
LDAPUserNameGörelili Konumu	Yok	0
LDAPUserNameUzunluğu	Yok	0

Çizelge 13. MQAIR ' deki alanların ilk değerleri (devamı var)

Alan adı	Değişmezin adı	Değişmezin değeri
LDAPPassword	Yok	Dizgi ya da boşluk boş değerli
OCSPResponderURL	Yok	Dizgi ya da boşluk boş değerli

Notlar:

1. ↵ simgesi tek bir boş karakteri temsil eder.
2. C programlama dilinde, makro değişkeniMQAIR_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlara ilişkin ilk değerleri sağlamak için bunu aşağıdaki şekilde kullanın:

```
MQAIR MyAIR = {MQAIR_DEFAULT};
```

C bildirimi

```
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 */
};
```

COBOL bildirimi

```
** 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).
```

Visual Basic bildirimi

```
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		

MQBMHO-Arabelleğiyle ileti tanıtıcısı seçenekleri

Aşağıdaki çizelge yapıdaki alanları özetlemektedir. MQBMHO structure-buffer to message handle options

Çizelge 14. MQBMHO 'daki alanlar		
Alan	Tanım	Konu
<i>StrucId</i>	Yapı tanıtıcısı	StrucId
<i>Version</i>	Yapı sürüm numarası	S\u00fcr\u00fcm
<i>Options</i>	MQBMHO 'nun eylemini denetleyen seçenekler	Seçenekler

MQBMHO 'ya Genel Bakış

Kullanılabilirlik: Tümü. Buffer to message handle options yapısı-genel bakış

Amaç: MQBMHO yapısı, uygulamaların, ileti tutamaçlarının arabelleklerden nasıl üretildiğini denetleyen seçenekleri belirlemesine olanak sağlar. Yapı, MQBUFMH çağrısına ilişkin bir giriş değiştirmedir.

Karakter kümesi ve kodlama: MQBMHO içindeki veriler, uygulamanın uygulama ve kodlamasının karakter kümesinde yer almalıdır (MQENC_NATIVE).

MQBMHO alanları

Arabellek-ileti tanıtıcısı seçenekleri yapısı-alanlar

MQBMHO yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

Seçenekler (MQUZE)

Arabellek-ileti tanıtıcısı yapısı-Seçenekler alanı

Değer şu şekilde olabilir:

MQBMHO_DELETE_PROPERTIES

İleti tutamaçına eklenen özellikler arabellekten silinir. Arama başarısız olursa özellikler silinmez.

Varsayılan seçenekler: Açıklanmanız gerekmiyorsa, aşağıdaki seçeneği kullanın:

MQBMHO_YOK

Seçenek belirtilmedi.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQBMHO_DELETE_PROPERTIES olarak değerlendirilmektedir.

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.

MQBMHO için ilk değerler ve dil bildirimleri

Arabellek-ileti tanıtıcısı yapısı-İlk değerler

Çizelge 15. MQBMHO 'daki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
StrucId	MQBMHO_STRUC_ID	' BMHO '
Version	MQBMHO_VERSION_1	1
Options	MQBMHO_YOK	0
Notlar: 1. C programlama dilinde, makro değişkeniMQBMHO_DEFAULT, tabloda listelenen değerleri içerir. Yapıdaki alanlara ilişkin ilk değerleri sağlamak için bunu aşağıdaki şekilde kullanın: <pre>MQBMHO MyBMHO = {MQBMHO_DEFAULT};</pre>		

C bildirimi

Arabellek-ileti tanıtıcısı yapısı-C dil 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 */  
};
```

COBOL bildirimi

Arabellek-ileti tanıtıcısı yapısı-COBOL dil 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.
```

PL/I bildirimi

Arabellek-ileti tanıtıcısı yapısı-PL/I dil 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 */

```

High Level Assembler bildirimi

Arabellekten ileti tanıtıcısı yapısı-Assembler dil 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)

```

MQBO-Başlat seçenekleri

Aşağıdaki çizelge yapıdaki alanları özetlemektedir.

Çizelge 16. MQBO 'daki alanlar		
Alan	Tanım	Konu
<i>StrucId</i>	Yapı tanıtıcısı	StrucId
<i>Version</i>	Yapı sürüm numarası	S\u00fcr\u00fcm
<i>Options</i>	MQBEGIN eylemini denetleyen seçenekler	Se\u00e7enekler

MQBO için genel bakış

Kullanılabilirlik: AIX, HP-UX, IBM i, Solaris, Linux, Windows ; IBM MQ MQI clients için kullanılamaz.

Amaç: MQBO yapısı, uygulamanın bir iş birimi yaratılmasıyla ilgili seçenekleri belirlemesine olanak sağlar. Yapı, MQBEGIN çağrısındaki bir giriş/çıkış parametresidir.

Karakter kümesi ve kodlama: MQBO 'daki veriler, *CodedCharSetId* kuyruk yöneticisi özniteliği tarafından verilen karakter kümesinde ve MQKEN_NATIVE tarafından verilen yerel kuyruk yöneticisinin kodlamasında olmalıdır. Ancak, uygulama bir MQ MQI istemcisi olarak çalışıyorsa, yapı istemcinin karakter kümesinde ve kodlamasında olmalıdır.

MQBO için alanlar

MQBO yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

Seçenekler (MQUZE)

Bu alan her zaman bir giriş alanıdır. İlk değeri MQBO_NONE olur.

Değer şu olmalıdır:

MQBO_NONE

Seçenek belirtilmedi.

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 tanıtıcı.

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.

MQBO için ilk değerler ve dil bildirimleri

Çizelge 17. MQBO için MQBO 'daki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
StrucId	MQBO_STRUC_ID	'B0¬¬'
Version	MQBO_VERSION_1	1
Options	MQBO_NONE	0
Notlar: 1. ¬ simgesi tek bir boş karakteri temsil eder. 2. C programlama dilinde, makro değişkeniMQBO_DEFAULT, tabloda listelenen değerleri içerir. Yapıdaki alanlara ilişkin ilk değerleri sağlamak için bunu aşağıdaki şekilde kullanın: <pre>MQBO MyBO = {MQBO_DEFAULT};</pre>		

C bildirimi

```
typedef struct tagMQBO MQBO;  
struct tagMQBO {  
    MQCHAR4 StrucId; /* Structure identifier */  
    MQLONG Version; /* Structure version number */  
    MQLONG Options; /* Options that control the action of MQBEGIN */  
};
```

COBOL bildirimi

```
** 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.
```

PL/I bildirimi

```
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 */
```

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

```

MQCBC-Callback bağlamı

Aşağıdaki çizelge yapıdaki alanları özetlemektedir. Geri çağırma yordamını tanımlayan yapı.

Çizelge 18. MQCBC 'deki alanlar		
Alan	Tanım	Konu
<i>StrucID</i>	Yapı tanıttıcısı	StrucID
<i>Version</i>	Yapı sürüm numarası	S\u00fcr\u00fcm
<i>CallType</i>	Neden işlev çağrıldı	CallType
<i>Hobj</i>	Nesne tanıttıcısı	Hobj
<i>CallbackArea</i>	Geri bildirme işlevi için kullanılacak alan	CallbackArea
<i>ConnectionArea</i>	Geri bildirme işlevi için kullanılacak alan	ConnectionArea
<i>CompCode</i>	Tamamlanma kodu	CompCode
<i>Reason</i>	Neden Kodu	Neden
<i>State</i>	Yürürlükteki tüketicinin durumunun göstergesi	STATE
<i>DataLength</i>	İleti uzunluğu	DataLength
<i>BufferLength</i>	İleti arabelleğinin bayt cinsinden uzunluğu	BufferLength
<i>Flags</i>	Genel işaretler	İşaretler
Not: Sürüm MQCBC_VERSION_2'den küçükse geri kalan alan yoksaılır.		
<i>ReconnectDelay</i>	Yeniden bağlanma girişiminden önce milisaniye sayısı	ReconnectDelay

MQCBC 'ye Genel Bakış

Kullanılabilirlik: Bu sistemlere bağlı AIX, HP-UX, IBM i, Solaris, Linux, Windows, z/OS, artı IBM MQ MQI clients .

Amaç: MQCBC yapısı, geri bildirme işlevine geçirilen bağlam bilgilerini belirtmek için kullanılır.

Yapı, bir ileti tüketici yordamuna yapılan çağrıdaki bir giriş/çıkış parametresidir.

Sürüm: MQCBC 'nin yürürlükteki sürümü MQCBC_VERSION_2' dir.

Karakter kümesi ve kodlama: MQCBC 'deki veriler, *CodedCharSetId* kuyruk yöneticisi özniteliği tarafından verilen karakter kümesinde ve MQKEN_NATIVE tarafından verilen yerel kuyruk yöneticisinin kodlamasında olmalıdır. Ancak, uygulama bir MQ MQI istemcisi olarak çalışıyorsa, yapı, istemcinin karakter kümesinde ve kodlamasında olur.

MQCBC için alanlar

MQCBC yapısına ilişkin alanların alfabetik listesi.

MQCBC yapısı aşağıdaki alanları içerir; alanlar alfabetik sırayla anlatılır:

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.

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 *HOB* 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.

CallType (MQUZE)

Bu işlevin neden çağrıldığı hakkında bilgi içeren alan; aşağıdaki tanımdır.

İleti teslim çağrısı 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 çağrıldı.

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 çağrıldı.

İleti, *MsgToken* komutunu kullanarak, nesne tanıtıcısından yok edici bir şekilde kaldırılabilir.

İleti şu nedeniyle 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 çağrısı tipleri: Bu çağrı tipleri, geri bildirimin 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 gerçekleştirmesine izin vermesidir.

Geri bildirme işlevi çağrıldığında, geri bildirme kaydedildikten hemen sonra, bu durumda MQCB çağrısından *Operation* alanı için bir değer kullanılarak bir MQCB 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 MQRC_NONE olur.

MQCBCT_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 MQOP_START ya da MQOP_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 MQRC_NONE olur.

MQCBCT_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, MQOP_STOP için *Operation* alanı için bir değer kullanılarak bir MQCTL çağrısı yayınlandığında çağrılır.

Bu arama tipi yalnızca ileti kullanıcıları için kullanılır.

Reason alanının değeri, durmanın nedenini gösterecek şekilde ayarlanır.

MQCBCT_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, MQOP_DEREGISTER ile bir MQCB çağrısı kullanılarak kayıttan kaldırılır.
- Kuyruk kapatıldı, örtülü bir kayıttan çıkarıcı neden oluyor. Bu örnekte, geri bildirme işlevi nesne tanıtıcısı olarak MQHO_UNUABLE_HOBJ ' i iletir.
- 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.

Reason alanının değeri, durmanın nedenini gösterecek şekilde ayarlanır.

MQCBCT_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* = MQCC_FAILED) saptandığında ileti tüketici işlevi çağrılmadan çağrılır; örneğin, *Reason* kodu = MQRC_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.

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.

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 çağırımlarda ortaktır.

Bu bir giriş ve çıkış alanıdır. Bu alanın başlangıç değeri, boş değerli ya da boş değerli bayttır.

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 yayınlatabilmenize 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 kullanılamaması 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.

İşaretler (MQUZE)

Bu tüketiciyle ilgili bilgileri içeren işaretler.

Aşağıdaki seçenek tanımlıdır:

MQBCF_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.

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.

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.

Durum	Kuyruk yöneticisi işlemi	Değişmez değeri
<i>MQCS_NONE</i> Bu neden kodu, ek neden bilgisi olmayan olağan çağrışı gösterir.	Hayır, bu normal bir işlemidir.	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

Durum	Kuyruk yöneticisi işlemi	Değişmezin değeri
<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

StrucId (MQCHAR4)

Bu alandaki değer yapı tanıttırıcısıdır.

Değer şu olmalıdır:

MQCBC_STRUC_ID

Geri çağırma bağlamı yapısına ilişkin tanıttıcı.

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.

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 19 sayfa 284 , olay işleyicisinden dönüşte kuyruk yöneticisinin davranışını değiştirmek için ayarlayabileceğiniz değerleri listeler.

Çizelge 19. ReconnectDelay değerleri		
Ad	Değer	Tanım
MQRD_NO_RECONNECT	-1	Başka bir yeniden bağlantı kurma girişiminde bulunmaz. Uygulamaya bir hata döndürüldü.
MQRD_NO_DELAY	0	Hemen yeniden bağlanmayı deneyin.
Milliseconds	>0	Bağlantıyı yeniden denemeden önce bu süre (milisaniye) beklenir.

MQCBC için ilk değerler ve dil bildirimleri

Geri çağırma bağlamı yapısı-ilk değerler

MQCBC yapısı için başlangıç değeri yoktur. Yapı, bir geri çağrı yordama parametre olarak geçirilir. Kuyruk yöneticisi yapıyı kullanıma hazırlar; uygulamalar bu yapıyı hiçbir zaman başlatmaz.

C bildirimi

Geri bildirme bağlamı yapısı-C dili 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 */
```

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 **
```

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 */

```

High Level Assembler 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
               ORG      MQCBC
MQCBC_AREA     DS CL(MQCBC_LENGTH)

```

MQCBD-Callback tanımlayıcısı

Aşağıdaki çizelge yapıdaki alanları özetlemektedir. Geri bildirme işlevini belirten yapı.

Çizelge 20. MQCBD içindeki alanlar		
Alan	Tanım	Konu
<i>StrucID</i>	Yapı tanıtıcısı	StrucID
<i>Version</i>	Yapı sürüm numarası	S\u00fcr\u00fcm
<i>CallbackType</i>	Geri bildirme işlevi tipi	CallbackType
<i>Options</i>	İleti tüketimini denetleyen seçenekler	Seçenekler
<i>Callback Area</i>	Geri bildirme işlevi için kullanılacak alan	CallbackArea
<i>CallbackFunction</i>	İşleve bir API çağrısı olarak çağrılıp çağrılmayacağı	CallbackFunction
<i>CallbackName</i>	İşleve dinamik olarak bağlantılı bir program olarak çağrılıp çağrılmayacağı	CallbackName
<i>MaxMsgLength</i>	Okunabilecek en uzun iletinin uzunluğu	MaxMsgUzunluğu

MQCBD 'ye Genel Bakış

Kullanılabilirlik: Bu sistemlere bağlı AIX, HP-UX, IBM i, Solaris, Linux, Windows, z/OS ve IBM MQ MQI clients .

Amaç: MQCBD yapısı, bir geri çağırma işlevi ve kuyruk yöneticisi tarafından kullanımını denetleyen seçenekleri belirlemek için kullanılır.

Yapı, MQCB çağrısına ilişkin bir giriş değiştirir.

Sürüm: MQCBD 'nin yürürlükteki sürümü MQCBD_VERSION_1' dir.

Karakter kümesi ve kodlama: MQCBD içindeki veriler, *CodedCharSetId* kuyruk yöneticisi özneliği tarafından verilen karakter kümesinde ve MQKEN_NATIVE tarafından verilen yerel kuyruk yöneticisinin kodlamasında olmalıdır. Ancak, uygulama bir MQ MQI istemcisi olarak çalışıyorsa, yapı istemcinin karakter kümesinde ve kodlamasında olmalıdır.

MQCBD alanları

MQCBD yapısına ilişkin alanların alfabetik listesi.

MQCBD yapısı aşağıdaki alanları içerir; alanlar alfabetik sırayla anlatılır:

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 çağırısı olarak çağrılır.

Geri bildirme işlevine ilişkin bir gösterge belirtmek için bu alanı kullanın.

CallbackFunction ya da *CallbackName* 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 üzerinde işlevin, OS bağ kuralları ile çağırılması 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 MQ V7.0.1, 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* 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.

UNIX sistemleri

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 araç 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 yöneltme 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 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.

CallbackType (MQUZE)

Geri çağırma tanımlayıcı yapısı- CallbackType alanı

Bu, geri bildirme işlevinin tipidir. Değer aşağıdakilerden biri olmalıdır:

MQCBT_MESSAGE_CONSUMER

Bu geri çağırıcı 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 çağırıcı zamanuysuz olay yordamı olarak tanımlar; bir tanıtıcı için iletileri tüketmek için yönlendirilmez.

Olay işleyicisini tanımlayan MQCB çağrısında *Hob j* 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.

MaxMsgUzunluğu (MQUZE)

Bu, tanıtıcıdan okunabilen ve geri çağırma yordamında okunabilen en uzun iletinin bayt cinsinden uzunluğudur. Geri Çağırıcı 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_BAŞARISIZ 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 çağrılır.

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 MQCBD_FULL_MSG_LENGTH değerine sahip.

Seçenekler (MQUZE)

Geri çağırma tanımlayıcı 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 destekliyse).

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.

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.

MQCBD için ilk değerler ve dil bildirimleri

Geri çağırma tanımlayıcısı yapısı-İlk değerler

Çizelge 21. MQCBD içindeki alanların ilk değerleri

Alan adı	Değişmezin adı	Değişmezin değeri
<i>StrucId</i>	MQCBD_STRUC_ID	'CBD↵'
<i>Version</i>	MQCBD_VERSION_1	1
<i>CallBackType</i>	MQCBT_MESSAGE_CONSUMER	1
<i>Options</i>	MQCBDO_NONE	0
<i>CallBackArea</i>	Yok	Boş değerli gösterge ya da boş boşluk
<i>CallBackFunction</i>	Yok	Boş değerli gösterge ya da boş boşluk
<i>CallBackName</i>	Yok	Dizgi ya da boşluk boş değerli
<i>MaxMsgLength</i>	MQCBD_FULL_MSG_LENGTH	-1

Notlar:

1. ↵ simgesi tek bir boş karakteri temsil eder.
2. Boş değerli dizgi ya da boşluk, C programlama dilinde boş değer, diğer programlama dillerindeki boş karakterleri gösterir.
3. C programlama dilinde, makro değişkeniMQCB_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlara ilişkin ilk değerleri sağlamak için bunu aşağıdaki şekilde kullanın:

```
MQCBD MyCBD = {MQCBD_DEFAULT};
```

C bildirimi

Geri çağırma tanımlayıcısı yapısı-C dili bildirimi

```
typedef struct tagMQCBD MQCBD;
struct tagMQCBD {
    MQCHAR4    StrucId;           /* 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 */
};
```

COBOL bildirimi

```
** MQCBDCD 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-MAXMSGLENGTH	PIC S9(9) BINARY.

PL/I bildirimi

```

dcl
1 MQCBD based,
3 StrucId          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 */

```

MQCHARV-Değişken Uzunluklu Dizgi

Aşağıdaki çizelge yapıdaki alanları özetlemektedir.

Alan	Tanım	Konu
<i>VSPtr</i>	Değişken uzunluk dizilimini gösteren gösterge	VSPtr
<i>VSOOffset</i>	Bu MQCHARV yapısını içeren yapının başlangıcındaki değişken uzunluklu dizginin bayt cinsinden görelî konumu	VSOOffset
<i>VSLength</i>	VSPtr ya da VSOOffset alanı tarafından adreslenen değişken uzunluktaki dizilimin bayt cinsinden uzunluğu.	VSKlength
<i>VSBufSize</i>	VSPtr ya da VSOOffset alanı tarafından adreslenen arabelleğin bayt cinsinden büyüklüğü.	VSBufSize
<i>VSCCSID</i>	VSPtr ya da VSOOffset alanı tarafından adreslenen değişken uzunluklu dizilimin karakter takımı tanıtıcısı.	VSCCSID

MQCHARV 'a Genel Bakış

Kullanılabilirlik: Bu sistemlere bağlı AIX, HP-UX, Solaris, Linux, IBM i, Windows, artı IBM MQ MQI clients .

Amaç: Değişken uzunluk dizesini tanımlamak için MQCHARV yapısını kullanın.

Karakter kümesi ve kodlama: MQCHARV içindeki veriler, MQENC_NATIVE tarafından verilen yerel kuyruk yöneticisinin kodlamasında ve yapı içindeki VSCCSID alanının karakter takımında olmalıdır. Uygulama bir MQ istemcisi olarak çalışıyorsa, yapının istemcinin kodlamasında olması gerekir. Bazı karakter kümelerinin, kodlamaya bağlı bir gösterimi vardır. VSCCSID, bu karakter kümelerinden biriye, kullanılan kodlama, MQCHARV içindeki diğer alanlarla aynı kodlamadır. VSCCSID ile tanıtılan 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. COBOL ' un tüm ortamlarda işaretçi veri tipini desteklemediğini unutmayın. Bu yüzden, veriler, MQCHARV içeren yapının başlangıcından elde edilen verilerin görelî konumunu içeren alanlar kullanılarak da adreslenebilir.

COBOL

Ortamlar arasında bir uygulamayı kapı içine almak istiyorsanız, işaretçi veri tipinin tüm amaçlanan ortamlarda kullanılabilir olup olmadığını kesin olarak bilmelisiniz. Yoksa, uygulamanın, işaretçi alanları yerine görelî konum alanlarını kullanarak verileri ele vermeleri gerekir.

İşaretçilerin desteklenmediği ortamlarda, işaretçi alanlarını uygun uzunluğun byte dizgileri olarak bildirebilirsiniz; başlangıç değeri, tüm boş değeri byte dizgisidir. Görelî konum alanlarını kullanıyorsanız,

bu ilk değeri değiştirmeyin. Sağlanan kopya kitaplarını değiştirmeden bunu yapmak için bir yol aşağıda belirtilenlerle birlikte kullanılmalıdır:

```
COPY CMQCHRVV REPLACING POINTER BY ==BINARY PIC S9(9)==.
```

Burada CMQCHRVV , kullanılacak kopya kitabı için değiş tokuş edilebilir.

MQCHARV alanları

MQCHARV yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

VSBufSize (MQUZE)

Bu, VSPtr ya da VSOOffset 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 arayana yalnızca VSBufSize byte veri döndürülebiliyor.

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.

VSCCSID (MQUZA)

Bu, **VSPtr** ya da **VSOOffset** alanı tarafından adreslenen değişken uzunluklu dizginin karakter takımı tanıtıcısıdır.

The initial value of this field is *MQCCSI_APPL*, which is defined by MQ to indicate that it should be changed to the true character set identifier of the current process. Sonuç olarak, *MQCCSI_APPL* değişiminin değeri bir değişken uzunluk dizgisiyle hiçbir zaman ilişkilendirilmemiştir.

Bu alanın ilk değeri, derleme biriminiz için değişmez *MQCCSI_APPL* için farklı bir değer tanımlanarak değiştirilebilir. Bunu nasıl yapacağınız, uygulamanızın programlama diline bağlıdır.

 z/OS sistemlerinde, *MQCCSI_APPL* tarafından kullanılan CCSID varsayılan uygulaması aşağıdaki gibi tanımlıdır:

- DLL arabirimini kullanan toplu iş LE uygulamaları için varsayılan değer, **MQCONN** kütüğünün yürürlükteki ülke değeriyle ilişkilendirilmiş olan CODESET değeridir (varsayılan değer 1047 'dir).
- Toplu iş MQ sınırlı kod öbeklerinden birine bağlı toplu LE uygulamaları için varsayılan değer, **MQCONN** sonrasında verilen ilk MQI çağrısı sırasında yürürlükteki ülke değeriyle ilişkilendirilmiş CODESET değeridir (varsayılan değer 1047 'dir).
- Bir USS iş parçasısında çalışan, toplu kipte olmayan uygulamalar için varsayılan değer, **MQCONN** ' den sonra verilen ilk MQI çağrısı sırasında THLICCSID değeridir (varsayılan değer 1047 'dir).
- Diğer toplu uygulamalar için varsayılan değer, kuyruk yöneticisinin CCSID ' idir.

Güç Gücü (MQUZA)

VSPtr ya da VSOOffset 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, dizeyi 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 (UCS-2 CCSID 1200 ve 13488), 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.

VSOOffset (MQUZE)

Görelî konum pozitif ya da negatif olabilir. Değişken uzunluklu dizgiyi belirtmek için, VSPtr ya da VSOOffset 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.

VSPtr (MQPTR)

Bu, değişken uzunluklu dizilimin göstergesini gösterir.

Değişken uzunluklu dizgiyi belirtmek için, VSPtr ya da VSOOffset alanını kullanabilirsiniz, ancak her ikisini birden belirleyemez.

Bu alanın başlangıç değeri, boş değerli ya da boş değerli bayttır.

MQCHARV için ilk değerler ve dil bildirimleri

MQCHARV içindeki alanların ilk değerleri

Alan adı	Değişmezin adı	Değişmezin değeri
<i>VSPtr</i>	Yok	Boş değerli gösterge ya da boş değerli byte.
<i>VSOOffset</i>	Yok	0
<i>VSBufSize</i>	MQVS_USE_VSLENGTH	0
<i>VSLength</i>	Yok	0
<i>VSCCSID</i>	MQCCSI_APPL	-3

Not: C programlama dilinde, MQCHARV_default makro değişkeni, çizelgede listelenen değerleri içerir. Yapıdaki alanlara ilk değerleri sağlamak için aşağıdaki şekilde kullanılabilir:

```
MQCHARV MyVarStr = {MQCHARV_DEFAULT};
```

C bildirimi

```
typedef struct tagMQCHARV MQCHARV;  
struct tagMQCHARV {  
    MQPTR    VSPtr;                /* Address of variable length string */  
    MQLONG   VSOOffset;            /* 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.
```

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 */
```

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
ORG  MQCHARV
MQCHARV_AREA      DS    CL(MQCHARV_LENGTH)
```

MQCCSI_APPL 'nin yeniden tanımlaması

Aşağıdaki örneklerde, çeşitli programlama dillerinde MQCCSI_APPL değerinin nasıl geçersiz kılınabileceği gösterilmektedir. MQCCSI_APPL değerini değiştirebileceğiniz gibi, her değişken uzunluk dizgisi için VSCCSID 'yi ayrı olarak ayarlama gereksinimini ortadan kaldırın.

Bu örneklerde CCSID, 1208 olarak ayarlanır; bu değeri gerek duyduğunuz değere çevirin. Bu değer, VSCCSID 'yi belirli bir MQCHARV yönetim ortamında ayarlayarak geçersiz kılınabileceğiniz 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);
```

System/390 çevirici kullanımı

MQCCSI_APPL EQU 1208
CMQA LIST=NO

MQCIH- CICS bridge üstbilgisi

Aşağıdaki çizelge yapıdaki alanları özetlemektedir.

Çizelge 22. MQCIH içindeki alanlar		
Alan	Tanım	Konu
<i>StrucId</i>	Yapı tanıtıcısı	StrucId
<i>Version</i>	Yapı sürüm numarası	S\u00fcr\u00fcm
<i>StrucLength</i>	MQCIH yapısının uzunluğu	StrucLength
<i>Encoding</i>	Ayrıldı	Kodlama
<i>CodedCharSetId</i>	Ayrıldı	CodedCharSetId
<i>Format</i>	MQ format name of data that follows MQCIH	Biçim
<i>Flags</i>	İşaretler	İşaretler
<i>ReturnCode</i>	Köprüden dönüş kodu	ReturnCode
<i>CompCode</i>	MQ tamamlama kodu ya da CICS EIBRESP	CompCode
<i>Reason</i>	MQ neden ya da geribildirim kodu ya da CICS EIBRESP2	Neden
<i>UOWControl</i>	İş birimi denetimi	UOWControl
<i>GetWaitInterval</i>	Köprü görevi tarafından MQGET çağrısına ilişkin bekleme aralığı	GetWaitAralığı
<i>LinkType</i>	Bağlantı tipi	LinkType
<i>OutputDataLength</i>	Çıkış COMMAREA veri uzunluğu	OutputDataUzunluğu
<i>FacilityKeepTime</i>	Köprü olanağı serbest bırakma süresi	FacilityKeepSüresi
<i>ADSDescriptor</i>	Gönderme/alma ADS tanımlayıcısı	ADSDescriptor
<i>ConversationalTask</i>	Görevin dönüştürülüp dönüştürülemeyeceği	ConversationalTask
<i>TaskEndStatus</i>	Görevin sonundaki durum	TaskEndDurumu
<i>Facility</i>	Köprü olanağı simgesi	Tesis
<i>Function</i>	MQ arama adı ya da CICS EIBFN işlevi	function
<i>AbendCode</i>	Olağandışı bitiş kodu	AbendCode
<i>Authenticator</i>	Parola ya da geçiş bileti	Kimliği doğrulayıcı
<i>Reserved1</i>	Ayrıldı	Reserved1
<i>ReplyToFormat</i>	MQ yanıt iletisinin adını biçimlendirir	ReplyToBiçimi
<i>RemoteSysId</i>	Kullanmak için uzak CICS sistem tanıtıcısı	RemoteSysTanıtıcısı
<i>RemoteTransId</i>	CICS RTRANSID to use	RemoteTransTanıtıcısı
<i>TransactionId</i>	Bağlanmaya ilişkin hareket	TransactionId

Çizelge 22. MQCIH içindeki alanlar (devamı var)

Alan	Tanım	Konu
<i>FacilityLike</i>	Uçbirim öykünülen öznitelikleri	FacilityLike
<i>AttentionId</i>	AID tuşu	AttentionId
<i>StartCode</i>	İşlem başlatma kodu	StartCode
<i>CancelCode</i>	Olağandışı işlem kodu	CancelCode
<i>NextTransactionId</i>	Eklenecek sonraki hareket	NextTransactionTanıtıcısı
<i>Reserved2</i>	Ayrıldı	Reserved2
<i>Reserved3</i>	Ayrıldı	Reserved3
Not: <i>Version</i> , MQCIH_VERSION_2değerinden küçükse, geri kalan alanlar yoktur.		
<i>CursorPosition</i>	İmleç konumu	CursorPosition
<i>ErrorOffset</i>	İletide hata görelî konumu	ErrorOffset
<i>InputItem</i>	Ayrıldı	InputItem
<i>Reserved4</i>	Ayrıldı	Reserved4

MQCIH ' ye Genel Bakış

MQCIH yapısı, CICS bridgeiçinde CICS ' e gönderilen bir iletiye ilişkin üstbilgi bilgilerini açıklar. For any IBM MQ supported platform you can create and transmit a message that includes the MQCIH structure, but only an IBM MQ for z/OS queue manager can use the CICS bridge. Therefore, for the message to get to CICS from a non-z/OS queue manager, your queue manager network must include at least one z/OS queue manager through which the message can be routed.

Kullanılabilirlik: Bu sistemlere bağlı AIX, HP-UX, z/OS, Solaris, Linux, Windows, artı IBM MQ MQI clients .

Amaç: MQCIH yapısı, CICS bridge ile IBM MQ for z/OSarasında gönderilen bir iletinin başlangıcında var olabilecek bilgileri açıklar.

Biçim adı: MQFMT_CICS.

Sürüm: MQCIH 'nin yürürlükteki sürümü MQCIH_VERSION_2' dir. Yalnızca yapının daha yeni sürümünde var olan alanlar, izleyen açıklamalarda olduğu gibi tanımlanır.

The header, COPY, and INCLUDE files provided for the supported programming languages contain the most-recent version of MQCIH, with the initial value of the *Version* field set to MQCIH_VERSION_2.

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.
- Diğer kuyruk yöneticilerine bağlanan uygulamalar, desteklenen karakter kümelerinin ve kodlamaların herhangi birinde bulunan bir MQCIH yapısı sağlayabilir; CICS bridge kuyruğunun sahibi olan kuyruk yöneticisine bağlı alan ileti kanalı aracısınin, MQCIH yapısını dönüştüren bir MQCIH yapısı olması gerekir.
- MQCIH yapısından sonra gelen uygulama iletisi verileri, MQCIH yapısıyla aynı karakter kümesinde ve kodlamada olmalıdır. MQCIH yapısındaki *CodedCharSetId* ve *Encoding* alanlarını, uygulama iletisi verilerinin karakter kümesini ve kodlamasını belirtmek için kullanamazsınız.

Veri, kuyruk yöneticisi tarafından desteklenen yerleşik biçimlerden biri değilse, uygulama iletisi verilerini dönüştürmek için bir veri dönüştürme çıkışı sağlamanız gerekir.

Kullanım: Uygulama, Çizelge 24 sayfa 306' ta gösterilen ilk değerlerle aynı olan değerleri gerektiriyorsa ve köprü AUTH=LOCAL ya da AUTH=TANT ile çalışıyorsa, MQCIH yapısını iletiden çıkarabilirsiniz. Diğer tüm durumlarda, yapının mevcut olması gerekir.

Köprü, version-1 ya da bir version-2 MQCIH yapısını kabul eder, ancak 3270 işlemleri için bir version-2 yapısı kullanmanız gerekir.

Uygulama, istek alanları olarak belgelenen alanların, köprüye gönderilen iletide uygun değerlere sahip olmasını sağlamalıdır; bu alanlar köprüye girişlerdir.

Yanıt alanları olarak belgelenmiş alanlar, köprünün uygulamaya gönderdiği yanıt iletisinde CICS bridge tarafından ayarlanır. Error information is returned in the *ReturnCode*, *Function*, *CompCode*, *Reason*, and *AbendCode* fields, but not all of them are set in all cases. Çizelge 23 sayfa 298 , farklı *ReturnCode* değerleri için hangi alanların belirlendiğini gösterir.

Çizelge 23. MQCIH için MQCIH yapısındaki hata bilgileri alanlarının içeriği				
ReturnCode	Function	CompCode	Reason	AbendCode
MQCRC_OK	-	-	-	-
MQCRC_BRIDGE_ERROR	-	-	MQFB_CICS_*	-
MQCRC_MQ_API_ERROR MQCRC_BRIDGE_TIMEOUT	MQ arama adı	MQ CompCode	MQ Reason	-
MQCRC_CICS_EXEC_ERROR MQCRC_SECURITY_ERROR MQCRC_PROGRAM_NOT_VAR MQCRC_TRANSID_NOT_VAR	CICS EIBFN	CICS EIBRESP	CICS EIBRESP2	-
MQCRC_BRIDGE_ABEND MQCRC_APPLICATION_ABEND	-	-	-	CICS ABCODE

MQCIH için alanlar

MQCIH yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

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.

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.

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.

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.

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.

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.

CompCode (MQUZN)

Bu alan bir yanıt alanıdır. İlk değeri MQCC_OK değeridir.

Bu alanda döndürülen değer *ReturnCode* ' a bağlıdır; bkz. [Çizelge 23 sayfa 298](#).

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.

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.

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.

ErrorOffset (MQUZE)

ErrorOffset alanı, köprü çıkışı tarafından saptanan geçersiz verilerin konumunu gösterir. Bu alan, iletinin başlangıcından, geçersiz verilerin yerine göreli 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.

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 iletisinde 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.

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.

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.

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.

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 kurallarla 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 CSQCBDC0 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.

İşlev (MQCHAR4)

Bu alan bir yanıt alanıdır. Bu alanın uzunluğu MQ_FUNCTION_LENGTH tarafından verilir. Bu alanın ilk değeri MQCFUNC_NONE olur.

Bu alanda döndürülen değer *ReturnCode* ' a bağlıdır; bkz. [Çizelge 23 sayfa 298](#). *Function* bir IBM MQ arama adı içerdiğinde aşağıdaki değerler kullanılabilir:

MQCFUNC_MQCONN

MQCONN çağrısı.

MQCFUNC_MQGET

MQGET çağrısı.

MQCFUNC_MQINQ

MQINQ çağrısı.

MQCFUNC_MQAC

MQOPEN çağrısı.

MQCFUNC_MQPUT

MQPUT çağrısı.

MQCFUNC_MQPUT1

MQPUT1 çağrısı.

MQCFUNC_NONE

Arama yok.

Tüm durumlarda, C programlama dili için, MQCFUNC_ * _ARRAY değişmezleri de tanımlanır; bu değişmezler, karşılık gelen MQCFUNC_ * değişmezleriyle aynı değerlere sahiptir, ancak dizgiler yerine karakterler dizileridir.

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.

InputItem (MQUZE)

Bu alan ayrılmış bir alandır. Değer 0 olmalıdır.

Version , MQCIH_VERSION_2değerinden küçükse bu alan yoktur.

LinkType (MQUZE)

Bu alan bir istek alanıdır. İlk değeri MQCLT_PROGRAR değeridir.

Bu değer, köprünün bağlanmayı denediği nesnenin tipini belirtir. Bu değer aşağıdaki değerlerden biri olmalıdır:

MQCLT_PROGRAM

DPL programı.

MQCLT_TRANSACTION

3270 hareketi.

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.

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.

Neden (MQUZE)

Bu alan bir yanıt alanıdır. İlk değeri MQRC_NONE olur.

Bu alanda döndürülen değer *ReturnCode* ' a bağlıdır; bkz. [Çizelge 23 sayfa 298](#).

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 iletilerinde döndürülür.

3270 sözde sohbetinde, sohbetteki sonraki tüm iletiler, ilk yanıtta döndürülen uzak SYSID ' yi belirtmelidir. Belirtilirse, 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.

Belirtilirse, alan, CICS START ' nin RTRANSID değeri olarak kullanılır.

ReplyToBiçimi (MQCHAR8)

Bu alanın değeri, yürürlükteki iletiye yanıt olarak gönderilen yanıt iletilerinin 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.

Reserved1 (MQCHAR8)

Bu alan ayrılmış bir alandır. Değer 8 boşluk olmalıdır.

Reserved2 (MQCHAR8)

Bu alan ayrılmış bir alandır. Değer 8 boşluk olmalıdır.

Reserved3 (MQCHAR8)

Bu alan ayrılmış bir alandır. Değer 8 boşluk olmalıdır.

Reserved4 (MQUZE)

Bu alan ayrılmış bir alandır. Değer 0 olmalıdır.

Version , MQCIH_VERSION_2değerinden küçükse bu alan yoktur.

ReturnCode (MQUZE)

Bu alanın değeri, köprü tarafından gerçekleştirilen işlemin sonucunu açıklayan CICS bridge ' den gelen dönüş kodudur. Bu alan bir yanıt alanıdır ve MQCRC_OK ilk değerine sahip olur.

Function , *CompCode* , *Reasonve AbendCode* alanları ek bilgi içerebilir (bkz. [Çizelge 23 sayfa 298](#)).

Değer aşağıdakilerden biridir:

MQCRC_APPLICATION_ABEND

(5, X'005 ') Uygulama olağandışı sona erdi.

MQCRC_BRIDGE_ABEND

(4, X'004 ') CICS bridge olağandışı şekilde sona erdi.

MQCRC_BRIDGE_ERROR

(3, X'003 ') CICS bridge bir hata algıladı.

MQCRC_BRIDGE_TIMEOUT

(8, X'008 ') Yürürlükteki iş birimi içinde ikinci ya da daha sonraki bir ileti belirlenen süre içinde alınmadı.

MQCRC_CICS_EXEC_ERROR

(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_NOT_YOK

(7, X'007 ') Program kullanılmıyor.

MQCRC_SECURITY_ERROR

(6, X'006 ') Güvenlik hatası oluştu.

MQCRC_TRANSID_NOT_VAR

(9, X'009 ') İşlem kullanılmıyor.

StartCode (MQCHAR4)

Bu alanın değeri, köprünün 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 Sürüm 1.2 için, bu alan yalnızca bir istek alanıdır; yanıtta değeri tanımsız olur.

CICS Transaction Server Sürüm 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.

StrucId (MQCHAR4)

Bu alan, ilk MQCIH_STRUC_ID değeri olan bir istek alanıdır.

Değer şu olmalıdır:

MQCIH_STRUC_ID

CICS bilgi üstbilgisi yapısına ilişkin tanıtıcı.

C programlama dili için, MQCIH_STRUC_ID_ARRAY değişmezi de tanımlanır; bu, MQCIH_STRUC_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisidir.

StrucLength (MQUZE)

Bu alan, başlangıç değeri olan MQCIH_LENGTH_2 olan bir istek alanıdır.

Değer aşağıdakilerden biri olmalıdır:

MQCIH_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.

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.

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.

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).

Sürüm (MQUZE)

Bu alan bir istek alanıdır. İlk değeri MQCIH_VERSION_2' dir.

Değer aşağıdakilerden biri olmalıdır:

MQCIH_VERSION_1

Version-1 CICS bilgi üstbilgisi yapısı.

MQCIH_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:

MQCIH_CURRENT_VERSION

CICS bilgi üstbilgisi yapısının yürürlükteki sürümü.

MQCIH için ilk değerler ve dil bildirimleri

Çizelge 24. MQCIH için MQCIH ' de alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
<i>StrucId</i>	MQCIH_STRUC_ID	' CIH '
<i>Version</i>	MQCIH_VERSION_2	2
<i>StrucLength</i>	MQCIH_LENGTH_2	180
<i>Encoding</i>	Yok	0
<i>CodedCharSetId</i>	Yok	0
<i>Format</i>	MQFMT_NONE	Boşluklar
<i>Flags</i>	MQCIH_NONE	0
<i>ReturnCode</i>	MQCRC_OK	0
<i>CompCode</i>	MQCC_OK	0
<i>Reason</i>	MQRC_NONE	0
<i>UOWControl</i>	YALNIZCIK	273
<i>GetWaitInterval</i>	MQCGWI_VARSAYILAN	-2
<i>LinkType</i>	MQCLT_PROGRAM	1

Çizelge 24. MQCIH için MQCIH ' de alanların ilk değerleri (devamı var)

Alan adı	Değişmezin adı	Değişmezin değeri
<i>OutputDataLength</i>	MQCODL_AS_INPUT	-1
<i>FacilityKeepTime</i>	Yok	0
<i>ADSDescriptor</i>	MQCADSD_NONE	0
<i>ConversationalTask</i>	MQCCT_NO	0
<i>TaskEndStatus</i>	MQCTES_NOSYNC	0
<i>Facility</i>	MQCFAC_NONE	Boş Değerler
<i>Function</i>	MQCFUNC_NONE	Boşluklar
<i>AbendCode</i>	Yok	Boşluklar
<i>Authenticator</i>	Yok	Boşluklar
<i>Reserved1</i>	Yok	Boşluklar
<i>ReplyToFormat</i>	MQFMT_NONE	Boşluklar
<i>RemoteSysId</i>	Yok	Boşluklar
<i>RemoteTransId</i>	Yok	Boşluklar
<i>TransactionId</i>	Yok	Boşluklar
<i>FacilityLike</i>	Yok	Boşluklar
<i>AttentionId</i>	Yok	Boşluklar
<i>StartCode</i>	MQCSC_NONE	Boşluklar
<i>CancelCode</i>	Yok	Boşluklar
<i>NextTransactionId</i>	Yok	Boşluklar
<i>Reserved2</i>	Yok	Boşluklar
<i>Reserved3</i>	Yok	Boşluklar
<i>CursorPosition</i>	Yok	0
<i>ErrorOffset</i>	Yok	0
<i>InputItem</i>	Yok	0
<i>Reserved4</i>	Yok	0

Notlar:

1. ↵ simgesi tek bir boş karakteri temsil eder.
2. C programlama dilinde, makro değişkeniMQCIH_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlara ilişkin ilk değerleri sağlamak için bunu aşağıdaki şekilde kullanın:

```
MQCIH MyCIH = {MQCIH_DEFAULT};
```

C bildirimi

```
typedef struct tagMQCIH MQCIH;
struct tagMQCIH {
    MQCHAR4 StrucId;          /* Structure identifier */
    MQLONG Version;          /* Structure version number */
};
```

```

MQLONG  StrucLength;      /* Length of MQCIH structure */
MQLONG  Encoding;        /* Reserved */
MQLONG  CodedCharSetId;   /* Reserved */
MQCHAR8  Format;          /* MQ format name of data that follows
                           MQCIH */
MQLONG  Flags;           /* Flags */
MQLONG  ReturnCode;       /* Return code from bridge */
MQLONG  CompCode;        /* MQ completion code or CICS EIBRESP */
MQLONG  Reason;          /* MQ reason or feedback code, or CICS
                           EIBRESP2 */
MQLONG  UOWControl;      /* Unit-of-work control */
MQLONG  GetWaitInterval; /* Wait interval for MQGET call issued
                           by bridge task */
MQLONG  LinkType;        /* Link type */
MQLONG  OutputDataLength; /* Output COMMAREA data length */
MQLONG  FacilityKeepTime; /* Bridge facility release time */
MQLONG  ADSDescriptor;    /* Send/receive ADS descriptor */
MQLONG  ConversationalTask; /* Whether task can be conversational */
MQLONG  TaskEndStatus;    /* Status at end of task */
MQBYTE8  Facility;       /* Bridge facility token */
MQCHAR4  Function;       /* MQ call name or CICS EIBFN
                           function */
MQCHAR4  AbendCode;      /* Abend code */
MQCHAR8  Authenticator;  /* Password or passticket */
MQCHAR8  Reserved1;      /* Reserved */
MQCHAR8  ReplyToFormat;   /* MQ format name of reply message */
MQCHAR4  RemoteSysId;     /* Reserved */
MQCHAR4  RemoteTransId;   /* Reserved */
MQCHAR4  TransactionId;   /* Transaction to attach */
MQCHAR4  FacilityLike;    /* Terminal emulated attributes */
MQCHAR4  AttentionId;     /* AID key */
MQCHAR4  StartCode;       /* Transaction start code */
MQCHAR4  CancelCode;      /* Abend transaction code */
MQCHAR4  NextTransactionId; /* Next transaction to attach */
MQCHAR8  Reserved2;      /* Reserved */
MQCHAR8  Reserved3;      /* Reserved */
MQLONG  CursorPosition;   /* Cursor position */
MQLONG  ErrorOffset;      /* Offset of error in message */
MQLONG  InputItem;       /* Reserved */
MQLONG  Reserved4;       /* Reserved */
};

```

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-STRUCLNGTH      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-REMOtetTRANSID 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.

```

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 */

```

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_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)

```

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		

MQCMHO-İleti tanıtıcısı yaratma seçenekleri

Aşağıdaki çizelge yapıdaki alanları özetlemektedir.

Çizelge 25. MQCMHO içindeki alanlar		
Alan	Tanım	Konu
<i>StrucId</i>	Yapı tanıtıcısı	StrucId
<i>Version</i>	Yapı sürüm numarası	S\u00fcr\u00fcm
<i>Options</i>	Seçenekler	Seçenekler

MQCMHO için genel bakış

Kullanılabilirlik: AIX, HP-UX, IBM i, Solaris, Linux, Windows, z/OS ve IBM MQ istemcileri.

Amaç: MQCMHO yapısı, uygulamaların ileti tutamaçlarının nasıl yaratıldığını denetleyen seçenekleri belirlemesine olanak sağlar. Yapı, **MQCRTMH** çağrısındaki bir giriş parametresidir.

Karakter kümesi ve kodlama: MQCMHO içindeki veriler, uygulamanın uygulama ve kodlamasının karakter kümesinde (**MQENC_NATIVE**) olmalıdır.

MQCMHO için alanlar

MQCMHO yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

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
 - VEYA
 - 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 hiçbirini 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.

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.

MQCMHO için ilk değerler ve dil bildirimleri

Çizelge 26. MQCMHO 'daki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
StrucId	MQCMHO_STRUC_ID	' CMHO '
Version	MQCMHO_VERSION_1	1
Options	MQCMHO_DEFAULT_VALIDATION	0
Notlar:		
1. C programlama dilinde, makro değişkeniMQCMHO_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlara ilk değerleri sağlamak için aşağıdaki şekilde kullanılabilir:		
<pre>MQCMHO MyCMHO = {MQCMHO_DEFAULT};</pre>		

C bildirimi

```
struct tagMQCMHO {
    MQCHAR4    StrucId;           /* Structure identifier */
    MQLONG     Version;          /* Structure version number */
    MQLONG     Options;          /* Options that control the action of MQCRTMH */
};
```

COBOL bildirimi

```
** 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.
```

PL/I bildirimi

```
dc1
```

```

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 */

```

High Level Assembler bildirimi

```

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)

```

MQCNO-Bağlantı seçenekleri

Aşağıdaki çizelge yapıdaki alanları özetlemektedir.

Çizelge 27. MQCNO içindeki alanlar		
Alan	Tanım	Konu
<i>StrucId</i>	Yapı tanıtıcısı	StrucId
<i>Version</i>	Yapı sürüm numarası	S\u00fcr\u00fcm
<i>Options</i>	MQCONNX işleminin denetlenmesini denetleyen seçenekler	Seçenekler
Not: <i>Version</i> , MQCNO_VERSION_2değerinden küçükse, geri kalan alanlar yoksayılır.		
<i>ClientConnOffset</i>	İstemci bağlantısı için MQCD yapısının göreceli konumu	ClientConnKayması
<i>ClientConnPtr</i>	İstemci bağlantısı için MQCD yapısının adresi	ClientConnPtr
Not: <i>Version</i> , MQCNO_VERSION_3değerinden küçükse, geri kalan alanlar yoksayılır.		
<i>ConnTag</i>	Kuyruk yöneticisi bağlantısı etiketi	ConnTag
Not: <i>Version</i> , MQCNO_VERSION_4değerinden küçükse, geri kalan alanlar yoksayılır.		
<i>SSLConfigPtr</i>	İstemci bağlantısı için MQSCO yapısının adresi	SSLConfigPtr
<i>SSLConfigOffset</i>	İstemci bağlantısı için MQSCO yapısının göreceli konumu	SSLConfigOffset
Not: <i>Version</i> , MQCNO_VERSION_5değerinden küçükse, geri kalan alanlar yoksayılır.		
<i>ConnectionId</i>	Benzersiz Bağlantı Tanıtıcısı	ConnectionId
<i>SecurityParmsOffset</i>	Güvenlik değiştiricileri	SecurityParmsKayması
<i>SecurityParmsPtr</i>	Güvenlik değiştiricileri	SecurityParmsPtr 'si

İlgili bilgiler

[MQCONN olanağının kullanılması](#)

MQCNO ile ilgili genel bakış

Kullanılabilirlik: MQCNO_VERSION_4: AIX, HP-UX, IBM i, Solaris, Linux, Windows, artı IBM MQ MQI clients dışındaki tüm sürümler bu sistemlere bağlı olarak.

Amaç: MQCNO yapısı, uygulamanın yerel kuyruk yöneticisiyle bağlantıyla ilgili seçenekleri belirlemesine olanak sağlar. Yapı, MQCONNX çağrısında bir giriş/çıkış değiştirgesi. Paylaşılan çekme noktaları ve MQCONNX çağrısıyla ilgili daha fazla bilgi için bkz. [Shared \(thread independent\) connections with MQCONNX](#).

Sürüm: Desteklenen programlama dilleri için sağlanan üstbilgi, COPY ve INCLUDE dosyaları, MQCNO 'nun en son sürümünü içerir; ancak, *Version* alanının ilk değeri MQCNO_VERSION_1olarak ayarlanmış olmalıdır. version-1 yapısında mevcut olmayan alanları kullanmak için, uygulamanın *Version* alanını, gereken sürümün sürüm numarasına ayarlaması gerekir.

Karakter kümesi ve kodlama: MQCNO içindeki veriler, *CodedCharSetId* kuyruk yöneticisi özniteliği tarafından verilen karakter kümesinde ve MQKEN_NATIVE tarafından verilen yerel kuyruk yöneticisinin kodlamasıyla olmalıdır. Ancak, uygulama bir IBM MQ MQI clientolarak çalışıyorsa, yapı istemcinin karakter kümesinde ve kodlamasında olmalıdır.

MQCNO alanları

MQCNO yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

ClientConnGörelİ Konumu (MQUZE)

ClientConnGörelİ Konumu, MQCNO yapısının başlangıcındaki bir MQCD kanal tanımlama yapısının bayt cinsinden görelİ konutdur. Görelİ 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 değerli göstergedir ve tersi durumda, boş değerli 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 diğerini 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 çizelgesi kullanılır.

Uygulama IBM MQ MQI client, *ClientConnOffset* ve *ClientConnPtr* olarak çalışmazsa, yoksayılır.

Uygulama bir MQCD yapısı sağlıyorsa, alanları gereken değerlere ayarlayın; MQCD 'deki diğer alanlar yoksayılır. Karakter dizimlerini, alanın uzunluğuna göre boşluklarla doldurur ya da boş değerli bir karakterle sonlandırabilirsiniz. MQCD yapısındaki alanlarla ilgili ek bilgi için [“Alanlar” sayfa 1498](#) belgesine bakın.

MQCD 'deki alan	Değer
<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.

MQCD 'deki alan	Değer
<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 bildirimi akışları arasındaki süre (saniye).
<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>	SSL CipherSpec.
<i>SSLPeerNamePtr</i>	SSL eşdüzey adının adresi.
<i>SSLPeerNameLength</i>	SSL eş adı uzunluğu.
<i>KeepAliveInterval</i>	Kanala ilişkin canlı tutma zamanlaması için iletişim yığınının geçirilen değer

MQCD 'deki alan**Değer***LocalAddress*

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 göreceli konumuna ayarlaması gerekir. Bu göreceli 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_2 değerinden 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.

ConnectionId (MQBYTE24)

ConnectionId , IBM MQ ' un bir uygulamayı güvenilir bir şekilde tanıtmayı 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*, *MsgID* ve *GroupId* gibi 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.

ConnTag (MQBYTE128)

ConnTag , kuyruk yöneticisinin bu bağlantı sırasında uygulamadan etkilenen kaynaklarla ilişkilendirdiği bir etikettir. Kuyruk yöneticisinin etkilenen kaynaklara erişimi doğru olarak diziselleştirebilmesi için, her uygulama ya da uygulama eşgörünümünün etiket için farklı bir değer kullanması gerekir. Bu alan bir giriş alanıdır ve başlangıçtaki değeri MQCT_NONE olur.

Farklı uygulamalar tarafından kullanılacak değerlere ilişkin ek ayrıntılar için, MQCNO_*_CONN_TAG_* seçeneklerinin açıklamalarına bakın. Etiket, uygulama sona erdirildiğinde ya da MQDISC çağrısını yayınlarken geçerli olacak şekilde durur.

Not: ASCII ya da EBCDIC 'de MQ ile başlayan bağlantı biçim imi değerleri, IBM ürünleri tarafından kullanılmak üzere ayrılmıştır ya da EBCDIC ya da EBCDIC' de ayrılır. Bu harflerle başlayan bağlantı biçim imi değerlerini kullanmayın.

Etiket gerektirmiyorsa aşağıdaki özel değeri kullanın:

MQCT_NONE

Değer, alanın uzunluğuna ilişkin ikili sıfır değeridir.

C programlama dili için, sabit MQCT_NONE_DIZISI de tanımlanır; bu değişmez, MQCT_NONE ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisidir.

Bu alan, bir z/OS kuyruk yöneticisine bağlanılırken kullanılır. Diğer ortamlarda MQCT_NONE değerini belirtin.

Bu alanın uzunluğu MQ_CONN_TAG_LENGTH tarafından verilmektedir. *Version* , MQCNO_VERSION_3değerinden küçükse bu alan yok sayılır.

Seçenekler (MQUZE)

MQCONN'ın işlemini denetleyen seçenekler.

Hesap seçenekleri

Aşağıdaki seçenekler, *AccountingConnOverride* kuyruk yöneticisi özniteliği MQMON_ENABLED değerine ayarlıysa, muhasebe tipini denetler:

MQCNO_ACCOUNTING_MQI_ENABLED

MQIAccounting özniteliğini MQMON_OFF olarak ayarlayarak, kuyruk yöneticisi tanımlamasında veri toplama oturumu kapatıldığında, bu işaret, MQI muhasebe verileri toplamasına olanak sağlar.

MQCNO_ACCOUNTING_MQI_DISABLLANT

MQIAccounting özniteliğini MQMON_OFF olarak ayarlayarak, kuyruk yöneticisi tanımlamasında veri toplama oturumu kapatıldığında, bu işaretin ayarlanması MQI muhasebe verileri derlemi durdurur.

MQCNO_ACCOUNTING_Q_ENABLED

Kuyruk yöneticisi tanımlamasında kuyruk-muhasebe verileri derlemi, *MQIAccounting* özniteliğini MQMON_OFF değerine ayarlayarak, kuyruk yöneticisi tanımlamasında kapalıysa, bu işaret ayarlandığında, kuyruk tanımlamasının *MQIAccounting* alanında bir kuyruk yöneticisi belirten kuyruklar için hesap verileri derlemi geçerli olur.

MQCNO_ACCOUNTING_Q_DISABET

Kuyruk yöneticisi tanımlamasında kuyruk-muhasebe verileri derlemi, *MQIAccounting* özniteliğini MQMON_OFF değerine ayarlayarak, kuyruk yöneticisi tanımlamasında kapalıysa, bu işaret, kuyruk tanımlamasının *MQIAccounting* alanında bir kuyruk yöneticisi belirten kuyruklar için muhasebe verileri derleminin kapatılmasına neden olur.

Bu işaretlerin hiçbirisi 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 destekliyorsa ve MQCNO_STANDARD_BINDING değerini belirlerseniz, kuyruk yöneticisi gerçek bağ tanımı tipini seçmek için `qm.ini` dosyasındaki `Connection` 'ta `DefaultBindType` özneliğini 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.

- MQ_CONNECT_TYPE ortam değişkeni belirtilirse, MQCB_STANDARD_BINDING belirtilirse, MQCONN ya da MQCONNX 'in davranışını değiştirmek için aşağıdaki seçeneklerden biri sağlanabilir. (Bu kural dışı durum, MQCNO_FASTPATH_BINDING MQ_CONNECT_TYPE ile LOCAL ya da STANDARD değerine ayarlandıysa, uygulama için ilgili bir değişiklik yapılmaksızın, fastpath bağlantılarının yönetici tarafından düşürülmesine izin verir:

Değer	Anlamı
CLIENT	Yalnızca istemci bağlantısı denenişir.
FastPath	Bu değeri önceki yayınlarda desteklendi, ancak belirtilirse yoksayılr.
LOCAL	Yalnızca sunucu bağlantısı denendi. Fastpath bağlantıları, standart bir sunucu bağlantısına düşürülebiliyor.
Standart	Önceki yayın düzeyleriyle uyumluluk için desteklenir. Bu değeri şimdi LOCAL olarak ele alınır.

- MQCONNX çağrıldığında MQ_CONNECT_TYPE ortam değışkeni ayarlanmazsa, varsayılan bağ tanımlama tipini kullanan bir standart sunucu bağlantısı girişiminde bulunulması denenişir. 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 yoksayılr; 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.
- UNIX sistemleri 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 üzerinde program, mqm grubunun bir üyesi olmalıdır. Fastpath bağ tanımı 64 bit uygulamaları için desteklenmez.

Şu ortamlarda MQCNO_FASTPATH_BINDING seçeneği desteklenir: AIX, HP-UX, IBM i, Solaris, Linux ve Windows.

 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:

-  MQCNO_CLIENT_BINDING, z/OS üzerinde yoksayıldı.
- MQCNO_CLIENT_BINDING, MQCNO_STANDARD_BINDING dışında herhangi bir MQCNO bağ tanımı seçeneğiyle belirtildiyse, 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:

-  MQCNO_LOCAL_BINDING, z/OS üzerinde yoksayıldı.

- MQCNO_RECONNECT_AS_DEF dışında herhangi bir MQCNO yeniden bağlanma seçeneğiyle belirtilirse, MQCNO_LOCAL_BINBIND, 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 AIX, HP-UX, Solaris, Linux, and Windows, 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. 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 verdiyseniz, fastpath bağ tanımı yalnızca *her ikisi* ortam değişkeni ve *Options* alanı, fastpath bağ tanımını belirtirse kullanılır.

Bağlantı biçim imi seçenekleri



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:

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. Çağrının sonucu, yerel kuyruk yöneticisinin ait olduğu kuyruk paylaşım grubunun başka bir yerindeki bağlantı etiketi kullanılarak etkilenmez.

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.

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. Arama işleminin sonucu, bağlantı etiketinin yerel kuyruk yöneticisinin ait olduğu kuyruk paylaşım grubunun başka bir yerinde 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.

MQCNO_RESTRICT_CONN_TAG_QSG

Bu seçenek, yerel kuyruk yöneticisinin ait olduğu kuyruk paylaşım grubu içindeki bağlantı etiketinin paylaşımlı 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

Bu seçenekler şu ortamlarda desteklenir: AIX, HP-UX, IBM i, Solaris, Linux ve 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) kullanılabileceğini 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 kullanılabileceği anlamına gelir. Ancak, her defasında yalnızca bir iş parçacığı belirli bir tanıtıcıyı kullanabilir; yani, yalnızca bir tutamaçla dizesel 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 tamamlanıncaya kadar MQCC_FAILED ve MQRC_CALL_IN_PROGRESS ile tamamlanıncaya kadar arama tamamlanır.

Bir iş parçacığında 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 parçacığında güncellenebilir.

Bir tutamaç paylaşımı seçeneği belirtmezseniz, varsayılan değer ortam tarafından belirlenir:

- 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, MQCONNXparametresinin 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 Javatarafından 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 paylaşılabileceğini gösteriyorsa, ancak kanalın sunucu bağlantısı ucundaki *SharingConversations* (SHARECNV) tanımlaması bir paylaşımaya ayarlıdır, paylaşım oluşmaz ve uygulamaya hiçbir uyarı verilmez.

Similarly, if the application indicates that sharing is permitted but the server-connection *SharingConversations* definition is set to zero, no warning is given, and the application exhibits the same behavior as a client in versions of the product earlier than Sürüm 7.0; the application setting relating to sharing conversations is ignored.

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 yoksayılır.

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 yoksayılır.

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.

SecurityParmsGörelili Konumu (MQUZE)

SecurityParmsGörelili Konumu, MQCNO yapısının başlangıcındaki MQCSP yapısının bayt cinsinden görelili konudur. Görelili 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 değiştirgeleri” sayfa 328içinde tanımlıdır.

SecurityParmsPtr (PMQCSP)

SecurityParmsPtr, yetkilendirme hizmeti tarafından kimlik doğrulaması için bir kullanıcı kimliği ve parola belirlemek üzere kullanılan MQCSP yapısının adresidir. Bu alan bir giriş alanıdır ve başlangıç değeri boş değerli ya da boş değerli bayttır.

Sürüm , MQCNO_VERSION_5değerinden küçükse bu alan yok sayılır.

MQCSP yapısı “MQCSP-Güvenlik değiştirgeleri” sayfa 328içinde tanımlıdır.

SSLConfigOffset (MQUZE)

SSLConfigOffset , MQCNO yapısının başlangıcındaki bir MQSCO yapısının bayt cinsinden görelili konudur. Görelili 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.

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* yoksayılır.

SSLConfigPtr ya da *SSLConfigOffset*değerini belirterek, artı *ClientConnPtr* ya da *ClientConnOffset*belirtilerek, uygulama istemci bağlantısı için SSL kullanımını denetleyebilir. SSL bilgileri bu şekilde belirtildiğinde, MQSSLKEYR ve MQSSLCRYP ortam değişkenleri yoksayılır; istemci kanal tanımlama çizelgesindeki (CCDT) SSL ile ilgili tüm bilgiler de dikkate alınmaz.

SSL bilgileri yalnızca aşağıdaki yerde belirtilebilir:

- İstemci işlemine ilişkin ilk MQCONN çağrısı ya da
- Kuyruk yöneticisine önceki tüm SSL/TLS bağlantıları MQDISC kullanılarak sonuçlanmış olduğunda, izleyen bir MQCONN çağrısı sonuçlandırıldı.

Bu, süreç genelindeki SSL ortamının kullanıma hazırlanabileceği tek durumlardır. SSL ortamı zaten varsa SSL bilgilerini belirten bir MQCONN çağrısı yayınlanırsa, çağrıdaki SSL bilgileri yoksayılır ve bağlantı, var olan SSL ortamı kullanılarak yapılır; çağrı, bu durumda MQCC_UYARI tamamlanma kodunu döndürür ve MQRC_SSL_ALREADY_INITIALLY 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_4değerinden küçükse bu alan yok sayılır.

Not: Programlama dilinin işaretçi veri tipini desteklemediği altyapılarda, bu alan uygun uzunluğun byte dizisi olarak bildirilir.

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 (MQUZE)

Sürüm her zaman bir giriş alanıdır. İlk değeri MQCNO_VERSION_1' dir.

Değer aşağıdakilerden biri olmalıdır:

MQCNO_VERSION_1

Version-1 bağ-seçenekleri yapısı.

MQCNO_VERSION_2

Version-2 bağ-seçenekleri yapısı.

MQCNO_VERSION_3

Version-3 bağ-seçenekleri yapısı.

MQCNO_VERSION_4

Version-4 bağlanma seçenekleri yapısı.

MQCNO_VERSION_5

Version-5 bağlantı seçenekleri yapısı.

MQCNO yapısının bu sürümü, z/OS'ta MQCNO_VERSION_3 ' i ve diğer tüm altyapılarda MQCNO_VERSION_4 ' i genişletir.

Yalnızca yapının daha yeni sürümlerinde var olan alanlar, alanların açıklamalarında olduğu gibi tanımlanır. Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

MQCE_CURRENT_VERSION

Connect-options yapısının yürürlükteki sürümü.

MQCNO için ilk değerler ve dil bildirimleri

Çizelge 28. MQCNO için MQCNO içindeki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
<i>StrucId</i>	MQCNO_STRUC_ID	' CNO↵ '
<i>Version</i>	MQCNO_VERSION_1	1
<i>Options</i>	MQCNO_NONE	0
<i>ClientConnOffset</i>	Yok	0
<i>ClientConnPtr</i>	Yok	Boş değerli gösterge ya da boş değerli byte
<i>ConnTag</i>	MQCT_NONE	Boş Değerler
<i>SSLConfigPtr</i>	Yok	Boş değerli gösterge ya da boş değerli byte
<i>SSLConfigOffset</i>	Yok	0
<i>ConnectionId</i>	Yok	Boş değerli gösterge ya da boş değerli byte
<i>SecurityParmsOffset</i>	Yok	Boş değerli gösterge ya da boş değerli byte
<i>SecurityParmsPtr</i>	Yok	Boş değerli gösterge ya da boş değerli byte
Notlar:		
1. ↵ simgesi tek bir boş karakteri temsil eder.		
2. C programlama dilinde, makro değişkeniMQCNO_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlara ilişkin ilk değerleri sağlamak için bunu aşağıdaki şekilde kullanın:		
<pre>MQCNO Mycno = {MQCNO_DEFAULT};</pre>		

C bildirimi

```
typedef struct tagMQCNO MQCNO;
struct tagMQCNO {
    MQCHAR4    StrucId;           /* Structure identifier */
    MQLONG     Version;           /* Structure version number */
    MQLONG     Options;           /* Options that control the action of
                                   MQCONN */
    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 */
};

```

COBOL bildirimi

```

** 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.

```

PL/I bildirimi

```

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 MQCONNX */
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 */

```

High Level Assembler bildirimi

```

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
*                      MQCONNX
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)	

Visual Basic bildirimi

```

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'
End Type

```

MQCSP-Güvenlik deęiřtirgeleri

Ařaęıdaki çizelge yapıdaki alanları özetlemektedir.

Uyarı: Bazı durumlarda, istemci uygulaması için bir MQCSP yapısındaki parola, düz metindeki bir aę üzerinden gönderilir. İstemci uygulaması parolalarının uygun řekilde korunmasını saęlamak için bkz. [IBM MQCSP parola koruması](#).

Çizelge 29. MQCSP içindeki alanlar		
Alan	Tanım	Konu
<i>StrucId</i>	Yapı tanıtıcısı	StrucId
<i>Version</i>	Yapı sürüm numarası	S\u00fcr\u00fcm
<i>AuthenticationType</i>	Kimlik doęrulama tipi	AuthenticationType
<i>Reserved1</i>	IBM üzerindeki iřaretçi hizalaması için gerekli	Reserved1
<i>CSPUserIdPtr</i>	Kullanıcı kimlięinin adresi	CSPUserIdPtr
<i>CSPUserIdOffset</i>	Kullanıcı kimlięinin görel konumu	CSPUserIdGörel Konumu
<i>CSPUserIdLength</i>	Kullanıcı kimlięi uzunluęu	CSPUserIdUzunluęu
<i>Reserved2</i>	IBM üzerindeki iřaretçi hizalaması için gerekli	Reserved2
<i>CSPPasswordPtr</i>	Parola adresi	CSPPasswordPtr
<i>CSPPasswordOffset</i>	Parola görel konumu	CSPPasswordOffset
<i>CSPPasswordLength</i>	Parola uzunluęu	CSPPasswordLength

MQCSP 'ye Genel Bakıř

Kullanılabilirlik: Tüm IBM MQ ürünleri.

Amaç: MQCSP yapısı, yetki hizmetinin bir kullanıcı kimliği ve parola doğrulamasına olanak sağlar. MQCONNX çağrısında MQCSP bağlantı güvenliği değiştirgeleri yapısını belirtmenizi sağlar.

Uyarı: Bazı durumlarda, istemci uygulaması için bir MQCSP yapısındaki parola, düz metindeki bir ağ üzerinden gönderilir. İstemci uygulaması parolalarının uygun şekilde korunmasını sağlamak için bkz. [IBM MQCSP parola koruması](#).

Karakter kümesi ve kodlama: MQCSP içindeki veriler, yerel kuyruk yöneticisinin karakter kümesinde ve kodlamasında olmalıdır; bunlar sırasıyla *CodedCharSetId* kuyruk yöneticisi özniteliği ve MQENC_NATIVE ile verilir.

MQCSP için alanlar

MQCSP yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

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'** yi ayarlamanız gerekir. **AuthenticationType** -MQCSP_AUTH_USER_ID_AND_PWD.

Ek bilgi için [MQCSP password protection](#) başlıklı konuya bakın.

CSPPasswordLength (MQUZE)

Bu alan, kimlik doğrulamasında kullanılacak parolanın uzunluğudur.

Parola uzunluğu üst sınırı MQ_CSP_PASSWORD_LENGTH ' dir. Parolanın uzunluğu izin verilen üst sınırdan büyükse, kimlik doğrulama isteği MQRC_NOT_AUTHENTICALE ile başarısız olur.

MQ_CSP_PASSWORD_LENGTH değeri 256.

Bu alan bir giriş alanıdır. Bu alanın ilk değeri 0 'tır.

CSPPasswordOffset (MQUZE)

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.

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_5değerinden küçükse bu alan yok sayılır.

This field can contain an empty password which is rejected by the operating system ya da ayara bağlı olarak LDAP parola denetimi, but is not rejected by IBM MQ before passing it the authentication method.

CSPUserIdUzunluğu (MQUZE)

Bu alan, kimlik doğrulamasında kullanılacak kullanıcı kimliğinin uzunluğudur.

Kullanıcı kimliğinin uzunluk üst sınırı altyapıya bağlıdır ve [Kullanıcı Kimlikleri](#) konusuna 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.

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.

CSPUserIdPtr (MQPTR)

Bu, kimlik doğrulamasında kullanılacak kullanıcı kimliğinin bayt cinsinden adresidir.

Bu bir giriş alanıdır. Bu alanın ilk değeri, göstergeleri destekleyen programlama dillerindeki boş değeri göstergeci ve tersi durumda, boş değeri byte dizilimidir. *Version* , MQCNO_VERSION_5değerinden 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.

Reserved1 (MQBYTE4)

A reserved field, required for pointer alignment on IBM i.

Bu bir giriş alanıdır. Bu alanın ilk değeri boş değeri.

Reserved2 (MQBYTE8)

A reserved field, required for pointer alignment on IBM i.

Bu bir giriş alanıdır. Bu alanın ilk değeri boş değeri.

StrucId (MQCHAR4)

Yapı tanıtıcısı.

Değer şu olmalıdır:

MQCSP_STRUC_ID

Güvenlik parametreleri yapısına ilişkin tanıtıcı.

C programlama dili için, MQCSP_STRUC_ID_ARRAY değeri de tanımlanır; bu, MQCSP_STRUC_ID ile aynı değeri 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 değeri, yürürlükteki sürümün sürüm numarasını belirtir:

MQCSP_XX_ENCODE_CASE_ONE current_version

Güvenlik değeri yapılarının yürürlükteki sürümü.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQCSP_VERSION_1' dir.

MQCSP için ilk değeri ve dil bildirimleri

Çizelge 30. MQCSP için MQCSP ' de alanların ilk değerleri

Alan adı	Değişmezin adı	Değişmezin değeri
StrucId	MQCSP_STRUC_ID	'CSP¬'
Version	MQCSP_VERSION_1	1
AuthenticationType	Yok	MQCSP_AUTH_NONE
Reserved1	Yok	Dizgi ya da boşluk boş değeri
CSPUserIdPtr	Yok	Boş değeri gösterge ya da boş değeri byte
CSPUserIdOffset	Yok	0
CSPUserIdLength	Yok	0
Reserved2	Yok	Dizgi ya da boşluk boş değeri
CSPPasswordPtr	Yok	Boş değeri gösterge ya da boş değeri byte
CSPPasswordOffset	Yok	0
CSPPasswordLength	Yok	0

Notlar:

1. ¬ simgesi tek bir boş karakteri temsil eder.
2. C programlama dilinde, makro değişkeni MQCSP_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlara ilk değerleri sağlamak için aşağıdaki şekilde kullanılabilir:

```
MQCSP MyCSP = {MQCSP_DEFAULT};
```

C bildirimi

```
typedef struct tagMQCSP MQCSP;
struct tagMQCSP {
    MQCHAR4    StrucId;           /* 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 */
};
```

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.

```

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 */

```

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

```

MQCTLO-Geri çağırma seçenekleri yapısı

Aşağıdaki çizelge yapıdaki alanları özetlemektedir. Denetim geri bildirme işlevini belirten Yapısı.

Çizelge 31. MQCTLO içindeki alanlar		
Alan	Tanım	Konu
<i>StrucID</i>	Yapı tanıtcısı	StrucID
<i>Version</i>	Yapı sürüm numarası	S\u00fcr\u00fcm
<i>Options</i>	Seenekler	Seenekler
<i>Reserved</i>	Ayrılmış alan	Seenekler
<i>ConnectionArea</i>	Geri bildirme işlevi için kullanılacak alan	ConnectionArea

MQCTLO için genel bakış

Kullanılabilirlik: Bu sistemlere bağlı AIX, HP-UX, IBM i, Solaris, Linux, Windows, z/OS ve IBM MQ MQI clients . MQCTLO yapısına genel bakış.

Amaç: MQCTLO yapısı, denetim çağrılarını işleviyle ilgili seçenekleri belirlemek için kullanılır.

Yapı, MQCTL çağrısındaki bir giriş ve çıkış parametresidir.

Sürüm: MQCTLO 'nun yürürlükteki sürümü MQCTLO_VERSION_1' dir.

Karakter kümesi ve kodlama: MQCTLO içindeki veriler, *CodedCharSetId* kuyruk yöneticisi özniteliği tarafından verilen karakter kümesinde ve MQKEN_NATIVE tarafından verilen yerel kuyruk yöneticisinin kodlamasında olmalıdır. Ancak, uygulama bir MQ MQI istemcisi olarak çalışıyorsa, yapı istemcinin karakter kümesinde ve kodlamasında olmalıdır.

MQCTLO alanları

MQCTLO yapısına ilişkin alanların alfabetik listesi.

MQCTLO yapısı aşağıdaki alanları içerir; alanlar alfabetik sırayla anlatılır:

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.

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.

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.

MQCTLO için ilk değerler ve dil bildirimleri

Denetim seçenekleri yapısı-İlk değerler

Çizelge 32. MQCTLO içindeki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
<i>StrucId</i>	MQCTLO_STRUC_ID	'CTLO'
<i>Version</i>	MQCTLO_VERSION_1	1
<i>Options</i>	MQCTLO_NONE	Boş Değerler
<i>Reserved</i>	Ayrılmış alan	
<i>ConnectionArea</i>	Yok	Boş değerli gösterge ya da boş değerli byte
Notlar: 1. C programlama dilinde, makro değişkeniMQCTLO_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlara ilişkin ilk değerleri sağlamak için bunu aşağıdaki şekilde kullanın: <pre>MQCTLO MyCTLO = {MQCTLO_DEFAULT};</pre>		

C bildirimi

Denetim Seçenekleri yapısı-C dili bildirimi

```
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 */
};
```

COBOL bildirimi

```
** 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
```

PL/I bildirimi

```
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 */
```

MQDH-Dağıtım üstbilgisi

Aşağıdaki çizelge yapıdaki alanları özetlemektedir.

Çizelge 33. MQDH 'deki alanlar		
Alan	Tanım	Konu
<i>StrucId</i>	Yapı tanıtıcısı	StrucId
<i>Version</i>	Yapı sürüm numarası	S\u00fcr\u00fcm
<i>StrucLength</i>	MQDH yapısı ve izleyen kayıtların uzunluğu	StrucLength
<i>Encoding</i>	MQPMR kayıtları dizisini izleyen verilerin sayısal kodlaması	Kodlama
<i>CodedCharSetId</i>	MQPMR kayıtları dizisini izleyen verilerin karakter kümesi tanıtıcısı	CodedCharSetId
<i>Format</i>	MQPMR kayıtları dizisini izleyen verilerin biçimi	Biçim
<i>Flags</i>	Genel işaretler	İşaretler
<i>PutMsgRecFields</i>	Hangi MQPMR alanlarının var olduğunu gösteren işaretler	PutMsgRecFields
<i>RecsPresent</i>	Var olan nesne kayıtlarının sayısı	RecsPresent
<i>ObjectRecOffset</i>	MQDH başlangıcından ilk nesne kaydının kayması	ObjectRecGörelili Konumu
<i>PutMsgRecOffset</i>	İlk put-ileti kaydının MQDH başlangıcından farkı	PutMsgRecOffset

MQDH 'ye Genel Bakış

Kullanılabilirlik: AIX, HP-UX, IBM i, Solaris, Linux, Windowsve ayrıca bu sistemlere bağlı IBM MQ istemcileri.

Amaç: MQDH yapısı, ileti bir iletim kuyruğunda saklanan bir dağıtım listesi iletisi olduğunda, bir iletide var olan ek verileri açıklar. Dağıtım listesi iletisi, 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 ileti kaldırma (örneğin, ileti kanalı aracılığıyla) 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 kullanmaları ya da tek tek hedeflere gönderilen iletilere ilişkin bilgileri almak zorunda olmalarıdır.

Biçim adı: MQFMT_DIST_HEADER.

Karakter kümesi ve kodlama: MQDH ' deki veriler, *CodedCharSetId* kuyruk yöneticisi özniteliği tarafından verilen karakter kümesinde ve MQKEN_NATIVE tarafından verilen yerel kuyruk yöneticisinin kodlamasında olmalıdır.

MQDH ' nin karakter kümesini ve kodlamasını *CodedCharSetId* ve *Encoding* alanlarına aşağıdaki şekilde kodlayın:

- 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 iletisi 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 iletisi verileri

Variş noktalarına bağlı olarak, kuyruk yöneticisi birden çok ileti oluşturabilir ve bu iletiyi 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 yerleştiren bir uygulamanın, daha önce açıklanan sıraya uyması ve MQDH yapısının doğru olduğundan emin olması gerekir. MQDH yapısı geçerli değilse, kuyruk yöneticisi MQPUT ya da MQPUT1 çağrısını başarısız olabilir ve neden kodu MQRC_DH_ERROR ile başarısız olur.

You can store messages on a queue in distribution-list form only if you have defined the queue as being able to support distribution list messages (see the *DistLists* queue attribute described in [“Kuyruklara ilişkin öznitelikler” sayfa 811](#)). Bir uygulama dağıtım listesi iletisini dağıtım listelerini desteklemeyen bir kuyruğa doğrudan koyarsa, kuyruk yöneticisi dağıtım listesi iletisini tek tek iletilere ayırır ve bunun yerine bunları kuyruğun üzerine yerleştirir.

MQDH için alanlar

MQDH yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

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 şu ortamlarda desteklenir: AIX, HP-UX, IBM i, Solaris, Linux, Windows, ayrıca bu sistemlere bağlı IBM MQ istemcileri.

Bu alanın ilk değeri MQCCSI_UNDEFED değerine sahip.

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.

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 iletiliyle akışı gereken denetim bilgileri miktarını en aza indirir.

Bir uygulama bir dağıtım listesine ileti koyduğunda, kuyruk yöneticisi, MQDHF_NEW_MSG_IDS 'yi aşağıdakilerin her ikisi de doğru olduğunda oluşturduğu MQDH' ye 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 değişmezin başka biriyle birlikte kullanılması amaçlanmaz, ancak değeri sıfır olduğu için bu tür bir kullanım saptanamaz.

Bu alanın ilk değeri MQDHF_NONE olur.

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 kurallarla aynıdır.

Bu alanın ilk değeri MQFMT_NONE olur.

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.

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.

PutMsgRecOffset (MQlong)

Bu, MQPMR dizesindeki ilk kaydın byte cinsinden görelî konumu verir, ileti özelliklerini içeren ileti kayıtlarını içerir. Varsa, bu dizideki *RecsPresent* kayıtları vardır. Bu kayıtlar (artı, ilk put iletisi 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.

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.

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.

StrucLength (MQUZE)

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 içerdği 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.

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.

MQDH için ilk değerler ve dil bildirimleri

Çizelge 34. MQDH için MQDH ' de alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
<i>StrucId</i>	MQDH_STRUC_ID	'DH↯'
<i>Version</i>	MQDH_VERSION_1	1
<i>StrucLength</i>	Yok	0
<i>Encoding</i>	Yok	0
<i>CodedCharSetId</i>	MQCCSI_UNDEDINED	0
<i>Format</i>	MQFMT_NONE	Boşluklar
<i>Flags</i>	MQDHF_NONE	0
<i>PutMsgRecFields</i>	MQPMRF_NONE	0
<i>RecsPresent</i>	Yok	0
<i>ObjectRecOffset</i>	Yok	0
<i>PutMsgRecOffset</i>	Yok	0
Notlar: ↯ simgesi tek bir boş karakteri temsil eder. - C programlama dilinde, makro değişkeni MQDH_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlara ilişkin ilk değerleri sağlamak için bunu aşağıdaki şekilde kullanın: <pre>MQDH MyDH = {MQDH_DEFAULT};</pre>		

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 */
};

```

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-STRUCLength      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-RECPRESENT       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.

```

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 */

```

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'
               'present'
RecsPresent  As Long     'Number of MQOR records present'
ObjectRecOffset As Long   'Offset of first MQOR record from start'
               'of MQDH'
PutMsgRecOffset As Long   'Offset of first MQPMR record from start'
               'of MQDH'
End Type

```

MQDLH-Dead-letter üstbilgisi

Aşağıdaki çizelge yapıdaki alanları özetlemektedir.

Çizelge 35. MQDLH içindeki alanlar		
Alan	Tanım	Konu
<i>StrucId</i>	Yapı tanıtıcısı	StrucId
<i>Version</i>	Yapı sürüm numarası	S\u00f0fcr\u00f0fcm
<i>Reason</i>	Neden iletisi ölü mektup kuyruğuna geldi	Neden
<i>DestQName</i>	Özgün hedef kuyruğun adı	DestQName
<i>DestQMGrName</i>	Özgün hedef kuyruk yöneticisinin adı	DestQMGrAdı
<i>Encoding</i>	MQDLH ' yi izleyen verilerin sayısal kodlaması	Kodlama
<i>CodedCharSetId</i>	MQDLH ' yi izleyen verilerin karakter kümesi tanıtıcısı	CodedCharSetId
<i>Format</i>	MQDLH ' yi izleyen verilerin adını biçimle	Biçim
<i>PutApplType</i>	İletinin gönderileceği ileti kuyruğuna ileti tipi uygulama	PutApplTipi
<i>PutApplName</i>	İletinin gönderileceği ileti kuyruğuna ileti koyan uygulamanın adı	PutApplAdı
<i>PutDate</i>	İletinin ölüme ilişkin kuyruğun üzerine konduğu tarih	PutDate
<i>PutTime</i>	İletinin ölüme ilişkin kuyruğa konduğu saat	PutTime

MQDLH için Genel Bakış

Kullanılabilirlik: Tüm IBM MQ altyapıları.

Amaç: MQDLH yapısı, ileti verilerini, ölü-mektup (teslim edilemeyen ileti) kuyruğunda bulunan iletilerin ileti verilerini önleyen bilgileri tanımlar. Bir ileti, kuyruk yöneticisi ya da ileti kanalı aracısı tarafından kuyruğa yeniden yönlendirdiği için ya da bir uygulama iletiyi doğrudan kuyruğa koyduğundan, ileti kuyruğunda bir ileti gelebilir.

Biçim adı: MQFMT_DEAD_LETTER_HEADER.

Karakter kümesi ve kodlama: MQDLH yapısındaki alanlar, *CodedCharSetId* ve *Encoding* alanları tarafından verilen karakter kümesinde ve kodlamalarında yer alıyor. Bunlar, MQDLH ' den önce gelen

üstbilgi yapısında ya da MQDLH uygulama ileti verilerinin başlangıcındaki MQDLH yapılıyorsa, bu değer MQMD yapısında belirtilir.

Karakter takımı, kuyruk adlarında geçerli olan karakterler için tek byte 'lık karakterler içeren bir karakter takımı olmalıdır.

If you are using the WMQ classes for Java/JMS, and the code page defined in the MQMD is not supported by the Java virtual machine, then the MQDLH is written in the UTF-8 character set.

Kullanım: İletilerin doğrudan ölü-mektup kuyruğuna yerleştiren uygulamalar, ileti verilerine bir MQDLH yapısıyla önek koymalı ve alanları uygun değerlerle kullanıma hazırlamalıdır. Ancak, kuyruk yöneticisi bir MQDLH yapısının var olmasını ya da alanlar için geçerli değerlerin belirtilmesini gerektirmez.

Bir ileti, kuyruklama-kuyruğu kuyruğuna konmak için çok uzunsa, uygulamanın aşağıdakilerden birini yapması gerekir:

- İleti verilerinin, ölü-mektup kuyruğuna sığabilmek için kesilmesi.
- İletiyi yardımcı belleğe kaydedin ve bir kural dışı durum raporu iletilisini, bu iletiyi gösteren ileti kuyruğuna yerleştirin.
- İletiyi atar ve bir hatayı orijinaline geri döndürür. İleti kritik bir iletiyse (ya da olabilir), bu iletiyi yalnızca, başlatan iletinin bir kopyası olduğu biliniyorsa (örneğin, bir iletişim kanalından ileti kanalı aracısı tarafından alınan bir ileti) varsa bunu yapın.

Önceki eylemlerden hangisi uygunsa (varsa), uygulamanın tasarımına bağlıdır.

Kuyruk yöneticisi, öndeki bir MQDLH yapısına sahip bir ileti olan bir ileti konduğunda özel işleme gerçekleştirir; ayrıntılar için MQMDE yapısının tanımına bakın.

İletilerin ölü-harf kuyruğuna yerleştirme: Bir ileti, ölü-mektup kuyruğuna konduğunda, MQPUT ya da MQPUT1 çağırısı için kullanılan MQMD yapısı, iletiyle ilişkilendirilmiş MQMD ile aynı olmalıdır (genellikle, MQGET çağırısının döndürdüğü MQMD), aşağıdaki kural dışı durum ile aynıdır:

- Set the *CodedCharSetId* and *Encoding* fields to whatever character set and encoding are used for fields in the MQDLH structure.
- Verilerin MQDLH yapısıyla başladığını göstermek için *Format* alanını MQFMT_DEAD_LETTER_HEADER olarak ayarlayın.
- Koşullar için uygun bir bağlam seçeneği kullanarak bağlam alanlarını (*AccountingToken*, *ApplIdentityData*, *ApplOriginData*, *PutApplName*, *PutApplType*, *PutDate*, *PutTime*, *UserIdentifier*) ayarlayın:
 - Ölü-mektup kuyruğuna yerleştirilecek bir uygulamanın, önceki herhangi bir iletiyle ilgili olmayan bir iletiyi, MQPMO_DEFAULT_CONTEXT seçeneğini kullanması gerekir; bu, kuyruk yöneticisinin ileti tanımlayıcısındaki bağlam alanlarının tümünü varsayılan değerlerine ayarlamasına neden olur.
 - Yalnızca almış olduğu bir ileti kuyruğuna gönderilen bir sunucu uygulaması, özgün bağlam bilgilerini korumak için MQPMO_PASS_ALL_CONTEXT seçeneğini kullanmalıdır.
 - Ölü-mektup kuyruğuna *yanıt* 'yi yeni aldığı bir iletiye koyan bir sunucu uygulaması, MQPMO_PASS_IDENTITY_CONTEXT seçeneğini kullanmalı; bu, kimlik bilgilerini korur, ancak sunucu uygulamasının kaynak kaynağını belirler.
 - Bir ileti kanalı aracısının, iletişim kanalından aldığı bir iletiyi ileti kuyruğuna yerleştirmesi, özgün bağlam bilgilerini korumak için MQPMO_SET_ALL_CONTEXT seçeneğini kullanmalıdır.

MQDLH yapısının kendisinde, alanları aşağıdaki gibi ayarlayın:

- *CodedCharSetId*, *Encoding* ve *Format* alanlarını, genellikle özgün ileti tanımlayıcısındaki değerler olan MQDLH yapısını açıklayan değerleri tanımlayan değerlere ayarlayın.
- *PutApplType*, *PutApplName*, *PutDate* ve *PutTime* bağlam alanlarını, iletiyi ölü-mektup kuyruğuna koyan uygulamaya uygun değerlere ayarlayın; bu değerler özgün iletiyle ilişkili değildir.
- Diğer alanları uygun şekilde ayarlayın.

Tüm alanların geçerli değerlere sahip olduğundan ve bu karakter alanlarının, alanın tanımlı uzunluğuna göre boşluklarla doldurulduğundan emin olun; kuyruk yöneticisi boş değerli ve sonraki karakterleri

MQDLH yapısındaki boşluklara dönüştürmediği için, boş bir karakter kullanarak karakter verilerini zamanından önce sonlanmayın.

Ölü-mektup kuyruğundan ileti alınması: Ölü harf kuyruğundan ileti alan uygulamalar, iletilerin MQDLH yapısıyla başladığını doğrulamalıdır. Uygulama, MQMD ileti tanımlayıcısındaki *Format* alanını inceleyerek bir MQDLH yapısının olup olmadığını saptayabilir; alan MQFMT_DEAD_LETTER_HEADER değerine sahipse, ileti verileri bir MQDLH yapısıyla başlar. Uygulamaların, kuyruktan uzun süredir uzun süre kalabilirlerse, ölümden mektup kuyruğundan elde edilen iletilerin kesilebileceğini de unutmayın.

MQDLH alanları

MQDLH yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

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.

This value is supported in the following environments: AIX, HP-UX, z/OS, IBM i, Solaris, Linux, Windows, plus IBM MQ MQI clients connected to these systems.

Bu alanın ilk değeri MQCCSI_UNDEFED değerine sahip.

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.

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.

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.

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.

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 tanımlayıcısı” sayfa 404](#) içindeki *PutApplName* alanının açıklamasına bakın.

Kuyruk yöneticisi iletiyi ölü harf kuyruğuna yeniden yönlendirirse, *PutApplName* kuyruk yöneticisi adının ilk 28 karakterini içerir, gerekirse boşluklarla doldurulabilir.

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.

PutApplTipi (MQUZE)

PutApplTipi, 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 tanımlayıcısı” sayfa 404](#) konusuna bakın).

Kuyruk yöneticisi iletiyi ölü harf kuyruğuna yeniden yönlendirirse, *PutApplType* MQAT_QMGR değerine sahiptir.

Bu alanın başlangıç değeri 0' dir.

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.

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_FULL). Oluşabilecek ortak MQFB_ * değerlerine ilişkin ayrıntılar için [“MQMD-İleti tanımlayıcısı” sayfa 404](#) 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-Tetikleme iletisi” sayfa 579](#)).

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-Tetikleme iletisi” sayfa 579](#)).

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.

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.

MQDLH için ilk değerler ve dil bildirimleri

Çizelge 36. MQDLH için MQDLH içindeki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
<i>StrucId</i>	MQDLH_STRUC_ID	'DLH~'
<i>Version</i>	MQDLH_VERSION_1	1
<i>Reason</i>	MQRC_NONE	0
<i>DestQName</i>	Yok	Dizgi ya da boşluk boş değerli
<i>DestQMgrName</i>	Yok	Dizgi ya da boşluk boş değerli
<i>Encoding</i>	Yok	0
<i>CodedCharSetId</i>	MQCCSI_UNDEDINED	0
<i>Format</i>	MQFMT_NONE	Boşluklar
<i>PutApplType</i>	Yok	0
<i>PutApplName</i>	Yok	Dizgi ya da boşluk boş değerli
<i>PutDate</i>	Yok	Dizgi ya da boşluk boş değerli
<i>PutTime</i>	Yok	Dizgi ya da boşluk boş değerli

Çizelge 36. MQDLH için MQDLH içindeki alanların ilk değerleri (devamı var)

Alan adı	Değişmezin adı	Değişmezin değeri
Notlar: 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. 3. C programlama dilinde, makro değişkeniMQDCH_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlara ilişkin ilk değerleri sağlamak için bunu aşağıdaki şekilde kullanın: <pre>MQDLH MyDLH = {MQDLH_DEFAULT};</pre>		

C bildirimi

```
typedef struct tagMQDLH MQDLH;
struct tagMQDLH {
    MQCHAR4    StrucId;          /* Structure identifier */
    MQLONG     Version;         /* Structure version number */
    MQLONG     Reason;          /* Reason message arrived on dead-letter
                                (undelivered-message) queue */
    MQCHAR48    DestQName;      /* Name of original destination queue */
    MQCHAR48    DestQMgrName;   /* Name of original destination queue
                                manager */
    MQLONG     Encoding;        /* Numeric encoding of data that follows
                                MQDLH */
    MQLONG     CodedCharSetId;  /* Character set identifier of data that
                                follows MQDLH */
    MQCHAR8     Format;          /* Format name of data that follows
                                MQDLH */
    MQLONG     PutApplType;      /* Type of application that put message on
                                dead-letter (undelivered-message)
                                queue */
    MQCHAR28    PutApplName;     /* Name of application that put message on
                                dead-letter (undelivered-message)
                                queue */
    MQCHAR8     PutDate;         /* Date when message was put on dead-letter
                                (undelivered-message) queue */
    MQCHAR8     PutTime;         /* Time when message was put on the
                                dead-letter (undelivered-message)
                                queue */
};
```

COBOL bildirimi

```
** MQDLH structure
10 MQDLH.
** Structure identifier
15 MQDLH-STRUCID PIC X(4).
** Structure version number
15 MQDLH-VERSION PIC S9(9) BINARY.
** Reason message arrived on dead-letter (undelivered-message) queue
15 MQDLH-REASON PIC S9(9) BINARY.
** Name of original destination queue
15 MQDLH-DESTQNAME PIC X(48).
** Name of original destination queue manager
15 MQDLH-DESTQMGRNAME PIC X(48).
** Numeric encoding of data that follows MQDLH
15 MQDLH-ENCODING PIC S9(9) BINARY.
** Character set identifier of data that follows MQDLH
15 MQDLH-CODEDCHARSETID PIC S9(9) BINARY.
** Format name of data that follows MQDLH
15 MQDLH-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).

```

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 */

```

High Level Assembler 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)	

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'

DestQName	As String*48	'(undelivered-message) queue'
DestQMgrName	As String*48	'Name of original destination queue'
Encoding	As Long	'Name of original destination queue'
CodedCharSetId	As Long	'manager'
Format	As String*8	'Numeric encoding of data that follows'
PutApplType	As Long	'MQDLH'
PutApplName	As String*28	'Character set identifier of data that follows MQDLH'
PutDate	As String*8	'Format name of data that follows MQDLH'
PutTime	As String*8	'Type of application that put message on'
End Type		'dead-letter (undelivered-message) queue'

MQDMHO-İleti tanıtıcısı seçeneklerini sil

Aşağıdaki çizelge yapıdaki alanları özetlemektedir.

Çizelge 37. MQDMHO içindeki alanlar		
Alan	Tanım	Konu
<i>StrucId</i>	Yapı tanıtıcısı	StrucId
<i>Version</i>	Yapı sürüm numarası	S\u00fcr\u00fcm
<i>Options</i>	Seçenekler	Seçenekler

MQDMHO için genel bakış

Kullanılabilirlik: Tüm IBM MQ sistemleri ve IBM MQ istemcileri.

Amaç: MQDMHO yapısı, uygulamaların ileti tutamaçlarının nasıl silineceğini denetleyen seçenekleri belirlemesine olanak sağlar. Yapı, **MQDLTMH** çağrısındaki bir giriş parametresidir.

Karakter kümesi ve kodlama: MQDMHO içindeki veriler, uygulamanın uygulama ve kodlamasının karakter kümesinde (**MQENC_NATIVE**) olmalıdır.

MQDMHO için alanlar

MQDMHO yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

Seçenekler (MQUZE)

Değer şu olmalıdır:

MQDMHO_NONE

Seçenek belirtilmedi.

Bu her zaman bir giriş alanıdır. Bu alanın başlangıç değeri **MQDMHO_NONE** dir.

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.

MQDMHO için ilk değerler ve dil bildirimleri

Çizelge 38. MQDMHO 'daki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
<i>StrucId</i>	MQDMHO_STRUC_ID	' DMHO '
<i>Version</i>	MQDMHO_VERSION_1	1
<i>Options</i>	MQDMHO_NONE	0
Notlar: 1. C programlama dilinde, makro değişkeniMQDMHO_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlara ilk değerleri sağlamak için aşağıdaki şekilde kullanılabilir: <pre>MQDMHO MyDMHO = {MQDMHO_DEFAULT};</pre>		

C bildirimi

```
typedef struct tagMQDMHO;  
struct tagMQDMHO {  
    MQCHAR4    StrucId;           /* Structure identifier */  
    MQLONG     Version;          /* Structure version number */  
    MQLONG     Options;          /* Options that control the action of MQDLTMH */  
};
```

COBOL bildirimi

```
**    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.
```

PL/I bildirimi

```
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          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)

```

MQDMMPO-İleti özelliği seçeneklerini sil

Aşağıdaki çizelge yapıdaki alanları özetlemektedir. MQDMPO yapısı-ileti özelliği seçeneklerini sil

Çizelge 39. MQDMPO 'daki alanlar		
Alan	Tanım	Konu
<i>StrucId</i>	Yapı tanıtıcısı	StrucId
<i>Version</i>	Yapı sürüm numarası	S\u00fcr\u00fcm
<i>Options</i>	MQDMMPO işleminin denetlenmesine ilişkin seçenekler	Seçenekler

MQDMPO ile ilgili genel bakış

Kullanılabilirlik: Tüm IBM MQ sistemleri ve IBM MQ istemcileri.

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 içindeki veriler, uygulamanın uygulama ve kodlamasının (MQENC_NATIVE) karakter kümesinde yer almalıdır.

MQDMPO ile ilgili alanlar

İleti özelliği seçenekleri yapısını sil-alanlar

MQDMPO yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

Seçenekler (MQUZE)

İleti özelliği seçenekleri yapısını sil-Seçenekler alanı

Konum seçenekleri: Aşağıdaki seçenekler, özellik imlecine göre özelliğin görel konularıyla 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_AVAILABLE neden olur. Özellik imlecinin işaret ettiği özellik zaten silindiyse, arama da MQCC_FAILED tamamlanma kodu ile başarısız olur ve MQRC_PROPERTY_NOT_AVAILABLE neden olur.

İki seçenekten hiçbirini 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.

StrucId (MQCHAR4)

İleti özelliği seçenekleri yapısını sil- StrucId alanı

Bu, yapı tanıtıcısıdır. Değer şu olmalıdır:

MQDMPO_STRUCT_ID

İleti özelliği seçenekleri yapısının silinmesine ilişkin tanıtıcı.

C programlama dili için, sabit MQDMPO_STRUCT_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.

MQDMPO için ilk değerler ve dil bildirimleri

İleti özelliği seçenekleri yapısını sil-Başlangıç değerleri

Çizelge 40. MQDPMO 'daki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
<i>StrucId</i>	MQDMPO_STRUCT_ID	' DMP0 '
<i>Version</i>	MQDMPO_VERSION_1	1
<i>Options</i>	MQDLTMP ' nin işlemini denetleyen seçenekler	MQDMPO_NONE
Notlar:		
1. C programlama dilinde, makro değişkeniMQDMPO_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlara ilişkin ilk değerleri sağlamak için bunu aşağıdaki şekilde kullanın:		
<pre>MQDMPO MyDMP0 = {MQDMPO_DEFAULT};</pre>		

C bildirimi

İleti özelliği seçenekleri yapısını sil-C dil bildirimi

```
typedef struct tagMQDMP0 MQDMP0;
struct tagMQDMP0 {
    MQCHAR4 StrucId;          /* Structure identifier */
    MQLONG Version;           /* Structure version number */
    MQLONG Options;           /* Options that control the action of
                               MQDLTMP */
};
```

COBOL bildirimi

İleti özelliği seçenekleri yapısını sil-COBOL dil bildirimi


```

** MQDMP0 structure
10 MQDMP0.
** Structure identifier
15 MQDMP0-STRUCID PIC X(4).
** Structure version number
15 MQDMP0-VERSION PIC S9(9) BINARY.
** Options that control the action of MQDLTMP
15 MQDMP0-OPTIONS PIC S9(9) BINARY.

```

PL/I bildirimi

İleti özelliği seçenekleri yapısını sil-PL/I dil bildirimi

```

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 */

```

High Level Assembler bildirimi

İleti özelliği seçenekleri yapısını sil-Assembler dil bildirimi

```

MQDMP0 DSECT
MQDMP0_STRUCID DS CL4 Structure identifier
MQDMP0_VERSION DS F Structure version number
MQDMP0_OPTIONS DS F Options that control the
* action of MQDLTMP
MQDMP0_LENGTH EQU *-MQDMP0
MQDMP0_AREA DS CL(MQDMP0_LENGTH)

```

MQEPH-Embedded PCF üstbilgisi

Aşağıdaki çizelge yapıdaki alanları özetlemektedir.

Çizelge 41. MQEPH içindeki alanlar		
Alan	Tanım	Konu
<i>StrucId</i>	Yapı tanıtıcısı	StrucId
<i>Version</i>	Yapı sürüm numarası	S\00fcr\00fcm
<i>StrucLength</i>	MQPH yapısının uzunluğu artı onu izleyen MQCFH ve değiştirge yapılarının uzunluğu	StrucLength
<i>Encoding</i>	Son PCF parametre yapısını izleyen verilerin sayısal kodlaması	Kodlama
<i>CodedCharSetId</i>	Son PCF parametre yapısını izleyen verilerin karakter kümesi tanıtıcısı	CodedCharSetId
<i>Format</i>	Son PCF parametre yapısını izleyen verilerin biçimi	Biçim
<i>Flags</i>	İşaretler	İşaretler
<i>PCFHeader</i>	Programlanabilir komut biçimi (PCF) üstbilgisi	PCFHeader

MQEPH 'ye Genel Bakış

Kullanılabilirlik: Tüm IBM MQ altyapıları.

Amaç: MQEPH yapısı, ileti bir programlanabilir komut biçimi (PCF) iletisi olduğunda, bir iletide var olan ek verileri açıklar. *PCFHeader* alanı, bu yapıyı izleyen PCF parametrelerini tanımlar ve bu, PCF iletisi verilerini diğer üstbilgilerle izlemenize olanak tanır.

Biçim adı: MQFMT_EMBEDDED_PCF

Karakter kümesi ve kodlama: MQEPH içindeki veriler, *CodedCharSetId* kuyruk yöneticisi özniteliği tarafından verilen karakter kümesinde ve MQENC_NATIVE tarafından verilen yerel kuyruk yöneticisinin kodlamasında olmalıdır.

MQEPH ' nin karakter kümesini ve kodlamasını *CodedCharSetId* ve *Encoding* 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 ettiği 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 oluşturmaz.

MQEPH alanları

MQEPH yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

CodedCharSetId (MQHOMER)

Bu, MQEPH yapısını ve ilişkili PCF parametrelerini izleyen verilerin karakter takımı tanıttıcısıdır; MQEPH yapısındaki karakter verileri için geçerli değildir.

Bu alanın ilk değeri MQCCSI_UNDEFED değerine sahip.

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.

İşaretler (MQUZE)

Aşağıdaki değerler kullanılabilir:

MQEPH_NONE

Herhangi bir işaret belirtilmedi. MQEPH_NONE, program belgelerine yardımcı olmak için tanımlıdır.

Bu değişimin 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.

Biçim (MQCHAR8)

Bu ad, MQEPH yapısını ve ilişkili PCF değiştirgelerini izleyen verilerin biçim adıdır.

Bu alanın ilk değeri MQFMT_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 42. 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.

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.

StrucLength (MQUZE)

Bu, bir sonraki üstbilgi yapısından önceki verilerin miktarıdır. Bu ürün aşağıdakileri içerir:

- MQEPH üstbilgisinin uzunluğu
- Üstbilgiyi izleyen tüm PCF parametrelerinin uzunluğu
- Bu değiştiricileri 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.

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.

MQEPH için ilk değerler ve dil bildirimleri

Çizelge 43. MQEPH için MQEPH 'de alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
StrucId	MQEPH_STRUCT_ID	'EPH↵'
Version	MQEPH_VERSION_1	1
StrucLength	MQEF_STRUCT_LENGTH_FIXED	68
Encoding	Yok	0
CodedCharSetId	MQCCSI_UNDEDINED	0
Format	MQFMT_NONE	Boşluklar
Flags	MQEPH_NONE	0
PCFHeader	Çizelge 42 sayfa 355 içinde tanımlandığı şekilde adlar ve değerler	0
Notlar: ↵ simgesi tek bir boş karakteri temsil eder. - C programlama dilinde, makro değişkeni MQEPPH_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlara ilişkin ilk değerleri sağlamak için bunu aşağıdaki şekilde kullanın: <pre>MQEPH MyEPH = {MQEPH_DEFAULT};</pre>		

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 */
};
```

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-STRULENGTH 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.

```

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 */

```

High Level Assembler bildirimi

MQEPH	DSECT	
MQEPH_STRUCID	DS	CL4 Structure identifier
MQEPH_VERSION	DS	F Structure version number
MQEPH_STRULENGTH	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_STRULENGTH	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
MQEPH_PCFHEADER_AREA        DS      CL(MQEPH_PCFHEADER_LENGTH)
*
MQEPH_LENGTH                EQU    *-MQEPH
MQEPH_AREA                  DS      CL(MQEPH_LENGTH)

```

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'
                                'and parameter structures that follow it'
    Encoding      As Long     'Numeric encoding of data that follows last'
                                'PCF parameter structure'
    CodedCharSetId As Long     'Character set identifier of data that'
                                'follows last PCF parameter structure'
    Format        As String*8 'Format name of data that follows last PCF'
                                'parameter structure'
    Flags         As Long     'Flags'
    PCFHeader     As MQCFH    'Programmable command format header'
End Type

Global MQEPH_DEFAULT As MQEPH

```

MQGMO-Get-message seçenekleri

Aşağıdaki çizelge yapıdaki alanları özetlemektedir.

Çizelge 44. MQGMO içindeki alanlar		
Alan	Tanım	Konu
<i>StrucId</i>	Yapı tanıtcısı	StrucId
<i>Version</i>	Yapı sürüm numarası	S\u00fcr\u00fcm
<i>Options</i>	MQGET işlemini denetleyen seçenekler	MQGMO-Seçenekler alanı
<i>WaitInterval</i>	Bekleme Aralığı	WaitInterval
<i>Signal1</i>	Sinyal	Signal1
<i>Signal2</i>	Sinyal tanıtcısı	Signal2
<i>ResolvedQName</i>	Hedef kuyruğun çözümlenmiş adı	ResolvedQName
Not: <i>Version</i> , MQGMO_VERSION_2değerinden küçükse, geri kalan alanlar yoksayılır.		
<i>MatchOptions</i>	MQGET için kullanılan seçim ölçütlerini denetleyen seçenekler	MatchOptions
<i>GroupStatus</i>	Alınan iletinin bir grup içinde olup olmadığını belirten işaret	GroupStatus
<i>SegmentStatus</i>	Alınan iletinin bir mantıksal ileti bölümü olup olmadığını belirten işaret	SegmentStatus
<i>Segmentation</i>	Alınan ileti için ek bölümlenmeye izin verilip verilmediğini belirten işaret	Segmentasyon
<i>Reserved1</i>	Ayrıldı	Reserved1
Not: <i>Version</i> , MQGMO_VERSION_3değerinden küçükse, geri kalan alanlar yoksayılır.		
<i>MsgToken</i>	İleti simgesi	MsgToken

Çizelge 44. MQGMO içindeki alanlar (devamı var)		
Alan	Tanım	Konu
<i>ReturnedLength</i>	Döndürülen ileti verilerinin uzunluğu (byte)	<u>ReturnedLength</u>
Not: <i>Version</i> , MQGMO_VERSION_4değerinden küçükse, geri kalan alanlar yoksayılır.		
<i>Reserved2</i>	Ayrıldı	<u>Reserved2</u>
<i>MsgHandle</i>	Kuyruktan alınan iletinin özellikleri ile doldurulacak bir iletinin tanıtıcısı.	<u>MsgHandle</u>

MQGMO 'ya Genel Bakış

Kullanılabilirlik: Tüm IBM MQ altyapıları.

Amaç: MQGMO yapısı, uygulamanın iletilerin kuyruktan nasıl kaldırılacağını denetlemesini sağlar. Yapı, MQGET çağrısındaki bir giriş/çıkış parametresidir.

Sürüm: MQGMO 'nun yürürlükteki sürümü MQGMO_VERSION_4' tür. Bazı alanlar yalnızca MQGMO 'nun belirli sürümlerinde kullanılabilir. Birkaç ortam arasındaki uygulamaları kapı olarak görmeniz gerekiyorsa, MQGMO sürümünün tüm ortamlarda tutarlı olduğundan emin olmanız gerekir. Yalnızca yapının belirli sürümlerinde bulunan alanlar, “MQGMO-Get-message seçenekleri” sayfa 358 içinde ve alan açıklamalarında olduğu gibi tanımlanır.

Desteklenen programlama dilleri için sağlanan üstbilgi, COPY ve INCLUDE dosyaları, ortam tarafından desteklenen MQGMO 'nun en son sürümünü içerir, ancak *Version* alanının ilk değeri MQGMO_VERSION_1olarak ayarlanmış olmalıdır. version-1 yapısında mevcut olmayan alanları kullanmak için, *Version* alanını, gereken sürümün sürüm numarasıyla ayarlayın.

Karakter kümesi ve kodlama: MQGMO 'daki veriler, *CodedCharSetId* kuyruk yöneticisi özniteliği tarafından verilen karakter kümesinde ve MQKEN_NATIVE tarafından verilen yerel kuyruk yöneticisinin kodlamasında olmalıdır. Ancak, uygulama bir MQ MQI istemcisi olarak çalışıyorsa, yapı istemcinin karakter kümesinde ve kodlamasında olmalıdır.

MQGMO için alanlar

MQGMO yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

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.

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. Karşılık gelen eşleşme seçeneğinin *belirtilmediği* alanlar, döndürülebilmek için ileti seçilirken

yoksayılr. MQGET çağrısında seçim ölçütü belirtmediyseniz (yani, *herhangi biri* iletisi kabul edilebilir), *MatchOptions* olarak MQMO_NONE olarak ayarlayın.

- z/OS' ta, kullanılabilecek seçim ölçütleri, kuyruk için kullanılan dizin tipi ile sınırlanmış olabilir. Ek ayrıntılar için *IndexType* kuyruk özniteliğine bakın.

MQGMO_LOGICAL_ORDER belirtilirse, sonraki MQGET çağrısıyla döndürülebilecek 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* , yalnızca gruptaki sonraki ileti ya da mantıksal iletinin sonraki bölümü 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, döndürülebilecek 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 470.

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 yoksayılr ve herhangi bir ileti tanıtıcısı eşleşir.

Not: MQMI_NONE ileti tanıtıcısı, ileti için MQMD ' deki *herhangi bir* ileti tanıtıcısıyla eşleşen özel bir değerdir. Bu nedenle, MQMI_NONE ile MQMO_MATCH_MSG_ID değerinin belirlenmesi, MQMO_MATCH_MSG_ID belirtilerek *değil* ile aynıdır.

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 yoksayılr ve herhangi bir ilinti tanımlayıcısı eşleşir.

Not: MQCI_NONE ilinti tanıtıcısı, ileti için MQMD ' deki *any* ilinti tanıtıcısıyla eşleşen özel bir değerdir. Bu nedenle, MQCI_NONE belirtimiyle MQMO_MATCH_COREL_ID değerinin belirlenmesi, MQMO_MATCH_COREL_ID belirtimiyle *değil* ile aynıdır.

MQMO_MATCH_GROUP_ID

Alınacak iletinin, MQGET çağrısının *MsgDesc* parametresindeki *GroupId* alanının değeriyle eşleşen bir grup tanıtıcısı olmalıdır. Bu eşleşme, geçerli olabilecek diğer eşleşmelere ek olarak (örneğin, ilinti tanıtıcısıdır).

Bu seçeneği çıkarırsanız, *MsgDesc* parametresindeki *GroupId* alanı yoksayılr ve herhangi bir grup tanıtıcısı eşleşir.

Not: MQGI_NONE grup tanıtıcısı, ileti için MQMD ' deki herhangi bir grup tanıtıcısıyla eşleşen özel bir değerdir. Bu nedenle, MQGI_NONE ile MQMO_MATCH_GROUP_ID değerinin belirlenmesi, MQMO_MATCH_GROUP_ID belirtimiyle değil ile aynıdır.

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 tanıtıcısı).

Bu seçeneği çıkarırsanız, *MsgDesc* parametresindeki *MsgSeqNumber* alanı yoksayılr 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 yoksayılr 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 belirtemezsiniz. 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 yoksayılr ve herhangi bir ileti simgesi eşleşir.

Tanımlanan seçeneklerin hiçbirini belirtmediyseniz, aşağıdaki seçeneği kullanabilirsiniz:

MQMO_NONE

Döndürülebilecek iletiyi seçerken eşleşme yok; kuyruksa 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 bilgiler

[JMS ' de ileti seçicileri](#)

MsgHandle (MQHMSG)

MQGMO_PROPERTIES_AS_Q_DEF seçeneği belirtilirse ve *PropertyControl* kuyruk özneliği 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 değiştirgesi 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_4 değerinden küçükse bu alan yok sayılır.

MsgToken (MQBYTE16)

MsgToken alanı-MQGMO 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 kaldırılncaya 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 MQGMO belirtildiğinde bir ileti döndürüldüğünde, kuyruk yöneticisi tarafından MQGMO ' ya kuyrukta iletiyi tanıtan *MsgToken* iletisi döndürülür. Bunun bir kural dışı durumu vardır: İleti, eşitleme noktası dışında kuyruktan 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 MQGMO_MSG_UNDER_CURSOR ya da MQGMO_BROWSE_MSG_UNDER_CURSOR belirtilirse, yalnızca bu *MsgToken* ile tanıtılan ileti döndürülebilmektedir. Bu seçenek, INDXTYPE ile bağımsız olarak tüm yerel kuyruklarda geçerlidir ve z/OS üzerinde, yalnızca İş Yüğü Yöneticisi (WLM) kuyruklarında INDXTYPE (MSGTOKEN) olanağını kullanmanız gerekir.

Belirtilen diğer *MatchOptions* sayısı imlenir ve eşleşmezse, MQRC_NO_MSG_AVAILABLE döndürülür. MQGMO_BROWSE_NEXT, MQMO_MATCH_MSG_TOKEN ile kodlandıysa, *MsgToken* ile tanıtılan ileti yalnızca, çağırın tanıtıcı için göz atma imlecinin ötesinde olduğunda döndürülür.

MQGMO_MSG_UNDER_CURSOR ya da MQGMO_BROWSE_MSG_UNDER_CURSOR belirtilirse, MQMO_MATCH_MSG_TOKEN yoksayılır.

MQMO_MATCH_MSG_TOKEN, aşağıdaki alma iletisi seçenekleriyle geçerli değil:

- MQGMO_BEKLE
- MQGMO_SET_SIGNAL

MQMO_MATCH_MSG_TOKEN MQGET çağrısına ilişkin çağrıya bir MQGMO 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_AVAILABLE ile MQCCT_BARINIZ geri döndürülür.

Seçenekler (MQUZE)

MQGMO seçenekleri, MQGET işlemini denetler. Seçeneklerden sıfır ya da daha fazlasını belirleyebilirsiniz. Birden fazla isteğe bağlı değer gerekliyse:

- Değerleri ekleyin (aynı değışmezi 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, kuyruğun varolması için bekleyen iletilerin beklenmesiyle ilgilidir:

MQGMO_WAIT

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:

Çizelge 45. Rules for activating MQGET calls on a shared queue.		
Etkinleştirilmeyi bekleyen MQGET çağrılarının sayısı		Sonuç
GÖZ AT seçeneği ile	BROWSE seçeneği olmadan ¹	
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.

¹ MQGMO_LOCK seçeneğini belirten bir MQGET çağrısı, göz atmayan bir çağrı olarak işlem görür.

Not:

- İlk kategori içinde, daha belirli alma bekleme isteklerine ek öncelik verilmez. Örneğin, hem *MsgId* hem de *CorrelId* belirtilmesini sağlar.
- 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' 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 iletisinin yıkıcı alma isteği olduğundan, *MsgId* ya da *CorrelId* eşleşmediği için, ECB işaretiniz 200 milisaniyeden sonra MQEC_MSG_VARDI iletisi 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 yoksayılır; herhangi bir hata ortaya çıktı.

MQGMO_NO_WAIT

Uygun bir ileti bulunmazsa, uygulama beklemes. 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 zaman uyumlu 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 değıştirmeleri tanımlı deęil. 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 etkinleştirildięi 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 isteęi olamaz.

Bu seçenek, aşağıdaki seçeneklerin hiçbirisiyle geçerli deęildir:

- MQGMO_UNLOCK
- MQGMO_WAIT

Bir MQGET çağrısı paylaşılan bir kuyruksa ve çağrı bir göz atma isteęi ya da bir grup iletisinin yıkıcı alma isteęi olduğundan, *MsgId* ya da *CorrelId* eşleştirilmezse, kullanıcının sinyal ECB değeri 200 milisaniyeden sonra MQEC_MSG_ARRIVED ' a gönderilir.

Bu durum, kuyruęa uygun bir ileti gönderilmemiş olsa da, kuyruk MQEC_WAIT_INTERVAL_EXPIRED ile birlikte gönderildiğinde bekleme süresi doluncaya kadar kuyruęa 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 deęildir. Göz atma durumunda bu ek yükü önlemek için, MQGET çağrısında *MsgId* (dizinlenmemiş ya da *MsgId* ile dizinlenen) ya da *CorrelId* (*CorrelId* ile dizinlendiyse) ile eşleşenifdeęerini belirtin.

Bu seçenek yalnızca z/OS üzerinde desteklenir.

MQGMO_FAIL_IF QUIESCING

Kuyruk yöneticisi susturulmuş durumdaysa, MQGET çağrısını başarısız olmak için zorlayın.

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 girdięi sırada olaęanüstüyorsa:

- 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 özğü bir sinyal tamamlanma kodu ile iptal edilir.

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: Aşağıdaki seçenekler, bir iş birimi içinde MQGET çağrısının katılımlarıyla ilgilidir:

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. Diğer 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 hiçbirisiyle 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

İstek, olağan çalışma birimi protokolleri içinde çalışmak, ancak alınan iletinin kalıcı olması durumunda *yalnızca* . 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 hiçbirisiyle 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 şu ortamlarda desteklenir: AIX, HP-UX, z/OS, IBM i, Solarisve Linux, artı IBM MQ MQI clients bu sistemlere bağlı olarak.

MQGMO_NO_SYNCPOINT

Bu istek, olağan çalışma birimi protokollerinin dışında işlem yapmak. Göz atma seçeneği olmayan bir ileti aldıysanız, ileti hemen kuyruktan silinir. İleti, iş biriminin yedeklenmesiyle yeniden kullanılabilir kılınmaz.

MQGMO_BROWSE_FIRST ya da MQGMO_BROWSE_NEXT belirtilirse bu seçenek varsayılan olarak kabul edilir.

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. Diğer 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 either MQGMO_SYNCPOINT or MQGMO_NO_SYNCPOINT explicitly.

Bu seçenek, aşağıdaki seçeneklerin hiçbirisiyle geçerli değildir:

- MQGMO_MARK_SKIP_BACKOUT
- MQGMO_SYNCPOINT
- MQGMO_SYNCPOINT_IF_PERSISTENT

MQGMO_MARK_SKIP_BACKOUT

Bu seçenekle işaretlenen iletiyi kuyruğa geri vermeden, bir iş birimini geri alma.

Bu seçenek yalnızca z/OS üzerinde desteklenir.

Bu seçenek belirtilirse, MQGMO_SYNCPOINT de belirtilmelidir. MQGMO_MARK_SKIP_BACKOUT aşağıdaki seçeneklerden herhangi biriyle geçerli değildir:

- 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. İşaretli iletiyi içeren yeni iş birimini kesinleştirmeden önce bir IBM MQ çağrısı yayınlamanız gerekir. Arama, istediğiniz herhangi bir IBM MQ çağrısı olabilir.

1. IMS üzerinde, IMS APAR PN60855 ' i uygulamadıysanız ve bir IMS MPP ya da BMP uygulaması çalıştırıyorsanız.
2. Herhangi bir uygulama çalıştırıyorsanız, CICS üzerinde.

Her iki durumda da, yedeklenmiş iletiyi içeren yeni iş birimini kesinleştirmeden önce herhangi bir IBM MQ çağrısı yayınlayın.

Not: Bir iş birimi içinde, geri alma işlemi atlanıyor olarak işaretlenen tek bir alma isteği yanı sıra, hiçbir ya da birden çok işaretlenmemiş alma isteği olabilir.

Bir uygulama, bir iş biriminden dışarı çıkmışsa, MQGMO_MARK_SKIP_BACKOUT kullanılarak alınan bir ileti önceki durumuna geri yüklenmez. Diğer kaynak güncellemeleri yedeklenir. İleti, geri alma isteği tarafından başlatılan yeni bir iş biriminde alınmış gibi işlem görür. İleti, MQGMO_MARK_SKIP_BACKOUT seçeneği 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 iletili 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 alınırsa, gruptaki ilk iletiyi işaretlemek avantajlıdır. İş birimi geriletirse, ilk (işaretli) ileti yeni iş birimine taşınır. Gruptaki ikinci ve sonraki iletiler kuyrukta yeniden yürürlüğe girilir. Kuyruğunda kalan iletiler, MQGMO_LOGICAL_ORDER kullanılarak başka bir uygulama tarafından alınamıyor. Gruptaki ilk ileti artık kuyrukta 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 kuyrukta 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 yoksayılır. Bu seçenek kullanılarak alınan tüm iletiler, bu seçenek olmadan alınan iletiyle aynı şekilde kuyrukta yeniden başlatılır.

Göz at seçenekleri: Aşağıdaki seçenekler kuyruklardaki iletilere göz atmak için ilişkilendirir:

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. Kuyrukta 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 seçenekle 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 çağrılarına ilişkin iki grup ve bölüm bilgilerini saklar. Göz atma çağrılarına 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
- FIFO ne olursa olsun önceliği (MQMDS_FIFO)

MsgDeliverySequence kuyruk özneliği, hangi yöntemin geçerli olduğunu belirtir (ayrıntılar için [“Kuyruklara ilişkin öznelikler”](#) sayfa 811 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 seçenekle 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, kuyrukta 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 de 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 çağrılarının hiçbiri 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 kilitlenmeden 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. İletin 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 çağrılarına 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 hiçbirisiyle 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 tanıtılan 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 döndürülebilir bile ileti durumu değişmeye devam eder.
- İletin 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 işbirliği 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

Kilit seçenekleri: Aşağıdaki seçenekler kuyruklardaki iletileri kilitlemek için ilişkilendir:

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 kilitlenmış olabilir. İleti, mantıksal bir ileti ya da fiziksel bir ileti olabilir:

- MQGMO_COMPLETE_MSG belirtilirse, mantıksal iletiyi oluşturan tüm ileti bölümleri kuyruk tanıtıcısı için kilitletir. İletiler, kuyruğun üzerinde olmalı ve alma için kullanılabilir olmalıdır.
- MQGMO_COMPLETE_MSG' i atlarsanız, kuyruk sapına yalnızca tek bir fiziksel ileti kilitletir. 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

TMF tarafından eşgüdümlü bir z/OS kuyruk yöneticisine HP Integrity NonStop Server üzerinde bir IBM MQ istemcisi kullanıyorsanız, MQGMO_LOCK bu olanakla mümkün değildir.

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 verisi seçenekleri: Aşağıdaki seçenekler, ileti kuyruktan okunduğunda ileti verilerinin işlenmesiyle ilgilidir:

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 işlemini 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” sayfa 879 .

- 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. Tamamlanma 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 tanımlayıcısı” sayfa 404 içinde açıklanan *Format* alanına bakın.

Grup ve bölüm seçenekleri: Aşağıdaki seçenekler, mantıksal iletilerin gruplarındaki ve kısımlarındaki iletilerin işlenmesiyle ilgilidir. 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 görel konumu, mantıksal iletiden verilerin başlangıcındaki fiziksel iletteki verilerin görel konutdur. Her bir bölüm fiziksel bir ileti olduğu için, mantıksal bir iletteki kesimlerin genellikle farklı ileti tanıtıcıları vardır.

Kesimlere ayrılmış olmayan, ancak gönderme uygulaması tarafından bölümlenmeye 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 tanıttıcısına 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 tanıttıcısı için ardışık MQGET çağrılarını 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 görelili konumlarından belirtilen sırayla döndürülür. Bu sipariş, kuyrukta ileti ve kesimlerin oluştuğu 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 çağrılarını arasında grup ve bölüm bilgilerini korur. Grup ve bölüm bilgileri, kuyruk tanıttıcısı 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 bilgilerini 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 çağrılarını, 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 olmadığındaki etkindir. Ek ayrıntılar için “MQGMO-Get-message seçenekleri” sayfa 358 içinde açıklanan *MatchOptions* alanına bakın.

Çizelge 46 sayfa 377 , 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 46. 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	Offset
Evet	Hayır	Hayır	MatchOption starafından denetlenen	MatchOption starafından denetlenen	MatchOption starafından 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 starafından denetlenen	MatchOption starafından denetlenen	MatchOption starafından denetlenen	MatchOption starafından denetlenen	MatchOption starafından 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 görel konum 0 olan fiziksel iletiler, uygun grupların döndürülebilecek sırayı belirler.

MQGMO_LOGICAL_ORDER seçeneği, iş birimlerini aşağıdaki gibi etkiler:

- Bir iş birimi içinde bir gruptaki ilk mantıksal ileti ya da bölüm alınır, 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 kesim, bir iş birimi içinde *alınmadıysa* ve aynı kuyruk tanıtıcısı kullanılırsa, gruptaki diğer mantıksal ileti ve kesimlerin hiçbirisi bir iş birimi içinde alınamaz.

Bu koşullar karşılanmazsa, MQGET çağrısı neden kodu MQRC_INCONSISTENT_UOW ile başarısız olur.

MQGMO_LOGICAL_ORDER belirtildiğinde, MQGET çağrısında belirtilen MQGMO , MQGMO_VERSION_2 değerinden küçük olamaz ve MQMD MQMD_VERSION_2 değerinden küçük olmamalıdır. Bu koşul karşılanmazsa, uygun olduğu şekilde, çağrı neden kodu MQRC_WRONG_GMO_VERSION ya da MQRC_WRONG_MD_VERSION ile başarısız olur.

If MQGMO_LOGICAL_ORDER is *değil* 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 iletimden gelen iletilerin ya da bölümlerin, diğer gruplardan ya da mantıksal iletilerden gelen iletilerle ya da bölümlerle ya da gruplarda olmayan ve kesimler olmayan iletilerle birlikte gönderilebileceği 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 358 ' de açıklanan *MatchOptions* alanına bakın).

Sistem başarısızlığı oluştuğundan sonra, ortadaki bir ileti grubunu ya da mantıksal iletiyi yeniden başlatmak için kullanılacak 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 *bu olmadan* 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 çağrısı için sakladığı grup ve kesim bilgileri, MQPUT çağrısına ilişkin sakladığı grup ve bölüm bilgilerinden ayrıdır. Ayrıca, kuyruk yöneticisi aşağıdakine 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, aşağıdaki noktalara dikkat edin:

- MQGMO_LOGICAL_ORDER' ı atlarsanız, her başarılı MQGET çağrısı, 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) değiştirilir.
- 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 47 sayfa 378 , oluşabilecek çeşitli vakaları gösterir. Bu durumlarda, tamamlanma kodu MQCC_OKdeğilse, neden kodu aşağıdakilerden biridir (uygun olduğu 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 47. 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 grupta 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.

For example, an application that wants to *ileri* physical messages that it receives, without regard for whether those messages are in groups or segments of logical messages, must *değil* specify MQGMO_LOGICAL_ORDER. 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 MQGMO_LOGICAL_ORDER belirtilirse, iletleme 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).

- 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.
- AIX, HP-UX, IBM i, Solaris, Linux, Windows ve bu sistemlere bağlı IBM MQ MQI clients işletim sistemlerinde, bu seçenek tüm yerel kuyruklar için desteklenir.

MQGMO_COMPLETE_MSG

MQGET çağrısı, yalnızca tam bir mantıksal ileti döndürülebiliyor. 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.

Kuyruk yöneticisi, *kalıcı* iletiler için, kısımları 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.

Kuyruk yöneticisi, *kalıcı olmayan* iletiler için yeniden birleştirme işlemini gerçekleştirmek için bir iş birimi 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 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* kesimine 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' ta bu seçenek özel ve paylaşılan kuyruklar için desteklenir, ancak kuyruğun bir MQIT_GROUP_ID dizin tipine sahip olması gerekir. Paylaşılan kuyruklar için, kuyruk eşleminin CFLEVEL (3) ya da daha sonraki bir yerde olması gereken CFSTRUCT nesnesinden biri.
- AIX, HP-UX, IBM i, Solaris, Linux, Windows ve bu sistemlere bağlı IBM MQ MQI clients işletim sistemlerinde, bu seçenek tüm yerel kuyruklar için desteklenir.

MQGMO_ALL_MSGS_AVAILABLE

Gruptaki iletiler, 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:

- If both options are specified, MQGMO_ALL_MSGS_AVAILABLE has an effect *yalnızca* when there is no current group or logical message. If there : a current group or logical message, MQGMO_ALL_MSGS_AVAILABLE is ignored. Bu, MQGMO_ALL_MSGS_AVAILABLE ' un iletileri mantıksal sırada işlerken üzerinde kalabileceği anlamına gelir.
- MQGMO_LOGICAL_ORDER belirtilmeden MQGMO_ALL_MSGS_AVAILABLE belirtilirse, MQGMO_ALL_MSGS_AVAILABLE *her zaman* bir etkiye sahiptir. 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 kilitlenmez).

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' 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.
- AIX, HP-UX, IBM i, Solaris, Linux, Windows ve bu sistemlere bağlı IBM MQ MQI clients işletim sistemlerinde, bu seçenek tüm yerel kuyruklar için desteklenir.

MQGMO_ALL_SEGMENTS_AVAILABLE

Mantıksal bir iletteki kesimler, mantıksal iletiyle yalnızca *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 özneliliğinin 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:

- If both options are specified, MQGMO_ALL_SEGMENTS_AVAILABLE has an effect *yalnızca* when there is no current logical message. If there : a current logical message, MQGMO_ALL_SEGMENTS_AVAILABLE is ignored. Bu, MQGMO_ALL_SEGMENTS_AVAILABLE ' un iletileri mantıksal sırada işlerken üzerinde kalabileceği anlamına gelir.
- MQGMO_LOGICAL_ORDER belirtilmeden MQGMO_ALL_SEGMENTS_AVAILABLE belirtilirse, MQGMO_ALL_SEGMENTS_AVAILABLE *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 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 başarısı 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 MQGMO_* 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 dizin tipine sahip olması gerekir. Paylaşılan kuyruklar için, kuyruk eşleminin CFLEVEL (3) ya da daha sonraki bir yerde olması gereken CFSTRUCT nesnesinden biri.
- AIX, HP-UX, IBM i, Solaris, Linux, Windows ve bu sistemlere bağlı IBM MQ MQI clients işletim sistemlerinde, bu seçenek tüm yerel kuyruklar için desteklenir.

Özellik seçenekleri: Aşağıdaki seçenekler, iletinin özellikleriyle ilgilidir:

MQGMO_PROPERTIES_AS_Q_DEF

İletin özellikleri (ya da uzantısı), ileti tanımlayıcısında (ya da uzantısında) bulunanlar dışında, *PropertyControl* kuyruk özneteliği tarafından tanımlandığı şekilde gösterilmelidir. Bir *MsgHandle* sağlandıysa, bu seçenek yoksayılır ve iletinin özellikleri, *PropertyControl* kuyruk özneteliğinin değeri MQPROP_FORCE_MQRFH2 değerse, *MsgHandle* aracılığıyla kullanılabilir.

Özellik seçeneği belirlenmezse, bu varsayılan işlemdir.

MQGMO_PROPERTIES_IN_HANDLE

İletin 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, iletin hiçbir özelliği alınacak. Bir *MsgHandle* belirtilirse, bu değer yoksayılacak.

MQGMO_PROPERTIES_FORCE_MQRFH2

İletin 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 yoksayılır.

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, iletin tüm özellikleri atılır ve artık bu uygulamanın erişilemeyecek şekilde bulunur.

Varsayılan seçenek: Tanımlanan seçeneklerden hiçbirisi 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.

Reserved1 (MQCHAR)

Bu ayrılmış bir alandır. Bu alanın ilk değeri boşluk karakteridir. *Version* , MQGMO_VERSION_2 değerinden küçükse bu alan yok sayılır.

Reserved2 (MQUZE)

Bu ayrılmış bir alandır. Bu alanın ilk değeri boşluk karakteridir. *Version* , **MQGMO_VERSION_4**değerinden küçükse bu alan yok sayılır.

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.

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* değil MQRL_UNDEFINED ise, 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.

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_INHIMATED 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.

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 tanımlama 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/OSüzerinde, 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.

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.

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 tanımlayıcısı” sayfa 404 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.

MQGMO için ilk değerler ve dil bildirimleri

Çizelge 48. MQGMO için MQGMO 'daki alanların ilk değerleri

Alan adı	Değişmezin adı	Değişmezin değeri
StrucId	MQGMO_STRUC_ID	'GMO↵'
Version	MQGMO_VERSION_1	1
Options	MQGMO_NO_BEKLEME	0
WaitInterval	Yok	0
Signal1	Yok	z/OS üzerinde boş değerli gösterge; 0 aksi takdirde
Signal2	Yok	0
ResolvedQName	Yok	Dizgi ya da boşluk boş değerli
MatchOptions	MQMO_MATCH_MSG_ID + MQMO_MATCH_COREL_ID	3
GroupStatus	MQGS_NOT_IN_GROUP	'↵'
SegmentStatus	MQSS_NOT_A_KESIMI	'↵'
Segmentation	MQSEG_INHIBITED	'↵'
Reserved1	Yok	'↵'
MsgToken	MQMTOK_NONE	Boş Değerler
ReturnedLength	MQRL_UNDEFED	-1
Reserved2	Yok	'↵'
MsgHandle	MQHM_NONE	0

Notlar:

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.
3. C programlama dilinde, makro değişkeniMQGMO_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlara ilk değerleri sağlamak için aşağıdaki şekilde kullanılabilir:

```
MQGMO MyGMO = {MQGMO_DEFAULT};
```

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 */
};

```

- z/OSüzerinde, *Signal1* alanı PMQLONGolarak bildirilir.

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.

```

- z/OSüzerinde, *Signal1* alanı POINTERolarak bildirilir.

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

```

```

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 */

```

- z/OS üzerinde, *Signal1* alanı pointer olarak bildirilir.

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
*                               reserved
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)

```

Visual Basic 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'
                                'used for MQGET'
    GroupStatus       As String*1    '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

```

MQIIH- IMS bilgi üstbilgisi

Aşağıdaki çizelge yapıdaki alanları özetlemektedir.

Çizelge 49. MQIIH 'taki alanlar

Alan	Tanım	Konu
<i>StrucId</i>	Yapı tanıtıcısı	StrucId
<i>Version</i>	Yapı sürüm numarası	S\u00f0fcr\u00f0cm
<i>StrucLength</i>	MQIIH yapısının uzunluğu	StrucLength
<i>Encoding</i>	Ayrıldı	Kodlama
<i>CodedCharSetId</i>	Ayrıldı	CodedCharSetId
<i>Format</i>	MQ format name of data that follows MQIIH	Biçim
<i>Flags</i>	İşaretler	İşaretler
<i>LTermOverride</i>	Mantıksal uçbirim geçersiz kılma değeri	LTermOverride
<i>MFSMapName</i>	İleti biçimi hizmetleri eşlemi adı	MFSMapName
<i>ReplyToFormat</i>	MQ yanıt iletisinin adını biçimlendirir	ReplyToBiçimi
<i>Authenticator</i>	RACF™ parolası ya da passticket	Kimliği doğrulayıcı
<i>TranInstanceId</i>	İşlem eşgörünümü tanıtıcısı	TranInstanceTanıtıcısı
<i>TranState</i>	Hareket durumu	TranState
<i>CommitMode</i>	Kesinleştirme kipi	CommitMode
<i>SecurityScope</i>	Güvenlik kapsamı	SecurityScope
<i>Reserved</i>	Ayrıldı	Ayrıldı

MQIIH 'a Genel Bakış

MQIIH yapısı, IMS köprüsünde IMS ' e gönderilen bir iletiye ilişkin üstbilgi bilgilerini açıklar. For any IBM MQ supported platform you can create and transmit a message that includes the MQIIH structure, but only an IBM MQ for z/OS queue manager can use the IMS bridge. Therefore, for the message to get to IMS from a non-z/OS queue manager, your queue manager network must include at least one z/OS queue manager through which the message can be routed.

Kullanılabilirlik: Tüm IBM MQ sistemleri ve IBM MQ istemcileri.

Amaç: MQIIH yapısı, IMS köprüsünde IBM MQ for z/OS köprüsünün gönderildiği bir iletinin başlangıcındaki var olması gereken bilgileri açıklar.

Biçim adı: MQFMT_IMS.

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.
- Diğer kuyruk yöneticilerine bağlanan uygulamalar, desteklenen karakter kümelerinin ve kodlamaların herhangi birinde bulunan bir MQIIH yapısı sağlayabilir; IMS köprü kuyruğunun sahibi olan kuyruk yöneticisine bağlı alıcı ileti kanalı aracısı, MQIIH ' yi dönüştürür.
- MQIIH yapısının ardından gelen uygulama iletisi verileri, MQIIH yapısıyla aynı karakter kümesinde ve kodlamada olmalıdır. MQIIH yapısındaki *CodedCharSetId* ve *Encoding* alanlarını, uygulama iletisi verilerinin karakter kümesini ve kodlamasını belirtmek için kullanmayın.

Veri, kuyruk yöneticisi tarafından desteklenen yerleşik biçimlerden biri değilse, uygulama iletisi verilerini dönüştürmek için bir veri dönüştürme çıkışı sağlamanız gerekir.

MQIIH alanları

MQIIH yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

Kimlik doğrulayıcı (MQCHAR8)

Bu, RACF parolasıdır ya da PassTicket' dir. Bu isteğe bağlıdır; belirtildiyse, 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.

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.

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 (TRANSACTION makrosu için MSGTYPE parametresi) çalışır. Bu, bir işlem anahtarı aracılığıyla başlatılan işlemler için geçerlidir.

Bu alanın ilk değeri MQICM_COMMIT_THEN_SEND olarak değerlendirilmektedir.

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.

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 CM0 transactions.

MQIIH_CM0_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.

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.

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 419.](#)

İstek iletisine varsayılan değer (MQIIH.replyToFormat= MQFMT_NONE) ise ve yanıt iletisi MQGMO_CONVERT kullanılarak alındıysa, veri dönüştürme gerçekleştirilmez.

Ayrılmış (MQCHAR)

Bu ayrılmış bir alandır; boş bırakılmalıdır.

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.

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.

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.

TranInstanceTanıtıcısı (MQBYTE16)

Bu, hareket eşgörünümü tanıtıcısıdır. Bu alan çıkış iletileri tarafından IMS' den kullanılır, bu nedenle ilk girişte yoksayılır. *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 yoksayılır, 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.

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.

MQIIH için ilk değerler ve dil bildirimleri

Çizelge 50. MQIIH için MQIIH içindeki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
<i>StrucId</i>	MQIIH_STRUC_ID	' IIH? '
<i>Version</i>	MQIIH_VERSION_1	1
<i>StrucLength</i>	MQIIH_LENGTH_1	84
<i>Encoding</i>	Yok	0
<i>CodedCharSetId</i>	Yok	0
<i>Format</i>	MQFMT_NONE	Boşluklar
<i>Flags</i>	MQIH_NONE	0
<i>LTermOverride</i>	Yok	Boşluklar
<i>MFSMapName</i>	Yok	Boşluklar
<i>ReplyToFormat</i>	MQFMT_NONE	Boşluklar
<i>Authenticator</i>	MQIAUT_NONE	Boşluklar
<i>TranInstanceId</i>	MQITII_NONE	Boş Değerler
<i>TranState</i>	MQITS_NOT_IN_SOHBET	' ? '
<i>CommitMode</i>	MQICM_COMMIT_THEN_SEND	' 0 '
<i>SecurityScope</i>	MQISS_CHECK	' C '
<i>Reserved</i>	Yok	' ? '
Notlar:		
1. Sembol mü? tek bir boş karakteri temsil eder.		
2. C programlama dilinde, makro değişkeniMQIIH_VARSAYILAND, tabloda listelenen değerleri içerir. Yapıdaki alanlara ilk değerleri sağlamak için aşağıdaki şekilde kullanılabilir:		
<pre>MQIIH MyIIH = {MQIIH_DEFAULT};</pre>		

C bildirimi

```
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  MFSMapName;     /* 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 */
};

```

COBOL bildirimi

```

** 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.

```

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 */

```

High Level Assembler 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)

```

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

```

MQIMPO-Sorgula ileti özelliği seçenekleri

Aşağıdaki çizelge yapıdaki alanları özetlemektedir. MQIMPO yapısı-sorgu iletisi özellik seçenekleri

Çizelge 51. MQIMPO içindeki alanlar		
Alan	Tanım	Konu
<i>StrucId</i>	Yapı tanıtıcısı	StrucId
<i>Version</i>	Yapı sürüm numarası	S\u00fcr\u00fcm
<i>Options</i>	MQINQMP ' nin işlemini denetleyen seçenekler	Seçenekler
<i>RequestedEncoding</i>	Sorgulayan özelliğin dönüştürüleceği kodlama	RequestedEncoding
<i>RequestedCCSID</i>	Sorgulamanın karakter kümesi	RequestedCCSID
<i>ReturnedEncoding</i>	Döndürülen değerin kodlaması	ReturnedEncoding
<i>ReturnedCCSID</i>	Döndürülen değerin karakter kümesi	ReturnedCCSID
<i>Reserved1</i>	Ayrılmış alan	ReturnedCCSID
<i>ReturnedName</i>	Sorgulanmış özelliğin adı	ReturnedName

Çizelge 51. MQIMPO içindeki alanlar (devamı var)		
Alan	Tanım	Konu
<i>TypeString</i>	Özelliğe ilişkin veri tipinin dizgi gösterimi	<u>TypeString</u>

MQIMPO için genel bakış

Sorgu iletisi özellikleri seçenekleri yapısı.

Kullanılabilirlik: Tüm IBM MQ sistemleri ve IBM MQ istemcileri.

Amaç: MQIMPO yapısı, uygulamaların, iletilerin özelliklerini nasıl sorguladığına ilişkin seçenekleri belirlemesine olanak sağlar. Yapı, MQINQMP çağrısında bir giriş değişikliğidir.

Karakter kümesi ve kodlama: MQIMPO 'daki veriler, uygulamanın uygulama ve kodlamasının karakter kümesinde (MQENC_NATIVE) olmalıdır.

MQIMPO alanları

Sorgu iletisi özellik seçenekleri yapısı-alanlar

MQIMPO yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

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 iletir.

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* değişikliği, 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 görel 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 iletir. 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 *VLength* 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 değerin 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* değiştirilmesi, uygulamanın dönüştürülen özellik değerini sığdırmak için gereken arabelleğin büyüklüğünü belirlemesine olanak vermek için, özellik değerinin dönüştürüleceğini döndürür. Özellik imleci değişmeden kalır.

MQINQMP çağrısının *Type* değiştirilmesinin 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 dönüştürmeleri 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

Özellik veri tipi	Desteklenen hedef veri tipleri
MQTYPE_STRING	MQTYPE_BOOLEAN, MQTYPE_INT8, MQTYPE_INT16, MQTYPE_INT32, MQTYPE_INT64, MQTYPE_FLOAT32, MQTYPE_FLOAT64
MQTYPE_NULL	Yok

Desteklenen dönüştürmeleri yöneten genel kurallar aşağıdaki gibidir:

- Sayısal özellik değeri, 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_SUPPORTED 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, değerın 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 özelliğe yeniden sorgulamak için).

Yalnızca bir özellik imlecini 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 çağrısıya, belirtilen ad ile eşleşen ilk özellik döndürülür.

MQIMPO_INQ_PROP_UNDER_CURSOR seçeneği, gerekirse bir MQINQMP çağrısıyla, aynı özelliği yeniden sorgulamak için kullanılabilir.

İmlecin 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 hiçbirisi 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.

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.

RequestedEncoding (MQUZE)

Sorgu iletisi özellik seçenekleri yapısı- RequestedEncoding alanı

Bu, sorgulayan özellik değerin MQIMPO_CONVERT_VALUE ya da MQIMPO_CONVERT_TYPE belirtildiğinde dönüştürüleceği kodlamadır.

Bu alanın ilk değeri MQENC_NATIVE 'dir.

Reserved1 (MQCHAR)

Bu ayrılmış bir alandır. Bu alanın ilk değeri boşluk karakteridir (4 byte 'lık alan).

ReturnedCCSID (MQHOT)

İleti özelliği seçenekleri yapısı- ReturnedCCSID alanı

Çıkışta, MQINQMP çağrısının *Type* değiştirgesi MQTYPE_STRING ise, döndürülen değerin karakter takıdır.

MQIMPO_CONVERT_VALUE seçeneği belirtilirse ve dönüştürme başarılı olursa, dönüş sırasında *ReturnedCCSID* alanı, geçirilen değerle aynı değerde olur.

Bu alanın ilk değeri sıfır.

ReturnedEncoding (MQlong)

İleti özelliği seçenekleri yapısı- ReturnedEncoding alanı

Çıkışta, döndürülen değerin 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.

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.

StrucId (MQCHAR4)

Sorgu iletisi özellik seçenekleri yapısı- StrucId alanı

Bu, yapı tanıtıcısıdır. Değer şu olmalıdır:

MQIMPO_STRUC_ID

Sorgu iletisi 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.

TypeString (MQCHAR8)

Sorgu iletisi özellik seçenekleri yapısı- TypeString alanı

Özellik 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.

Sürüm (MQUZE)

Sorgu iletisi 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.

MQIMPO için ilk değerler ve dil bildirimleri

Sorgu iletisi özellik seçenekleri yapısı-İlk değerler

Çizelge 52. MQIMPO 'daki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
<i>StrucId</i>	MQIMPO_STRUC_ID	'IMPO'
<i>Version</i>	MQIMPO_VERSION_1	1
<i>Options</i>	MQIMPO_INQ_FIRST	
<i>RequestedEncoding</i>	MQENC_NATIVE	
<i>RequestedCCSID</i>	MQCCSI_APPL	
<i>ReturnedEncoding</i>	MQENC_NATIVE	
<i>ReturnedCCSID</i>	0	
<i>Reserved1</i>	0	
<i>ReturnedName</i>	MQCHARV_DEFAULT	
<i>TypeString</i>	Dizgi ya da boşluk boş değerli	
Notlar:		
1. Boş değerli dizgi ya da boşluklar, C içindeki boş değerli dizeyi ve diğer programlama dillerindeki boş karakterleri belirtir.		
2. C programlama dilinde, makro değişkeniMQIMPO_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlara ilişkin ilk değerleri sağlamak için bunu aşağıdaki şekilde kullanın:		
<pre>MQIMPO MyIMPO = {MQIMPO_DEFAULT};</pre>		

C bildirimi

Sorgu iletisi özellik seçenekleri yapısı-C dili bildirimi

```
typedef struct tagMQIMPO MQIMPO;  
struct tagMQIMPO {
```

```

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 */
};

```

COBOL bildirimi

Sorgu iletisi özellik seçenekleri yapısı-COBOL dil bildirimi

```

** MQIMPO structure
10 MQIMPO.
**   Structure identifier
15 MQIMPO-STRUCID          PIC X(4).
**   Structure version number
15 MQIMPO-VERSION         PIC S9(9) BINARY.
**   Options that control the action of MQINQMP
15 MQIMPO-OPTIONS         PIC S9(9) BINARY.
**   Requested encoding of VALUE
15 MQIMPO-REQUESTEDENCODING PIC S9(9) BINARY.
**   Requested character set identifier of VALUE
15 MQIMPO-REQUESTEDCCSID   PIC S9(9) BINARY.
**   Returned encoding of VALUE
15 MQIMPO-RETURNEDENCODING PIC S9(9) BINARY.
**   Returned character set identifier of VALUE
15 MQIMPO-RETURNEDCCSID   PIC S9(9) BINARY.
**   Reserved field
15 MQIMPO-RESERVED1
**   Returned property name
15 MQIMPO-RETURNEDNAME.
**   Address of variable length string
20 MQIMPO-RETURNEDNAME-VSPTR POINTER.
**   Offset of variable length string
20 MQIMPO-RETURNEDNAME-VSOFFSET PIC S9(9) BINARY.
**   CCSID of variable length string
20 MQIMPO-RETURNEDNAME-VSCCSID PIC S9(9) BINARY.
**   Property data type as string
15 MQIMPO-TYPESTRING      PIC S9(9) BINARY.

```

PL/I bildirimi

Sorgu iletisi özellik seçenekleri yapısı-PL/I dil bildirimi

```

dcl
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          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
ORG 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)

```

MQMD-İleti tanımlayıcısı

Aşağıdaki çizelge yapıdaki alanları özetlemektedir.

Çizelge 53. MQMD 'deki alanlar		
Alan	Tanım	Konu
<i>StrucId</i>	Yapı tanıtıcısı	StrucId
<i>Version</i>	Yapı sürüm numarası	S\u00fcr\u00fcm
<i>Report</i>	Rapor iletileri için seçenekler	Rapor
<i>MsgType</i>	İleti tipi	MsgType
<i>Expiry</i>	İleti kullanım süresi	MQMD-Süre bitimi alanı
<i>Feedback</i>	Geribildirim ya da neden kodu	MQMD-Geribildirim alanı
<i>Encoding</i>	İleti verilerinin sayısal kodlaması	Kodlama
<i>CodedCharSetId</i>	İleti verilerinin karakter kümesi tanıtıcısı	CodedCharSetId
<i>Format</i>	İleti verilerinin adını biçimle	Biçim
<i>Priority</i>	İleti önceliği	Öncelik
<i>Persistence</i>	İleti kalıcılığı	Kalıcılık
<i>MsgId</i>	İleti Tanıtıcısı	MQMD- MsgId alanı
<i>CorrelId</i>	İlinti tanıtıcısı	CorrelId
<i>BackoutCount</i>	Geriletme sayacı	BackoutCount
<i>ReplyToQ</i>	Yanıt kuyruğunun adı	ReplyToQ
<i>ReplyToQMgr</i>	Yanıt kuyruğu yöneticisinin adı	ReplyToQMgr
<i>UserIdentifier</i>	Kullanıcı kimliği	UserIdentifier
<i>AccountingToken</i>	Hesap simgesi	AccountingToken

Çizelge 53. MQMD ' deki alanlar (devamı var)

Alan	Tanım	Konu
<i>ApplIdentityData</i>	Kimlikle ilgili uygulama verileri	ApplIdentityVerileri
<i>PutApplType</i>	İletiyi koyan uygulamanın tipi	PutApplTipi
<i>PutApplName</i>	İletiyi koyan uygulamanın adı	PutApplAdı
<i>PutDate</i>	İletinin konulduğu tarih	PutDate
<i>PutTime</i>	İletinin konulduğu saat	PutTime
<i>ApplOriginData</i>	Köken ile ilgili uygulama verileri	ApplOriginVerileri
Not: <i>Version</i> , MQMD_VERSION_2değerinden küçükse, geri kalan alanlar yoksayılır.		
<i>GroupId</i>	Grup tanıtıcısı	GroupId
<i>MsgSeqNumber</i>	Grup içindeki mantıksal iletinin sıra numarası	MsgSeqNumarası
<i>Offset</i>	Mantıksal iletinin başlangıcındaki fiziksel iletilerde verilerin görelî konumu	Görelî Konum
<i>MsgFlags</i>	İleti İşaretleri	MQMD- MsgFlags alanı
<i>OriginalLength</i>	Özgün iletinin uzunluğu	OriginalLength

MQMD için genel bakış

Uygunluk: Tüm IBM MQ sistemleri, artı IBM MQ MQI clients bu sistemlere bağlı.

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 deęiştiricidir.

Sürüm: MQMD 'nin yürürlükteki sürümü MQMD_VERSION_2' dir. Birkaç ortam arasında taşınabilir olması amaçlanan uygulamalar, ilgili tüm ortamlarda gerekli olan MQMD sürümünün desteklenmesini sağlamalıdır. Yalnızca yapının daha yeni sürümlerinde var olan alanlar, izleyen açıklamalarda olduęu gibi tanımlanır.

Desteklenen programlama dilleri için sağlanan üstbilgi, COPY ve INCLUDE dosyaları, ortam tarafından desteklenen MQMD 'nin en son sürümünü içerir, ancak *Version* alanının ilk değeri MQMD_VERSION_1olarak ayarlıdır. version-1 yapısında mevcut olmayan alanları kullanmak için, uygulamanın *Version* alanını, gereken sürümün sürüm numarasına ayarlaması gerekir.

version-1 yapısına ilişkin bir bildirge MQMD1adıyla kullanılabilir.

Karakter kümesi ve kodlama: MQMD 'deki veriler, yerel kuyruk yöneticisinin karakter kümesinde ve kodlamasında olmalıdır; bunlar *CodedCharSetId* kuyruk yöneticisi öznitelięi ve MQENC_NATIVE tarafından verilir. 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: Bir version-2 MQMD, bir version-1 MQMD 'si kullanılarak ve ileti verilerinin MQMDE yapısıyla önceden düzeltilmesine eşdeęerdir. Ancak, MQMDE yapısındaki tüm alanların varsayılan değeri varsa, MQMDE atlanabilir. Bir version-1 MQMD artı MQMDE, açıklandığı biçimde kullanılır:

- MQPUT ve MQPUT1 çağrılarında, uygulama bir version-1 MQMD sağlıyorsa, uygulama isteęe baęlı olarak bir MQMDE ile ileti verilerini önleyebilir, MQMD 'de *Format* alanını MQMDE'ye ayarlayarak MQMDE 'yi (MQMDE) ayarlayarak, MQMDE ile birlikte ileti verilerini MQFMT_MD_EXTENSION ile önleyebilir.

Uygulama bir MQMDE sağlamazsa, kuyruk yöneticisi MQMDE içindeki alanlar için varsayılan değerleri 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 *değil*, MQPUT ve MQPUT1 çağrılarının çıktısındaki MQMD 'deki eşdeğer alanlara herhangi bir değer döndürür; uygulama bu çıkış değerlerini gerektiriyorsa, bu değer bir version-2 MQMD kullanmalıdır.

- MQGET çağrısında, uygulama bir version-1 MQMD sağlıyorsa, kuyruk yöneticisi örnekleri bir MQMDE ile döndürülür; ancak, MQMDE içindeki alanlardan biri ya da daha fazlası varsayılan olmayan bir değere sahip olur. MQMD 'deki *Format* alanının, MQMDE 'nin bir MQMDE olduğunu belirtmek için MQFMT_MD_EXTENSION değeri olur.

MQMDE 'deki alanlar için kuyruk yöneticisinin kullandığı varsayılan değerler, [Çizelge 58 sayfa 457](#) içinde gösterilen bu alanların ilk değerlerinin aynıdır.

Bir ileti iletim kuyruğunda olduğunda, MQMD 'deki bazı alanlar belirli değerlere ayarlanır; ayrıntılar için [“MQXQH-İletim kuyruğu üstbilgisi” sayfa 599](#) konusuna bakın.

İleti bağlamı: MQMD 'deki bazı alanlar ileti bağlamını içerir. İki tip ileti bağlamı vardır: *kimlik bağlamı* ve *köken bağlamı*. Tipik olarak:

- Kimlik bağlamı, iletiyi *ilk olarak* koyan uygulamayla ilgilidir.
- Çıkış noktası bağlamı, iletiyi *en son* 'in koyduğu uygulamayla ilgilidir.

Bu iki uygulama aynı uygulama olabilir, ancak bunlar aynı zamanda farklı uygulamalar da olabilir (örneğin, bir ileti bir uygulamadan başka bir uygulamaya iletildiğinde).

Tanıtıcı ve kaynak bağlamı tipik olarak tanımlansa da, MQMD 'deki bağlam alanlarının her iki tipinin içeriği, ileti konduğunda belirtilen MQPMO_ * _CONTEXT seçeneklerine bağlıdır. Sonuç olarak, kimlik bağlamı, iletiyi ilk olarak koyan uygulamayla ilişkili değildir ve kaynak bağlamı, en son iletiyi en son yerleştiren uygulamayla ilişkili değildir; uygulama takımının tasarımına bağlıdır.

Message Channel Agent (MCA) ileti bağlamını hiçbir zaman değiştirmez. Uzak kuyruk yöneticilerinden gelen iletileri alan MCA 'lar, MQPUT ya da MQPUT1 çağrısında MQPMO_SET_ALL_CONTEXT bağlam seçeneğini kullanır. Bu işlem, MCA 'nın gönderilen MCA' dan gelen iletiyle birlikte gönderilen ileti bağlamını tam olarak korumasını sağlar. Ancak sonuç, kaynak bağlamı, iletiyi gönderen ve alınan MCA 'larla ilişkilendirilmediğini de sağlar. Kaynak bağlamı, iletiyi koyan daha önceki bir uygulamaya gönderme yapıyor. Tüm ara uygulamalar ileti bağlamını geçirdiyse, kaynak bağlamı, kaynak uygulamanın kendisini gösterir.

Tanımlarda, bağlam alanları, önceden açıklandığı gibi kullanılsa da açıklanmıştır. İleti bağlamına ilişkin ek bilgi için [İleti bağlamı](#) başlıklı konuya bakın.

MQMD için alanlar

MQMD yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

Çizelge 54. MQMD 'deki alanlar		
Alan	Tanım	Konu
<i>StrucId</i>	Yapı tanıtıcısı	StrucId
<i>Version</i>	Yapı sürüm numarası	S\u00fcr\u00fcr
<i>Report</i>	Rapor iletileri için seçenekler	Rapor
<i>MsgType</i>	İleti tipi	MsgType
<i>Expiry</i>	İleti kullanım süresi	MQMD-Süre bitimi alanı
<i>Feedback</i>	Geribildirim ya da neden kodu	MQMD-Geribildirim alanı
<i>Encoding</i>	İleti verilerinin sayısal kodlaması	Kodlama

Çizelge 54. MQMD ' deki alanlar (devamı var)

Alan	Tanım	Konu
<i>CodedCharSetId</i>	İleti verilerinin karakter kümesi tanıtıcısı	CodedCharSetId
<i>Format</i>	İleti verilerinin adını biçimle	Biçim
<i>Priority</i>	İleti önceliği	Öncelik
<i>Persistence</i>	İleti kalıcılığı	Kalıcılık
<i>MsgId</i>	İleti Tanıtıcısı	MQMD- MsgId alanı
<i>CorrelId</i>	İlinti tanıtıcısı	CorrelId
<i>BackoutCount</i>	Geriletme sayacı	BackoutCount
<i>ReplyToQ</i>	Yanıt kuyruğunun adı	ReplyToQ
<i>ReplyToQMGr</i>	Yanıt kuyruğu yöneticisinin adı	ReplyToQMGr
<i>UserIdentifier</i>	Kullanıcı kimliği	UserIdentifier
<i>AccountingToken</i>	Hesap simgesi	AccountingToken
<i>ApplIdentityData</i>	Kimlikle ilgili uygulama verileri	ApplIdentityVerileri
<i>PutApplType</i>	İletiyi koyan uygulamanın tipi	PutApplTipi
<i>PutApplName</i>	İletiyi koyan uygulamanın adı	PutApplAdı
<i>PutDate</i>	İletinin konulduğu tarih	PutDate
<i>PutTime</i>	İletinin konulduğu saat	PutTime
<i>ApplOriginData</i>	Köken ile ilgili uygulama verileri	ApplOriginVerileri
Not: <i>Version</i> , MQMD_VERSION_2değerinden küçükse, geri kalan alanlar yoksayılır.		
<i>GroupId</i>	Grup tanıtıcısı	GroupId
<i>MsgSeqNumber</i>	Grup içindeki mantıksal iletinin sıra numarası	MsgSeqNumarası
<i>Offset</i>	Mantıksal iletinin başlangıcındaki fiziksel iletilerde verilerin göreceli konumu	Göreceli Konum
<i>MsgFlags</i>	İleti İşaretleri	MQMD- MsgFlags alanı
<i>OriginalLength</i>	Özgün iletinin uzunluğu	OriginalLength

AccountingToken (MQBYTE32)

Bu, iletinin **kimlik bağlamının** bir parçası olan muhasebe simgidir. İleti bağlamına ilişkin daha fazla bilgi için bkz. “MQMD için genel bakış” sayfa 405 ; ayrıca bkz. [İleti bağlamı](#).

AccountingToken , bir uygulamanın, iletinin bir sonucu olarak çalışma için uygun bir şekilde ücret ödemesine olanak sağlar. Kuyruk yöneticisi bu bilgileri bir bit dizisi olarak değerlendirir ve içeriğini denetmez.

Kuyruk yöneticisi bu bilgileri aşağıdaki gibi oluşturur:

- 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 kurtarma simgesiyle bitleştirilen 8 karakterlik PSB adı.
- IBM i' ta, muhasebe bilgileri işe ilişkin muhasebe koduna ayarlanır.
- UNIX sistemleri' ta, hesap bilgileri ASCII karakterlerinde sayısal kullanıcı tanımlayıcıyla belirlenir.
- Windows üzerinde, muhasebe bilgileri sıkıştırılmış biçimde bir Windows güvenlik tanıtıcısına (SID) ayarlıdır. SID, *UserIdentifier* alanında saklanan kullanıcı kimliğini benzersiz olarak tanımlar. SID, *AccountingToken* alanında saklandığında, 6 baytlık Tanıtıcı Yetkisi (SID ' nin üçüncü ve sonraki baytlarında bulunur) atlanır. For example, if the Windows SID is 28 bytes long, 22 bytes of SID information are stored in the *AccountingToken* field.
- Hesap alanının son baytı (byte 32), hesap belirteci tipine ayarlanır (bu durumda MQACTT_NT_SECURITY_ID, x '0b'):

MQACTT_CICS_LUOW_ID

CICS LUOW tanıtıcısı.

MQACTT_NT_SECURITY_ID

Windows güvenlik tanıtıcısı.

MQACTT_OS400_ACCOUNT_TOKEN

IBM i hesap simgesi.

MQACTT_UNIX_NUMERIC_ID

UNIX sistemleri sayısal tanıtıcısı.

MQACTT_USER

Kullanıcı tanımlı hesap belirteci.

MQACTT_UNKNOWN

Bilinmeyen muhasebe belirteci tipi.

The accounting-token type is set to an explicit value only in the following environments: AIX, HP-UX, IBM i, Solaris, Linux, Windows, plus IBM MQ MQI clients connected to these systems. Diğer ortamlarda, muhasebe belirteci tipi MQACTT_UNKNOWN değerine ayarlanır. Bu ortamlarda, alınan muhasebe simgesinin tipini algılamak için *PutAppLType* alanını kullanın.

- Diğer tüm baytlar ikili sıfıra ayarlanır.

MQPUT ve MQPUT1 çağrıları için, *PutMsgOpts* parametresinde MQPMO_SET_IDENTITY_CONTEXT ya da MQPMO_SET_ALL_CONTEXT belirtilmişse, bu giriş/çıkış alanıdır. Ne MQPMO_SET_IDENTITY_CONTEXT ne de MQPMO_SET_ALL_CONTEXT belirtilirse, bu alan girişte yoksayılr ve yalnızca çıkış alanıdır. İleti bağlamına ilişkin ek bilgi için [İleti bağlamı](#) başlıklı konuya bakın.

Bir MQPUT ya da MQPUT1 çağrısının başarıyla tamamlanmasından sonra, bu alan, bir kuyruğa konduysa, iletiyle birlikte gönderilen *AccountingToken* ögesini içerir. Bu, alıkonursa iletiyle birlikte tutulan *AccountingToken* değeri (alıkonan yayınlara ilişkin ek ayrıntılar için “MQPMO seçenekleri (MQUWT)” sayfa 488 içindeki MQPMO_RETAIN ile ilgili açıklamaya bakın) ancak bu ileti, gönderilen tüm yayınlarda *AccountingToken* ' i geçersiz kılmak için bir değer sağladığından, bu ileti abonelere bir yayın olarak gönderildiğinde *AccountingToken* olarak kullanılmaz. İletinin bağlamı yoksa, alan tamamen ikili olarak sıfırdır.

Bu, MQGET çağrısına ilişkin bir çıkış alanıdır.

Bu alan, kuyruk yöneticisinin karakter kümesine dayalı olarak herhangi bir çeviriye tabi değildir; alan, bir karakter dizgisi olarak değil, bitler dizgisi olarak işlenir.

Kuyruk yöneticisi, bu alandaki bilgilerle hiçbir şey yapmaz. Hesap, muhasebe amacıyla bilgileri kullanmak istiyorsa, bilgileri yorumlamalıdır.

AccountingToken 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.

Bu alanın uzunluğu MQ_ACCOUNTING_TOKEN_LENGTH tarafından verilir. Bu alanın ilk değeri MQACT_NONE olur.

ApplIdentityVerileri (MQCHAR32)

Bu, iletinin **kimlik bağlamının** bir parçasıdır. İleti bağlamına ilişkin daha fazla bilgi için bkz. “MQMD için genel bakış” sayfa 405 ve [İleti bağlamı](#).

ApplIdentityData , 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.

MQPUT ve MQPUT1 çağrılarını için, *PutMsgOpts* parametresinde MQPMO_SET_IDENTITY_CONTEXT ya da MQPMO_SET_ALL_CONTEXT belirtilmişse, bu giriş/çıkış alanıdır. Boş değerli bir karakter varsa, boş değer ve izleyen karakterler kuyruk yöneticisi tarafından boşluklara dönüştürülür. Ne MQPMO_SET_IDENTITY_CONTEXT ne de MQPMO_SET_ALL_CONTEXT belirtilirse, bu alan girişte yoksayılr ve yalnızca çıkış alanıdır. İleti bağlamına ilişkin ek bilgi için [İleti bağlamı](#) başlıklı konuya bakın.

Bir MQPUT ya da MQPUT1 çağrısının başarıyla tamamlanmasından sonra, bu alan, bir kuyruğa konduysa, iletiyle birlikte gönderilen *ApplIdentityData* ögesini içerir. Bu, alıkonursa iletiyle alıkonan *ApplIdentityData* değerinin (alıkonan yayınlarla ilgili daha fazla ayrıntı için bkz. MQPMO_RETAIN ile ilgili açıklamalara bakın) değeri olur, ancak bu ileti, gönderilen tüm yayınlarda *ApplIdentityData* ' i geçersiz kılmak için bir değer sağladıkları için, bu ileti abonelere bir yayın olarak gönderildiğinde *ApplIdentityData* olarak kullanılmaz. İletinin bağlamı yoksa, alan tümüyle boştur.

Bu, MQGET çağrısına ilişkin bir çıkış alanıdır. 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.

ApplOriginVerileri (MQCHAR4)

Bu, iletinin **başlangıç bağlamının** bir parçasıdır. İleti bağlamına ilişkin daha fazla bilgi için bkz. “MQMD için genel bakış” sayfa 405 ve [İleti bağlamı](#).

ApplOriginData , iletinin kökeniyle ilgili ek bilgi sağlamak için kullanılabilecek uygulama takımı tarafından tanımlanan bilgilerdir. 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.

MQPUT ve MQPUT1 çağrılarını için, *PutMsgOpts* parametresinde MQPMO_SET_ALL_CONTEXT belirtilmişse, bu bir giriş/çıkış alanıdır. Alan içindeki boş değerli bir karakteri izleyen herhangi bir bilgi atılır. Kuyruk yöneticisi boş değerli karakteri ve izleyen herhangi bir karakteri boşluklara dönüştürür. MQPMO_SET_ALL_CONTEXT belirtilmediyse, bu alan girişte yoksayılr 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 değerli dizedir ve diğer programlama dillerindeki 4 boş karakterdir.

İleti yayınlandığında, *ApplOriginData* değeri ayarlansa da, aldığı abonelikte boş olur.

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 sayının doğruluğu, *HardenGetBackout* kuyruk özneliden etkilenir; bkz. “[Kuyruklara ilişkin öznelilikler](#)” sayfa 811.

z/OS üzerinde 255 değeri, iletinin 255 ya da daha fazla kez gerileteceği anlamına gelir; döndürülen değer 255 'ten büyük bir değer değildir.

Bu, MQGET çağrısına ilişkin bir çıkış alanıdır. Bu, MQPUT ve MQPUT1 çağrıları için yoksayılr. Bu alanın başlangıç değeri 0' 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 öznelilikler](#)” sayfa 778 '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_CCSDID_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_VARS

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 iletisi içinde yer alan her MQCFST, MQCFSL ve MQCFSF 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 ileti içinde bulunan her MQCFST, MQCFSL ve MQCFSF 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-Embedded PCF üstbilgisi” sayfa 353.](#)

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.

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_INHERIT 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 çağrılarında ilişkin bir giriş alanı olur. Bu alanın ilk değeri MQCCSI_Q_MGR ' dir.

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 oluşturulmaması 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 belirtilerek *not* ile 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 değeri *değil* belirtildiyse, bu bir giriş alanıdır ve MQPMO_NEW_CORREL_ID : belirtilmişse 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.

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.

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 değişmez 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 872](#) .

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.

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 kuyrukta geçirdiği süreyi ve uzak bir kuyruğa konursa ara iletim kuyruklarında da yer alan süreyi yansıtacak şekilde belirlenir. İleti kanalı aracılığıyla, 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 azaltabilir. Ancak, süre bitim süresi yaklaşık olarak kabul edilir ve değer, küçük zaman aralıklarını yansıtacak 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 hiçbirisi 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 letisi 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 letisi 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 yoksayılr, 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 tetiklemesi de dahil olmak üzere, kuyruğun tüm amaçlar için kuyruklardaki ileti sayısına kadar sayılabilir.
 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_UNSIINSINLI 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_UNESSISIT 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 iletisi verilerinin 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 iletisi için süre bitimi sınırsız olur. Ek bilgi için [İşaretler](#) konusuna bakın.

Şu özel değer tanındı:

MQE_UNSSINSIZ

İletinin süre bitimi sınırsız.

Bu, MQGET çağrısına ilişkin bir çıkış alanıdır ve MQPUT ve MQPUT1 çağrılarına 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

EXPYRINT (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, EXPYINT 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.

V 8.0.0.4 *Düşük süre bitim süreleri zorlamalı*

Sistem yöneticileri, kuyrukta ya da konuyla ilgili **CUSTOM** özniteisinde belirtilen CAPEXPER özniteliğini kullanarak, bir kuyruğa ya da konuya ilişkin süre bitimi süresini sınırlayabilirler.

z/OS Bu özelliği z/OS'ta etkinleştirmek için, APAR PI50761 ' i uygulamalı ve yeni özelliği MQSC komutu **RECOVER QMGR (TUNE CAPEXPY ON)** ile etkinleştirmeniz gerekir.

Önemli: RECOVER QMGR komutu, IBM tarafından kullanılmak üzere ayrılmıştır ve olağan durumda yalnızca IBM desteği tarafından kılavuz altında kullanılmalıdır.

z/OS Kuyruk yöneticisi başlatma sırasında CSQINP2 bitişirmesinde işlenen bir dosyaya ekleyerek, komutu bir konsolda ya da kuyruk yöneticisi yapısınıdaki bir dosyaya ekleyebilirsiniz. Komutun etkisinin bir kuyruk yöneticisi yeniden başlatıldığını kabul etmediğine dikkat edin.

MQMD ' nin **Expiry** alanında belirtilen süre bitimi, bir uygulama tarafından, kuyrukta ya da konudaki **CUSTOM** özniteisinde belirtilen CAPEXPY değerinden daha büyük olan bir süre, bu CAPEXPY değeriyle değiştirilir. Bir uygulama tarafından belirtilen süre bitimi (CAPEXPY değerinden daha düşük bir değer 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 konduğunda, ileti süre bitimi için üst sınır olarak tüm CAPEXPY değerleri en düşük değeri kullanılır.

CAPEXPY değerlerinde yapılan değişiklikler hemen yürürlüğe girsin. 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 kümenin MQOO_BIND_NOT_FIXED ile açılan bir kuyruğa konduğu bir kümede, ileti seçilen hedef kuyruk yöneticisine iletiyi gönderen, kanal tarafından kullanılan iletim kuyruğu için ayarlanan CAPEXPY değerine bağlı olarak, her bir kanta farklı süre bitimi değerleri atanabilir.

Note that a put to a queue or topic by a JMS application specifying a delivery delay fails with MQRC_EXPIRY_ERROR, if the delivery delay is beyond the resolved expiry time for the target queue or topic. Bir JMS Hedefi için çözölen bir kuyruğun CAPEXPY öznitelik kümesi bu hataya neden olabilir.

Not: CAPEXPY 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.

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 konmadığında 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 değerin MQFB_NONE olması ya da sistem aralığı ya da uygulama aralığı içinde olması gerekir. Bu, MsgTypedegeri 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 bildirimi (MQRO_PAN 'a bakın).

MQFB_NAN

Negatif işlem bildirimi (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 iletisi gönderilmesi, o yönetim ortamına işlemeyi 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 0x1A değilse, 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

İletinin ilişkili olduğu etkinliklerin sayısı maksimum etkinlik sınırını aştığından izleme rotası iletisi atıldığında döndürülür.

MQFB_NOT_ILETILDI

İzleme rotası iletisi atıldığında, izleme rotası iletilerini desteklemeyen uzak bir kuyruk yöneticisine gönderilmek üzere atıldığında döndürülür.

MQFB_NOT_TESLİM

İzleme rotası iletisi yerel bir kuyruğa konmak üzere olduğu için atıldığında döndürülür.

MQFB_UNSUPPORTED_FORWARDING

İletme (iletme) değiştirgesindeki bir değer tanınmadığı için, izleme rotası iletisi atıldığında ve reddedilen bit maskesinde yer aldığından bu ileti yayınlanır.

MQFB_UNSUPPORTED_DELIVERY

Teslim parametresindeki bir değer tanınmadığı için izleme rotası iletisi atıldığında ve reddedilen bit maskesinde yer aldığından, döndürülen rota iletisi 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 kullanılabılır 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.

Biçim (MQCHAR8)

Bu ad, iletiyi gönderenin, iletteki verilerin doğasını alıcıya göstermek için kullandığı bir addır. 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 büyüklüklerde MQ ' un başındaki adlar, kuyruk yöneticisi tarafından tanımlanan anlamlara sahiptir; kendi biçimlerin 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 [Programlanır 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 nesnelerle 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 iletiden gelen veriler, MQDLH yapısından hemen sonra gelir. Özgün ileti verilerinin biçim adı, MQDLH yapısındaki *Format* alanı tarafından verilir; bu yapının ayrıntıları için “MQDLH-Dead-letter üstbilgisi” sayfa 341 ' 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 335 ' e bakın. MQGET çağrısında MQGMO_CONVERT seçeneği belirtildiyse, MQFMT_DIST_HEADER biçimi olan iletiler dönüştürülebilmektedir.

Bu biçim şu ortamlarda desteklenir: AIX, HP-UX, IBM i, Solaris, Linux, Windows, artı IBM MQ MQI clients bu sistemlere bağlı olarak.

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 iletileri biçimi. Bu biçimle, uygulamalarının önceki PCF parametreleriyle başa çıkabilmesi koşuluyla, kullanıcı verilerinin izleme rotası iletileriyle birlikte gönderilmesine izin verilir.

PCF üstbilgisi ilk üstbilgi **olmalıdır** , ya da ileti bir izleme rotası iletileri olarak işlenmez. Bu, iletinin bir grupta yer almayacağı ve izleme rotası iletilerinin kesimlenemeyeceği anlamına gelir. Bir grup içinde izleme rotası iletileri gönderilirse, ileti, MQRC_MSG_NOT_ALLOWED_IN_GROUP neden koduyla reddedilir.

MQFMT_ADMIN, izleme rotası iletilerinin biçimi için de kullanılabilir; ancak, bu durumda izleme rotası iletileriyle 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 dönüştürülebiliyor. Version-2 olay iletileri yalnızca z/OS' ta dönüştürülebiliyor.

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 391 ve “ReplyToBiçimi (MQCHAR8)” sayfa 391 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, 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 , *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 sınırdan 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 iletileri verileri) izler. MQMDE 'yi izleyen verilerin biçim adı, karakter kümesi ve kodlaması, MQMDE' deki *Format, CodedCharSetIdve Encoding* alanları tarafından verilir. Bu yapıyla ilgili ayrıntılar için bkz. “MQMDE-İleti tanımlayıcı uzantısı” sayfa 453 . 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 iletileri 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 iletileri üstbilgisi” sayfa 528 . MQGET çağrısında MQGMO_CONVERT seçeneği belirtildiyse, bu biçimdeki iletiler dönüştürülebilmektedir.

Bu biçim şu ortamlarda desteklenir: AIX, HP-UX, IBM i, Solaris, Linux, Windows, artı IBM MQ MQI clients bu sistemlere bağlı olarak.

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 MQRFH2 ile başlar ve isteğe bağlı olarak diğer veriler tarafından takip edilir. İsteğe bağlı verilerin biçim adı, karakter kümesi ve kodlanması (varsa), MQRFH2 içinde *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-Tetikleme iletisi” sayfa 579 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/OSüzerinde, MQFMT_WORK_INFO_HEADER biçimi olan iletelerde *kullanıcı verilerini* dönüştürmek için MQGET çağrısında MQGMO_CONVERT seçeneğini belirtin. Ancak, MQWIIH yapısının kendisi her zaman kuyruk yöneticisinin karakter kümesinde 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 iletiden 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 599 .

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 çağrılarına ilişkin bir giriş alanı. Bu alanın uzunluğu MQ_FORMAT_LENGTH tarafından verilir. Bu alanın ilk değeri MQFMT_NONE olur.

GroupId (MQBYTE24)

Bu, fiziksel iletinin ait olduğu belirli ileti grubunu ya da mantıksal iletiyi tanımlamak için kullanılan bayt dizilimidir. *GroupId* , ileti için bölümlmeye 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 *değil* belirtildi.

Bunlar, rapor iletisi olmayan iletiler için bu çağrılar kullanmanın önerilen yollarıdır. Ancak, uygulama daha fazla denetim gerektiriyorsa ya da çağrı MQPUT1 ise, 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 *değil* 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 46 sayfa 377 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_2 değerinden küçükse bu alan yok sayılır.

MsgFlags (MQUZE)

MsgFlags , iletinin özniteliklerini belirten işaretlerdir ya da işlemeyi denetler.

MsgFlags , aşağıdaki kategorilere ayrılır:

- Bölümleme işaretleri
- Durum işaretleri

Bölümleme işaretleri: Bir ileti kuyruk için çok büyük olduğunda, iletiyi kuyruğa koyma girişimi genellikle başarısız olur. Bölümleme, 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_COMPLETE_MSG seçeneğinin ayrıntıları için "MQGMO-Get-message seçenekleri" sayfa

358 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 çağırısı, 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 çağırısında 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 çağırısından 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 yönlenebilirler. *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 aşağıdaki gibi işler:

- Kullanıcı tanımlı biçimler, 16 bayttan oluşan sınırlarda bölünmeye başlanır; kuyruk yöneticisi 16 bayttan küçük (son bölüm dışında) bölüm oluşturmaz.
- 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 baytından 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, bölümleme 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 bölümleme gerçekleştiremez. İleti bölümlmeye 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 bölümlemenin 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 aracısı (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 yansıtabiliyor.

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 bölüğüdür. 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 iletilerden 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 sıfır, 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 bölümdür.

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 belirtilinceye 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 belirtilinceye kadar açık kalmalıdır. Bu koşul karşılanmazsa, çağrı neden kodu MQRC_INCOMPLETE_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_INCOMPLETE_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: İletin 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 bölünmektedir; ayrıntılar için [“Rapor seçenekleri ve ileti işaretleri” sayfa 875'](#) e bakın.

Bu alanın ilk değeri MQMF_NONE olur. *Version* , MQMD_VERSION_2 değerinden küçükse bu alan yok sayılır.

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 bir kuyruk yöneticisinden diğerine ileti akışı olduğunda ileti tanıtıcısı 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.²İleti 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 döndürülebilmek 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 488](#) 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 çağrıdan geri dönmeye değişmeden 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 değer 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 şu hariç olmak üzere bu alanı kullanmıyor:

- Yukarıda açıklandığı gibi, istenirse benzersiz bir değer oluştur
- İletiyeye ilişkin alma isteğini içeren uygulamaya değer sağlayın

² 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 ' ta 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.

- Değeri, bu ileti hakkında oluşturduğu herhangi bir rapor iletisinin *CorrelId* alanına kopyalayın (*Report* seçeneklerine bağlı olarak).

Kuyruk yöneticisi ya da bir ileti kanalı aracısı bir rapor iletisi 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 çağrısına ilişkin *MsgId* , kuyruktan belirli bir iletiyi almak için kullanılabilecek beş alandan biridir. Olağan durumda MQGET çağrısı kuyrukta 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 değerlere ayarlar ve ardından, bu alanları seçim ölçütü olarak kullanmak için MQGMO 'daki *MatchOptions* alanındaki karşılık gelen MQMO_ * eşleşme seçeneklerini ayarlar. Yalnızca, bu alanlarda belirtilen değerlere sahip iletiler alma için adaydır. *MatchOptions* alanı için varsayılan değer (uygulama tarafından değiştirilmediyse) 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, imlecin 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:

- an MQGET call by *CorrelId* on 64-bit distributed platforms where the *CorrelId* index eliminates the need to perform a true sequential scan.
- 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 46 sayfa 377](#) .

Specifying MQMI_NONE as the message identifier has the same effect as *değil* specifying MQMO_MATCH_MSG_ID, that is, *herhangi biri* message identifier matches.

MQGET çağrısının *GetMsgOpts* parametresinde MQGMO_MSG_UNDER_CURSOR seçeneği belirtilirse bu alan yoksayılır.

MQGET çağrısından dönüşte, *MsgId* alanı, döndürülen iletinin ileti tanıtıcısına ayarlanır (varsa).

Aşağıdaki özel değer 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 çağrılarına ilişkin bir giriş/çıkış alanıdır. Bu alanın uzunluğu MQ_MSG_ID_LENGTH tarafından verilir. Bu alanın ilk değeri MQMI_NONE olur.

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, *değil* belirtildi.

Bunlar, rapor iletisi olmayan iletiler için bu çağrılar kullanmanın önerilen yollarıdır. Ancak, uygulama daha fazla denetim gerektiriyorsa ya da çağrı MQPUT1 ise, uygulama *MsgSeqNumber* ' in uygun bir değere ayarlandığını doğrulamalıdır.

MQPUT ve MQPUT1 çağrılarında giriş sırasında, kuyruk yöneticisi, Kuyrukta fiziksel sıralama alanında açıklanan değeri kullanır. MQPUT ve MQPUT1 çağrılarının çıktısında, kuyruk yöneticisi bu alanı, iletiyle birlikte gönderilen değere ayarlar.

MQGET çağrısına giriş sırasında, kuyruk yöneticisi Çizelge 46 sayfa 377 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_2 değerinden küçükse bu alan yok sayılır.

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 tanımlanmıştır:

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 aracı 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 çağrıları 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 çağrıları için bir giriş alanı. Bu alanın ilk değeri MQMT_DATAGRAM ' tır.

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 MQPUT1 ise, uygulama *Offset* ' in uygun bir değere ayarlandığından emin olmalıdır.

MQPUT ve MQPUT1 çağrılarına giriş sırasında, kuyruk yöneticisi, Kuyrukta fiziksel sıralama alanında 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 46 sayfa 377 içinde gösterilen değeri kullanır. MQGET çağrısından çıktıda, kuyruk yöneticisi bu alanı, alınan ileti için değere ayarlar.

Bu alanın ilk değeri sıfır. *Version* , MQMD_VERSION_2 değerinden 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

İletin 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.

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_PERSISTENT

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 kaldırılincaya 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 başlatılamadığından emin değildir. Bu, kuyruk yöneticisi yeniden başlatıldığında yardımcı belleğimde, iletinin el değmemiş bir kopyası bulunsu 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 yöneticisi tarafından kuyruk paylaşımı grubunda yeniden başlatılır, ancak paylaşılan kuyruklarda iletileri depolamak için kullanılan bağlaşım tesisinin başarısızlıklarını atlatmaz.

MQPER_PERSISTENCE_AS_Q_DEF

- Kuyruk bir küme kuyruğıysa, 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 *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 *yemel* queue manager, even if the destination queue manager is remote.

Kuyruk adı çözümleme yolunda birden çok tanımlama varsa, yoldaki *ilk* tanımındaki bu özniteliğin 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 ileti için istek ileti için 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 çağrılarına ilişkin bir giriş alanı. Bu alanın ilk değeri MQPER_PERSISTENCE_AS_Q_DEF 'dir.

Öncelik (MQUBE)

MQPUT ve MQPUT1 çağrıları için, değerin 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, 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 *Priority* alanına kopyalanır. *DefPriority* daha sonra değiştirilirse, kuyruğa önceden yerleştirmiş olan iletiler etkilenmez.

- Kuyruk bir küme kuyruğı değilse, ileti için öncelik, hedef kuyruk yöneticisi uzak olsa bile, *yerel* kuyruk yöneticisinde tanımlandığı şekilde *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 kuyruğı
- Yerel kuyruk
- Uzak kuyruğun yerel tanımlaması
- Kuyruk yöneticisi diğer adı
- Bir iletim kuyruğı (örneğin, *DefXmitQName* kuyruğı)

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 özniteliği 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 izin 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 izin tipi ile ilgili ek bilgi için [MsgDeliverySequence attribute](#) başlıklı konuya bakın.

Bir iletiyi yanıtlarken, uygulamaların yanıt ileti için istek ileti için 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 çağrılarına ilişkin bir giriş alanı. Bu alanın ilk değeri MQPRI_PRIORITY_AS_Q_DEF olur.

PutApplAd (MQCHAR28)

Bu, iletiyi koyan uygulamanın adıdır ve iletinin *başlangıç bağlamı* 'nin bir parçasıdır. İçerik, platformlar arasında farklılık gösterir ve yayınlar arasında farklılık gösterebilir.

İleti bağlamına ilişkin daha fazla bilgi için bkz. [“MQMD için genel bakış” sayfa 405](#) ve [İleti bağlamı](#).

PutApplName biçimi, *PutApplType* değerine bağlıdır ve bir yayından diğerine değişebilir. Değişiklikler nadirdir, ancak ortam değişirse, bunlar gerçekleşir.

Kuyruk yöneticisi bu alanı (yani, MQPMO_SET_ALL_CONTEXT dışındaki tüm seçenekler için) ayarlarken, alanı, ortam tarafından belirlenen bir değere ayarlar:

- **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 ölümektup kuyruğuna yerleştirilecek bir iş adı.

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** On Windows systems, the queue manager uses the following names:
 - 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.
- **UNIX** UNIX sistemleri üzerinde, kuyruk yöneticisi aşağıdaki adları kullanır:
 - Bir CICS uygulaması için, CICS işlem adı
 - CICS dışı bir uygulama için MQ , işletim sisteminde işlemin adını sorar. Bu, tam yolu olmayan program dosyası adı olarak döndürülür. Sonra MQ , bu süreç adını MQMD.PutApplName alanı aşağıdaki gibi olur:

AIX

Ad 28 bayttan küçükse ya da bu baytlardan küçükse, boşluk ile sağa doldurulmuş olarak ad eklenir.

Ad 28 byte 'tan büyükse, adın en solundaki 28 byte 'ı eklenir.

Linux ve Solaris

Ad 15 bayttan küçükse ya da bu baytlardan küçükse, boşluk ile sağa doldurulmuş olarak ad eklenir.

Ad 15 bayttan büyükse, adın en soldaki 15 baytı eklenir ve boşluklarla sağa doldurulur.

HP-UX

Ad 14 bayttan küçükse ya da bu bayttan küçükse ad, boşluklarla doldurulur ve boşlukla doldurulur.

Ad 14 bayttan büyükse, adın en solundaki 14 byte 'ı eklenir ve boşluklarla doldurulur.

Örneğin, /opt/mqm/samp/bin/amqsput QNAME QMNAME komutunu çalıştırırsanız, PutApplAdı 'amqsput' olur. Bu MQCHAR28 alanında 21 boşluk karakteri doldurma karakteri vardır. /opt/mqm/samp/bin ' un da içinde bulunduğu tam yolun PutApplAdı eklenmediğini unutmayın.

MQPUT ve MQPUT1 çağrılarını için, *PutMsgOpts* parametresinde MQPMO_SET_ALL_CONTEXT belirtilmişse, bu bir giriş/çıkış alanıdır. Alan içindeki boş değerli bir karakteri izleyen herhangi bir bilgi atılır. Boş değerli karakter ve izleyen karakterler kuyruk yöneticisi tarafından boşluklara dönüştürülür. MQPMO_SET_ALL_CONTEXT belirtilmediyse, bu alan girişte yoksayılr ve yalnızca çıkış alanıdır.

PutApplTipi (MQUZE)

Bu, iletiyi koyan uygulamanın tipidir ve iletinin **köken bağlamının** bir parçasıdır. İleti bağlamına ilişkin daha fazla bilgi için bkz. [“MQMD için genel bakış” sayfa 405](#) ve [İleti bağlamı](#).

PutApplType , aşağıdaki standart tiplerden birine sahip olabilir. Ayrıca, yalnızca MQAT_USER_FIRST ile MQAT_USER_SON aralığındaki değerlerle kendi tiplerinizi tanımlayabilirsiniz.

MQAT_AIX

AIX uygulaması (MQAT_UNIX ile aynı değer).

V 8.0.0.4 MQAT_AMQP

AMQP protokol uygulaması

MQAT_BROKER

Aracı.

MQAT_CICS

CICS işlemi.

MQAT_CICS_KöPRÜSÜ

CICS bridge.

MQAT_CICS_VSE

CICS/VSE işlemi.

MQAT_DOS

PC DOS üzerinde IBM MQ MQI client uygulaması.

MQAT_DQM

Dağıtılmış kuyruk yöneticisi aracı.

MQAT_GUARDIAN

Tandem Guardian uygulaması (MQAT_NSK ile aynı değer).

MQAT_IMS

IMS uygulaması.

MQAT_IMS_BRIDGE

IMS köprüsü.

MQAT_JAVA

Java.

MQAT_MVS

MVS ya da TSO uygulaması (MQAT_ZOS ile aynı değer).

MQAT_NOTES_AGENT

Lotus Notes Agent uygulaması.

MQAT_NSK

HP Integrity NonStop Server uygulaması.

MQAT_OS390

OS/390 uygulaması (MQAT_ZOS ile aynı değer).

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 iş yükü yöneticisi uygulaması.

MQAT_XCF

XCF.

MQAT_ZOS

z/OS uygulaması.

MQAT_DEFAULT

Varsayılan uygulama tipi.

Bu, uygulamanın çalışmakta olduğu altyapıya ilişkin varsayılan uygulama tipidir.

Not: Bu değişmez değeri ortamdan özgüdür. Bu yüzden, uygulamayı her zaman, uygulamanın çalışacağı altyapıya uygun olan üstbilgi, include ya da COPY dosyalarını kullanarak derleyin.

MQAT_BILINMIYOR

Bu değer, diğer bağlam bilgileri mevcut olsa da, uygulama tipinin bilinmediğini belirtmek için kullanılır.

MQAT_USER_FIRST

Kullanıcı tanımlı uygulama tipi için en düşük değer.

MQAT_USER_SON

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 bağlamına sahip olup olmadığına karar vermek için *PutApplType* bu değer için test edilebilir (*PutApplType* ' in hiçbir zaman MQPMO_SET_ALL_CONTEXT kullanılarak bir uygulama tarafından MQAT_NO_CONTEXT olarak ayarlanmaması önerilir; diğer bağlam alanlarından biri boş değilse).

Kuyruk yöneticisi bir uygulamanın sonucu olarak bu bilgileri oluşturduğunda, alan, ortam tarafından belirlenen bir değere ayarlanır. On IBM i, it is set to MQAT_OS400; the queue manager never uses MQAT_CICS on IBM i.

MQPUT ve MQPUT1 çağrıları için, *PutMsgOpts* parametresinde MQPMO_SET_ALL_CONTEXT belirtilmişse, bu bir giriş/çıkış alanıdır. MQPMO_SET_ALL_CONTEXT belirtilmediyse, bu alan girişte yoksayılr ve yalnızca çıkış alanıdır.

Bu, MQGET çağrısına ilişkin bir çıkış alanıdır. Bu alanın ilk değeri MQAT_NO_CONTEXT ' dir.

PutDate (MQCHAR8)

Bu, iletinin konulduğu tarihtir ve iletinin **başlangıç bağlamı** ' nin bir parçasıdır. İleti bağlamına ilişkin daha fazla bilgi için bkz. “MQMD için genel bakış” sayfa 405 ve [İleti bağlamı](#).

Bu alanın kuyruk yöneticisi tarafından oluşturulduğu tarih için kullanılan biçim şöyledir:

- YYYYAAGG

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.

İleti bir iş biriminin bir parçası olarak konulduysa, tarih, iş biriminin kesinleştirildiği tarih değil, ileti bulunduğu olduğu tarih olur.

MQPUT ve MQPUT1 çağrılar için, *PutMsgOpts* parametresinde MQPMO_SET_ALL_CONTEXT belirtilmişse, bu bir giriş/çıkış alanıdır. Alan içeriği kuyruk yöneticisi tarafından denetlenmez; ancak, alan içindeki boş değerli bir karakteri izleyen herhangi bir bilgi atılır. Kuyruk yöneticisi boş değerli karakteri ve izleyen herhangi bir karakteri boşluklara dönüştürür. MQPMO_SET_ALL_CONTEXT belirtilmediyse, bu alan girişte yoksayılr ve yalnızca çıkış alanıdır.

Bu, MQGET çağrısına ilişkin bir çıkış alanıdır. Bu alanın uzunluğu MQ_PUT_DATE_LENGTH tarafından verilir. Bu alanın başlangıç değeri, C ' deki boş değerli dizedir ve diğer programlama dillerinde 8 boş karakter olur.

PutTime (MQCHAR8)

Bu, iletinin bulunduğu ve iletinin **başlangıç bağlamının** bir parçası olduğu zamandır. İleti bağlamına ilişkin daha fazla bilgi için bkz. [“MQMD için genel bakış”](#) sayfa 405 ve [İleti bağlamı](#).

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 *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.

İleti, bir iş biriminin bir parçası olarak konulduysa, iş biriminin kesinleştirildiği zaman değil, ileti bulunduğu saat o zaman olur.

MQPUT ve MQPUT1 çağrılar için, *PutMsgOpts* parametresinde MQPMO_SET_ALL_CONTEXT belirtilmişse, bu bir giriş/çıkış alanıdır. Kuyruk yöneticisi alanın içeriğini denetmez; ancak, alan içindeki boş bir karakter izleyen herhangi bir bilgi atılır. Kuyruk yöneticisi boş değerli karakteri ve izleyen herhangi bir karakteri boşluklara dönüştürür. MQPMO_SET_ALL_CONTEXT belirtilmediyse, bu alan girişte yoksayılr ve yalnızca çıkış alanıdır.

Bu, MQGET çağrısına ilişkin bir çıkış alanıdır. Bu alanın uzunluğu MQ_PUT_TIME_LENGTH tarafından verilir. Bu alanın başlangıç değeri, C ' deki boş değerli dizedir ve diğer programlama dillerinde 8 boş karakter olur.

ReplyToQ (MQCHAR48)

Bu, ileti için alma isteğini yayınlayan uygulamanın MQMT_REPLY ve MQMT_REPORT iletilerini göndereceği ileti kuyruğunun adıdır. Ad, *ReplyToQMGr* ile tanımlanan kuyruk yöneticisinde tanımlı olan bir kuyruğun yerel adıdır. Kuyruk yöneticisi, ileti bulunduğu bunu doğrulama 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 iletili 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 iletili gerektiren bir ileti teslim edilemezse ve rapor iletili de belirtilen kuyruğa teslim edilemezse, hem özgün ileti, hem de rapor iletili ölü-mektup (teslim edilemeyen ileti) kuyruğuna gider ("Kuyruk yöneticisine ilişkin öznitelikler" sayfa 778 ' ta açıklanan *DeadLetterQName* özniteliğini göreceldir).

Bu, MQGET çağrısına ilişkin bir çıkış alanıdır ve MQPUT ve MQPUT1 çağrılarına 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 çağrılarına 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.

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 bildirimi (PAN)
- Negatif işlem bildirimi (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 olağan ileti kanalları üzerinden seyahat eden kalıcı olmayan iletiler için, hata koşuluna ilişkin gönderme uygulaması tarafından belirtilen işlem başarıyla tamamlanabiliyorsa, kural dışı durum raporu *yalnızca* oluşturulur. 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 zamanuymulu olarak MQPUT ya da MQPUT1 çağırısı 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 üstbilgisi yapısı 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 üstbilgisi yapısı 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 kesinleştirilirse 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 üstbilgisi yapısı 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 üstbilgisi yapısı 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/OSüzerinde, 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 oluşturmaları gerekir.

MQRO_NAN

Bu rapor tipi, iletii alan ve üzerinde işlem yapan uygulama tarafından oluşturulur. İletide istenen işlemin *değil* işlemi başarıyla gerçekleştirildiğini 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 oluşturmaları 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 değerin 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 raporlarına geçiş yapamazsa, kuyruk yöneticisi yöneticileri tarafından sağlanan bir API geçiş çıkışı kullanılarak raporlar açılabilir.

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 Sürüm 6.0 dışı ya da daha sonraki bir kuyruk yöneticisi tarafından işlenmiş olsa bile, herhangi bir kullanıcı iletime ayarlanmasına olanak sağlar.
5. Bir süreç, bir kuyruk yöneticisi ya da kullanıcı süreci, bu seçenek kümesi olan bir iletime 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 ileti oluşturulmadı.

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 875](#).

Bir rapor ileti istenirse, raporun gönderileceği kuyruğun adı, *ReplyToQ* alanında belirtilmeli. Bir rapor ileti alındığında, raporun niteliği, ileti tanımlayıcısındaki *Feedback* alanı incelenerek saptanabilir.

2. Bir rapor ileti oluşturan kuyruk yöneticisi ya da MCA, rapor iletimini yanıt kuyruğuna koyamıyorsa (örneğin, yanıt kuyruğu ya da iletim kuyruğu dolu olduğu için), rapor ileti ölü harf kuyruğunda yerine yerleştirilir. Aynı zamanda da başarısız olursa ya da herhangi bir ölü harf kuyruğu yoksa, işlem, rapor iletiminin tipine bağlıdır:

- Rapor ileti bir kural dışı durum bildirimiye, kural dışı durum raporunu oluşturan ileti iletim kuyruğunda bırakılır; bu, iletinin kaybolmamasını sağlar.
- Diğer tüm rapor tipleri için, rapor ileti 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 ileti 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, *ReplyToQ* kuyruğu açılır ve aşağıdaki durumlar dışında, rapora neden olan iletinin MQMD ' de *UserIdentifier* yetki yetkisini kullanarak rapor ileti açılır:
 - Alıcı bir MCA tarafından oluşturulan kural dışı durum raporları, rapora neden olan iletiyi yerleştirmeye çalıştığında, 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 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 uygulamaların, yanıt oluşturmak için kullandıkları yetkiyi kullanmaları gerekir; bu genellikle özgün iletime 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 ileti istenirse:

- Rapor ileti her zaman, özgün iletinin göndereni tarafından istenen veri miktarıyla oluşturulur. Rapor ileti yanıt kuyruğu için çok büyükse, yukarıda açıklanan işleme gerçekleşir; rapor ileti 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 *değil* alındıysa, 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 alınmadır.
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 engelle 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, NPMGDLV 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 çağrılarını tarafından *farklı tip* raporlarının döndürülebileceğini göz atın. 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 dönmesine neden olabilir. MQGMO_COMPLE_MSG seçeneğini kullanarak (isteğe bağlı 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 iletilisine geçeceği bağlıdır.

3. Kesimler yerleştiren uygulamalar, her bir bölüm için farklı rapor seçenekleri belirtebilir. Ancak, aşağıdaki noktalara dikkat edin:
 - Kesimler MQGMO_COMPLE_MSG seçeneği kullanılarak alınırsa, 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_AVAIIIIIIIIIIK 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 iletilisi oluşturulursa, kuyruk yöneticisi ya da MCA varsayılan olarak rapor iletilisinde 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 iletilisiyle veri isteyerek bu zorluğu önlemekten kaçının. Ancak, MQRO_ *_WITH_DATA belirtilirse, rapor iletilisi bir kuyruk yöneticisi tarafından oluşturulduysa ya da bölümlere desteğini desteklemeyen MCA tarafından oluşturulduysa, rapor iletilisini alan uygulamaya *küçüktür* 100 bayt uygulama iletilisi verisi döndürülebileceğini unutmayın.

Rapor iletilisi için ileti tanımlayıcısının içeriği: Kuyruk yöneticisi ya da ileti kanalı aracısı (MCA) bir rapor iletilisi oluşturduğunda, ileti tanımlayıcısındaki alanları aşağıdaki değerlere ayarlar ve sonra iletiyi olağan şekilde yerleştirir.

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_UNSNISIZ
<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
<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

MQMD ' de alan	Kullanılan değer
<i>UserIdentifier</i>	MQPMO_PASS_IDENTITY_CONTEXT seçeneği tarafından belirlendiği gibi
<i>AccountingToken</i>	MQPMO_PASS_IDENTITY_CONTEXT seçeneği tarafından belirlendiği gibi
<i>ApplIdentityData</i>	MQPMO_PASS_IDENTITY_CONTEXT seçeneği tarafından belirlendiği gibi
<i>PutApplType</i>	MQAT_QMGR ya da ileti kanalı aracısının uygun olduğu şekilde
<i>PutApplName</i>	Kuyruk yöneticisi adı 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>ApplOriginData</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 bulunduğu 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 877' 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.

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.

UserIdentifier (MQCHAR12)

Bu, iletinin **kimlik bağlamının** bir parçasıdır. İleti bağlamına ilişkin daha fazla bilgi için bkz. “MQMD için genel bakış” sayfa 405 ve [İleti bağlamı](#).

UserIdentifier , iletiyi oluşturan uygulamanın kullanıcı tanımlayıcını belirtir. Kuyruk yöneticisi bu bilgileri karakter verisi olarak değerlendirir, ancak biçiminin biçimini tanımlamaz.

Bir ileti alındıktan sonra, sonraki bir MQPLAIN ya da MQPUT1 çağrısının *ObjDesc* parametresinin *AlternateUserId* alanında, açık olan uygulama yerine *UserIdentifier* kullanıcısı için yetki denetimini gerçekleştirmek için *UserIdentifier* değıştirgesini kullanın.

Kuyruk yöneticisi, bir MQPUT ya da MQPUT1 çağrısı için bu bilgileri oluşturduğunda:

- On z/OS, the queue manager uses the *AlternateUserId* from the *ObjDesc* parameter of the MQOPEN or MQPUT1 call if the MQOO_ALTERNATE_USER_AUTHORITY or MQPMO_ALTERNATE_USER_AUTHORITY option was specified. İlgili seçenek belirtilmediyse, kuyruk yöneticisi ortamdan saptanan bir kullanıcı kimliğini kullanır.
- Diğer ortamlarda, kuyruk yöneticisi her zaman ortamdan belirlenen bir kullanıcı kimliğini kullanır.

Kullanıcı kimliği ortamdan saptanırken:

- On z/OS, the queue manager uses:
 - MVS (toplu iş) için, JES JOB kartından kullanıcı kimliği ya da başlatılan görev
 - TSO için, kullanıcı kimliği, iş gönderimi sırasında işe geçirildi.
 - 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 *yayınlanmamış* 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:
 - *sahip* ' in başarılı bir GU çağrısı yaptığı ileti BMP ve ileti IFP bölgeleri
 - MPP bölgeleri
- Kuyruk yöneticisi aşağıdakilerden birini kullanır:
- İletiyile ilişkilendirilen oturum açmış kullanıcı kimliği
 - Mantıksal uçbirim (LTERM) adı
 - Bölge JES JOB kartından kullanıcı kimliği
 - TSO kullanıcı kimliği
 - PSB adı

- On IBM i, the queue manager uses the name of the user profile associated with the application job.
- On UNIX sistemleri, the queue manager uses:
 - Uygulamanın oturum açma adı
 - Oturum açma kullanılamıyorsa, işlemin etkin kullanıcı kimliği
 - Uygulama bir CICS hareketiyse, işlemle ilişkili kullanıcı kimliği
- Windows sistemlerinde kuyruk yöneticisi, oturum açmış kullanıcı adının ilk 12 karakterini kullanır.

Bu alan, genellikle kuyruk yöneticisi tarafından üretilen bir çıkış alanıdır, ancak bir MQPUT ya da MQPUT1 çağrısı için bu alanı bir giriş/çıkış alanı yapabilir ve kuyruk yöneticisinin bu bilgileri oluşturmasına izin vermek yerine UserIdentification alanını belirleyebilirsiniz. Kuyruk yöneticisinin bir MQPUT ya da MQPUT1 çağrısı için UserIdentifier alanını oluşturmasını istemiyorsanız, PutMsgOpts parametresinde MQPMO_SET_IDENTITY_CONTEXT ya da MQPMO_SET_ALL_CONTEXT değerini belirtin ve UserIdentifier alanında bir kullanıcı kimliği belirtin.

MQPUT ve MQPUT1 çağrıları için, *PutMsgOpts* parametresinde MQPMO_SET_IDENTITY_CONTEXT ya da MQPMO_SET_ALL_CONTEXT belirtilmişse, bu giriş/çıkış alanıdır. Alan içindeki boş değeri bir karakteri izleyen herhangi bir bilgi atılır. Kuyruk yöneticisi boş değeri karakteri ve izleyen herhangi bir

karakteri boşluklara dönüştürür. MQPMO_SET_IDENTITY_CONTEXT ya da MQPMO_SET_ALL_CONTEXT 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 *UserIdentifier* ögesini içerir. Bu, alıkonursa iletiyle alıkonan *UserIdentifier* değerinin (alıkonan yayınlarla ilgili daha fazla ayrıntı için bkz. MQPMO_RETAIN ile ilgili açıklamalara bakın) değeri olur, ancak bu ileti, gönderilen tüm yayınlarda *UserIdentifier* 'i geçersiz kılmak için bir değer sağladıkları için, bu ileti abonelere bir yayın olarak gönderildiğinde *UserIdentifier* olarak kullanılmaz. İletinin bağlamı yoksa, alan tümüyle boştur.

Bu, MQGET çağrısına ilişkin bir çıkış 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.

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.

MQMD için ilk değerler ve dil bildirimleri

Çizelge 55. MQMD için MQMD ' de alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
<i>StrucId</i>	MQMD_STRUC_ID	' MD '
<i>Version</i>	MQMD_VERSION_1	1
<i>Report</i>	MQRO_NONE	0
<i>MsgType</i>	MQMT_DATAGRAM	8
<i>Expiry</i>	MQE_UNSiNSiZ	-1
<i>Feedback</i>	MQFB_YOK	0
<i>Encoding</i>	MQENC_NATIVE	Ortama göre değişir
<i>CodedCharSetId</i>	MQCCSI_Q_MGR	0
<i>Format</i>	MQFMT_NONE	Boşluklar
<i>Priority</i>	MQPRI_PRIORITY_AS_Q_DEF	-1
<i>Persistence</i>	MQPER_PERSISTENCE_AS_Q_DEF	2
<i>MsgId</i>	MQMI_NONE	Boş Değerler

Çizelge 55. MQMD için MQMD ' de alanların ilk değerleri (devamı var)

Alan adı	Değişmezin adı	Değişmezin değeri
<i>CorrelId</i>	MQCI_NONE	Boş Değerler
<i>BackoutCount</i>	Yok	0
<i>ReplyToQ</i>	Yok	Dizgi ya da boşluk boş değerli
<i>ReplyToQMgr</i>	Yok	Dizgi ya da boşluk boş değerli
<i>UserIdentifier</i>	Yok	Dizgi ya da boşluk boş değerli
<i>AccountingToken</i>	MQACT_NONE	Boş Değerler
<i>ApplIdentityData</i>	Yok	Dizgi ya da boşluk boş değerli
<i>PutApplType</i>	MQAT_NO_CONTEXT	0
<i>PutApplName</i>	Yok	Dizgi ya da boşluk boş değerli
<i>PutDate</i>	Yok	Dizgi ya da boşluk boş değerli
<i>PutTime</i>	Yok	Dizgi ya da boşluk boş değerli
<i>ApplOriginData</i>	Yok	Dizgi ya da boşluk boş değerli
<i>GroupId</i>	MQGI_NONE	Boş Değerler
<i>MsgSeqNumber</i>	Yok	1
<i>Offset</i>	Yok	0
<i>MsgFlags</i>	MQMF_NONE	0
<i>OriginalLength</i>	MQOL_TANIMSIZ	-1

Notlar:

1. Boş değerli dizgi ya da boşluklar, C içindeki boş değerli dizeyi ve diğer programlama dillerindeki boş karakterleri belirtir.
2. C programlama dilinde, makro değişkeniMQMD_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlara ilk değerleri sağlamak için aşağıdaki şekilde kullanılabilir:

```
MQMD MyMD = {MQMD_DEFAULT};
```

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    PutApplType; /* Type of application that put the
                        message */
MQCHAR28  PutApplName; /* 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 */
};

```

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-ORIGINALLLENGTH PIC S9(9) BINARY.

```

PL/I bildirim

```

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 */

```

High Level Assembler bildirim

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)	

Visual Basic bildirimi

```

Type MQMD
    StructId      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

```

MQMDE-İleti tanımlayıcı uzantısı

Aşağıdaki çizelge yapıdaki alanları özetlemektedir.

Çizelge 56. MQMDE içindeki alanlar		
Alan	Tanım	Konu
<i>StructId</i>	Yapı tanıttıcısı	StructId
<i>Version</i>	Yapı sürüm numarası	S\u00f0fcr\u00f0cm
<i>StructLength</i>	MQMDE yapısının uzunluğu	StructLength
<i>Encoding</i>	MQMDE ' yi izleyen verilerin sayısal kodlaması	Kodlama
<i>CodedCharSetId</i>	MQMDE ' yi izleyen verilerin karakter kümesi tanıttıcısı	CodedCharSetId
<i>Format</i>	MQMDE ' yi izleyen verilerin adı	Biçim
<i>Flags</i>	Genel işaretler	İşaretler

Çizelge 56. MQMDE içindeki alanlar (devamı var)		
Alan	Tanım	Konu
<i>GroupId</i>	Grup tanıtıcısı	<u>GroupId</u>
<i>MsgSeqNumber</i>	Grup içindeki mantıksal iletinin sıra numarası	<u>MsgSeqNumarası</u>
<i>Offset</i>	Mantıksal iletinin başlangıcındaki fiziksel iletilerde verilerin görelî konumu	<u>Görelî Konum</u>
<i>MsgFlags</i>	İleti İşaretleri	<u>MsgFlags</u>
<i>OriginalLength</i>	Özgün iletinin uzunluğu	<u>OriginalLength</u>

MQMDE ile ilgili genel bakış

Kullanılabilirlik: Tüm IBM MQ sistemleri, ayrıca bu sistemlere bağı IBM MQ istemcileri.

Amaç: MQMDE yapısı, bazen uygulama iletisi verilerinden önce ortaya çıkan verileri açıklar. Yapı, version-2 MQMD 'de var olan ancak version-1 MQMD' de bulunmayan MQMD alanlarını içerir.

Biçim adı: MQFMT_MD_EXTENSION.

Karakter kümesi ve kodlama: MQMDE içindeki veriler, yerel kuyruk yöneticisinin karakter kümesinde ve kodlamasında olmalıdır; bunlar, C programlama dili için *CodedCharSetId* kuyruk yöneticisi özneliği ve MQENC_NATIVE tarafından verilir.

MQMDE ' nin karakter kümesini ve kodlamasını *CodedCharSetId* ve *Encoding* alanlarına aşağıdaki şekilde kodlayın:

- MQMD (MQMDE yapısı ileti verilerinin 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 ileti verileri olarak kabul edilir.

Not: Windowsüzerinde, Micro Focus COBOL ile derlenen uygulamalar, kuyruk yöneticisinin kodlamasından farklı olan MQENC_NATIVE değerini kullanır. MQPUT, MQPUT1ve MQGET çağrılarında MQMD yapısındaki sayısal alanların Micro Focus COBOL kodlamasında olması gerekir; ancak, MQMDE yapısındaki sayısal alanların kuyruk yöneticisi kodlamasında olması gerekir. Bu ikincisi, C programlama dili için MQENC_NATIVE tarafından verilir ve değeri 546 'sına sahiptir.

Kullanım: version-2 MQMD kullanan uygulamalar bir MQMDE yapısıyla karşılaşmayacaktır. 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: MQPUT ve MQPUT1 çağrılarında, uygulama bir version-1 MQMD sağlıyorsa, uygulama isteğe bağı olarak bir MQMDE ile ileti verilerini önleyebilir, MQMD 'de *Format* alanını MQMD' ye ayarlayarak MQMDE 'yi (MQMDE) ayarlayarak, MQMD' ye ayarlayabilir. Uygulama bir MQMDE sağlamazsa, kuyruk yöneticisi MQMDE içindeki alanlar için varsayılan değerleri devralır. Kuyruk yöneticisinin kullandığı varsayılan değerler, yapıya ilişkin başlangıç değerleriyle aynıdır; bkz. Çizelge 58 sayfa 457.

Uygulama bir version-2 MQMD ve örnekleri sağlıyorsa, uygulama ileti verilerini bir MQMDE ile ekler; yapılar, Çizelge 57 sayfa 455içinde gösterildiği gibi işlenir.

Çizelge 57. MQPUT üzerinde MQMDE ya da MQMDE için MQPUT1 için MQMDE belirlendiğinde 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
Not: On z/OS, if the application specifies a version-1 MQMD with an MQMDE, the queue manager validates the MQMDE only if the queue has an <i>IndexType</i> of MQIT_GROUP_ID.			

Özel bir vaka var. Uygulama bir kesimin (yani, MQMF_SEGMENT ya da MQMF_LAST_SEGMENT işareti ayarlandıysa) bir iletiyi koymak için bir version-2 MQMD kullanıyorsa ve MQMD ' deki biçim adı MQFMT_DEAD_LETTER_HEADER 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 *değil* , MQPUT ve MQPUT1 çağrılarının çıktısındaki MQMD ' deki eşdeğer alanlara herhangi bir değer döndürür; uygulama bu çıkış değerlerini gerektiriyorsa, bu değer bir version-2 MQMD kullanmalıdır.

MQGET çağrısı tarafından döndürülen MQMDE çağrısı: 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 içindeki *Format* alanını MQFM_md_extension değerine ayarlar ve bir MQMDE' nin var olduğunu gösterir.

Uygulama, *Buffer* parametresinin başlangıcındaki bir MQMDE sağlıyorsa, MQMDE yoksayılır. MQGET çağrısından dönüşte, ileti için MQMDE ile değiştirilir (gerekliyorsa) ya da üzerine uygulama iletileri verileri (MQMDE gerekmiyorsa) üzerine yazılır.

MQGET çağrısı bir MQMDE döndürürse, MQMDE ' deki veriler genellikle kuyruk yöneticisinin karakter kümesi ve kodlamasıdır. Ancak, MQMDE aşağıdaki durumlarda başka bir karakter kümesinde ve kodlamada da olabilir:

- MQMDE, MQPUT ya da MQPUT1 çağrısında veri olarak değerlendirildi (bu duruma neden olacak durumlar için Çizelge 57 sayfa 455 konusuna bakın).
- İleti, bir TCP bağlantısıyla bağlantılı uzak bir kuyruk yöneticisinden alındı ve alıcı ileti kanalı aracısı (MCA) doğru ayarlanmadı.

Not: Windows üzerinde, Micro Focus COBOL ile derlenen uygulamalar, kuyruktan yöneticinin kodlamasından farklı (yukarıya bakın) MQENC_NATIVE değerini kullanır.

İletim kuyruklarına ilişkin iletilerde MQMDE: Messages on transmission queues are prefixed with the MQXQH structure, which contains within it a version-1 MQMD. MQXQH yapısı ve uygulama iletileri verileri arasında konumlandırılmış bir MQMDE da var olabilir, ancak genellikle MQMDE içindeki alanlardan biri ya da daha fazlasının varsayılan dışı bir değeri varsa vardır.

Diğer MQ üstbilgi yapıları, MQXQH yapısı ile uygulama ileti verileri arasında da oluşabilir. Örneğin, MQDLH adlı ölü-mektup üstbilgisi varsa ve ileti bir bölüm değilse, sıralama düzeni şöyledir:

- MQXQH (bir version-1 MQMD içeren)
- MQMDE
- MQDLH
- uygulama ileti verileri

MQMDE alanları

MQMDE yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

CodedCharSetId (MQHOMER)

Bu, MQMDE yapısını izleyen verilerin karakter kümesi tanıttıcısını 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:

MQCCSI_INHERIT

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 tanıttıcısına 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 şu ortamlarda desteklenir: AIX, HP-UX, IBM i, Solaris, Linux, Windows, ayrıca bu sistemlere bağlı IBM MQ istemcileri.

Bu alanın ilk değeri MQCCSI_UNDEFED değerine sahip.

Kodlama (MQUZE)

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-İleti tanımlayıcısı” sayfa 404](#) içinde açıklanan *Encoding* alanına bakın.

Bu alanın ilk değeri MQENC_NATIVE 'dir.

İşaretler (MQUZE)

Aşağıdaki işaret belirlenebilir:

MQMDEF_NONE

Bayrak yok.

Bu alanın ilk değeri MQMDEF_NONE olur.

Biçim (MQCHAR8)

Bu, MQMDE yapısını izleyen verilerin biçim adını belirtir.

MQPUT 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 tanımlayıcısı” sayfa 404](#) içinde açıklanan *Format* alanına bakın.

Bu alanın ilk değeri MQFMT_NONE olur.

GroupId (MQBYTE24)

“MQMD-İleti tanımlayıcısı” sayfa 404’inde açıklanan *GroupId* alanına bakın. Bu alanın ilk değeri MQGI_NONE olur.

MsgFlags (MQUZE)

“MQMD-İleti tanımlayıcısı” sayfa 404’inde açıklanan *MsgFlags* alanına bakın. Bu alanın ilk değeri MQMF_NONE olur.

MsgSeqNumarası (MQUBE)

“MQMD-İleti tanımlayıcısı” sayfa 404’inde açıklanan *MsgSeqNumber* alanına bakın. Bu alanın ilk değeri 1 'dir.

Görelî Konum (MQLONG)

“MQMD-İleti tanımlayıcısı” sayfa 404’inde açıklanan *Offset* alanına bakın. Bu alanın ilk değeri 0 'tır.

OriginalLength (MQlong)

“MQMD-İleti tanımlayıcısı” sayfa 404’inde açıklanan *OriginalLength* alanına bakın. Bu alanın ilk değeri MQOL_UNDEFINED olur.

StrucId (MQCHAR4)

Değer şu olmalıdır:

MQMDE_STRUC_ID

İleti tanımlayıcı uzantısı yapısına ilişkin tanıttıcı.

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.

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.

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.

MQMDE için ilk değerler ve dil bildirimleri

Çizelge 58. MQMDE için MQMDE 'deki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
<i>StrucId</i>	MQMDE_STRUC_ID	'MDE↵'
<i>Version</i>	MQMDE_VERSION_2	2
<i>StrucLength</i>	MQMDE_LENGTH_2	72

Çizelge 58. MQMDE için MQMDE 'deki alanların ilk değerleri (devamı var)

Alan adı	Değişmezin adı	Değişmezin değeri
Encoding	MQENC_NATIVE	Ortama göre değişir
CodedCharSetId	MQCCSI_UNDEDINED	0
Format	MQFMT_NONE	Boşluklar
Flags	MQMDEF_NONE	0
GroupId	MQGI_NONE	Boş Değerler
MsgSeqNumber	Yok	1
Offset	Yok	0
MsgFlags	MQMF_NONE	0
OriginalLength	MQOL_TANIMSIZ	-1

Notlar:

1. ↪ simgesi tek bir boş karakteri temsil eder.
2. C programlama dilinde, makro değişkeniMQMDE_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlara ilk değerleri sağlamak için aşağıdaki şekilde kullanılabilir:

```
MQMDE MyMDE = {MQMDE_DEFAULT};
```

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 */
};
```

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-STRUCLNGTH   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.

```

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 */

```

High Level Assembler bildirimi

MQMDE	DSECT	
MQMDE_STRUCID	DS	CL4 Structure identifier
MQMDE_VERSION	DS	F Structure version number
MQMDE_STRUCLNGTH	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 within group
*		
MQMDE_OFFSET	DS	F 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)

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 MQMDE'
CodedCharSetId	As Long 'Character-set identifier of data that follows MQMDE'
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	'group' '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		

MQMHBO-İleti tanıtıcısı-arabellek seçenekleri

Aşağıdaki çizelge yapıdaki alanları özetlemektedir. MQMHBO structure-message handle to buffer options

Çizelge 59. MQMHBO 'daki alanlar		
Alan	Tanım	Konu
<i>StrucId</i>	Yapı tanıtıcısı	StrucId
<i>Version</i>	Yapı sürüm numarası	S\u00fcr\u00fcm
<i>Options</i>	MQMHBUF işleminin denetlenmesine ilişkin seçenekler	Seçenekler

MQMHBO 'ya Genel Bakış

Kullanılabilirlik: Tüm IBM MQ sistemleri ve IBM MQ MQI clients.

Amaç: MQMHBO yapısı, uygulamaların, ileti tanıtıcılarından arabelleklerin nasıl üretildiğini denetleyen seçenekleri belirlemesine olanak sağlar. Yapı, MQMHBUF çağrısına ilişkin bir giriş değiştirmedir.

Karakter kümesi ve kodlama: MQMHBO içindeki veriler, uygulamanın uygulama ve kodlamasının karakter kümesinde olmalıdır (MQENC_NATIVE).

MQMHBO ile ilgili alanlar

İleti tanıtıcısı-arabellek seçenekleri yapısı-alanlar

MQMHBO yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

Seçenekler (MQUZE)

İleti tanıtıcısı-arabellek seçenekleri yapısı-Seçenekler alanı

Bu seçenekler MQMHBUF ' un işlemini denetler.

Aşağıdaki seçeneği belirlemelisiniz:

MQMHBO_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).

MQMHBO_DELETE_PROPERTIES

Arabelleğe eklenen özellikler ileti tanıtıcısından silinir. Arama başarısız olursa özellikler silinmez.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQMHBO_PROPERTIES_IN_MQRFH2' dir.

StrucId (MQCHAR4)

İleti tanıtıcısı-arabellek seçenekleri yapısı- StrucId alanı

Bu, yapı tanıtıcısıdır. Değer şu olmalıdır:

MQMHBO_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:

MQMHBO_VERSION_1

Arabellek seçenekleri yapısıyla ileti tanıtıcısı için sürüm numarası.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

MQMHBO_CURRENT_VERSION

İleti tutamacının arabellek seçenekleri yapısına ilişkin yürürlükteki sürümü.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQMHBO_VERSION_1' dir.

MQMHBO için ilk değerler ve dil bildirimleri

İleti tanıtıcısı-arabellek yapısı-İlk değerler

Çizelge 60. MQMHBO 'daki alanların ilk değerleri

Alan adı	Değişmezin adı	Değişmezin değeri
StrucId	MQMHBO_STRUCT_ID	' MHBO '
Version	MQMHBO_VERSION_1	1
Options	MQMHBO_PROPERTIES_IN_MQRFH2	

Notlar:

- Boş değerli dizgi ya da boşluklar, C içindeki boş değerli dizeyi ve diğer programlama dillerindeki boş karakterleri belirtir.
- C programlama dilinde, makro değişkeni MQMHBO_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlara ilişkin ilk değerleri sağlamak için bunu aşağıdaki şekilde kullanın:

```
MQMHBO MyMHBO = {MQMHBO_DEFAULT};
```

C bildirimi

İleti tanıtıcısı-arabellek seçenekleri yapısı-C dil 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 */
};
```

COBOL bildirimi

İleti tanıtıcısı-arabellek seçenekleri yapısı-COBOL dil 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.
```

PL/I bildirimi

İleti tanıtıcısı-arabellek seçenekleri yapısı-PL/I dil 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 */

```

High Level Assembler bildirimi

İleti tanıtıcısı-arabellek seçenekleri yapısı-Assembler dil bildirimi

```

MQMHBO          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(MQMHBO_LENGTH)

```

MQOD-Nesne tanımlayıcısı

Aşağıdaki çizelge yapıdaki alanları özetlemektedir.

Alan	Tanım	Konu
<i>StrucId</i>	Yapı tanıtıcısı	StrucId
<i>Version</i>	Yapı sürüm numarası	S\u00fcr\u00fcm
<i>ObjectType</i>	Nesne tipi	ObjectType
<i>ObjectName</i>	Nesne adı	ObjectName
<i>ObjectQMgrName</i>	Nesne kuyruğu yöneticisi adı	ObjectQMgrAdı
<i>DynamicQName</i>	Dinamik kuyruk adı	DynamicQName
<i>AlternateUserId</i>	Diğer kullanıcı kimliği	AlternateUserTanıtıcısı
Not: <i>Version</i> , MQOD_VERSION_2değerinden küçükse, geri kalan alanlar yoksayılır.		
<i>RecsPresent</i>	Var olan nesne kayıtlarının sayısı	RecsPresent
<i>KnownDestCount</i>	Başarıyla açılan yerel kuyrukların sayısı	KnownDestSayı
<i>UnknownDestCount</i>	Başarıyla açılan uzak kuyrukların sayısı	UnknownDestSayısı
<i>InvalidDestCount</i>	Açılamayan kuyrukların sayısı	InvalidDestSayısı
<i>ObjectRecOffset</i>	MQOD başlangıcından ilk nesne kaydının konumu	ObjectRecGörelî Konumu
<i>ResponseRecOffset</i>	MQOD başlangıcından ilk yanıt kaydının kayma konumu	ResponseRecKayması
<i>ObjectRecPtr</i>	İlk nesne kaydının adresi	ObjectRecPtr 'si
<i>ResponseRecPtr</i>	İlk yanıt kaydının adresi	ResponseRecPtr 'si
Not: <i>Version</i> , MQOD_VERSION_3değerinden küçükse, geri kalan alanlar yoksayılır.		
<i>AlternateSecurityId</i>	Diğer güvenlik tanıtıcısı	AlternateSecurityTanıtıcısı
<i>ResolvedQName</i>	Çözülmüş kuyruk adı	ResolvedQName
<i>ResolvedQMgrName</i>	Çözülmüş kuyruk yöneticisi adı	ResolvedQMgrAdı

Alan	Tanım	Konu
Not: <i>Version</i> , MQOD_VERSION_4değerinden küçükse, geri kalan alanlar yoksayılr.		
<i>ObjectString</i>	Uzun nesne adı	ObjectString
<i>SelectionString</i>	Seçim dizgisi	SelectionString
<i>ResObjectString</i>	Çözülmüş uzun nesne adı	ResObjectDizgisi
<i>ResolvedType</i>	Çözülmüş nesne tipi	ResolvedType

MQOD için genel bakış

Uygunluk: Tüm IBM MQ sistemleri, artı IBM MQ MQI clients bu sistemlere bağlı.

Amaç: MQOD yapısı, ada göre bir nesneyi belirtmek için kullanılır. Aşağıdaki nesne tipleri geçerlidir:

- Kuyruk ya da dağıtım listesi
- Ad Listesi
- Süreç tanımlaması
- Kuyruk yöneticisi
- Konu

Yapı, MQOPER ve MQPUT1 çağrılarında bir giriş/çıkış değiştirir.

Sürüm: MQOD 'nin yürürlükteki sürümü MQOD_VERSION_4' dir. Birkaç ortam arasında bağlantı noktası yapmak istediğiniz uygulamalar, ilgili MQOD sürümünün ilgili tüm ortamlarda desteklenmesini sağlamalıdır. Yalnızca yapının daha yeni sürümlerinde var olan alanlar, izleyen açıklamalarda olduğu gibi tanımlanır.

Desteklenen programlama dilleri için sağlanan üstbilgi, COPY ve INCLUDE dosyaları, ortam tarafından desteklenen en son MQOD sürümünü içerir; ancak, *Version* alanının ilk değeri MQOD_VERSION_1olarak ayarlanmış olmalıdır. version-1 yapısında mevcut olmayan alanları kullanmak için, uygulamanın *Version* alanını, gereken sürümün sürüm numarasına ayarlaması gerekir.

Bir dağıtım listesi açmak için *Version* , MQOD_VERSION_2 ya da daha büyük bir değer olmalıdır.

Karakter kümesi ve kodlama: MQOD içindeki veriler, *CodedCharSetId* kuyruk yöneticisi özneliği tarafından verilen karakter kümesinde ve MQENC_NATIVE tarafından verilen yerel kuyruk yöneticisinin kodlamasında olmalıdır. Ancak, uygulama bir MQ MQI istemcisi olarak çalışıyorsa, yapı istemcisinin karakter kümesinde ve kodlamasında olmalıdır.

MQOD alanları

MQOD yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla**anlatılır:

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_3 değerinden küçükse bu alan yok sayılır.

AlternateUserTanıtıcısı (MQCHAR12)

MQOPEN çağrısı 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 yoksayılır.

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.

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 dizisiyle 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/OS üzerinde değer 'CSQ.*' dir.
- Diğer platformlarda değer 'AMQ.*' dir.

Değer, C ' de boş karakterle biten bir dizedir ve diğer programlama dillerinde boşlukla doldurulmuş bir dizedir.

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* değiştirgesi MQCC_OK ya da MQCC_WARNING; ise, *CompCode* değiştirgesi MQCC_FAILED ise *ayarlanmaz* .

Bu bir çıkış alanıdır. Bu alanın ilk değeri 0 'tır. *Version* , MQOD_VERSION_1değerinden 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.

ObjectName (MQCHAR48)

Bu, *ObjectQMgrName* ile tanımlanan kuyruk yöneticinde 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 (/), alt çizgi (_), yüzde (%)

Ad, baştaki ya da gömülü boşluklar içermemeli, ancak sondaki boşlukları içermeyebilir. Addaki önemli verilerin sonunu belirtmek için boş değerli bir karakter kullanın; boş değer ve izleyen karakterler boşluk olarak işlenir. Belirtilen ortamlarda aşağıdaki kısıtlamalar geçerlidir:

- EBCDIC Katakana kullanan sistemlerde küçük harfli karakterler kullanılamaz.
- Açıkz/OS:
 - Bir alt çizgiyle başlayan ya da biten adlardan kaçının; işlemler ve denetim panoları tarafından işlenemez.

- The percent character has a special meaning to RACF. Dış güvenlik yöneticisi olarak RACF kullanılırsa, adlar yüzde değerini içermemelidir. Bu adlar kullanılırsa, RACF soysal profilleri kullanıldığında bu adlar hiçbir güvenlik denetiminde yer almaz.
- IBM üzerinde, komutlar üzerinde belirtildiğinde, küçük harf, eğik çizgi ya da yüzde içeren adların tırnak işareti içine alınması gerekir. Bu tırnak işaretleri, yapılarda alan olarak ya da çağrılarda parametre olarak ortaya çıkan adlar için belirtilmemelidir.

Tam konu adı iki farklı alandan oluşturulabiliyor: *ObjectName* ve *ObjectString*. Bu iki alanın nasıl kullanılma hakkında ayrıntılı bilgi için bkz. [“Konu dizelerinin kullanılması” sayfa 560](#).

Aşağıdaki noktalar, gösterilen nesne tipleri için geçerlidir:

- *ObjectName* bir model kuyruğunun adıysa, kuyruk yöneticisi, model kuyruğunun öznitelikleriyle dinamik bir kuyruk yaratır ve *ObjectName* alanında, yaratılan kuyruğun adını döndürür. Model kuyruğu yalnızca MQOPEN çağrısında belirtilebilir; MQPUT1 çağrısında bir model kuyruğu geçerli değildir.
- *ObjectName*, TARGTYPE (KONU) olan bir diğer ad kuyruğunun adıysa, adı belirtilen diğer ad kuyruğunda ilk kez güvenlik denetimi yapılır; diğer ad kuyrukları kullanıldığında bu olağandır. Güvenlik denetimi başarıyla tamamlandığında, MQOPEN çağrısı devam eder ve MQOT_TOPIC; üzerinde bir MQOPEN çağrısı gibi davranacaktır; bu durum, denetim konusu nesnesine ilişkin bir güvenlik denetiminin gerçekleştirilmesini içerir.
- *ObjectName* ve *ObjectQMGrName*, kuyruk paylaşım grubunun iyeliğindeki bir paylaşılan kuyruğu yerel kuyruk yöneticisinin ait olduğu bir kuyruğu tanıyorsa, yerel kuyruk yöneticisinde aynı adı içeren bir kuyruk tanımlaması da olmamalıdır. Böyle bir tanımlama (yerel bir kuyruk, diğer ad kuyruğu, uzak kuyruk ya da model kuyruğu) varsa, çağrı neden kodu MQRC_OBJECT_NOT_UNIQUE neden ile başarısız olur.
- Açılmakta olan nesne bir dağıtım listese (yani, *RecsPresent* mevcut ve sıfırdan büyük), *ObjectName* boş olmalı ya da boş dizgi olmalıdır. Bu koşul karşılanmazsa, çağrı neden kodu MQRC_OBJECT_NAME_ERROR neden ile başarısız olur.
- *ObjectType* MQOT_Q_MGR ise, özel kurallar geçerlidir; bu durumda, adın ilk boş karakteri ya da alanın sonuna kadar tümüyle boş olması gerekir.

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_NAME_LENGTH değeri tarafından verilir. Bu alanın başlangıç değeri, C ' de boş değer dizgisidir ve diğer programlama dillerindeki 48 boş karakter dizilimidir.

ObjectQMGrAdı (MQCHAR48)

Bu ad, *ObjectName* nesnesinin tanımlandığı kuyruk yöneticisinin adıdır. Adda geçerli olan karakterler, *ObjectName* ile aynıdır (bkz. [“ObjectName \(MQCHAR48\)” sayfa 465](#)). 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 grubuna ait, *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/OS üzerinde desteklenir.

- *ObjectName* , bir uzak kuyruk paylaşım grubunun (yerel kuyruk yöneticisinin *değil* ait olduğu bir kuyruk paylaşım grubu) sahip olduğu bir paylaşılan kuyruğun adı ise, *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 listesi (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.

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* sıfırsa, alan yoksayılır.

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 sıfırsa 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.

ObjectRecPtr (MQPTR)

Bu, ilk MQOR nesne kaydının adresidir. *ObjectRecPtr* , yalnızca bir dağıtım listesi açılırken kullanılır. *RecsPresent* sıfırsa, alan yoksayılır.

Nesne kayıtlarını belirtmek için *ObjectRecPtr* ya da *ObjectRecOffset* 'yi kullanabilirsiniz, ancak her ikisi de değil; *ObjectRecOffset* alanının açıklaması için [“ObjectRecGörelili Konumu \(MQUZE\)” sayfa 467](#) 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_2 değerinden 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.

ObjectString (MQCHARV)

ObjectString alanı, uzun nesne adını belirtir.

Bu, kullanılacak uzun nesne adını belirtir. Bu alan yalnızca *ObjectType* 'un belirli değerleri için başvurur ve diğer tüm değerler için yoksayılır. Hangi değerlerin bu alanın kullanıldığını gösterdiğine ilişkin ayrıntılar için *ObjectType* ' in açıklamasına bakın.

ObjectString yanlış belirtildiyse, MQCHARV yapısının nasıl kullanılacağı ya da uzunluk üst sınırını aştıysa, çağrı neden kodu MQRC_OBJECT_STRING_ERROR ile başarısız olur.

Bu bir giriş alanıdır. Bu yapıdaki alanların ilk değerleri, MQCHARV yapısındaki alanlarla aynıdır.

Tam konu adı iki farklı alandan oluşturulabiliyor: *ObjectName* ve *ObjectString*. Bu iki alanın nasıl kullanılma hakkında ayrıntılı bilgi için bkz. [“Konu dizelerinin kullanılması” sayfa 560](#).

ObjectType (MQUZE)

Nesne tanımlayıcısında adlandırılmakta olan nesnenin tipi. Olası değerler şunlardır:

MQOT_CLNTCONN_CHANNEL

İstemci bağlantı kanalı. Nesnenin adı, *ObjectName* alanında bulunur.

MQOT_Q

Kuyruk. Nesnenin adı, *ObjectName* alanında bulunur.

MQOT_NAMELIST

İsim listesi. 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_KONUSU

Konu. Tam konu adı iki farklı alandan oluşturulabiliyor: *ObjectName* ve *ObjectString*.

Bu iki alanın nasıl kullanılma hakkında ayrıntılı bilgi için bkz. [“Konu dizelerinin kullanılması” sayfa 560](#).

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQOT_Q ' dur.

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/OS üzerinde bu alan sıfır olmalıdır.

Bu bir giriş alanıdır. Bu alanın ilk değeri 0 'tır. *Version* , MQOD_VERSION_2 değerinden küçükse bu alan yok sayılır.

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 aşılıyorsa, çağrı neden kodu *MQRC_RES_OBJECT_STRING_ERROR* ile başarısız olur.

ResolvedQMgrAdı (MQCHAR48)

Bu ad, yerel kuyruk yöneticisinin adı çözdükten sonra hedef kuyruk yöneticisinin adıdır. Döndürülen ad, *ResolvedQName* ile tanıtilan 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_3* değerinden 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_3* değerinden küçükse bu alan yok sayılır.

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.

ResponseRecGörelİ Konumu (MQUZE)

Bu, MQOD yapısının başlangıcındaki ilk MQRR yanıt kaydının bayt cinsinden görelİ konutdur. Görelİ konum pozitif ya da negatif olabilir. *ResponseRecOffset* , yalnızca bir dağıtım listesi açılırken kullanılır. *RecsPresent* sıfırda, alan yoksaılır.

Bir dağıtım listesi açılırken, açılmadığını (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 görelİ konum belirtilerek ya da *ResponseRecPtr* ' ta bir adres belirterek sağlanabilir. Bunu nasıl yapacağına ilişkin ayrıntılar için bkz. "[ObjectRecGörelİ Konumu \(MQUZE\)](#)" sayfa 467. 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_2değerinden küçükse bu alan yok sayılır.

ResponseRecPtr (MQPTR)

Bu, ilk MQRR yanıt kaydının adresidir. *ResponseRecPtr* , yalnızca bir dağıtım listesi açılırken kullanılır. *RecsPresent* sıfırda, alan yoksaılır.

Yanıt kayıtlarını belirtmek için *ResponseRecPtr* ya da *ResponseRecOffset* seçeneğini kullanın, ancak her ikisini de belirtmeyin; ayrıntılar için bkz. "[ResponseRecGörelİ Konumu \(MQUZE\)](#)" sayfa 470. *ResponseRecPtr* kullanmayacaksa, boş değeri ya da boş byte 'a ayarlayın.

Bu bir giriş alanıdır. Bu alanın ilk değeri, göstergeleri destekleyen programlama dillerindeki boş değeri gösterir ve tersi durumda, boş değeri byte dizilimidir. *Version* , MQOD_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 değeri byte dizisi olarak kabul edilir.

SelectionString (MQCHARV)

Bu dizilim, kuyruğun dışına ileti alınırken kullanılan seçim ölçütlerinin sağlanması için kullanılan dizidir.

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 aşıysa, çağrı neden kodu MQRC_SELECTION_STRING_ERROR ile başarısız olur. *SelectionString* uzunluğu üst sınırı MQ_SELECTOR_LENGTH.

SelectionString kullanımı, [Seçiciler](#)' de açıklanır.

StrucId (MQCHAR4)

Bu, yapı tanıttıcısıdır; değer şöyle olmalıdır:

MQOD_STRUC_ID

Nesne tanımlayıcı yapısına ilişkin tanıttıcı.

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.

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.

Sürüm (MQUZE)

Bu, yapı sürüm numarasıdır; değer 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.

MQOD için ilk değerler ve dil bildirimleri

Alan adı	Değişmezin adı	Değişmezin değeri
<i>StrucId</i>	MQOD_STRUC_ID	'OD- -'
<i>Version</i>	MQOD_VERSION_1	1

Alan adı	Değişmezin adı	Değişmezin değeri
<i>ObjectType</i>	MQOT_Q	1
<i>ObjectName</i>	Yok	Dizgi ya da boşluk boş değerli
<i>ObjectQMgrName</i>	Yok	Dizgi ya da boşluk boş değerli
<i>DynamicQName</i>	Yok	z/OS üzerinde 'CSQ.*' ; 'AMQ.*' aksi takdirde
<i>AlternateUserId</i>	Yok	Dizgi ya da boşluk boş değerli
<i>RecsPresent</i>	Yok	0
<i>KnownDestCount</i>	Yok	0
<i>UnknownDestCount</i>	Yok	0
<i>InvalidDestCount</i>	Yok	0
<i>ObjectRecOffset</i>	Yok	0
<i>ResponseRecOffset</i>	Yok	0
<i>ObjectRecPtr</i>	Yok	Boş değerli gösterge ya da boş değerli byte
<i>ResponseRecPtr</i>	Yok	Boş değerli gösterge ya da boş değerli byte
<i>AlternateSecurityId</i>	MQSID_NONE	Boş Değerler
<i>ResolvedQName</i>	Yok	Dizgi ya da boşluk boş değerli
<i>ResolvedQMgrName</i>	Yok	Dizgi ya da boşluk boş değerli
<i>ObjectString</i>	MQCHARV_DEFAULT	MQCHARV için tanımlandığı gibi
<i>SelectionString</i>	MQCHARV_DEFAULT	MQCHARV için tanımlandığı gibi
<i>ResObjectString</i>	MQCHARV_DEFAULT	MQCHARV için tanımlandığı gibi
<i>ResolvedType</i>	MQOT_NONE	0

Notlar:

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.
3. C programlama dilinde, makro değişkeniMQOD_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlara ilk değerleri sağlamak için aşağıdaki şekilde kullanılabilir:

```
MQOD MyOD = {MQOD_DEFAULT};
```

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 */
};

```

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-OBJECTQMGRNAME              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-KNOWNDESTCOUNT             PIC S9(9) BINARY.
** Number of remote queues opened successfully
15  MQOD-UNKNOWNDESTCOUNT          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-OBJECTRECPtr                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.

```

PL/I bildirim

```

dcl
1 MQOD based,
3 StructId          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
                                     successfully */
3 InvalidDestCount  fixed bin(31),    /* Number of queues that failed to
                                     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,
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,
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,
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 */

High Level Assembler 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
*                                  successfully
MQOD_UNKNOWNDDESTCOUNT             DS    F     Number of remote queues opened
*                                  successfully
MQOD_INVALIDDESTCOUNT              DS    F     Number of queues that failed to
*                                  open
MQOD_OBJECTRECOFFSET                DS    F     Offset of first object record from
*                                  start of MQOD
MQOD_RESPONSERECOFFSET              DS    F     Offset of first response record
*                                  from start of MQOD
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_VSBUFFSIZE        DS    F     size of buffer
MQOD_OBJECTSTRING_VSLENGTH          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_VSBUFFSIZE     DS    F     size of buffer
MQOD_SELECTIONSTRING_VSLENGTH       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_VSBUFFSIZE     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)

```

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'

```

UnknownDestCount	As Long	'successfully' 'Number of remote queues opened'
InvalidDestCount	As Long	'successfully' 'Number of queues that failed to'
ObjectRecOffset	As Long	'open' 'Offset of first object record from'
ResponseRecOffset	As Long	'start of MQOD' 'Offset of first response record'
ObjectRecPtr	As MQPTR	'from start of MQOD' '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		

MQOR-Nesne kaydı

Aşağıdaki çizelge yapıdaki alanları özetlemektedir.

Çizelge 61. MQOR içindeki alanlar		
Alan	Tanım	Konu
<i>ObjectName</i>	Nesne adı	ObjectName
<i>ObjectQMGrName</i>	Nesne kuyruğu yöneticisi adı	ObjectQMGrAdı

MQOR için genel bakış

Kullanılabilirlik: Bu sistemlere bağlı AIX, HP-UX, IBM i, Solaris, Linux, Windows, artı IBM MQ MQI clients .

Amaç: Tek bir hedef kuyruğun kuyruk adını ve kuyruk yöneticisi adını belirtmek için MQOR yapısını kullanın. MQOR, MQOPEN ve MQPUT1 çağrılarını için bir giriş yapısıdır.

Karakter kümesi ve kodlama: MQOR içindeki veriler, *CodedCharSetId* kuyruk yöneticisi özniteliği tarafından verilen karakter kümesinde ve MQKEN_NATIVE tarafından verilen yerel kuyruk yöneticisinin kodlamasında olmalıdır. Ancak, uygulama bir MQ MQI istemcisi olarak çalışıyorsa, yapı istemcinin karakter kümesinde ve kodlamasında olmalıdır.

Kullanım: MQOPER çağrısında bu yapılardan oluşan bir dizi sağlayarak, kuyruklar listesini açabilir; bu listeye *dağıtım listesi* adı verilir. Kuyruğun başarıyla açılması koşuluyla, listedeki her bir kuyruğun döndürdüğü kuyruk tanıtıcısı kullanılarak döndürülen her ileti, listedeki kuyrukların her birine yerleştirilir.

MQOR için alanlar

MQOR yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

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.

MQOR için ilk değerler ve dil bildirimleri

Çizelge 62. MQOR için, MQOR ile ilgili alanların ilk değerleri

Alan adı	Değişmezin adı	Değişmezin değeri
<i>ObjectName</i>	Yok	Dizgi ya da boşluk boş değeri
<i>ObjectQMgrName</i>	Yok	Dizgi ya da boşluk boş değeri
Notlar: - Boş değeri dizgi ya da boşluklar, C içindeki boş değeri dizeyi ve diğer programlama dillerindeki boş karakterleri belirtir. - C programlama dilinde, makro değişkeniMQOR_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlara ilk değerleri sağlamak için aşağıdaki şekilde kullanılabilir: <pre>MQOR MyOR = {MQOR_DEFAULT};</pre>		

C bildirimi

```
typedef struct tagMQOR MQOR;
struct tagMQOR {
    MQCHAR48 ObjectName; /* Object name */
    MQCHAR48 ObjectQMgrName; /* Object queue manager name */
};
```

COBOL bildirimi

```
** MQOR structure
10 MQOR.
** Object name
15 MQOR-OBJECTNAME PIC X(48).
** Object queue manager name
15 MQOR-OBJECTQMGRNAME PIC X(48).
```

PL/I bildirimi

```
dcl
1 MQOR based,
3 ObjectName char(48), /* Object name */
3 ObjectQMgrName char(48); /* Object queue manager name */
```

Visual Basic bildirimi

```
Type MQOR
    ObjectName As String*48 'Object name'
    ObjectQMgrName As String*48 'Object queue manager name'
End Type
```

MQPD-Özellik tanımlayıcısı

Aşağıdaki çizelge yapıdaki alanları özetlemektedir.

Çizelge 63. MQPD 'deki alanlar		
Alan	Tanım	Konu
<i>StrucId</i>	Yapı tanıttıcısı	StrucId
<i>Version</i>	Yapı sürüm numarası	S\u00fcr\u00fcm
<i>Options</i>	Seçenekler	Seçenekler

Çizelge 63. MQPD ' deki alanlar (devamı var)		
Alan	Tanım	Konu
<i>Support</i>	İleti özelliği için gereken destek	Destek
<i>Context</i>	Özelliğin ait olduğu ileti bağlamı	Bağlam
<i>CopyOptions</i>	Özelliğin ait olduğu seçenekleri kopyala	CopyOptions

MQPD ' ye Genel Bakış

Kullanılabilirlik: AIX, HP-UX, IBM i, Solaris, Linux, Windows, z/OS ve IBM MQ MQI clients.

Amaç: MQPD , bir özelliğin özniteliklerini tanımlamak için kullanılır. Yapı, MQSETMP çağrısında bir giriş/çıkış değiştirgesi ve MQINQMP çağrısındaki bir çıkış değiştirgesinde bulunuyor.

Karakter kümesi ve kodlama: MQPD içindeki veriler, uygulamanın uygulama ve kodlamasının karakter kümesinde (**MQENC_NATIVE**) olmalıdır.

MQPD ile ilgili alanlar

MQPD yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

Bağlam (MQUZE)

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 *Context* alanının değerini düzeltir.

Aşağıdaki seçenek belirlenebilir:

MQPD_USER_CONTEXT

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, MQOO_SAVE_ALL_CONTEXT için açıklandığı şekilde saklanır. MQPMO_PASS_ALL_CONTEXT ile MQPUT çağrısı, özelliğin 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 değışımezi bir kereden fazla eklemeyin) ya da bit ya da işlemi kullanarak değerleri birleştirin (programlama dili, bit işlemleri destekliyse).

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.

Seçenekler (MQUZE)

Değer şu olmalıdır:

MQPD_NONE

Seçenek belirtilmedi

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQPD_NONE olur.

StrucId (MQCHAR4)

Bu, yapı tanıtcısıdır; değer şöyle olmalıdır:

MQPD_STRUC_ID

Özellik tanımlayıcı yapısına ilişkin tanıtcı.

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.

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 tanıtcısı için MQSETMP çağrısını kullanarak IBM MQtanımlı bir özellik ayarlarken, *Support* 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.

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 alırsa, özellik şu şekilde ele alınır:

- 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 belirlendiği 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_REQUIREY

Özellikle ilişkin destek gereklidir. İleti, IBM MQtanımlı özelliği desteklemeyen bir kuyruk yöneticisi tarafından reddedilir. MQPUT ya da MQPUT1 çağrısı, tamamlanma kodu MQCC_FAILED ve neden kodu MQRC_UNUNSUPPORTED_EXTY ile başarısız olur.

MQPD_SUPPORT_REQUIRED_IF_LOCAL

İleti, yerel bir kuyruk için yazılmışsa, IBM MQtanımlı özelliği desteklemeyen bir kuyruk yöneticisi tarafından reddedilir. MQPUT ya da MQPUT1 çağrısı, tamamlanma kodu MQCC_FAILED ve neden kodu MQRC_UNUNSUPPORTED_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.

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.

MQPD ' ye ilişkin ilk değerler ve dil bildirimleri

Çizelge 64. MQPD ' deki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
<i>StrucId</i>	MQPD_STRUC_ID	' PD '
<i>Version</i>	MQPD_VERSION_1	1
<i>Options</i>	MQPD_NONE	0
<i>Support</i>	MQPD_SUPPORT_OPTIONAL	0
<i>Context</i>	MQPD_NO_CONTEXT	0

Çizelge 64. MQPD 'deki alanların ilk değerleri (devamı var)

Alan adı	Değişmezin adı	Değişmezin değeri
CopyOptions	MQCOPY_DEFAULT	0
Notlar: 1. C programlama dilinde, MQPD_default makro değişkeninde çizelgede listelenen değerler bulunur. Yapıdaki alanlara ilk değerleri sağlamak için aşağıdaki şekilde kullanılabilir: <pre>MQPD MyPD = {MQPD_DEFAULT};</pre>		

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 */
};
```

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.
```

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 */
```

High Level Assembler 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

MQPMO-put-message seçenekleri

Aşağıdaki çizelge yapıdaki alanları özetlemektedir.

Çizelge 65. MQPMO yapısı		
Alan	Tanım	Konu
<i>StrucId</i>	Yapı tanıtıcısı	StrucId
<i>Version</i>	Yapı sürüm numarası	S\u00f0fcr\u00f0fcm
<i>Options</i>	MQPUT ve MQPUT1işleminin denetlenmesini denetleyen seçenekler	Seçenekler
<i>Timeout</i>	Ayrıldı	Zamanaşımı
<i>Context</i>	Giriş kuyruğunun nesne tanıtıcısı	Bağlam
<i>KnownDestCount</i>	Yerel kuyruklara başarıyla gönderilen ileti sayısı	KnownDestSayı
<i>UnknownDestCount</i>	Uzak kuyruklara başarıyla gönderilen ileti sayısı	UnknownDestSayısı
<i>InvalidDestCount</i>	Gönderilememiş iletilerin sayısı	InvalidDestSayısı
<i>ResolvedQName</i>	Hedef kuyruğun çözümlenmiş adı	ResolvedQName
<i>ResolvedQMgrName</i>	Hedef kuyruk yöneticisinin çözümlenmiş adı	ResolvedQMgrAdı
Not: <i>Version</i> , MQPMO_VERSION_2değerinden küçükse, geri kalan alanlar yoksayılır.		
<i>RecsPresent</i>	Gönderilen ileti kaydı ya da yanıt kayıtlarının sayısı	RecsPresent
<i>PutMsgRecFields</i>	Hangi MQPMR alanlarının var olduğunu gösteren işaretler	PutMsgRecFields
<i>PutMsgRecOffset</i>	MQPMO ' nun başlangıcından gelen ilk put-ileti kaydı konumu	PutMsgRecOffset
<i>ResponseRecOffset</i>	MQPMO ' nun başlangıcından ilk yanıt kaydının kayması	ResponseRecKayması
<i>PutMsgRecPtr</i>	İlk ileti kaydı kaydının adresi	PutMsgRecPtr
<i>ResponseRecPtr</i>	İlk yanıt kaydının adresi	ResponseRecPtr 'si
Not: <i>Version</i> , MQPMO_VERSION_3değerinden küçükse, geri kalan alanlar yoksayılır.		
<i>OriginalMsgHandle</i>	Özgün ileti tanıtıcısı	OriginalMsgtanıtıcı değeri
<i>NewMsgHandle</i>	Yeni ileti tanıtıcısı	NewMsgtanıtıcısı
<i>Action</i>	Gerçekleştirilmekte olan ve <i>OriginalMsgHandle</i> alanı tarafından belirtilen özgün ileti ile <i>NewMsgHandle</i> alanı tarafından belirtilen yeni ileti arasındaki ilişkiyi içeren bir put tipi.	İşlem
<i>PubLevel</i>	Yayın tarafından hedeflenen abonelik düzeyi	PubLevel

MQPMO 'ya Genel Bakış

Kullanılabilirlik: Tüm IBM MQ sistemleri, ayrıca bu sistemlere bağlı IBM MQ istemcileri.

Amaç: MQPMO yapısı, uygulamanın, iletilerin kuyruklara nasıl yerleştirileceğini denetleyen ya da konu başlıklarına ilişkin seçenekleri belirleyebilmesini sağlar. Yapı, MQPUT ve MQPUT1 çağrılarında bir giriş/çıkış değiştirir.

Sürüm: MQPMO 'nun yürürlükteki sürümü MQPMO_VERSION_3' tür. Belirli alanlar yalnızca MQPMO 'nun belirli sürümlerinde kullanılabilir. Birkaç ortam arasındaki uygulamaları kapı olarak görmeniz gerekiyorsa, MQPMO sürümünün tüm ortamlarda tutarlı olduğundan emin olmanız gerekir. Yalnızca yapının belirli sürümlerinde bulunan alanlar, [“MQPMO-put-message seçenekleri”](#) sayfa 482 içinde ve alan açıklamalarında olduğu gibi tanımlanır.

Desteklenen programlama dilleri için sağlanan üstbilgi, COPY ve INCLUDE dosyaları, ortam tarafından desteklenen MQPMO 'nun en son sürümünü içerir, ancak *Version* alanının ilk değeri MQPMO_VERSION_1olarak ayarlıdır. version-1 yapısında mevcut olmayan alanları kullanmak için, uygulamanın *Version* alanını, gereken sürümün sürüm numarasına ayarlaması gerekir.

Karakter kümesi ve kodlama: MQPMO içindeki veriler, *CodedCharSetId* kuyruk yöneticisi özniteliği tarafından verilen karakter kümesinde ve MQKEN_NATIVE tarafından verilen yerel kuyruk yöneticisinin kodlamasında olmalıdır. Ancak, uygulama bir MQ MQI istemcisi olarak çalışıyorsa, yapı istemcinin karakter kümesinde ve kodlamasında olmalıdır.

MQPMO için alanlar

MQPMO yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

İşlem (MQUZE)

Bu, gerçekleştirilmekte olan put tipini ve OriginalMsgtanıtıcı 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 belirtik olarak ayarlanan ileti tanımlayıcı alanları, özgün ileti tanıtıcısındaki önceliklerden öncelikli olarak uygulanır.

İleti verileri, MQPUT ya da MQPUT1 arabellek değiştirgesinden alınır.

MQAKP_ILERİ

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 değiştirgesinden 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 seçilir:

Çizelge 66. Yanıt iletisi tanıtıcısı dönüşümü	
MQMD ' de alan	Kullanılan değer
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_UNSNISZ
Geribildirim	MQFB_YOK

Çizelge 66. 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:

Çizelge 67. İleti tanıtıcısı dönüşümünü raporla	
MQMD ' de alan	Kullanılan değer
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	BufferLengtholarak 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 özellik ile 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 ' ın bir Feedback değeri, MQPUT ya da MQPUT1 çağrısının, MQRC_FEEDBACK_ERROR neden koduyla başarısız olmasına neden olur.

Rapor iletisinin kullanıcı verilerini seçmek için IBM MQ , sonuçtaki MQMD ' deki Rapor ve Geri Bildirim alanlarını ve MQPUT ya da MQPUT1 çağrısının Arabellek ve BufferLength değıştirmeleriyle iletişim 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
- Aşağıdaki durumlardan biri geçerliyse, 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 seçeneklerini içermiyorsa, bu iletiyle hiçbir kullanıcı verisi içerilmez.
- Geribildirim, yukarıda listelenen değerdan farklı bir değeri alırsa, arabellek ve BufferLength değeri normal olarak kullanılır.

Kullanıcı verilerinin türetme işlemi aşağıdaki çizelgede gösterilmiştir:

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.

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.

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.

NewMsgTanıtıcısı (MQHMSG)

Bu, işlemin işlem alanının değerine tabi olduğu isteğe bağlı bir tanıtıcıdır. Bu ileti, iletinin özelliklerini tanımlar ve belirtildiyse, *OriginalMsgHandled* 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.

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 destekliyse). 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, yayımlandığı 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 yayınlı değiştirilir. Bu nedenle, aynı konu üzerinde birden fazla yayınlı 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 775](#) çağrısının açıklamasına bakın.

Tutulan yayınların abonelik düzeyleriyle nasıl etkileşimde bulunabilmesiyle ilgili bilgi edinmek için [Yayınlı Arası Yayınlar](#) konusuna bakın.

Bu seçenek kullanılırsa ve yayın saklanamazsa, ileti yayınlı olmaz 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 yayınlı 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, yayınlı herhangi bir aboneliğe teslim edilemedi. Ancak, koyma işleminde MQPMO_RETAIN seçeneği belirtilirse, ileti alıkonur ve daha sonra tanımlanmış eşleşen abonelikte teslim edilir.

Aşağıdaki koşullardan herhangi biri karşılanırsa, konuyla ilgili bir abonelik yayınlı eşleşir:

- İleti, abonelik kuyruğuna teslim edilir
- İleti abonelik kuyruğuna teslim edildi; ancak, kuyrukla ilgili bir sorun, iletinin kuyruğa konamadığı 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 yayınlı ilgili bir abonelik yayınlı olmaz:

- Aboneliğin, yayınlı eşleşmeyen bir seçim dizgisi var.
- Abonelik, MQSO_PUBLICATION_ON_REQUEST seçeneğini belirtti.
- Put işleminde MQPMO_NOT_OWN_SUBS seçeneği belirtildiğinden ve abonelik yayınlı yayınlıcının kimliğiyle eşleştiğinden yayınlı 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 iletiyle birlikte gönderilir ve MQPUT ya da MQPUT1 çağrısından çıkış için uygulamaya geri gönderilir.

MQPMO_NEW_MSG_ID seçeneği, ileti bir dağıtım listesine konduğunda 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 iletiyle birlikte gönderilir ve MQPUT ya da MQPUT1 çağrısından çıkış için uygulamaya geri gönderilir.

MQPMO_NEW_CORREL_ID seçeneği, ileti bir dağıtım listesine konduğunda 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 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*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 ilettten oluşur. Kesimler, mantıksal ilettide verilerin başlangıcındaki fiziksel ilettteki 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 ilettteki kesimler genellikle farklı ileti tanıtıcılarına sahiptir.

Kesimlere ayrılmış olmayan, ancak gönderme uygulaması tarafından bölümlenmeye 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 belirtildiyse, uygulamanın art arda gelen MQPUT çağrılarını kullandığını gösterir:

1. Her bir mantıksal iletiye, boşluk olmadan, 0 'dan başlayarak, artan kesim görel konumu sırasına göre kesimler yerleştirin.
2. Bölümleri sonraki mantıksal iletiye koymadan önce, tüm bölümleri bir mantıksal iletiye koyun.

3. Herhangi bir boşluk olmadan, 1 'den başlayarak, ileti sıra sayısı artırımı sırasına göre, her ileti grubuna mantıksal iletileri yerleştirin. IBM MQ , ileti sıra numarasını otomatik olarak artırır.
4. Sonraki ileti grubuna mantıksal iletiler 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_BAĞLAMı

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:

MQMD ' de alan	Kullanılan değer
<i>UserIdentifier</i>	Olanaklıysa, ortamdan saptanır; tersi durumda boşluklara ayarlanır.
<i>AccountingToken</i>	Olanaklıysa, ortamdan saptanır; tersi durumda MQACT_NONE değerine ayarlanır.
<i>ApplIdentityData</i>	Boşluklara ayarlayın.
<i>PutApplType</i>	Ortamdan saptanır.
<i>PutApplName</i>	Olanaklıysa, ortamdan saptanır; tersi durumda boşluklara ayarlanır.
<i>PutDate</i>	İletinin konulduğu tarihe ayarlanır.
<i>PutTime</i>	İletinin konulduğu zamana ayarlanır.
<i>ApplOriginData</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 yükseltebilir. 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 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. Bir istemci uygulaması Sürüm 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, DEFRESP özneliliğinin 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 eşanlamlıdır.

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.

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.

PubLevel (MQUTR)

Bu alanın ilk değeri 9 'tır. Bu yayın tarafından hedeflenen abonelik düzeyi. Yalnızca, bu değere eşit ya da bu değerden küçük olan en yüksek SubLevel abonelikleri bu yayını alır. Bu değer, sıfır ile 9 aralığında olmalıdır; sıfır, en düşük düzeydir. Ancak, bir yayın saklandıysa, PubLevel 1 düzeyinde yeniden yayınlandığından, bu yayın artık daha yüksek düzeylerde aboneler için kullanılabilir değildir.

Bilgi için bkz. [Intercepting yayınları](#).

PutMsgRecFields (MQlong)

Bu alan, uygulama tarafından sağlanan ileti kaydı kayıtlarında hangi MQPMR alanlarının bulunduğunu belirten işaretler içerir. *PutMsgRecFields* yalnızca ileti bir dağıtım listesine konduğunda kullanın. *RecsPresent* sıfırsa ya da hem *PutMsgRecOffset* , hem de *PutMsgRecPtr* 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 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 değişimin başka biriyle birlikte kullanılması amaçlanmaz, ancak değeri sıfır olduğu 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* 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
- İlti 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 olduğu 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*indeki bir görelî konum belirterek ya da *PutMsgRecPtr* içinde bir adres belirterek, MQOD 'daki nesne kayıtlarına benzer bir şekilde sağlayın. Bunu nasıl yapacağını ilişkin ayrıntılar için, "[MQOD-Nesne tanımlayıcısı](#)" sayfa 462içinde 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.

PutMsgRecPtr (MQPTR)

Bu, ileti kaydının ilk MQPMR 'nin adresidir. *PutMsgRecPtr* yalnızca ileti bir dağıtım listesine konduğunda kullanılır. *RecsPresent* sıfırsa, alan yoksayılır.

Put ileti kayıtlarını belirtmek için *PutMsgRecPtr* ya da *PutMsgRecOffset* kullanılabilir, ancak her ikisi de değil; ayrıntılar için "[PutMsgRecOffset \(MQlong\)](#)" sayfa 495'e bakın. *PutMsgRecPtr*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* , MQPMO_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.

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 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.

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.

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.

ResponseRecGörelİ Konumu (MQUZE)

Bu, MQPMO yapısının başlangıcındaki ilk MQRR yanıt kaydının bayt cinsinden görelİ konudur. Görelİ konum pozitif ya da negatif olabilir. *ResponseRecOffset* yalnızca ileti bir dağıtım listesine konduğunda kullanılır. *RecsPresent* sıfırsa, alan yoksayılr.

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; MQRC_MULTIPLE_REASONS çağrıda neden kodu bu vakayı gösterir. 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 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 işlemde 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 MQOD, 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 “MQOD-Nesne tanımlayıcısı” sayfa 462. Ancak, *ResponseRecOffset* ve *ResponseRecPtr* ' den en fazla birini kullanın; Her ikisi de sıfır dışında bir durumsa, çağrı neden kodu MQRC_RESPONSE_RECORDS_ERROR ile başarısız olur.

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. *Version* , MQPMO_VERSION_2değerinden küçükse bu alan yok sayılır.

ResponseRecPtr (MQPTR)

Bu, ilk MQRR yanıt kaydının adresidir. *ResponseRecPtr* yalnızca ileti bir dağıtım listesine konduğunda kullanılır. *RecsPresent* sıfırsa, alan yoksayılır.

Yanıt kayıtlarını belirtmek için *ResponseRecPtr* ya da *ResponseRecOffset* seçeneğini kullanın, ancak her ikisini de belirtmeyin; ayrıntılar için bkz. “*ResponseRecGörelili Konumu (MQUZE)*” sayfa 496. *ResponseRecPtr* kullanmazsa, boş değeri ya da boş değeri baytlara ayarlayın.

MQPUT1 çağrısı için, bu alan boş değeri ya da boş değeri byte olmalıdır. Bunun nedeni, yanıt bilgilerinin (istendiye), MQOD nesne tanımlayıcısı tarafından belirtilen yanıt kayıtlarında döndürülmesinden kaynaklanır.

Bu bir giriş alanıdır. Bu alanın ilk değeri, göstergeleri destekleyen programlama dillerindeki boş değeri göstergedir ve tersi durumda, boş değeri byte dizilimidir. *Version* , MQPMO_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 değeri byte dizgisi olarak kabul edilir.

StrucId (MQCHAR4)

Bu, yapı tanıtıcısıdır; değer şöyle olmalıdır:

MQPMO_STRUC_ID

Put-message options yapısına ilişkin tanıtıcı.

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.

Zamanaşımı (MQUZE)

Bu ayrılmış bir alandır; değeri anlamlı değildir. Bu alanın başlangıç değeri -1' dir.

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.

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 şu ortamlarda desteklenir: AIX, HP-UX, IBM i, Solaris, Linux, Windows, artı IBM MQ MQI clients bu sistemlere bağlı olarak.

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 için ilk değerler ve dil bildirimleri

Çizelge 68. MQPMO 'daki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
<i>StrucId</i>	MQPMO_STRUC_ID	'PMO~'
<i>Version</i>	MQPMO_VERSION_1	1
<i>Options</i>	MQPMO_NONE	0
<i>Timeout</i>	Yok	-1
<i>Context</i>	Yok	0
<i>KnownDestCount</i>	Yok	0
<i>UnknownDestCount</i>	Yok	0
<i>InvalidDestCount</i>	Yok	0
<i>ResolvedQName</i>	Yok	Dizgi ya da boşluk boş değerli
<i>ResolvedQMgrName</i>	Yok	Dizgi ya da boşluk boş değerli
<i>RecsPresent</i>	Yok	0
<i>PutMsgRecFields</i>	MQPMRF_NONE	0
<i>PutMsgRecOffset</i>	Yok	0
<i>ResponseRecOffset</i>	Yok	0

Çizelge 68. MQPMO 'daki alanların ilk değerleri (devamı var)

Alan adı	Değişmezin adı	Değişmezin değeri
PutMsgRecPtr	Yok	Boş değerli gösterge ya da boş değerli byte
ResponseRecPtr	Yok	Boş değerli gösterge ya da boş değerli byte
OriginalMsgHandle	MQHM_NONE	0
NewMsgHandle	MQHM_NONE	0
Action	MQACTP_YENİ	0
PubLevel	Yok	9

Notlar:

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.
3. C programlama dilinde, makro değişkeniMQPMO_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlara ilişkin ilk değerleri sağlamak için bunu aşağıdaki şekilde kullanın:

```
MQPMO MyPMO = {MQPMO_DEFAULT};
```

C bildirimi

```
typedef struct tagMQPMO MQPMO;
struct tagMQPMO {
    MQCHAR4 StrucId; /* 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 */
};
```

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-KNOWNDESTCOUNT PIC S9(9) BINARY.
** Number of messages sent successfully to remote queues
15 MQPMO-UNKNOWNDESTCOUNT PIC S9(9) BINARY.
** Number of messages that could not be sent
15 MQPMO-INVALIDDESTCOUNT 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.

```

PL/I bildiriimi

```

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 */

High Level Assembler 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_UNKNOWNDESTCOUNT	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)	

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 'MQPUT and MQPUT1'
		'Reserved'
Timeout	As Long	'Object handle of input queue'
Context	As Long	'Number of messages sent successfully' to local queues'
KnownDestCount	As Long	'Number of messages sent successfully' to remote queues'
		'Number of messages that could not be sent'
InvalidDestCount	As Long	'Resolved name of destination queue'
ResolvedQName	As String*48	'Resolved name of destination queue manager'
ResolvedQMgrName	As String*48	'Number of put message records or response records present'
RecsPresent	As Long	'Flags indicating which MQPMR fields are present'
PutMsgRecFields	As Long	'Offset of first put message record from start of MQPMO'
PutMsgRecOffset	As Long	'Offset of first response record from start of MQPMO'
ResponseRecOffset	As Long	'Address of first put message record'
PutMsgRecPtr	As MQPTR	'Address of first response record'
ResponseRecPtr	As MQPTR	
End Type		

MQPMR-Put-message kaydı

Aşağıdaki çizelge yapıdaki alanları özetlemektedir.

Çizelge 69. MQPMR içindeki alanlar		
Alan	Tanım	Konu
<i>MsgId</i>	İleti Tanıtıcısı	MsgId
<i>CorrelId</i>	İlinti tanıtıcısı	CorrelId
<i>GroupId</i>	Grup tanıtıcısı	GroupId
<i>Feedback</i>	Geribildirim ya da neden kodu	Geribildirim
<i>AccountingToken</i>	Hesap simgesi	AccountingToken

MQPMR 'ye Genel Bakış

Kullanılabilirlik: AIX, HP-UX, IBM i, Solaris, Linux, Windowsve ayrıca bu sistemlere bağlı IBM MQ istemcileri.

Amaç: Bir dağıtım listesine ileti yerleştirilirken tek bir hedefe ilişkin çeşitli ileti özelliklerini belirtmek için MQPMR yapısını kullanın. MQPMR, MQPUT ve MQPUT1 çağrılarına ilişkin bir giriş/çıkış yapısıdır.

Karakter kümesi ve kodlama: MQPMR içindeki veriler, *CodedCharSetId* kuyruk yöneticisi özniteliği tarafından verilen karakter kümesinde ve MQKEN_NATIVE tarafından verilen yerel kuyruk yöneticisinin kodlamasında olmalıdır. 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 kuyruğu için farklı değerler belirleyebilirsiniz. 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 *PutMsgRecFields* alanındaki işaretlerle gösterilir. Fields that are present **aşağıdaki sırada olmalıdır** :

- *MsgId*
- *CorrelId*
- *GroupId*
- *Feedback*
- *AccountingToken*

Olmayan alanlar, kayıta yer kapmaz.

MQPMR 'nin değişmez yerleşim düzeni olmadığı için, desteklenen programlama dillerine ilişkin üstbilgide, COPY ve INCLUDE kütüklerinde hiçbir tanım sağlanmaz. Uygulama programcısı, uygulamanın gerektirdiği alanları içeren bir bildirim yaratmalı ve var olan alanları belirtmek için *PutMsgRecFields* içindeki işaretleri ayarlamalıdır.

MQPMR için alanlar

MQPMR yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

AccountingToken (MQBYTE32)

Bu, MQOPER ya da MQPUT1 çağrısında bulunan MQOR yapılarının dizideki ilgili öge tarafından belirtilen adla kuyruğa gönderilen ileti için kullanılacak hesap simgesindeki bir simgedir. Bu, tek bir kuyruk için MQMD 'deki *AccountingToken* alanı ile aynı şekilde işlenir. Bu alanın içeriğiyle ilgili bilgi için “MQMD-İleti tanımlayıcısı” sayfa 404 içindeki *AccountingToken* açıklamalarına bakın.

Bu alan mevcut değilse, MQMD 'deki değer kullanılır.

Bu bir giriş alanıdır.

CorrelId (MQBYTE24)

Bu, MQOPER ya da MQPUT1 çağrısında bulunan MQOR yapılarının dizideki ilgili öge tarafından belirtilen bir adla kuyruğa 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 şeklinden farklıdır (*MsgId* alanına bakın).

Bu bir giriş/çıkış alanıdır.

Geribildirim (MQUZE)

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 *Feedback* alanı ile aynı şekilde işlenir.

Bu alan mevcut değilse, MQMD ' deki değer kullanılır.

Bu bir giriş alanıdır.

GroupId (MQBYTE24)

GroupId , MQOPER ya da MQPUT1 çağrısında bulunan MQOR yapılarının dizideki ilgili öge tarafından belirtilen ada sahip kuyruğa 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. Değer, Kuyruklardaki fiziksel sıralama içinde belgelendiği şekilde işlenir, ancak aşağıdaki farklar söz alır:

- GroupId , QMName 'den ve bir zaman damgasından yaratılır. Bu nedenle, bir GroupId benzersiz tutma kuyruğu yöneticisi adlarını da benzersiz tutmanız gerekir. Ayrıca, kuyruk yöneticileri makinesinde saatleri geri ayarlamayın.
- Yeni bir grup tanıtıcısının kullanılabileceği durumlarda, kuyruk yöneticisi her hedef için farklı bir grup tanıtıcısı oluşturur (yani, iki hedef aynı grup tanıtıcısına sahip değildir).
- Alandaki değer kullanılabileceği durumlarda, çağrı neden kodu MQRC_GROUP_ID_ERROR ile başarısız olur.

Bu bir giriş/çıkış alanıdır.

MsgId (MQBYTE24)

Bu ileti, MQOPER ya da MQPUT1 çağrısında bulunan MQOR yapılarının dizisindeki 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 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 *MsgId* field. Bu değer MQMI_NONE ise, bu hedeflerin *her biri* için yeni bir ileti tanıtıcısı oluşturulur (yani, bu hedeflerin ikisi de aynı ileti tanıtıcısına sahip değildir).

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.

MQPMR için ilk değerler ve dil bildirimleri

Bu yapı için herhangi bir başlangıç değeri tanımlanmadığı için, desteklenen programlama dillerine ilişkin üstbilgide, COPY ve INCLUDE kütüklerinde hiçbir yapı bildirimi sağlanmadığı için, bu değerler tanımlanmaz. Örnek bildirimlerde, tüm alanlar gerekliyse, yapının nasıl bildirileceği gösterilir.

C bildirimi

```
typedef struct tagMQPMR MQPMR;
struct tagMQPMR {
    MQBYTE24 MsgId;           /* Message identifier */
    MQBYTE24 CorrelId;        /* Correlation identifier */
    MQBYTE24 GroupId;         /* Group identifier */
    MQLONG Feedback;          /* Feedback or reason code */
    MQBYTE32 AccountingToken; /* Accounting token */
};
```

COBOL bildirimi

```
** MQPMR structure
10 MQPMR.
** Message identifier
15 MQPMR-MSGID PIC X(24).
** Correlation identifier
15 MQPMR-CORRELID PIC X(24).
** Group identifier
15 MQPMR-GROUPID PIC X(24).
** Feedback or reason code
15 MQPMR-FEEDBACK PIC S9(9) BINARY.
** Accounting token
15 MQPMR-ACCOUNTINGTOKEN PIC X(32).
```

PL/I bildirimi

```
dcl
1 MQPMR 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 */
```

Visual Basic bildirimi

```
Type MQPMR
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
```

MQRFH-Kurallar ve biçimleme üstbilgisi

Bu bölümde, kurallar ve biçimlendirme üstbilgisi, içerdiği alanlar ve bu alanların ilk değerleri ele alınmıştır.

MQRFH 'ye Genel Bakış

Uygunluk: Tüm IBM MQ sistemleri, artı IBM MQ MQI clients bu sistemlere bağlı.

Amaç: MQRFH yapısı, kural ve biçimlendirme üstbilgisinin düzenini tanımlar. Dizgi verisini ad/değer çiftleri biçiminde göndermek için bu başlığı kullanın.

Biçim adı: MQFMT_RF_HEADER.

Karakter kümesi ve kodlama: MQRFH yapısındaki alanlar (*NameValueString* içinde olmak üzere), MQRFH 'den önce gelen üstbilgi yapısındaki *CodedCharSetId* ve *Encoding* alanları tarafından verilen karakter kümesi ve kodlamadır. MQRFH, uygulama iletisi 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.

MQRFH için alanlar

MQRFH yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

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 *PutApplType* alanının değeri MQAT_BROKER ise, MQCCSI_INHERIT kullanılamaz.

Bu alanın ilk değeri MQCCSI_UNDEFED değerine sahip.

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.

İşaretler (MQUZE)

Şu öge belirtilebilir:

MQRFH_NONE

Bayrak yok.

Bu alanın ilk değeri MQRFH_NONE olur.

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.

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 düzelterek 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 yoksayılır.

Not: Bu alanın uzunluğu değişmez olduğu için, alan, desteklenen programlama dilleri için sağlanan yapının bildirimlerinden çıkarılır.

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.

StrucLength (MQUZE)

Bu, yapının sonundaki *NameValueString* alanı da dahil olmak üzere, MQRFH yapısının bayt cinsinden uzunluğudur. Uzunluk, *NameValueString* alanını izleyen herhangi bir kullanıcı verisi içermemektedir .

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.

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.

MQRFH için ilk değerler ve dil bildirimleri

Çizelge 70. MQRFH için MQRFH içindeki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
StrucId	MQRFH_STRUC_ID	'RFH↵'
Version	MQRFH_VERSION_1	1
StrucLength	MQRFH_STRUC_LENGTH_FIXED	32
Encoding	MQENC_NATIVE	Ortama göre değişir
CodedCharSetId	MQCCSI_UNDEDINED	0
Format	MQFMT_NONE	Boşluklar
Flags	MQRFH_NONE	0
Notlar: 1. ↵ simgesi tek bir boş karakteri temsil eder. 2. C programlama dilinde, makro değişkeniMQRFH_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlara ilk değerleri sağlamak için aşağıdaki şekilde kullanılabilir: <pre>MQRFH MyRFH = {MQRFH_DEFAULT};</pre>		

C bildirimi

```
typedef struct tagMQRFH MQRFH;  
struct tagMQRFH {  
    MQCHAR4 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 */  
    MQCHAR8 Format;            /* Format name of data that follows  
                               NameValueString */  
    MQLONG Flags;              /* Flags */  
};
```

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.

```

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 */

```

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 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)

```

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'
                          'NameValueString'
  Encoding     As Long     'Numeric encoding of data that follows'
                          'NameValueString'
  CodedCharSetId As Long   'Character set identifier of data that'
                          'follows NameValueString'
  Format       As String*8 'Format name of data that follows'
                          'NameValueString'
  Flags       As Long     'Flags'
End Type

```

MQRFH2 -Kural ve biçimlendirme üstbilgisi 2

Bu bölümde, kurallar ve biçimlendirme üstbilgisi 2, içerdiği alanlar ve bu alanların ilk değerleri ele alınmıştır.

MQRFH2için genel bakış

Kullanılabilirlik

Tüm IBM MQ sistemleri, artı IBM MQ MQI clients bu sistemlere bağlı olmalıdır.

Amaç

MQRFH2 üstbilgisi, MQRFH üstbilgisine dayalıdır; ancak, Unicode dizgilerinin çeviri olmadan taşınmasını sağlar ve sayısal veri tiplerini taşıyabilir.

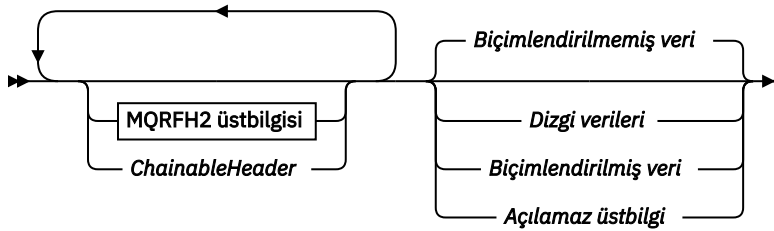
MQRFH2 yapısı, version-2 kuralları ve biçimlendirme üstbilgisinin biçimini tanımlar. XML benzeri bir sözdizimi kullanarak kodlanan verileri göndermek için bu üstbilgiyi kullanın. Bir ileti, serideki son MQRFH2 yapısından sonra isteğe bağlı olarak kullanıcı verileri ile seride iki ya da daha fazla MQRFH2 yapısı içerebilir.

Biçim Adı

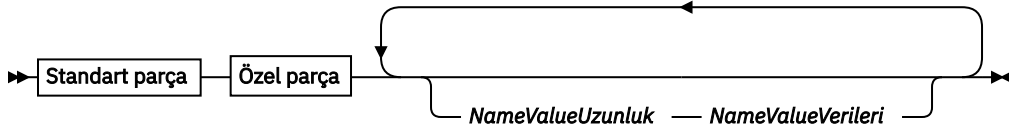
MQFMT_RF_HEADER_2

Sözdizimi

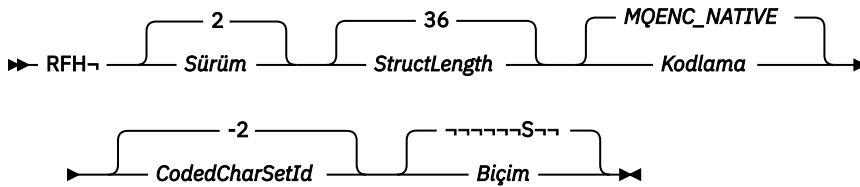
IBM MQ İleti



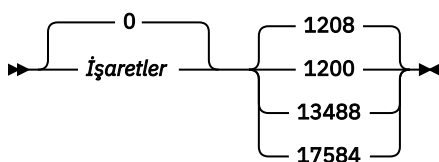
MQRFH2 üstbilgisi



Standart parça



Özel kısım



Karakter kümesi ve kodlama

Özel kurallar, MQRFH2 yapısı için kullanılan karakter kümesi ve kodlamaya uygulanır:

- Fields other than *NameValueData* are in the character set and encoding given by the *CodedCharSetId* and *Encoding* 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 MQGMO_CONVERT belirtildiğinde, kuyruk yöneticisi, *NameValueData* dışındaki MQRFH2 alanlarını istenen karakter kümesine ve kodlamaya dönüştürür.

- *NameValueData* , *NameValueCCSID* alanı tarafından verilen karakter kümesinde yer alıyor. Yalnızca listelenen Unicode karakter kümeleri *NameValueCCSID* için geçerlidir; Ayrıntılar için *NameValueCCSID* ' in açıklamalarına bakın.

Bazı karakter kümelerinin, kodlamaya bağlı bir gösterimi vardır. *NameValueCCSID* bu karakter kümelerinden biriye, *NameValueData* , MQRFH2' deki diğer alanlarla aynı kodlamada olmalıdır.

MQGET çağrısında MQGMO_CONVERT belirtildiğinde, kuyruk yöneticisi *NameValueData* ' yi istenen kodlamaya dönüştürür, ancak karakter kümesini değiştirmez.

MQRFH2alanları

MQRFH2 yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

CodedCharSetId (MQHOMER)

Bu, son *NameValueData* 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:

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_INHERIT olarak kabul edilir.

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.

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.

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.

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	UCS-2 açık uçlu
13488	UCS-2 2.0 altkümüsi
17584	UCS-2 2.1 altkümüsi (Euro simgesini içerir)
1208	UTF-8

UCS-2 karakter takımları için, *NameValueData* kodlamasının kodlaması (byte sırası), MQRFH2 yapısındaki diğer alanların kodlamasıyla aynı olmalıdır. Vekil karakterler (X'D800'-X'DFFF') desteklenmez.

Not: *NameValueCCSID* yukarıda listelenen değerlerden birine sahip değilse ve MQRFH2 yapısı MQGET çağrısına dönüştürülmesini gerektiriyorsa, çağrı neden kodu MQRC_SOURCE_CCSSID_ERROR ile tamamlanır ve ileti dönüştürülmez olarak döndürülür.

Bu alanın ilk değeri 1208 'dir.

NameValueVerileri (MQCHARn)

NameValueData, ileti özellikleri adlarını/değer çiftlerini içeren bir klasör içeren değişken uzunluklu bir alandır. Klasör, XML benzeri bir sözdizimi kullanılarak kodlanmış verileri içeren değişken uzunluklu bir karakter dizilimidir. Karakter dizisinin bayt cinsinden uzunluğu, *NameValueData* alanının önünde yer alan *NameValueLength* alanı tarafından verilir. Uzunluk, dörtten birden çok olmalıdır.

NameValueLength ve *NameValueData* alanları isteğe bağlıdır, ancak varsa çift olarak oluşmalıdır ve bitişik olmalıdır. Alan çifti, gerektiği kadar yinelenebilir; örneğin:

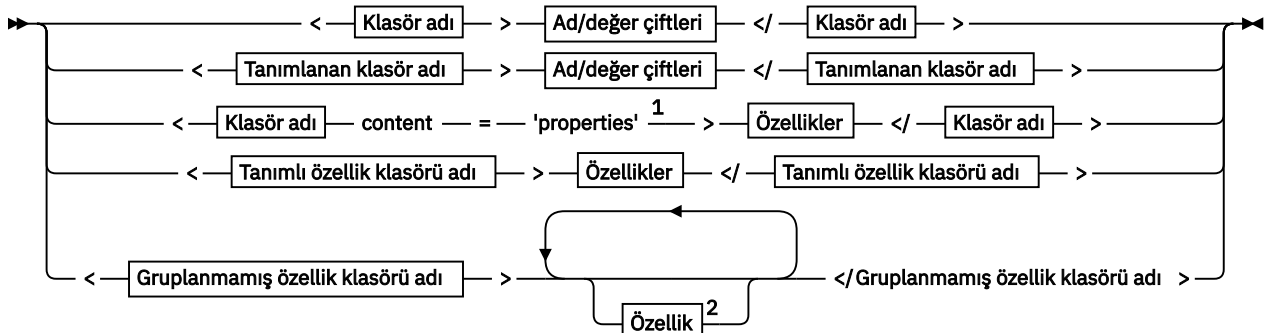
```
length1 data1 length2 data2 length3 data3
```

NameValueData, MQGET çağrısında belirtilen karakter kümesine dönüştürülmez. İleti, *NameValueData* etkisinde MQGMO_CONVERT seçeneğiyle alınsa bile, özgün karakter kümesinde kalır. Ancak, *NameValueData* is, MQGET çağrısında belirtilen kodlamaya dönüştürüldü.

Not:

- Bu alanlar isteğe bağlı olduğu için, 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 ayrılmış olduğu anlamına gelir.

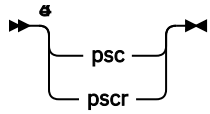
***NameValueData* sözdizimi**



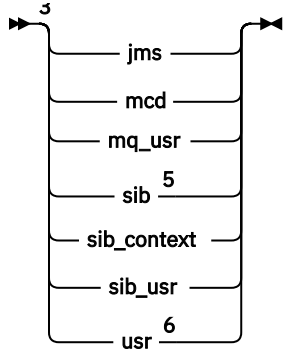
Klasör adı



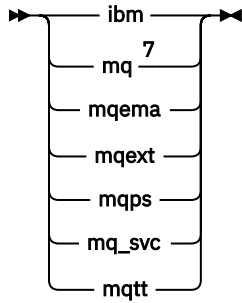
Tanımlanan klasör adı



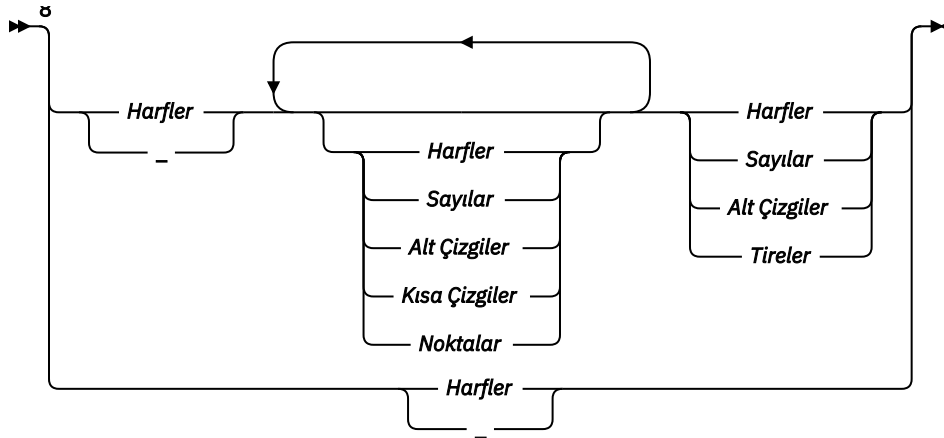
Tanımlı özellik klasörü adı



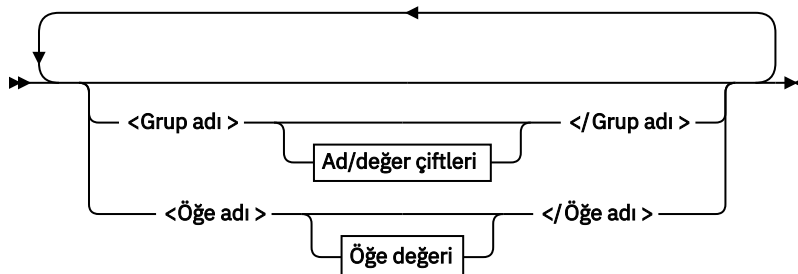
Gruplanmamış özellik klasörü adı



Ad



Ad/değer çiftleri



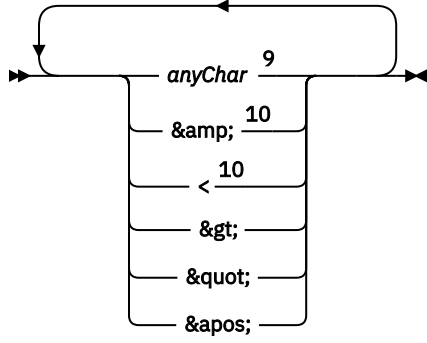
Grup adı

NAME

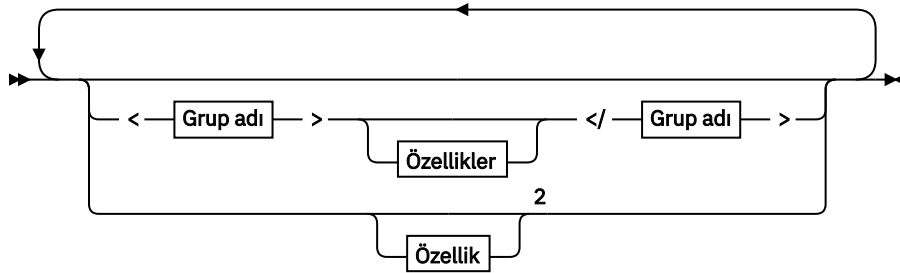
Öğe adı

NAME

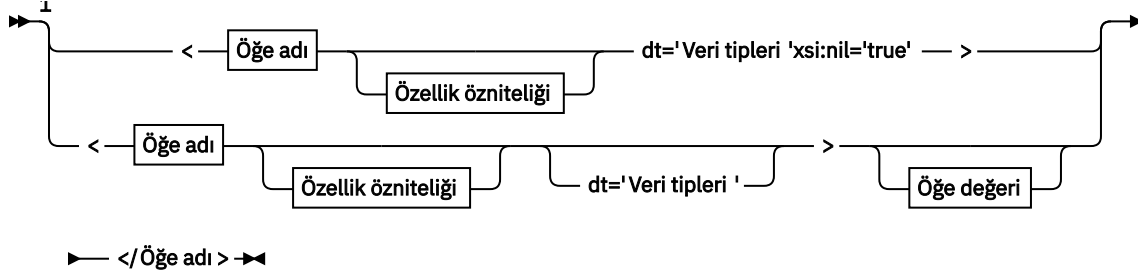
Öğe değeri



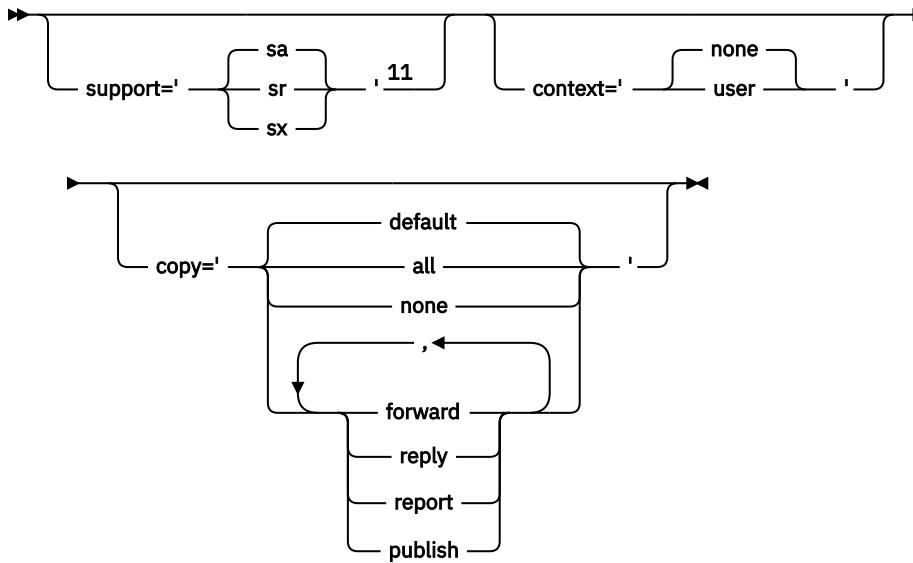
Özellikler



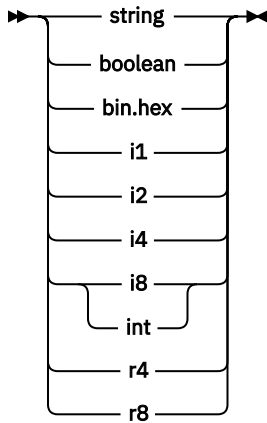
Özellik



Özellik özneliği



Veri tipleri



Notlar:

- ¹ Çift tırnak işaretleri ya da tek tırnak işaretleri geçerlidir.
- ² Geçersiz bir özellik adı kullanmayın; bkz. “Özellik adı geçersiz” sayfa 525. Ayrılmış bir özellik adını yalnızca tanımlı amacı için kullanın; bkz. “Tanımlı özellik adları” sayfa 525.
- ³ Adın küçük harfli olması gerekir.
- ⁴ Yalnızca bir psc ve pscı klasörü desteklenmektedir.
- ⁵ Yalnızca ilk MQRFH2 üstbilgisindeki özellikler önemlidir. WebSphere Application Server Service Integration Bus , sonraki MQRFH2 üstbilgilerinde sib, sib_contextve sib_usr klasörlerini yoksayar.
- ⁶ Bir MQRFH2 içinde birden fazla usı klasörü bulunmalı. usı klasöründeki özellikler bir kereden fazla geçmemelidir.
- ⁷ Yalnızca ilk mq klasöründeki özellikler önemlidir. Klasör UTF-8 ise, yalnızca tek baytlık UTF-8 karakterleri desteklenir. Tek beyaz alan karakteri Unicode U+0020' dir.
- ⁸ Valid characters are defined in the W3C XML specification, and consist essentially of Unicode categories Ll, Lu, Lo, Lt, Nl, Mc, Mn, Lm, and Nd ; see [Unicode karakter kategorileri](#).
- ⁹ Tüm karakterler önemlidir. Baştaki ve sondaki boşluklar, öge değerinin bir parçasıdır.
- ¹⁰ Geçersiz bir karakter kullanmayın; bkz. “Geçersiz karakterler” sayfa 525. Bu geçersiz karakterler yerine bir kaçış dizisi kullanın.
- ¹¹ Destek özelliği özneteliği yalnızca mq klasöründe geçerlidir.

Klasör adı

NameValueData , tek bir klasör içerir. Birden çok klasör yaratmak için birden çok *NameValueData* alanı yaratın. Bir ileti içinde tek bir MQRFH2 üstbilgisinde birden çok *NameValueData* alanı oluşturabilirsiniz. Diğer bir seçenek olarak, her biri birden çok *NameValueData* alanı içeren birden çok zincirleme MQRFH2 üstbilgisi yaratabilirsiniz.

MQRFH2 üstbilgilerinin sırası ve *NameValueData* alanlarının sırası, bir klasörün mantıksal içeriğiyle hiçbir farklılık göstermez. Bir iletide aynı klasör bir kereden fazla varsa, klasör bir bütün olarak ayrıştırılır. Aynı özellik aynı klasörün birden çok örneğinde ortaya çıkarsa, bu özellik bir liste olarak ayrıştırılır.

A correct parse of an MQRFH2 is not affected by the alternative ways a folder can be physically stored in a message.

Dört klasör bu kuralı izlemez. Yalnızca mq, sib, sib_context ve sib_usr klasörlerinin ilk örneği ayrıştırılır.

Aynı özellik, zincirlenmiş MQRFH2 üstbilgilerinin birleşik içeriğindeki birden çok kez oluşursa, özelliğin yalnızca ilk örneği ayrıştırılır. Bir özellik, MQSETMP gibi bir API çağrısı kullanılarak ayarlanır ve doğrudan bir uygulama tarafından MQRFH2 'a eklenirse, API çağrısı öncelikli olur.

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 515](#). Bir grup adını ve bir öge adını birleştirmeyin; bkz. [Şekil 2 sayfa 515](#)

```
<group1><nvp1>value</nvp1></group1><group2><nvp2>value</nvp2></group2>
<group3><nvp1> value </nvp1></group3><nvp3> value </nvp3>
```

Şekil 1. Grup ve ad/değer çiftlerinin doğru kullanımları

```
<group1><nvp1>value</nvp1>value</group1>
```

Şekil 2. Grup ve ad/değer çiftlerinin yanlış kullanımı

Geçersiz ya da ayrılmış bir klasör adı kullanmayın; bkz. “Yol adı geçersiz” sayfa 525 ve “Ayrılmış klasör ya da özellik klasörü adı” sayfa 524. Tanımlı bir klasör adını yalnızca tanımlı amaç için kullanın; bkz. “Tanımlanan klasör adı” sayfa 516.

Klasör adı etiketine 'content=properties' özniteliğini eklerseniz, klasör bir özellik klasörü olur; bkz. [Şekil 3 sayfa 515](#).

```
<myFolder></myFolder>
<myPropertyFolder contents='properties'></myPropertyFolder>
```

Şekil 3. Bir klasör ve bir özellik klasörü örneği

Klasör adları büyük ve küçük harfe duyarlıdır. Klasör adları ve özellik klasörü adları aynı ad alanını paylaşır. Farklı isimleri olmalı. [Şekil 4 sayfa 515](#) içindeki Folder1 , [Şekil 5 sayfa 516](#) içinde Folder2 olarak 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 516](#) içindeProperty1 , [Şekil 6 sayfa 516](#) içindeki Property1 için farklı bir özeldir.

```
<Folder3 content='properties'><Property1>value</Property1></Folder3>
```

Şekil 6. Folder3 ad alanı

Özellik klasörleri, iki önemli bakımdan özellik dışı klasörlerden farklıdır:

1. Özellik klasörleri özellikler içerir ve özellik dışı klasörler ad/değer çiftleri içerir. Klasörler biraz farklı, sözdizimsel olarak.
2. İleti özelliklerine erişmek için, örneğin, MQI ya da JMS ileti özellikleri gibi tanımlı arabirimleri kullanın. Arabirimler, MQRFH2 içindeki özellik klasörlerinin düzgün biçimlendirildiğinden emin olur. Düzgün biçimlendirmiş bir özellik klasörü, farklı platformlardaki ve farklı yayın düzeylerindeki kuyruk yöneticileri arasında birlikte çalışabilen bir klasördür.

MQI ileti özelliği, bir MQRFH2'yi okumak ve yazmak için sağlam bir yoldur ve bir MQRFH2 ' ı doğru bir şekilde ayrıştırmaya ilişkin zorlukları önler.

Tanımlanan klasör adı

Tanımlı klasör adı, IBM MQtarafından kullanılmak üzere ayrılmış bir klasörün ya da başka bir ürün adıdır. Aynı adı içeren bir klasör oluşturmayın ve klasörlere kendi ad/değer çiftlerinizi eklemeyin. Tanımlı klasörler psc ve psc1 ' dir.

psc ve psc1 , kuyruğa alınan yayınlama/abone olma yoluyla kullanılır.

MQMF_SEGMENT ya da MQMF_SEGMENTATION_ALLOWED ile bölümlenmiş bir ileti, tanımlı bir klasör adına sahip MQRFH2 içeremez. MQPUT , neden kodu 2443, MQRC_SEGMENTATION_NOT_ALLOWEDile başarısız olur.

Tanımlı özellik klasörü adı

Tanımlı özellik klasörü adı, IBM MQtarafından ya da başka bir ürün tarafından kullanılan bir özellik klasörünün adıdır. Klasörlerin ve içindekilerin adları için bkz. [Özellik klasörleri](#). Tanımlı özellik klasörü adları, IBM MQtarafından ayrılan tüm klasör adlarının bir alt kümedir; bkz. [“Ayrılmış klasör ya da özellik klasörü adı” sayfa 524](#).

Tanımlı bir özellik klasöründe saklanan herhangi bir öge bir özeldir. Tanımlı bir özellik klasöründe saklanan bir ögenin content='properties' özniteliği bulunmamalıdır.

Özellikleri yalnızca tanımlı özellik klasörlerine usr , mq_usr ve sib_usr ' a ekleyebilirsiniz. mq ve sib gibi diğer özellik klasörlerinde, IBM MQ tanımadığı ya da algılamadığı özellikleri yok sayar.

The description of each defined property folder lists the properties that IBM MQ has defined that can be used by application programs. Bazı özelliklere dolaylı olarak bir JMS özelliği ayarlanarak ya da alınırken dolaylı olarak erişilir ve bazıları doğrudan MQSETMP ve MQINQMP MQI çağrılarını kullanarak erişilir.

Tanımlı özellik klasörleri, IBM MQ ' un ayrılmış olduğu, ancak uygulamaların erişimi olmayan diğer özellikleri de içerir. Ayrılmış özelliklerin adları listelenmez. usr , mq_usr ve sib_usr özellik klasörlerinde ayrılmış özellikler yoktur. Ancak geçersiz özellik adları içeren özellikler oluşturmayın; bkz. [“Özellik adı geçersiz” sayfa 525](#).

Özellik klasörleri

jms

jms , JMS üstbilgi alanlarını ve MQMD' da tam olarak ifade edilemeyen JMSX özelliklerini içerir. jms klasörü her zaman bir JMS MQRFH2içinde bulunur.

Çizelge 71. jms özellik adı, eşanlamlı, veri tipi ve klasör			
Özellik eşanlamlı	Özellik adı	Veri türü	Klasör
JMStinati on	jms.Dst	string	<jms><Dst>destination</Dst></jms>
JMSSüresi	jms.Exp	i8	<jms><Exp>expiration</Exp></jms>
JMSCor	jms.Cid	string	<jms><Cid>correlationId</Cid></jms>
JMSTeslim i	jms.Dlv	i4	<jms><Dlv>delivery</Dlv></jms>
JMSÖnceli k	jms.Pri	i4	<jms><Pri>priority</Pri></jms>
JMSReplyT o	jms.Rto	string	<jms><Rto>replyToURI</Rto></jms>
JMSTimeSt amp	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, JMSBytesMessage, JMSStreamMessage, JMSMapMessage, JMSObjectMessageya da boş değerli olarak tanımlar.

mcd klasörü her zaman, MQRFH2içeren bir JMS iletisinde bulunur.

Bu, her zaman IBM Integration Bus' tan gönderilen bir MQRFH2 iletisi içeren bir iletide bulunur. Bu belge, bir iletinin etki alanını, biçimini, tipini ve ileti kümesini açıklar.

Çizelge 72. mcd özellik adı, eşanlamlı, veri tipi ve klasör			
Özellik eşanlamlı	Özellik adı	Veri türü	Klasör
	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 özellikler olarak gösterilmeyen uygulama tanımlı özellikleri içerir. JMS gereksinimlerini karşılamayan özellikler bu klasöre yerleştirilebilir.

You can create properties in the mq_usr folder. mq_usr içinde oluşturduğunuz özellikler, content='properties' öznitelikle yeni klasörlerde oluşturduğunuz özellikler gibidir.

sib

sib , WebSphere Application Server hizmet bütünleştirme veriyolu (WAS/SIB) sistem iletisi özelliklerini içerir. sib properties are not exposed as JMS properties to IBM MQ JMS applications because they are not of the supported types. Örneğin, bazı sib özellikleri byte dizileri olduğu için JMS özellikleri olarak gösterilemez. Some sib properties are exposed to WAS/SIB applications as JMS_IBM_* properties; these include forward and reverse routing paths properties.

sib klasörüne kendi özelliklerinizi eklemeyin.

sib_context

sib_context , WAS/SIB kullanıcı uygulamalarında ya da JMS özellikleri olarak gösterilmeyen WAS/SIB sistem iletisi özelliklerini içerir. sib_context , web hizmetleri için kullanılan güvenlik ve işlemsel özellikleri içerir.

sib_context klasörüne kendi özelliklerinizi eklemeyin.

sib_usr

sib_usr , desteklenen tiplerden olmadıkları için, JMS kullanıcı özellikleri olarak gösterilmeyen WAS/SIB kullanıcı iletisi özelliklerini içerir. sib_usr , SIMessage arabirimindeki WAS/SIB uygulamalarıyla gösterilir. [Developing Service Integration](#) konusuna bakın.

Bir sib_usr özelliğinin tipi bin . hex olmalı ve değer doğru biçimde olmalıdır. If an IBM MQ application writes a bin . hex typed element to the folder in the wrong format, the application receives an IOException. If the data type of the property is not bin . hex the application receives a ClassCastException.

Bu klasörü kullanmak yerine, JMS kullanıcı özelliklerini WAS/SIB için kullanıma hazırlamaya çalışmayın; bunun yerine usr klasörünü kullanın.

You can create properties in the sib_usr folder.

usr

usr , iletiyle ilişkili uygulama tanımlı JMS özelliklerini içerir. usr klasörü yalnızca, uygulama için uygulama tanımlı bir özellik ayarlandıysa bulunur.

usr , varsayılan özellik klasördür. Bir özellik, klasör adı olmadan ayarlanmışsa, bu özellik usr klasörüne konur.

<i>Çizelge 73. usr özellik adı, eşanlamlı, veri tipi ve klasör.</i>			
Web hizmetleri özellik değerleri MQRFH2 SOAP ayarları içinde açıklanmıştır.			
Özellik eşanlamlı	Özellik adı	Veri türü	Klasör
	usr.contentType	string	<usr><contentType>text/xml; charset=utf-8</contentType></usr>
	usr.endpointURL	string	<usr><endpointURL>URI</endpointURL></usr>

Çizelge 73. *usr* özellik adı, eşanlamlı, veri tipi ve klasör.

Web hizmetleri özellik değerleri [MQRFH2 SOAP ayarları](#) içinde açıklanmıştır.

(devamı var)

Özellik eşanlamlı	Özellik adı	Veri türü	Klasör
	usr.targetService	string	<usr><targetService>serviceName</targetService></usr>
	usr.soapAction	string	<usr><soapAction>name</soapAction></usr>
	usr.transportVersion	string	<usr><transportVersion>version</transportVersion></usr>

You can create properties in the *usr* folder.

MQMF_SEGMENT ya da MQMF_SEGMENTATION_ALLOWED ile bölümlenmiş bir ileti, tanımlı bir özellik klasörü adı olan bir MQRFH2 içeremez. MQPUT , neden kodu 2443, MQRC_SEGMENTATION_NOT_ALLOWED ile başarısız olur.

Gruplanmamış özellik klasörü adı

ibm

ibm , yalnızca IBM MQ tarafından kullanılan özellikleri içerir.

Çizelge 74. *ibm* özellik adı, eşanlamlı, veri tipi ve klasör

Özellik eşanlamlı	Özellik adı	Veri türü	Klasör
	ibm.rfp	string	<ibm><rfp>fingerprint</rfp></ibm>

ibm klasörüne kendi özelliklerinizi eklemeyin.

mq

mq , yalnızca IBM MQ tarafından kullanılan özellikleri içerir.

mq klasöründeki özellikler için aşağıdaki kısıtlamalar geçerlidir:

- Yalnızca, iletteki ilk önemli mq klasöründeki özellikler MQ tarafından üzerinde hareket edilir; iletteki diğer mq klasörlerindeki özellikler yoksaılır.
- 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.
- Klasör içinde çıkış dizgileri kullanmayın. Çıkış dizgisi, ögenin gerçek değeri olarak işlem görür.
- Yalnızca Unicode karakter U+0020 , klasör içinde beyaz alan olarak kabul edilir. 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 gözlemlemiyorsa, ileti 2527, MQRC_RFH_RESTRICTED_FORMAT_ERRneden koduyla reddedilir.

mq klasörüne kendi özelliklerinizi eklemeyin.

mqema

mqema , yalnızca WebSphere Application Server tarafından kullanılan özellikleri içerir. Klasör, mqext ile değiştirildi.

mqema klasörüne kendi özelliklerinizi eklemeyin.

mqext

mqext aşağıdaki özellik tiplerini içerir:

- Yalnızca WebSphere Application Server tarafından kullanılan özellikler.
- İletilerin gecikmeli olarak teslimi ile ilgili özellikler.

Uygulama, IBM tanımlı özelliklerden en az birini ayarlarsa ya da teslim gecikmesi olarak ayarlandıysa, klasör vardır.

Çizelge 75. mqext özellik adı, eşanlamlı, veri tipi ve klasör			
Özellik eşanlamlı	Özellik adı	Veri türü	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 yoluyla kullanılan özellikleri içerir. Bu klasör, yalnızca uygulama tümleşik yayınlama/abone olma özelliklerinden en az birini ayarladıysa var olur.

Çizelge 76. mqps özellik adı, eşanlamlı, veri tipi ve klasör			
Özellik eşanlamlı	Özellik adı	Veri türü	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>
MQPubTime	mqps.Pts	string	<mqps><Pts>publicationTime</Pts></mqps>
MQPubSeqNum	mqps.Seq	i8	<mqps><Seq>publicationSequenceNumber</Seq></mqps>
MQPubStrIntData	mqps.Sid	string	<mqps><Sid>publicationData</Sid></mqps>

Çizelge 76. mqps özellik adı, eşanlamlı, veri tipi ve klasör (devamı var)			
Özellik eşanlamlı	Özellik adı	Veri türü	Klasör
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.

mqtt

mqtt , IBM MQ Telemetrytarafından kullanılan özellikleri içerir.

Çizelge 77. mqtt özellik adı, eşanlamlı, veri tipi ve klasör			
Özellik eşanlamlı	Özellik adı	Veri türü	Klasör
	mqtt.clientId	string	<mqtt><clientId>topicString</clientId></mqtt>
	mqtt.qos	i4	<mqtt><qos>qualityOfService</qos></mqtt>
	mqtt.msgid	string	<mqtt><msgid>messageIdentifier</msgid></mqtt>

mqtt klasörüne kendi özelliklerinizi eklemeyin.

MQMF_SEGMENT ya da MQMF_SEGMENTATION_ALLOWED ile belirtilen bölümlenmiş bir ileti, gruplanmamış bir özellik klasörü adı olan bir MQRFH2 içeremez. MQPUT , neden kodu 2443, MQRC_SEGMENTATION_NOT_ALLOWEDile başarısız olur.

Ad/değer çiftleri

Sözdizimi şemasında, "Ad/değer çiftleri" olağan bir klasörün içeriğini açıklar. Sıradan bir klasör gruplar ve öğeler içerir. Bir öğe, bir ad/değer çiftidir. Bir grup, öğeleri ve diğer grupları içerir.

Ağaçlar, öğeler yaprak düğümlerdir ve gruplar iç düğümlerdir. Bir iç düğüm ve kök düğüm olan klasör, iç düğümlerin ve yaprak düğümlerin bir karışımının içermesini içerebilir. A node cannot be both an internal node and a leaf node at the same time; see [Şekil 2 sayfa 515](#).

Özellikler

Sözdizimi şemasında "Özellikler" , bir özellik klasörünün içeriğini açıklar. Bir özellik klasörü gruplar ve özellikler içerir. Özellik, isteğe bağlı veri tipi özniteliği olan bir ad/değer çiftidir. Bir grup, özellikleri ve diğer grupları içerir.

Ağaç bakımından, özellikler yaprak düğümlerdir ve gruplar iç düğümlerdir. Bir iç düğüm ve kök düğüm olan özellik klasörü, iç düğümlerin ve yaprak düğümlerin bir karışımının içermesini içerebilir. A node cannot be both an internal node and a leaf node at the same time; see [Şekil 2 sayfa 515](#).

Özellik

Bir ileti özelliği, özellik klasöründeki bir ad/değer çiftidir. İsteğe bağlı olarak, bir veri tipi özniteliği ve bir özellik özniteliği de içerilebilir; örneğin, aşağıdaki kodu görebilirsiniz. Veri tipi özniteliği atılırsa, özellik tipi `string` olur.

```
<pf><p1 dt='i8' > value </p1></pf>
```

Bir ileti özelliğinin adı, XML benzeri, `<>` sözdizimiyle, noktalarla değiştirilen tam yol adıdır. Örneğin, `myPropertyFolder1.myGroup1.myGroup2.myProperty1` aşağıdaki gibi bir `NameValueData` dizisiyle eşlenir. Daha kolay okuma için dize biçimlendirilir.

```
<myPropertyFolder1>
  <myGroup1>
    <myGroup2>
      <myProperty1>value</myProperty1>
    </myGroup2>
  </myGroup1>
</myPropertyFolder1>
```

Bir özellik klasörü birden çok özellik içerebilir. Örneğin, Şekil 7 sayfa 522 içindeki özellikler, Şekil 8 sayfa 522 içindeki özellik klasörüne eşlenir.

```
myPropertyFolder1.myProperty4
myPropertyFolder1.myGroup1.myGroup2.myProperty1
myPropertyFolder1.myGroup1.myGroup2.myProperty2
myPropertyFolder1.myGroup1.myProperty3
```

Şekil 7. Kök adı aynı olan 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

Ad, bir *Letter* ya da *Underscore* ile başlamalıdır. It must not contain a *İki nokta*, not end in a *Dönem* and contain only *Harfler*, *Sayısallar*, *Alt çizgiler*, *Kısa çizgiler* ve *Noktalar*. Valid characters are defined in the W3C XML specification, and consist essentially of Unicode categories L1, Lu, Lo, Lt, N1, Mc, Mn, Lm, and Nd ; see [Unicode karakter kategorileri](#).

Bir özelliğin ya da ad/değer çiftinin tam yolu, “Yol adı geçersiz” sayfa 525 içinde açıklanan kuralı bozmamalıdır. Paths are restricted to 4095 bytes, must not contain Unicode compatibility characters, and must not start with the string XML.

Grup adı

Grup adı, bir adla aynı sözdizimine sahiptir. Grup adları isteğe bağlıdır. Özellikler ve ad/değer çiftleri, bir klasörün köküne yerleştirilebilir. Özellikleri ve ad/değer çiftlerini düzenlemeye yardımcı olursa, grupları kullanın.

Öğe adı

Bir öğe adı, bir adla aynı sözdizimine sahip.

Öğe değeri

Bir öğe değeri, < *Element name* > etiketi ile < */Element name* > arasındaki tüm beyaz alanı içerir. Do not use the two characters < and & in a value. Daha sonra, < ve & ile değiştirin.

Özellik öznitelikleri

Özellik öznitelikleri eşleme özelliği tanımlayıcı alanları: Eşlemeler aşağıdaki gibidir:

Destek

<u>sa</u>	MQPD_SUPPORT_OPTIONAL
sr	MQPD_SUPPORT_REQUIRED
sx	MQPD_SUPPORT_REQUIRED_IF_LOCAL

Bağlam

<u>none</u>	MQPD_NO_CONTEXT
user	MQPD_USER_CONTEXT

CopyOptions

forward	MQPD_COPY_FORWARD
reply	MQPD_COPY_REPLY
report	MQPD_COPY_REPORT
publish	MQPD_COPY_PUBLISH
all	MQPD_COPY_ALL
all ' u diğer seçeneklerle birlikte kullanmayın.	
default	MQPD_COPY_DEFAULT
default ' u diğer seçeneklerle birlikte kullanmayın. default , forward + report + publish ile aynıdır.	
none	MQPD_COPY_NONE
none ' u diğer seçeneklerle birlikte kullanmayın.	

Destek özelliği öznitelikleri yalnızca mq klasöründeki özellikler için geçerlidir.

Bağlam ve CopyOptions özellik öznitelikleri tüm özellik klasörleri için geçerli olur.

Veri tipleri

MQRFH2 veri tipleri, ileti özelliği tiplerinde aşağıdaki gibi eşlenir:

Çizelge 78. Veri tipi eşlemeleri

MQRFH2 veri tipi	İleti özelliği tipi
bin.hex	MQBYTE[]
boolean	MQB00L
i1	MQINT8
i2	MQINT16
i4	MQINT32
i8	MQINT64
r4	MQFLOAT32
r8	MQFLOAT64
string	MQCHAR[]

Veri tipi olmayan herhangi bir ögenin, stringtipinde olduğu varsayılır.

Boş değer, `xsi:nil='true'` öge özniteyle gösterilir. Boş değer olmayan değerler için `xsi:nil='false'` özniteliğini kullanmayın. Örneğin, aşağıdaki özelliğin boş bir değeri vardır:

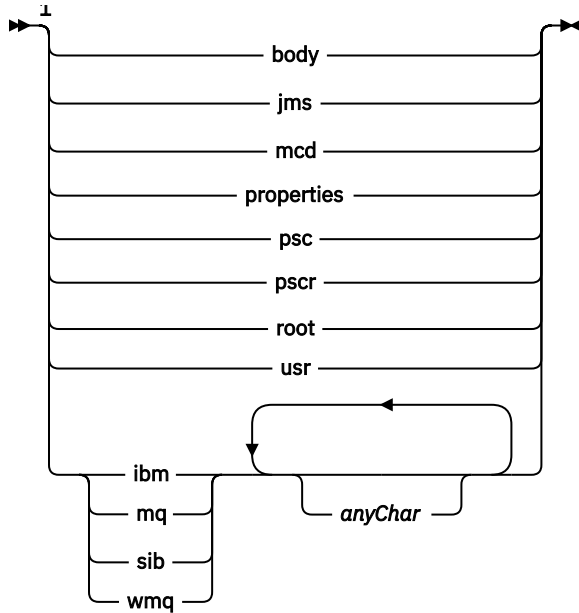
```
<NullProperty
  xsi:nil='true'></NullProperty>
```

Bir bayt ya da karakter dizilimi özelliği boş bir değere sahip olabilir. Boş bir değer, sıfır uzunluklu bir öge değerine sahip bir MQRFH2 ögesi tarafından temsil edilir. Örneğin, aşağıdaki özelliğin boş bir değeri vardır:

```
<EmptyProperty></EmptyProperty>
```

Ayrılmış klasör ya da özellik klasörü adı

Bir klasör ya da özellik klasörünün adını, aşağıdaki dizgilerden biriyle başlamayacak şekilde sınırlayın. Önekler, IBM tarafından oluşturulan klasör ya da özellik adları için ayrılmıştır.

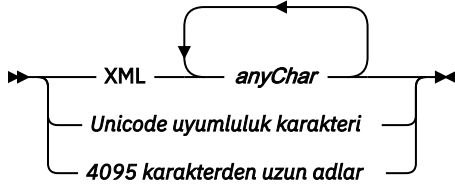


Notlar:

¹ Ayrılmış bir klasör ya da özellik adı alt ve büyük harflerin herhangi bir karışımının içermesini içerir.

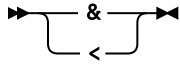
Yol adı geçersiz

Bir ad/değer çiftinin ya da bir özelliğin tam yolunu izleyen dizgilerden hiçbirini içermeyecek şekilde sınırlayın.



Geçersiz karakterler

Always use the escape sequences & and < instead of the literals "&" and "<".

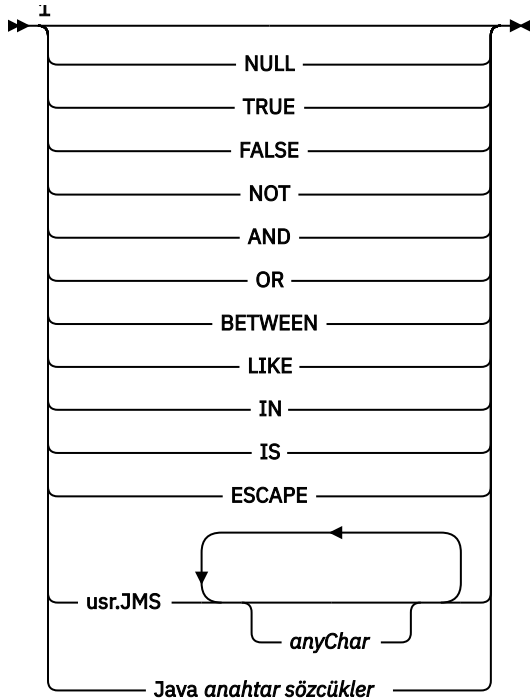


Tanımlı özellik adları

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. Tanımlı özellik adları, özellik klasörlerinin tanımlamasında açıklanır; bkz. [Özellik klasörleri](#).

Özellik adı geçersiz

Aşağıdaki kuralla eşleşen özellik adlarını oluşturmayın. Kural, yalnızca özellik ögesi adı için değil, bir özelliği adlara alan tam özellik yolu için geçerlidir.



Notlar:

¹ Geçersiz bir özellik adı, üst ve küçük harflerin herhangi bir birleşimini içerebilir.

Öznitelikler geçersiz

Özellik klasörleri ve özellikleri yalnızca desteklenen “Özellik öznitelikleri” sayfa 523 ve “Veri tipleri” sayfa 523 özelliklerini içerebilir.

Desteklenmeyen XML benzeri öznitelikler (örneğin, tırnak içine alınmış dizgi değerlerine sahip adlar), özellik klasörlerinde ya da özelliklerde yer alan adlar kaldırılabilir.

XML-like attributes included in non-property folders or non-property elements that remain in MQRFH2 headers.

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.

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.

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 verilerini *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.

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.

MQRFH2 için ilk değerler ve dil bildirimleri

Çizelge 79. Initial values of fields in MQRFH2 for MQRFH2		
Alan adı	Değişmezin adı	Değişmezin değeri
StrucId	MQRFH_STRUC_ID	'RFH↵'
Version	MQRFH_VERSION_2	2
StrucLength	MQRFH_STRUC_LENGTH_FIXED_2	36
Encoding	MQENC_NATIVE	Ortama göre değişir
CodedCharSetId	MQCCSI_INHERIT	-2
Format	MQFMT_NONE	Boşluklar
Flags	MQRFH_NONE	0
NameValueCCSID	Yok	1208
Notlar:		
1. ↵ simgesi tek bir boş karakteri temsil eder.		
2. C programlama dilinde, makro değişkeni MQRFH2_DEFAULT , çizelgede listelenen değerleri içerir. Yapıdaki alanlara ilişkin ilk değerleri sağlamak için bunu aşağıdaki şekilde kullanın:		
<pre>MQRFH2 MyRFH2 = {MQRFH2_DEFAULT};</pre>		

C bildirimi

```
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 */
};
```

COBOL bildirimi

```
** 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.

```

PL/I bildirimi

```

dcl
  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 */

```

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)	

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

```

MQRMH-Başvuru iletisi üstbilgisi

Aşağıdaki çizelge yapıdaki alanları özetlemektedir.

Çizelge 80. MQRMH ' deki alanlar

Alan	Tanım	Konu
<i>StrucId</i>	Yapı tanıtıcısı	StrucId
<i>Version</i>	Yapı sürüm numarası	S\u00fcr\u00fcm
<i>StrucLength</i>	Sabit alanların sonunda dizgiler de dahil olmak üzere MQRMH ' nin toplam uzunluğu, ancak toplu veriler değil	StrucLength
<i>Encoding</i>	Toplu verilerin sayısal kodlaması	Kodlama
<i>CodedCharSetId</i>	Toplu verilerin karakter kümesi tanıtıcısı	CodedCharSetId
<i>Format</i>	Toplu verilerin biçim adı	Biçim
<i>Flags</i>	Başvuru iletisi işaretleri	İşaretler
<i>ObjectType</i>	Nesne tipi	ObjectType
<i>ObjectInstanceId</i>	Nesne eşgörünümü tanıtıcısı	ObjectInstanceTanıtıcısı
<i>SrcEnvLength</i>	Kaynak ortam verilerinin uzunluğu	SrcEnvUzunluğu
<i>SrcEnvOffset</i>	Kaynak ortam verilerinin görelî konumu	SrcEnvKayma
<i>SrcNameLength</i>	Kaynak nesne adı uzunluğu	SrcNameUzunluk
<i>SrcNameOffset</i>	Kaynak nesne adının görelî konumu	SrcNameGörelî Konumu
<i>DestEnvLength</i>	Hedef ortam verilerinin uzunluğu	DestEnvUzunluğu
<i>DestEnvOffset</i>	Hedef ortam verilerinin görelî konumu	DestEnvGörelî Konumu
<i>DestNameLength</i>	Hedef nesne adı uzunluğu	DestNameLength
<i>DestNameOffset</i>	Hedef nesne adının görelî konumu	DestNameGörelî Konumu
<i>DataLogicalLength</i>	Toplu verilerin uzunluğu	DataLogicalLength
<i>DataLogicalOffset</i>	Toplu verilerin düşük görelî konumu	DataLogicalGörelî Konumu
<i>DataLogicalOffset2</i>	Toplu verilerin yüksek görelî konumu	DataLogicalOffset2

MQRMH ' ye Genel Bakış

Kullanılabilirlik: AIX, HP-UX, IBM i, Solaris, Linux, Windowsve ayrıca bu sistemlere bağlı IBM MQ istemcileri.

Amaç: MQRMH yapısı, başvuru iletisi üstbilgisinin biçimini tanımlar. Bu üstbilgi, çok 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, MQ kaynaklarının küçük sayıda son derece büyük ileti tarafından tükenme olasılığını azaltır.

Biçim adı: MQFMT_REF_MSG_HEADER.

Karakter kümesi ve kodlama: MQRMH ' de karakter verileri ve görelî konum alanları tarafından adreslenen dizgiler, yerel kuyruk yöneticisinin karakter kümesinde olmalıdır; bu, *CodedCharSetId* kuyruk yöneticisi özneteliği tarafından verilir. MQRMH ' deki sayısal veriler yerel makine kodlamasında olmalıdır; bu değer C programlama dili için MQENC_NATIVE değeri tarafından verilir.

MQRMH ' nin karakter kümesini ve kodlamasını *CodedCharSetId* ve *Encoding* alanlarına aşağıdaki şekilde kodlayın:

- 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 kanalı aracısı (MCA), ileti iletim kuyruğundan iletiyi okuduğunda, başvuru ileti üstbilgisini işlemek için kullanıcı tarafından sağlanan bir ileti çıkışı çağırılır. Çıkış, MQRMH yapısı tarafından tanımlanan toplu verileri, MCA iletiyi kanaldan bir sonraki kuyruk yöneticisine göndermeden önce, başvuru iletiğine sonuna eklenebilir.

Alıcı uçta, başvuru iletileri için bekleyen bir ileti çıkışı var olmalıdır. Bir başvuru ileti alındığında, çıkış, iletteki MQRMH ' yi izleyen toplu verilerden nesneyi yaratmalı ve daha sonra, toplu veriler olmadan başvuru iletiğine geçmelidir. Başvuru ileti daha sonra, bir kuyruktan başvuru ileti (toplu veriler olmadan) okunarak bir uygulama tarafından alınabilir.

Olağan durumda, MQRMH yapısı iletilerde yer alan tüm yapıdır. Ancak, ileti bir iletim kuyruğalıysa, MQRMH yapısından önce bir ya da daha çok ek üstten önce yer alan bir ya da daha çok ek üstbilgi vardır.

Ayrıca, bir dağıtım listesine bir başvuru ileti de gönderilebilir. Bu durumda, MQDH yapısı ve ilgili kayıtları, ileti bir iletim kuyruğunda olduğunda MQRMH yapısından önce gelir.

Not: İleti çıkışı doğru işleyemediğinden, bölümlenmiş ileti olarak bir başvuru ileti göndermeyin.

Veri dönüştürme: Veri dönüştürme amacıyla, MQRMH yapısının dönüştürülmesi, kaynak ortam verilerinin, kaynak nesne adının, hedef ortam verilerinin ve hedef nesne adının dönüştürülmesini içerir. Yapının başlangıcındaki *StrucLength* byte 'ındaki diğer baytlar atılır ya da veri dönüştürmesinden sonra tanımsız değerler içerilir. Toplu veriler, aşağıdakilerin tümünün doğru olması koşuluyla dönüştürülür:

- Toplu veriler, veri dönüştürme işlemi gerçekleştirildiğinde iletilerde bulunur.
- MQRMH ' deki *Format* alanının değeri MQFMT_NONE dışında bir değer içeriyor.
- Kullanıcı tarafından yazılan bir veri dönüştürme çıkışı, belirtilen biçim adıyla var olur.

Ancak, ileti kuyrukta olduğunda genellikle toplu verilerin *olmadığını* ve sonuç olarak, toplu verilerin MQGMO_CONVERT seçeneği tarafından dönüştürülmesini dikkate almayın.

MQRMH alanları

MQRMH yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

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

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 kullanmayın.

Bu değer şu ortamlarda desteklenir: AIX, HP-UX, IBM i, Solaris, Linux, Windows, ayrıca bu sistemlere bağlı IBM MQ istemcileri.

Bu alanın ilk değeri MQCCSI_UNDEFED değerine sahip.

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 görelî 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.

DataLogicalGörelî Konumu (MQUZE)

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, toplu verilerin MQRMH yapısının başlangıcından fiziksel olarak görelî olarak *değil* , görelî konum *StrucLength* tarafından verilir.

Büyük nesnelerin başvuru iletileri kullanılarak gönderilmesine izin vermek için, mantıksal görelî konum iki alana ayrılır ve gerçekte mantıksal görelî konum, bu iki alanın toplamını verir:

- *DataLogicalOffset* , mantıksal görelî konum 1 000 000 000 olarak bölündüğünde elde edilen kalanı temsil eder. Böylece, 0 ile 999 999 999 aralığında bir değer vardır.
- *DataLogicalOffset2* , 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.

DataLogicalOffset2 (MQUZE)

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. *DataLogicalOffset* .

Bu alanın ilk değeri 0 'tır.

DestEnvUzunluğu (MQUZE)

Bu, hedef ortam verilerinin uzunluğudur. Bu alan sıfırsa, hedef ortam verileri yoktur ve *DestEnvOffset* yoksayılr.

DestEnvGörelî Konumu (MQUZE)

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, 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* yoksayılr. 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* yoksayılr.

DestNameGörelî Konumu (MQUZE)

Bu alan, MQRMH yapısının başlangıcından hedef nesne adının kayma değerini belirtir. Hedef nesne adı, bu veriler oluşturucuda biliniyorsa, başvuru iletişinin 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* yoksayılır. 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.

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.

İşaretler (MQUZE)

Bunlar, başvuru ileti işaretleridir. Aşağıdaki işaretler tanımlanır:

MQRMHF_SON

Bu işaret, başvuru iletişinin başvuruda bulunulan nesnenin son bölümünü temsil ettiğini ya da içerdiğini gösterir.

MQRMHF_NOT_SON

Başvuru iletişisi 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.

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.

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.

ObjectType (MQCHAR8)

Bu ad, ileti çıkışının desteklediği başvuru iletişisi tiplerini tanımak için kullanılabileceği bir addır. Ad, *Format* alanıyla aynı kurallara uygun olmalıdır, bkz. "[Biçim \(MQCHAR8\)](#)" sayfa 532.

Bu alanın ilk değeri 8 boşluktur.

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 değeri belirtir. Kaynak ortam verileri, bu veriler yaratan kişi tarafından biliniyorsa, başvuru iletişinin 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 değeri belirtir. Kaynak nesne adı, bu veriler oluşturucuda biliniyorsa, başvuru iletişinin 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* yoksayılr. 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.

StrucId (MQCHAR4)

Bu, yapı tanıtıcısıdır; değer şöyle olmalıdır:

MQRMH_STRUC_ID

Başvuru iletişi üstbilgi yapısına ilişkin tanıtıcı.

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.

StrucLength (MQUZE)

Sabit alanların sonundaki dizgiler de dahil olmak üzere MQRMH ' nin toplam uzunluğu, ancak toplu veriler değil.

Bu alanın ilk değeri sıfır.

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.

MQRMH için ilk değerler ve dil bildirimleri

Çizelge 81. MQRMH için MQRMH ' de alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
<i>StrucId</i>	MQRMH_STRUC_ID	' RMH↵ '
<i>Version</i>	MQRMH_VERSION_1	1
<i>StrucLength</i>	Yok	0
<i>Encoding</i>	MQENC_NATIVE	Ortama göre değişir
<i>CodedCharSetId</i>	MQCCSI_UNDEDINED	0
<i>Format</i>	MQFMT_NONE	Boşluklar
<i>Flags</i>	MQRMHF_NOT_SON	0
<i>ObjectType</i>	Yok	Boşluklar
<i>ObjectInstanceId</i>	MQOL_NONE	Boş Değerler
<i>SrcEnvLength</i>	Yok	0
<i>SrcEnvOffset</i>	Yok	0
<i>SrcNameLength</i>	Yok	0
<i>SrcNameOffset</i>	Yok	0
<i>DestEnvLength</i>	Yok	0
<i>DestEnvOffset</i>	Yok	0
<i>DestNameLength</i>	Yok	0
<i>DestNameOffset</i>	Yok	0
<i>DataLogicalLength</i>	Yok	0
<i>DataLogicalOffset</i>	Yok	0
<i>DataLogicalOffset2</i>	Yok	0
Notlar:		
1. ↵ simgesi tek bir boş karakteri temsil eder.		
2. C programlama dilinde, makro değişkeniMQRMH_VARSAYILAN, çizelgede listelenen değerleri içerir. Yapıdaki alanlara ilişkin ilk değerleri sağlamak için bunu aşağıdaki şekilde kullanın:		
<pre>MQRMH MyRMH = {MQRMH_DEFAULT};</pre>		

C bildirimi

```
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 */
};

```

COBOL bildirimi

```

**   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 data
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-DATALOGICALLength 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.

```

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 */

```

High Level Assembler 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_DATALOGICALLLENGTH	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)

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'


```

Encoding          As Long      'not the bulk data'
CodedCharSetId    As Long      'Numeric encoding of bulk data'
Format            As String*8   'Character set identifier of bulk data'
Flags             As Long      'Format name of bulk data'
ObjectType        As String*8   'Reference message flags'
ObjectInstanceId  As MQBYTE24   'Object type'
SrcEnvLength       As Long      'Object instance identifier'
SrcEnvOffset      As Long      'Length of source environment data'
SrcNameLength     As Long      'Offset of source environment data'
SrcNameOffset     As Long      'Length of source object name'
DestEnvLength     As Long      'Offset of source object name'
DestEnvOffset     As Long      'Length of destination environment'
DestNameLength    As Long      'data'
DestNameOffset    As Long      'Offset of destination environment'
DataLogicalLength As Long      'data'
DataLogicalOffset As Long      'Length of destination object name'
DataLogicalOffset2 As Long     'Offset of destination object name'
End Type          As Long      'Length of bulk data'
                  As Long      'Low offset of bulk data'
                  As Long      'High offset of bulk data'

```

MQRR-Yanıt kaydı

Aşağıdaki çizelge yapıdaki alanları özetlemektedir.

Çizelge 82. MQRR 'deki alanlar		
Alan	Tanım	Konu
CompCode	Kuyruğa ilişkin tamamlanma kodu	CompCode
Reason	Kuyruğun neden kodu	Neden

MQRR 'ye Genel Bakış

Kullanılabilirlik: AIX, HP-UX, IBM i, Solaris, Linux, Windowsve ayrıca bu sistemlere bağlı IBM MQ istemcileri.

Amaç: Hedef dağıtım listesi olduğunda, tek bir hedef kuyruğa ilişkin açma ya da koyma işleminden kaynaklanan tamamlanma kodunu ve neden kodunu almak için MQRR yapısını kullanın. MQRR, MQOPEN, MQPUT ve MQPUT1 çağrılarına ilişkin bir çıkış yapısıdır.

Karakter kümesi ve kodlama: MQRR içindeki veriler, *CodedCharSetId* kuyruk yöneticisi özniteliği tarafından verilen karakter kümesinde ve MQKEN_NATIVE tarafından verilen yerel kuyruk yöneticisinin kodlamasında olmalıdır. Ancak, uygulama bir MQ MQI 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, arama sonucunun karma olduğu durumlarda bir dağıtım listesindeki tüm kuyruklara ilişkin tamamlanma kodlarını ve neden kodlarını belirleyebilirsiniz. Bu durumda, çağrı listedeki bazı kuyruklar için başarılı olur, ancak diğerleri için başarısız olur. Çağrıdan MQRC_MULTIPLE_REAMANS neden kodu, yanıt kayıtlarının (uygulama tarafından sağlandıysa) kuyruk yöneticisi tarafından ayarlandığını gösterir.

MQRR için alanlar

MQRR yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

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.

MQRR için ilk değerler ve dil bildirimleri

Çizelge 83. MQRR için MQRR 'de alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
CompCode	MQCC_OK	0
Reason	MQRC_NONE	0
Notlar: 1. C programlama dilinde, makro değişkeniMQRR_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlara ilişkin ilk değerleri sağlamak için bunu aşağıdaki şekilde kullanın: <pre>MQRR MyRR = {MQRR_DEFAULT};</pre>		

C bildirimi

```
typedef struct tagMQRR MQRR;  
struct tagMQRR {  
    MQLONG CompCode; /* Completion code for queue */  
    MQLONG Reason; /* Reason code for queue */  
};
```

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.
```

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 */
```

Visual Basic bildirimi

```
Type MQRR  
CompCode As Long 'Completion code for queue'  
Reason As Long 'Reason code for queue'  
End Type
```

MQSCO-SSL yapılanışı seçenekleri

Aşağıdaki çizelge yapıdaki alanları özetlemektedir.

Çizelge 84. MQSCO içindeki alanlar

Alan	Tanım	Konu
<i>StrucId</i>	Yapı tanıtıcısı	StrucId
<i>Version</i>	Yapı sürüm numarası	S\u00f0fcr\u00f0cm
<i>KeyRepository</i>	Anahtar havuzunun yeri	KeyRepository
<i>CryptoHardware</i>	Şifreleme donanımın ayrıntıları	CryptoHardware
<i>AuthInfoRecCount</i>	Var olan MQAIR kayıt sayısı	AuthInfoRecCount
<i>AuthInfoRecOffset</i>	MQSCO başlangıcından ilk MQAIR kaydının kayması	AuthInfoRecOffset
<i>AuthInfoRecPtr</i>	İlk MQAIR kaydının adresi	AuthInfoRecPtr
Not: <i>Version</i> , MQSCO_VERSION_2 değerinden küçükse, şu iki alan yoksayılır.		
<i>KeyResetCount</i>	SSL gizli anahtar ilk duruma getirme sayısı	KeyResetSayı
<i>FipsRequired</i>	IBM MQ' ta FIPS onaylı şifreleme algoritmaları kullan	“FipsRequired (MQlong)” sayfa 541
Not: <i>Version</i> , MQSCO_VERSION_3 değerinden küçükse aşağıdaki alan yoksayılır.		
<i>EncryptionPolicySuiteB</i>	Yalnızca Suite B şifreleme algoritmalarını kullan	EncryptionPolicySuiteB
Not: <i>Version</i> , MQSCO_VERSION_4 değerinden küçükse aşağıdaki alan yoksayılır.		
<i>CertificateValPolicy</i>	Sertifika geçerlilik denetimi ilkesi	CertificateValİlkesi
Not: <i>Version</i> , MQSCO_VERSION_5 değerinden küçükse aşağıdaki alan yoksayılır.		
<i>CertificateLabel</i>	Kullanılmakta olan sertifika etiketini ayrıntılı bir şekilde içerir.	CertificateLabel

İlgili başvurular

[“MQCNO-Bağlantı seçenekleri” sayfa 314](#)

Aşağıdaki çizelge yapıdaki alanları özetlemektedir.

[“MQSCO ile ilgili genel bakış” sayfa 539](#)

Kullanılabilirlik: AIX, HP-UX, IBM i, Solaris, Linux ve Windows istemcileri.

[“MQSCO alanları” sayfa 540](#)

[“MQSCO için ilk değerler ve dil bildirimleri” sayfa 543](#)

MQSCO ile ilgili genel bakış

Kullanılabilirlik: AIX, HP-UX, IBM i, Solaris, Linux ve Windows istemcileri.

Amaç: MQSCO yapısı (MQCD yapısındaki SSL alanlarıyla birlikte), bir uygulamanın IBM MQ MQI client olarak çalışan bir uygulamanın, kanal protokolü TCP/IP olduğunda istemci bağlantısı için SSL kullanımını denetleyen yapılandırma seçeneklerini belirlemesine olanak sağlar. Yapı, MQCONNX çağrısına ilişkin bir giriş değişikliğidir.

İstemci kanalına ilişkin kanal protokolü TCP/IP değilse, MQSCO yapısı yoksayılır.

Karakter kümesi ve kodlama: MQSCO içindeki veriler, *CodedCharSetId* kuyruk yöneticisi özniteliği tarafından verilen karakter kümesinde ve MQKEN_NATIVE ile verilen yerel kuyruk yöneticisinin kodlamasında olmalıdır.

MQSCO alanları

MQSCO yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

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 270. 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 görelidir. Görelilik pozitif ya da negatif olabilir. *AuthInfoRecCount* sıfırsa, alan yoksayılır.

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* sıfırsa, alan yoksayılır.

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* ' i 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 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 *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 sıfırsa, arama neden kodu MQRC_AUTH_INFO_REC_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 dizisi olarak bildirilir.

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 `ibmwebspheremq<user_id>`.

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.

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.

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.

FipsRequired (MQlong)

IBM MQ , şifreleme donanımıyla, kullanılan şifreleme modüllerinin donanım ürünü tarafından sağlananlar olması için yapılandırılabilir; bunlar, kullanılmakta olan şifreleme donanımı ürününe bağlı olarak, belirli bir

düzeyle 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 bir SSL şifrelemesi 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 SSL kanalı bağlantıları yalnızca aşağıdaki şifre belirtimlerinden 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, MQRC_SSL_INITIALIZATION_ERROR ile FIPS dışı CipherSpec belirten bağlantıları reddeder. IBM MQ , bu tür bağlantıların tümünü reddetmeyi garanti etmez ve IBM MQ yapılandırmanızın FIPS-uyumlu olup olmadığını belirlemek sizin sorumluluğunuzda.

ULW *KeyRepository (MQCHAR256)*

Bu alan yalnızca UNIX, Linuxve Pencerelemler 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 SSL 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.

KeyResetCount (MQLONG)

Bu, gizli anahtar yeniden anlaşılmadan önce bir SSL ya da TLS etkileşimi içinde gönderilen ve alınan şifrelenmemiş baytların toplam sayısını temsil eder.

Bayt sayısı, MCA tarafından gönderilen denetim bilgilerini içerir.

SSL ya da TLS gizli anahtar sıfırlama sayısı, 1 bayt-32 KB arasında bir değer belirlerseniz, SSL ya da TLS kanalları 32 KB 'lık bir gizli anahtar sıfırlama sayısını kullanır. Bu, küçük SSL ya da 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 anlaşılamadığını belirtmek için 0 değerini kullanın.

StrucId (MQCHAR4)

Bu, yapı tanıtıcısıdır; değer şöyle olmalıdır:

MQSCO_STRUC_ID

SSL 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 SSL yapılandırma seçenekleri yapısı.

MQSCO_VERSION_2

Version-2 SSL yapılandırma seçenekleri yapısı.

MQSCO_VERSION_3

Version-3 SSL yapılandırma seçenekleri yapısı.

MQSCO_VERSION_4

Version-4 SSL yapılandırma seçenekleri yapısı.

MQSCO_VERSION_5

Version-5 SSL yapılandırma seçenekleri yapısı.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

MQSCO_CURRENT_VERSION

SSL yapılandırma seçenekleri yapısının yürürlükteki sürümü.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri MQSCO_VERSION_1'dir.

MQSCO için ilk değerler ve dil bildirimleri

Çizelge 85. Initial values of fields in MQSCO		
Alan adı	Değişmezin adı	Değişmezin değeri
<i>StrucId</i>	MQSCO_STRUC_ID	'SCO-'
<i>Version</i>	MQSCO_CURRENT_VERSION	1
<i>KeyRepository</i>	Yok	Dizgi ya da boşluk boş değerli

Çizelge 85. Initial values of fields in MQSCO (devamı var)

Alan adı	Değişmezin adı	Değişmezin değeri
<i>CryptoHardware</i>	Yok	Dizgi ya da boşluk boş değeri
<i>AuthInfoRecCount</i>	Yok	0
<i>AuthInfoRecOffset</i>	Yok	0
<i>AuthInfoRecPtr</i>	Yok	Boş değeri gösterge ya da boş değeri byte
<i>KeyResetCount</i>	MQSCO_RESET_COUNT_DEFAULT	0
<i>FipsRequired</i>	MQSSL_FIPS_NO	0
<i>EncryptionPolicySuiteB</i>	MQ_SUITE_B_NONE, MQ_SUITE_B_NOT_AVAILABLE, MQ_SUITE_B_NOT_AVAILABLE, MQ_SUITE_B_NOT_AVAILABLE	1, 0, 0, 0
<i>CertificateValPolicy</i>	MQ_CERT_VAL_POLICY_DEFAULT	0

Notlar:

1. – simgesi tek bir boş karakteri temsil eder.
2. C programlama dilinde, makro değişkeni MQSCO_DEFAULT , çizelgede listelenen değerleri içerir. Yapıdaki alanlara ilişkin ilk değerleri sağlamak için bunu aşağıdaki şekilde kullanın:

```
MQSCO MySCO = {MQSCO_DEFAULT};
```

C bildirimi

```
typedef struct tagMQSCO MQSCO;
struct tagMQSCO {
    MQCHAR4    StrucId;           /* Structure identifier */
    MQLONG     Version;           /* Structure version number */
    MQCHAR256  KeyRepository;     /* Location of SSL key */
    MQCHAR256  CryptoHardware;    /* repository */
    MQCHAR256  CryptoHardware;    /* Cryptographic hardware */
    MQLONG     AuthInfoRecCount;  /* configuration string */
    MQLONG     AuthInfoRecOffset; /* Number of MQAIR records */
    PMQAIR     AuthInfoRecPtr;    /* present */
    MQLONG     KeyResetCount;     /* Offset of first MQAIR */
    MQLONG     FipsRequired;      /* record from start of */
    MQLONG     EncryptionPolicySuiteB[4]; /* MQSCO structure */
    MQLONG     CertificateValPolicy; /* Address of first MQAIR */
    MQCHAR64  CertificateLabel;    /* record */
    MQLONG     KeyResetCount;     /* Number of unencrypted */
    MQLONG     FipsRequired;      /* bytes sent/received */
    MQLONG     EncryptionPolicySuiteB[4]; /* before secret key is */
    MQLONG     CertificateValPolicy; /* reset */
    MQCHAR64  CertificateLabel;    /* Using FIPS-certified */
    MQLONG     KeyResetCount;     /* algorithms */
    MQLONG     FipsRequired;      /* Use only Suite B */
    MQLONG     EncryptionPolicySuiteB[4]; /* cryptographic algorithms */
    MQLONG     CertificateValPolicy; /* Certificate validation */
    MQCHAR64  CertificateLabel;    /* policy */
    MQLONG     KeyResetCount;     /* Ver:1 */
    MQLONG     FipsRequired;      /* Ver:2 */
    MQLONG     EncryptionPolicySuiteB[4]; /* Ver:3 */
    MQLONG     CertificateValPolicy; /* Ver:4 */
    MQCHAR64  CertificateLabel;    /* Ver:5 */
    MQLONG     KeyResetCount;
    MQLONG     FipsRequired;
    MQLONG     EncryptionPolicySuiteB[4];
    MQLONG     CertificateValPolicy;
    MQCHAR64  CertificateLabel;
}
```


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 SSL 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-AUTHINFORECPtr 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

```

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 SSL 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 */

```

Visual Basic bildirimi

```

Type MQSCO
StrucId As String*4 'Structure identifier'
Version As Long 'Structure version number'
KeyRepository As String*256 'Location of SSL key repository'
CryptoHardware As String*256 'Cryptographic hardware configuration'
'string'
AuthInfoRecCount As Long 'Number of MQAIR records present'
AuthInfoRecOffset As Long 'Offset of first MQAIR record from'
'start of MQSCO structure'
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

```

MQSD-Abonelik tanımlayıcısı

Aşağıdaki çizelge yapıdaki alanları özetlemektedir.

Alan	Tanım	Konu
<i>StrucId</i>	Yapı tanıttıcısı	StrucId
<i>Version</i>	Yapı sürüm numarası	S\u00fcr\u00fcm
<i>Options</i>	Seenekler	Seenekler
<i>ObjectName</i>	Nesne adı	ObjectName
<i>AlternateUserId</i>	Diğer Kullanıcı Kimliği	AlternateUserTanıtıcısı
<i>AlternateSecurityId</i>	Diğer Güvenlik Tanıtıcısı	AlternateSecurityTanıtıcısı
<i>SubExpiry</i>	Abonelik Süre Bitimi	SubExpiry
<i>ObjectString</i>	Nesne Dizgisi	ObjectString
<i>SubName</i>	Abonelik adı	SubName
<i>SubUserData</i>	Abonelik kullanıcı verileri	SubUserVerileri
<i>SubCorrelId</i>	Abonelik İltinti Tanıtıcısı	SubCorrelTanıtıcısı
<i>PubPriority</i>	Yayın önceliği	PubPriority
<i>PubAccountingToken</i>	Yayın Muhasebe Simgesi	PubAccountingSimgesi
<i>PubAppIdentityData</i>	Yayın uygulaması kimlik verileri	PubAppIdentityData
<i>SelectionString</i>	Seim ölçütleri sağlayan dizgi	SelectionString
<i>SubLevel</i>	Abonelik Düzeyi	SubLevel
<i>ResObjectString</i>	Uzun nesne adı	ResObjectDizgisi

MQSD için genel bakış

Kullanılabilirlik: Bu sistemlere baėlı AIX, HP-UX, IBM i, Solaris, Linux, Windows, z/OS, artı IBM MQ MQI clients .

Ama: MQSD yapısı, yapılmakta olan aboneliėe ilişkin ayrıntıları belirtmek için kullanılır.

Yapı, MQSUB aėırısında bir giriř/ıkıř deėiřtirgesi. Daha fazla bilgi için bkz. [MQSUB kullanım notları](#).

Yönetilen abonelikler: Bir uygulamanın, aboneliėiyle eřleşen yayınların hedefi olarak belirli bir kuyruėu kullanmak zorunda kalmaması durumunda, yönetilen abonelik özelliėini kullanabilir. Bir uygulama yönetilen aboneliėi kullanmayı seerse, kuyruk yöneticisi, MQSUB aėırısından ıkıř olarak bir nesne tanıttıcısı saėlayarak, yayınlanan iletilerin gönderildiėi hedef ile ilgili aboneyi bilgilendirir. Daha fazla bilgi için bkz. [Hobj \(MQHOBJ\)-giriř/ıkıř](#).

Abonelik kaldırıldıėında, kuyruk yöneticisi ařaėıdaki durumlarda, yönetilen hedeften alınmamıř olan iletileri temizlemek için de gerekir:

- Abonelik kaldırıldıėında-MQCO_REMOVE_SUB ile MQCLOSE kullanımı ve yönetilen Hobj kapatılır.
- Örtük olarak, kalıcı olmayan bir abonelik kullanan bir uygulamaya baėlantı kaybolduėunda (MQSO_NON_DAYANIKLI)
- Süresi dolduėu için abonelik kaldırıldıėında ve yönetilen Hobj kapatıldıėında sona erme tarihine kadar.

Bu temizleme iřlemi gerekleřebilmesi için, kalıcı olmayan aboneliklerle yönetilen abonelikleri kullanmanız gerekir; böylece, kapalı olmayan aboneliklere ilişkin iletiler kuyruk yöneticinizde yer kaplamaz. Dayanıklı abonelikler de yönetilen hedefleri kullanabilir.

Sürüm: MQSD 'nin yürürlükteki sürümü MQSD_VERSION_1' dir.

Karakter kümesi ve kodlama: MQSD ' deki veriler, *CodedCharSetId* kuyruk yöneticisi özniteliği tarafından verilen karakter kümesinde ve MQENC_NATIVE tarafından verilen yerel kuyruk yöneticisinin kodlamasında olmalıdır. Ancak, uygulama bir MQ MQI istemcisi olarak çalışıyorsa, yapı istemcisinin karakter kümesinde ve kodlamasında olmalıdır.

MQSD için alanlar

MQSD yapısı aşağıdaki alanları içerir; alanlar alfabetik sırayla anlatılır:

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 463 tanımlamasına bakın.

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.

ObjectName (MQCHAR48)

Bu ad, 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 (/), alt çizgi (_), yüzde (%)

Ad, baştaki ya da gömülü boşluklar içermemeli, ancak sondaki boşlukları içermeyebilir. Addaki önemli verilerin sonunu belirtmek için boş değerli bir karakter kullanın; boş değer ve izleyen karakterler boşluk olarak işlenir. Belirtilen ortamlarda aşağıdaki kısıtlamalar geçerlidir:

- EBCDIC Katakana kullanan sistemlerde küçük harfli karakterler kullanılamaz.
- Açıkz/OS:
 - Bir alt çizgiyle başlayan ya da biten adlardan kaçının; işlemler ve denetim panoları tarafından işlenemez.
 - The percent character has a special meaning to RACF. Dış güvenlik yöneticisi olarak RACF kullanılırsa, adlar yüzde değerini içermemelidir. Bu adlar kullanılırsa, RACF soysal profilleri kullanıldığında bu adlar hiçbir güvenlik denetiminde yer almaz.
- IBM üzerinde, komutlar üzerinde belirtildiğinde, küçük harf, eğik çizgi ya da yüzde içeren adların tırnak işareti içine alınması gerekir. Bu tırnak işaretleri, yapılar da alan olarak ya da çağrılarda parametre olarak ortaya çıkan adlar için belirtilmemelidir.

ObjectName , tam konu adını oluşturmak için kullanılır.

Tam konu adı iki farklı alandan oluşturulabiliyor: *ObjectName* ve *ObjectString*. Bu iki alanın nasıl kullanılma hakkında ayrıntılı bilgi için bkz. [“Konu dizelerinin kullanılması” sayfa 560](#).

ObjectName alanı tarafından tanımlanan nesne bulunamazsa, *ObjectString* içinde belirtilmiş bir dizgi olsa bile, arama neden kodu MQRC_UNKOWN_OBJECT_NAME neden koduyla başarısız olur.

MQSO_RESUME seçeneğini kullanarak bir MQSUB çağrısından bu alan değiştirilmeden döndür.

Bu alanın uzunluğu MQ_TOPIC_NAME_LENGTH tarafından verilir. Bu alanın başlangıç değeri, C ' de boş değer dizgisidir ve diğer programlama dillerindeki 48 boş karakter dizilimidir.

Var olan bir aboneliğin MQSO_ALTER seçeneğini kullanarak değiştirilmesi durumunda, abone olunan konu nesnesinin adı değiştirilemez. Bu alan ve *ObjectString* alanı atlanabilir. Sağlandıysa, aynı tam konu adını çözmeleri gerekir. Bu işlemi yapmazlarsa, çağrı MQRC_TOPIC_NOT_ALTERNATEABLE ile başarısız olur.

ObjectString (MQCHARV)

Bu, kullanılacak uzun nesne adıdır.

ObjectString , tam konu adını oluşturmak için kullanılır.

Tam konu adı iki farklı alandan oluşturulabiliyor: *ObjectName* ve *ObjectString*. Bu iki alanın nasıl kullanılma hakkında ayrıntılı bilgi için bkz. [“Konu dizelerinin kullanılması” sayfa 560](#).

ObjectString uzunluğu üst sınırı 10240 'tır.

ObjectString doğru şekilde belirtilmediyse, MQCHARV yapısının nasıl kullanılacağı ya da uzunluk üst sınırını aştıysa, çağrı neden kodu MQRC_OBJECT_STRING_ERROR ile başarısız olur.

Bu bir giriş alanıdır. Bu yapıdaki alanların ilk değerleri, MQCHARV yapısındaki alanlarla aynıdır.

ObjectString içinde joker karakterler varsa, bu genel arama karakterlerinin yorumu, MQSD ' nin Seçenekler alanında belirtilen Joker seçenekleri kullanılarak denetlenebilir.

MQSO_RESUME seçeneğini kullanarak bir MQSUB çağrısından bu alan değiştirilmeden döndür. The full topic name used is returned in the *ResObjectString* field if a buffer is provided.

Var olan bir aboneliğin MQSO_ALTER seçeneğini kullanarak değiştirilmesi durumunda, abone olunan konu nesnesinin uzun adı değiştirilemez. Bu alan ve *ObjectName* alanı atlanabilir. Sağlanırsa, aynı tam konu adına ya da çağrı MQRC_TOPIC_NOT_ALTERNATEABLE ile aynı konu adına çözümleri gerekir.

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. Çizelgede, geçerli erişim ve yaratma seçenekleri birleşimleri görüntülenir.

Seçenek bileşimi	Notlar
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 çağırılması, 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 önlenebilir. *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 döndürülebiliyor. 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 AlternateUserTanıtıcısı 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 'in MQSD ve MQSUB' da öznitelik değerlerini değiştirebilme yeteneğini göstermektedir.

Çizelge 86. MQSD ve MQSUB ' da değiştirilebilen öznitelikler			
Veri tipi tanımlayıcısı ya da işlev çağırısı	Alan adı	Bu öznitelik MQSO_ALTER kullanılarak değiştirilebilir mi?	Neden Kodu
MQSD	Dayanıklılık seçenekleri	Hayır	MQRC_DURABILITY_NOT_ALTERABLE

Çizelge 86. MQSD ve MQSUB ' da değiştirilebilen öznitelikler (devamı var)

Veri tipi tanımlayıcısı ya da işlev çağırısı	Alan adı	Bu öznitelik MQSO ALTER kullanılarak değiştirilebilir mi?	Neden Kodu
MQSD	Hedef Seçenekleri	Evet	Yok
MQSD	Kayıt seçenekleri	Evet (bkz. not "1" sayfa 551)	MQRC_GROUPING_NOT_ALTERABLE, MQSO_GROUP_SUB değiştirmeyi denesiyor
MQSD	Yayın seçenekleri	Evet (bkz. not "2" sayfa 551)	Yok
MQSD	Joker seçenekleri	Hayır	MQRC_TOPIC_NOT_ALTERABLE
MQSD	Diğer seçenekler	Hayır (bkz. not "3" sayfa 551)	Yok
MQSD	ObjectName	Hayır	MQRC_TOPIC_NOT_ALTERABLE
MQSD	AlternateUserTanıtıcısı	Hayır (bkz. not "4" sayfa 551)	Yok
MQSD	AlternateSecurityTanıtıcısı	Hayır (bkz. not "4" sayfa 551)	Yok
MQSD	SubExpiry	Evet	Yok
MQSD	ObjectString	Hayır	MQRC_TOPIC_NOT_ALTERABLE
MQSD	SubName	Hayır (bkz. not "5" sayfa 551)	Yok
MQSD	SubUserVerileri	Evet	Yok
MQSD	SubCorrelTanıtıcısı	Evet (bkz. not "6" sayfa 551)	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 551)	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 çağırısından 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 belirttik 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* değiştirgesinde 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 iletimi sağlanmasına neden olacak şekilde gruplandırılması, yalnızca bir iletinin kuyruğa gönderilmesine neden olur. Bu seçenek kullanılmıyorsa, her bir benzersiz abonelik (SubName ile 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 SubName olmalıdır; 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 değerini belirtmeyin. İkisi de belirlenmezse, varsayılan değer MQSO_FIXED_USERID 'dir.

Yayın seçenekleri: Aşağıdaki seçenekler, yayınların bu aboneye gönderilmesine yol açma şeklini denetler. Var olan bir aboneliğin MQSO_ALTER seçeneğini kullanarak değiştirilmesi durumunda, bu yayın seçenekleri değiştirilebilir.

MQSO_NOT_OWN_PUBS

Aracıya uygulamanın kendi yayınlarından herhangi birini görmek istemediğini belirtir. Bağlantı tanıtıcıları aynıysa, yayınların aynı uygulamadan kaynaklandığı düşünülür. MQSO_RESUME komutunu kullanarak bir MQSUB çağrısından dönüşte bu seçenek ayarlıdır (uygun).

MQSO_NEW_PUBLICATIONS_ONLY

Bu abonelik oluşturulduğunda, yalnızca yeni yayınlar gönderildiğinde, yürürlükte tutulan yayınların gönderilmemesine neden olmaz. Bu seçenek yalnızca, MQSO_CREATE belirtildiğinde geçerlidir. Abonelikte yapılacak her türlü değişiklik, yayınların akışını değiştirmez ve bu nedenle bir konuda tutulan yayınlar, aboneye yeni yayınlar olarak gönderilecektir.

Bu seçenek MQSO_CREATE olmadan belirtilirse, çağrı MQRC_OPTIONS_ERROR ile başarısız olur. MQSO_RESUME komutunu kullanarak bir MQSUB çağrısından dönüşte, bu seçenek bu seçenek kullanılarak yaratıldıysa bile bu seçenek belirlenmez.

Bu seçenek kullanılmıyorsa, önceden saklanan iletiler, sağlanan hedef kuyruğa gönderilir. Bu işlem bir hata nedeniyle başarısız olursa, MQRC_RETAINED_MSG_Q_ERROR ya da MQRC_RETAINED_NOT_TESLIM edilemedi, aboneliğin yaratılması başarısız olur.

MQSO_PUBLICATIONS_ON_REQUEST

Bu seçeneğin belirlenmesi, abonenin gerektiğinde özellikle bilgi isteyeceğini belirtir. Kuyruk yöneticisi aboneye istenmeyen iletiler göndermez. Önceki bir MQSUB çağrısından Hsub tanıtıcısı kullanılarak bir MQSUBRQ çağrısı yapıldığında, alıkonan yayın (ya da konu içinde bir genel arama karakteri belirtilirse, büyük olasılıkla birden çok yayın) aboneye gönderilir. Bu seçeneği kullanarak MQSUB çağrısının sonucu olarak hiçbir yayın gönderilmez. MQSO_RESUME komutunu kullanarak bir MQSUB çağrısından dönüşte bu seçenek ayarlıdır (uygun).

Bu seçenek, 1 'den büyük bir SubLevel ile birlikte geçerli değildir.

İleriye okuma seçenekleri: Aşağıdaki seçenekler, kalıcı olmayan iletilerin, uygulama isteğinde bulunan uygulama öncesinde bir uygulamaya gönderilip gönderilmeyeceğini denetler.

MQSO_READ_AHEAD_AS_Q_DEF

MQSUB çağrısı yönetilen bir tanıtıcı kullanıyorsa, uygulamanın uygulamaya istekte bulunmadan önce uygulamaya ileti gönderilip gönderilmeyeceğini belirlemek için, abone olunan konu ile ilişkili model kuyruğunun varsayılan okuma değeri özniteliği kullanılır.

Bu varsayılan değerdir.

MQSO_NO_READ_AHEAD

MQSUB çağrısı yönetilen bir tanıtıcı kullanıyorsa, uygulama bu tanıtıcıyı uygulamadan önce uygulamaya gönderilmez.

MQSO_READ_AHEAD

MQSUB çağrısı yönetilen bir tanıtıcı kullanıyorsa, uygulama bunları istemeden önce uygulamaya ileti gönderilebilir.

Not:

Öndeki okuma seçenekleri için aşağıdaki notlar geçerlidir:

1. Bu seçeneklerden yalnızca biri belirlenebilir. Hem MQO_READ_AHEAD, hem de MQOO_NO_READ_AHEAD belirtilirse, neden kodu MQRC_OPTIONS_ERROR döndürülür. Bu seçenekler yalnızca MQSO_MANAGED belirtilirse geçerlidir.
2. Önceden açılmış bir kuyruk geçirildiğinde, bunlar MQSUB için geçerli değildir. İstendiğinde okuma seçeneği etkinleştirilmemiş olabilir. İlk MQGET çağrısında kullanılan MQGET seçenekleri, önünün etkinleştirilmesini engelleyebilir. Ayrıca, istemcinin önden okuma desteklenmediği 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.

Özel Karakter	Davranış
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/level2/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 davranışdır.

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 547](#).

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 558](#).

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:

MQMD ' de alan	Kullanılan değer
<i>UserIdentifier</i>	Aboneliğin yapıldığı sırada, abonelik 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 yoksayılır, 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/OSüzerinde, CICS ya da IMS uygulaması için bu seçenek, bağlantı susturma durumundaysa MQSUB çağrısını da zorlar.

PubAccountingSimgesi (MQBYTE32)

Bu, bu abonelik 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 407](#)

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 409

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.

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 433.

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 433.

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.

ResObjectDizgisi (MQCHARV)

Kuyruk yöneticisi, *ObjectName*' ta sağlanan adı çözdükten sonra uzun nesne adıdır.

ObjectString 'ta uzun nesne adı sağlanırsa ve *ObjectName*' ta hiçbir şey sağlanmıyorsa, bu alanda döndürülen değer, *ObjectString*' de sağlanan değerle aynıdır.

Bu alan atılırsa (*ResObjectString.VSBufSize* sıfırdır), *ResObjectString* döndürülmez, ancak uzunluk *ResObjectString.VSLength*' te döndürülür. Uzunluk tam *ResObjectDizgisinden* kısaysa, kısaltılır ve sağlanan uzunluğa sığabileceği en sağdaki en sağdaki karakter döndürülür.

ResObjectString yanlış belirtildiyse, *MQCHARV* yapısının nasıl kullanılacağını ya da uzunluk üst sınırını aşıyorsa, çağrı neden kodu MQRC_RES_OBJECT_STRING_ERROR ile başarısız olur.

SelectionString (MQCHARV)

Bu, bir konudaki iletiler için abone olurken kullanılan seçim ölçütlerini sağlamak için kullanılan dizgidir.

Bu değişken uzunluk alanı, bir arabellek sağlandıysa ve *VSBufSize* içinde artı bir arabellek uzunluğu da varsa, MQSU_RESUME seçeneği kullanılarak bir MQSUB çağrısından çıkışa döndürülecektir. Çağrıda arabellek sağlanmazsa, *MQCHARV* ' in *VSLength* alanında yalnızca seçim dizgisinin uzunluğu döndürülür. Sağlanan arabellek, alanı döndürmek için gereken alandan küçükse, sağlanan arabellekte yalnızca *VSBufSize* byte değeri döndürülür.

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 kullanımı, *Seçicileri* içinde açıklanmaktadır.

StrucId (MQCHAR4)

Bu, yapı tanıtıcısıdır; değer şöyle olmalıdır:

MQSD_STRUC_ID

Abonelik Tanımlayıcısı yapısına ilişkin tanıtıcı.

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.

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 geçirilebilir.

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.

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 tanıdılır:

MQE_UNSNISZ

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.

SubLevel (MQUZE)

Bu, abonelikte ilişkili düzeydir. Bu abonelik yalnızca, yayınlama sırasında kullanılan PubLevel değerinden küçük ya da bu değere eşit olan en yüksek SubLevel değerine sahip abonelikler kümesinde yer alıyorsa, bu abonelik için yalnızca bu abonelikte teslim edilir. Ancak, bir yayın saklandıysa, PubLevel 1 düzeyinde yeniden yayınlandığından, bu yayın artık daha yüksek düzeylerde aboneler için kullanılabilir değildir.

Değer, sıfır ile 9 aralığında olmalıdır. Sıfır en düşük düzeydir.

Bu alanın ilk değeri 1 'dir.

Daha fazla bilgi için bkz. [Yayınlar Arası Yayınlar](#).

Varolan bir aboneliği MQSO_ALTER seçeneğini kullanarak değiştiriyorsa, SubLevel değeri değiştirilemez.

MQSO_PUBLICATIONS_ON_REQUEST seçeneğiyle bir SubLevel değerinin 1 'den büyük bir değere sahip olmasına izin verilmez.

MQSO_RESUME komutunu kullanarak bir MQSUB çağrısından dönüşte, bu alan abonelik için kullanılmakta olan yürürlükteki düzeye ayarlanır.

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.

SubName (MQCHARV)

Bu, abonelik adını belirtir. Bu alan yalnızca, *Options* MQSO_DAYANIKLI seçeneğini belirtiyorsa gereklidir; ancak, sağlandıysa, kuyruk yöneticisi tarafından MQSO_NON_MUKATIAL için de kullanılır.

If specified, *SubName* must be unique within the queue manager, because it is the method used to identify the subscription.

SubName uzunluğu üst sınırı 10240 'tır.

Bu alan iki amaca hizmet eder. Bir MQSO_DAYANIKLI aboneliği için, aboneliği tanımlamak için bu alanı kullanıyorsunuz; böylece, aboneliği aboneliğe kapatmış (MQCO_KEEP_SUB seçeneği kullanılarak) ya da kuyruk yöneticisinden bağlantısı kesildiyse, bu aboneliği sürdürebilmenizi sağlar. Bu, MQSO_RESUME seçeneğiyle MQSUB çağrısını kullanarak yapılır. Ayrıca, DISPLAY SBSTATUS içindeki SUBNAME alanındaki aboneliklerin yönetim görünümünde de görüntülenir.

If *SubName* is specified incorrectly, according to the description of how to use the MQCHARV structure, is left out when it is required (that is *SubName*. *VSLength* sıfır) ya da uzunluk üst sınırını aşarsa, çağrı neden kodu MQRC_ALT_NAME_ERROR neden ile başarısız olur.

Bu bir giriş alanıdır. Bu yapıdaki alanların ilk değerleri, MQCHARV yapısındaki alanlarla aynıdır.

Var olan bir aboneliği MQSO ALTER seçeneğini kullanarak değiştiriyorsa, gönderme yapılan aboneliği bulmak için kullanılan tanımlayıcı alan olduğu için, abonelik adı değiştirilemez. MQSO_RESUME seçeneği ile bir MQSUB çağrısının çıktısında değiştirilmez.

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.

Konu dizelerinin kullanılması

Konu, bir konu nesnesinde tanımlanan alt konu ve bir uygulama tarafından sağlanan bir alt konu ile oluşturulur. Konu adı olarak alt konuyu kullanabilir ya da yeni bir konu adı oluşturmak için bunları birleştirebilirsiniz.

Bir MQI programında, tam konu adı MQOPEN tarafından yaratılır. Yayınlama/abone olma MQI çağrılarında kullanılan iki alandan oluşur ve listelenen sırada:

1. **ObjectName** alanında adlandırılan konu nesnesinin **TOPICSTR** özniteliği.
2. Uygulama tarafından sağlanan alt konuyu tanımlayan **ObjectString** parametresi.

Sonuçtaki konu dizgisi, **ResObjectString** parametresine döndürülür.

Her alanın ilk karakteri boş değerli ya da boş bir karakter değilse ve alan uzunluğu sıfırdan büyükse, bu alanlar var olarak kabul edilir. Alanlardan yalnızca biri varsa, bu alan konu adı olarak değiştirilmeden kullanılır. Her iki alanda bir değer yoksa, konu adı geçerli değilse, çağrı neden kodu MQRC_UNKNOWN_OBJECT_NAME ya da MQRC_TOPIC_STRING_ERROR neden ile başarısız olur.

Her iki alan da varsa, bir ' / ' resultant birleşik konu adının iki ögesi arasına karakter eklenir.

Çizelge 87 sayfa 561 , konu dizgi bitleştirme örneklerini gösterir:

Çizelge 87. Konu dizgi bitleştirme örnekleri			
TOPICSTR	ObjectString	Tam konu adı	Açıklama
Futbol/İskoçya	' '	Futbol/İskoçya	TOPICSTR tek başına kullanılır
' '	Futbol/İskoçya	Futbol/İskoçya	ObjectString tek başına kullanılır
Futbol	Puanlar	Futbol/İskoçya	Bitleştirme noktasına bir ' / ' karakteri eklenir
Futbol	/İskolar	Futbol/İskoçya	İki dizgi arasında 'boş düğüm' üretildi
/Futbol	Puanlar	/Futbol/Scores	Konu 'boş düğüm' ile başlıyor

The ' / ' character is considered as a special character, providing structure to the full topic name in Konu ağaçları, and must not be used for any other reason as the structure of the topic tree is affected. Konu " /Football " konu ile aynı değil " dedi. Football ".

Aşağıdaki genel arama karakterleri özel karakterlerdir:

- artı işareti ' + '
- sayı işareti ' # '
- yıldız işareti ' * '
- soru işareti ' ? '

Bu karakterler geçersiz olarak kabul edilmiyor, ancak bunların nasıl kullanıldığını anladığınızdan emin olun. Yayınlama sırasında bu karakterleri konu dizginizde kullanmamayı tercih edebilirsiniz. Bir konu dizgisinde 'with' ile yayınlama # ' ya da ' + ' bir konu düzeyi içinde diğer karakterlerle (kendileri de içinde olmak üzere) karışık olarak, her iki joker şema ile abone olunabilir. Bir konu dizgisinde 'with' ile yayınlama # ' ya da ' + ' iki ' arasındaki tek karakter olarak / ' Bu karakterler, MQSO_WILDCARD_TOPIC genel arama karakteri kullanılarak bir uygulama tarafından belirtik olarak abone olunamayacak bir konu dizgisi üretir. Bu durum, uygulamanın beklenenden daha fazla yayın elde etmesi sonucunu elde eder.

Örnek kod parçacığı

Örnek program Örnek 2: Değişken konusuna yayınlayıcıörnek programından çıkarılan bu kod parçacığı, bir konu nesnesini bir değişken konu dizgisiyle birleştirir.

```
MQOD      td = {MQOD_DEFAULT}; /* Object Descriptor      */
td.ObjectType = MQOT_TOPIC; /* Object is a topic */
td.Version = MQOD_VERSION_4; /* Descriptor needs to be V4 */
strcpy(td.ObjectName, topicName, MQ_TOPIC_NAME_LENGTH);
```

```
td.ObjectString.VSPtr = topicString;
td.ObjectString.VSLength = (MQLONG)strlen(topicString);
td.ResObjectString.VSPtr = resTopicStr;
td.ResObjectString.VSBufSize = sizeof(resTopicStr)-1;
MQOPEN(Hconn, &td, MQOO_OUTPUT | MQOO_FAIL_IF_QUIESCING, &Hobj, &CompCode, &Reason);
```

MQSD için ilk değerler ve dil bildirimleri

Alan adı	Değişmezin adı	Değişmezin değeri
<i>StrucId</i>	MQSD_STRUC_ID	'SD↵↵'
<i>Version</i>	MQSD_VERSION_1	1
<i>Options</i>	MQSO_NON_DAYANIKLI	0
<i>ObjectName</i>	Yok	Dizgi ya da boşluk boş değerli
<i>AlternateUserId</i>	Yok	Dizgi ya da boşluk boş değerli
<i>AlternateSecurityId</i>	MQSID_NONE	Boş Değerler
<i>SubExpiry</i>	MQE_UNSNISIZ	-1
<i>ObjectString</i>	Yok	MQCHARV için tanımlandığı şekilde adlar ve değerler
<i>SubName</i>	Yok	MQCHARV için tanımlandığı şekilde adlar ve değerler
<i>SubUserData</i>	Yok	MQCHARV için tanımlandığı şekilde adlar ve değerler
<i>SubCorrelId</i>	MQCI_NONE	Boş Değerler
<i>PubPriority</i>	MQPRI_PRIORITY_AS_Q_DEF	-3
<i>PubAccountingToken</i>	MQACT_NONE	Boş Değerler
<i>PubApplIdentityData</i>	Yok	Dizgi ya da boşluk boş değerli
<i>SelectionString</i>	Yok	MQCHARV için tanımlandığı şekilde adlar ve değerler
<i>SubLevel</i>	Yok	1
<i>ResObjectString</i>	Yok	MQCHARV için tanımlandığı şekilde adlar ve değerler

Notlar:

- ↵ simgesi tek bir boş karakteri temsil eder.
- Boş değerli dizgi ya da boşluklar, C içindeki boş değerli dizeyi ve diğer programlama dillerindeki boş karakterleri belirtir.
- C programlama dilinde, makro değişkeniMQSD_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlara ilk değerleri sağlamak için aşağıdaki şekilde kullanılabilir:

```
MQSD MySD = {MQSD_DEFAULT};
```

C bildirimi

```
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 */
};

```

COBOL bildirimi

```

** 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.

```

PL/I bildirimi

```

dcl
1 MQSD based,
3 StrucId      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 VSOOffset    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 VSOOffset    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 VSOOffset    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 VSOOffset    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 VSOOffset    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 */

```

High Level Assembler 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_VSBUFFSIZE DS  F   size of buffer
MQSD_SUBUSERDATA_VSLNGTH 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_VSBUFFSIZE DS  F   size of buffer
MQSD_SELECTIONSTRING_VSLNGTH 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_VSBUFFSIZE DS  F   size of buffer
MQSD_RESOBJECTSTRING_VSLNGTH 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)

```

MQSMPO-İleti özelliği seçeneklerini belirle

Aşağıdaki çizelge yapıdaki alanları özetlemektedir.

Çizelge 88. MQSMPO içindeki alanlar		
Alan	Tanım	Konu
<i>StrucId</i>	Yapı tanıttıcısı	StrucId
<i>Version</i>	Yapı sürüm numarası	S\u00fcr\u00fcm
<i>Options</i>	Seenekler	Seenekler
<i>ValueEncoding</i>	Özellik değeri kodlaması	ValueEncoding
<i>ValueCCSID</i>	Özellik değeri karakter kümesi	ValueCCSID

MQSMPO için genel bakış

Kullanılabilirlik: Tüm IBM MQ sistemleri ve IBM MQ istemcileri.

Amaç: MQSMPO yapısı, uygulamaların, iletilerin özelliklerini nasıl ayarlayacağını denetleyen seçenekleri belirlemesine olanak sağlar. Yapı, MQSETMP çağrısındaki bir giriş parametresidir.

Karakter kümesi ve kodlama: MQSMPO içindeki veriler, uygulamanın uygulama ve kodlamasının karakter kümesinde (MQENC_NATIVE) olmalıdır.

MQSMPO alanları

MQSMPO yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

Seçenekler (MQUZE)

Konum seçenekleri: Aşağıdaki seçenekler, özelliğin özellik imlecine karşılaştırılan görel konularıyla ilgilidir:

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.

MQSMPO_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.

MQSMPO_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.

MQSMPO_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.

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.

ValueCCSID (MQHOT)

Değer bir karakter dizilimiye, ayarlanacak özellik değerinin karakter kümesi.

Bu her zaman bir giriş alanıdır. Bu alanın başlangıç değeri **MQCCSI_APPL**' dir.

ValueEncoding (MQUZE)

Değer sayıysa, ayarlanacak özellik değerinin kodlaması.

Bu her zaman bir giriş alanıdır. Bu alanın başlangıç değeri **MQENC_NATIVE**' 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.

MQSMPO için ilk değerler ve dil bildirimleri

Çizelge 89. MQSMPO içindeki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
<i>StrucId</i>	MQSMPO_STRUC_ID	' SMPO '
<i>Version</i>	MQSMPO_VERSION_1	1
<i>Options</i>	MQSMPO_NONE	0
<i>ValueEncoding</i>	MQENC_NATIVE	Ortama göre değişir
<i>ValueCCSID</i>	MQCCSI_APPL	-3

Çizelge 89. MQSMPPO içindeki alanların ilk değerleri (devamı var)

Alan adı	Değişmezin adı	Değişmezin değeri
Notlar: - Boş değerli dizgi ya da boşluklar, C içindeki boş değerli dizeyi ve diğer programlama dillerindeki boş karakterleri belirtir. - C programlama dilinde, makro değişkeniMQSMPPO_VARSAYILAN, çizelgede listelenen değerleri içerir. Yapıdaki alanlara ilk değerleri sağlamak için aşağıdaki şekilde kullanılabilir: <pre>MQSMPPO MySMPPO = {MQSMPPO_DEFAULT};</pre>		

C bildirimi

```
typedef struct tagMQSMPPO MQSMPPO;
struct tagMQSMPPO {
    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 */
};
```

COBOL bildirimi

```
** MQSMPPO structure
10 MQSMPPO.
** Structure identifier
15 MQSMPPO-STRUCID PIC X(4).
** Structure version number
15 MQSMPPO-VERSION PIC S9(9) BINARY.
** Options that control the action of MQSETMP
15 MQSMPPO-OPTIONS PIC S9(9) BINARY.
** Encoding of VALUE
15 MQSMPPO-VALUEENCODING PIC S9(9) BINARY.
** Character set identifier of VALUE
15 MQSMPPO-VALUECCSID PIC S9(9) BINARY.
```

PL/I bildirimi

```
dcl
1 MQSMPPO 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 */
```

High Level Assembler bildirimi

```
MQSMPPO          DSECT
MQSMPPO_STRUCID  DS CL4 Structure identifier
MQSMPPO_VERSION  DS F   Structure version number
MQSMPPO_OPTIONS  DS F   Options that control the action of
*                MQSETMP
MQSMPPO_VALUEENCODING DS F Encoding of VALUE
MQSMPPO_VALUECCSID DS F Character set identifier of VALUE
MQSMPPO_LENGTH   EQU *-MQSMPPO
MQSMPPO_AREA     DS CL(MQSMPPO_LENGTH)
```


MQSRO-Abonelik isteđi seçenekleri

Bu bölümde, abonelik isteđi seçenekleri, bu alanların içerdđi alanlar ve bu alanların ilk değeri açıklanır.

Alan	Tanım	Konu
<i>StrucId</i>	Yapı tanıtıcısı	StrucId
<i>Version</i>	Yapı sürüm numarası	S\u00fcr\u00fcm
<i>Options</i>	Seçenekler	Seçenekler
<i>NumPubs</i>	Yayın sayısı	NumPubs

MQSRO 'ya Genel Bakış

Kullanılabilirlik: Bu sistemlere bađlı AIX, HP-UX, IBM i, Solaris, Linux, Windows, z/OS ve IBM MQ MQI clients .

Amaç: MQSRO yapısı, uygulamanın bir abonelik isteđinin nasıl yapıldđını denetleyen seçenekleri belirlemesine olanak sađlar. Yapı, MQSUBRQ çağrısındaki bir giriş/çıkış parametresidir.

Sürüm: MQSRO 'nun yürürlükteki sürümü MQSRO_VERSION_1' dir.

Karakter kümesi ve kodlama: MQSRO içindeki veriler, *CodedCharSetId* kuyruk yöneticisi özniteliđi tarafından verilen 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.

MQSRO alanları

MQSRO yapısı ařađıdaki alanları içeri; alanlar alfabetik sırayla anlatılır:

NumPubs (MQUZE)

Bu bir çıkış alanıdır ve bu çağrının sonucu olarak abonelik kuyruđına gönderilen yayın sayısını belirtmek için uygulamaya geri döndürölür. bu çağrı sayısının bir sonucu olarak bu sayıda yayın gönderilmesine rađmen, bu kadar çok mesajın, özellikle de kalıcı olmayan mesajlar olması halinde, uygulamanın elde edilmesi için bu kadar çok mesaja ulaşabileceđine dair bir garanti yoktur.

Konu joker karakterle içerdilseyse 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.

Seçenekler (MQUZE)

Ařađıdaki 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, bađlantının susturulmuş durumda olması durumunda MQSUBRQ çağrısını da zorlar.

Varsayılan seçenek: Önceden açıklanan seçenek gerekmiyorsa, ařađıdaki seçeneđin kullanılması gerekir:

MQSRO_YOK

Başka bir seçenek belirtilmemeyi belirtmek için bu değeri kullanın; tüm seçenekler varsayılan değeri kabul eder.

MQSRO_NONE, program belgelerine yardımcı olur. Bu seçeneđin başka biriyle birlikte kullanılması amaçlanmasa da, değeri sıfır olduđu için bu kullanım saptanamaz.

StrucId (MQCHAR4)

Bu, yapı tanıtcısıdır; değer şöyle olmalıdır:

MQSRO_STRUC_ID

Abonelik İsteği Seçenekleri yapısına ilişkin tanıtcı.

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.

MQSRO için ilk değerler ve dil bildirimleri

Alan adı	Değişmezin adı	Değişmezin değeri
<i>StrucId</i>	MQSRO_STRUC_ID	' SRO↵ '
<i>Version</i>	MQSRO_VERSION_1	1
<i>Options</i>	MQSRO_YOK	0
<i>NumPubs</i>	Yok	0
Notlar: 1. ↵ simgesi tek bir boş karakteri temsil eder. 2. C programlama dilinde, makro değişkeniMQSRO_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlara ilk değerleri sağlamak için aşağıdaki şekilde kullanılabilir: <pre>MQSRO MySRO = {MQSRO_DEFAULT};</pre>		

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 */  
};
```

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.

```

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 */

```

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)

```

MQSTS-Durum raporlama yapısı

Aşağıdaki çizelge yapıdaki alanları özetlemektedir.

Çizelge 90. MQSTS 'deki alanlar		
Alan	Tanım	Konu
<i>StrucId</i>	Yapı tanıtıcısı	StrucId
<i>Version</i>	Yapı sürüm numarası	S\u00fcr\u00fcm
<i>CompCode</i>	İlk hatanın tamamlanma kodu	CompCode
<i>Reason</i>	İlk hatanın neden kodu	Neden
<i>PutSuccessCount</i>	Başarılı zamanuyumsuz koyma çağrılarının sayısı	SuccessCount
<i>PutWarningCount</i>	Uyarı içeren zamanuyumsuz koyma çağrılarının sayısı	WarningCount
<i>PutFailureCount</i>	Başarısız olan zamanuyumsuz koyma çağrılarının sayısı	FailureCount
<i>ObjectType</i>	Hatalı nesnenin tipi	ObjectType
<i>ObjectName</i>	Başarısız olan nesnenin adı	ObjectName
<i>ObjectQMGrName</i>	Başarısız olan nesneye sahip olan kuyruk yöneticisinin adı	ObjectQMGrAdı
<i>ResolvedObjectName</i>	Hedef kuyruğun çözümlenmiş adı	ResolvedObjectAdı
<i>ResolvedQMGrName</i>	Hedef kuyruk yöneticisinin çözümlenmiş adı	ResolvedQMGrAdı
Not: Sürüm MQSTS_VERSION_2değerinden küçükse, geri kalan alanlar yoksayılr.		
<i>ObjectString</i>	Başarısız olan nesnenin uzun nesne adı	ObjectString

Çizelge 90. MQSTS ' deki alanlar (devamı var)		
Alan	Tanım	Konu
<i>SubName</i>	Aboneliğin başarısız olduğu abonelik adı	<u>SubName</u>
<i>OpenOptions</i>	Hatayla ilişkili açma seçenekleri	<u>OpenOptions</u>
<i>SubOptions</i>	Hata ile ilişkili abonelik seçenekleri	<u>SubOptions</u>

MQSTS ' ye Genel Bakış

Amaç: MQSTS yapısı, MQSTAT komutundan bir çıkış değiştirir.

Karakter kümesi ve kodlama: MQSTS ' deki karakter verileri, yerel kuyruk yöneticisinin karakter takımında; bu değer *CodedCharSetId* kuyruk yöneticisi özneliği tarafından verilir. MQSTS ' deki sayısal veriler yerel makine kodlamasıdır; bu, *Kodlamata* tarafından verilir.

Kullanım: MQSTAT komutu, durum bilgilerini almak için kullanılır. Bu bilgiler bir MQSTS yapısıyla döndürülür. MQSTAT ile ilgili bilgi için bkz. “MQSTAT-Durum bilgilerini al” sayfa 764.

MQSTS için alanlar

MQSTS yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

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_OK olur.

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_OK olur.

CompCode her zaman bir çıkış alanıdır. İlk değer MQCC_OK'dir.

ObjectName (MQCHAR48)

Raporlanmakta olan nesnenin adı.

ObjectName ' ın 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 başlantı kurulamazsa, yeniden başlantı kurulamasına 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).

MQSTAT_TYPE_RECONNECTION

Boş.

MQSTAT_TYPE_RECONNECTION_ERROR

Başlantı yeniden başlantı kurulamazsa, yeniden başlantı kurulamasına neden olan nesne adlarıdır. Başarısızlığın nedeni, MQSTS yapısındaki *CompCode* ve *Reason* alanlarında raporlanır.

ObjectQMgrName , bir çıkış alanıdır. Bu değer, C içindeki boş değerli dizidir ve diğer programlama dillerinde 48 boş karakter içerir.

ObjectString (MQCHARV)

Raporlanmakta olan başarısız nesnenin uzun nesne adı. Yalnızca MQSTS ya da daha sonraki sürümlerde bulunan 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.

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_ Qdeğeridir.

OpenOptions (MQUZE)

The *OpenOptions* used to open the object being reported upon. Yalnızca MQSTS ya da daha sonraki sürümlerde bulunan 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.

PutFailureCount (MQLONG)

Başarısız olan zamanuyumsuz koyma işlemlerinin sayısı.

PutFailureCount değeri, MQSTAT Type parametresinin değerine bağlıdır.

MQSTAT_TYPE_ASYNC_ERROR

MQCC_FAILED ile tamamlanan MQSTS yapısındaki nesneye zamanuyumsuz koyma işlemlerinin sayısı.

MQSTAT_TYPE_RECONNECTION

Sıfır.

MQSTAT_TYPE_RECONNECTION_ERROR

Sıfır.

PutFailureCount , bir çıkış alanıdır. Başlangıç değeri sıfır.

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 zamanuyumsuz 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 zamanuyumsuz koyma işlemlerinin sayısı.

PutWarningCount değeri, MQSTAT Type parametresinin değerine bağlıdır.

MQSTAT_TYPE_ASYNC_ERROR

MQCC_WARNING ile tamamlanan MQSTS yapısındaki nesneye zamanuyumsuz koyma işlemlerinin sayısı.

MQSTAT_TYPE_RECONNECTION

Sıfır.

MQSTAT_TYPE_RECONNECTION_ERROR

Sıfır.

PutWarningCount , bir çıkış alanıdır. Başlangıç değeri sıfır.

SubName (MQCHARV)

Başarısız olan aboneliğin adı. Yalnızca MQSTS ya da daha sonraki sürümlerde bulunan Sürüm 2 'de bulunur.

SubName ' in 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 abonelikle ilgili değilse, bu sıfır uzunluklu bir dizgidir.

SubName , bir çıkış alanıdır. Başlangıç değeri sıfır uzunluklu bir dizilimdir.

SubOptions (MQUZE)

The SubOptions used to open the failing subscription. Yalnızca MQSTS ya da daha sonraki sürümlerde bulunan Sürüm 2 'de bulunur.

SubOptions ' ın 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.

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_NONE olur.

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_NONE olur.

Reason , bir çıkış alanıdır. İlk değer MQRC_NONE'dir.

ResolvedObjectAdı (MQCHAR48)

Yerel kuyruk yöneticisi adı çözümledikten sonra, *ObjectName* içinde belirtilen nesnenin adı.

ResolvedObjectName ' ın yorumu, MQSTAT Type parametresinin değerine bağlıdır.

MQSTAT_TYPE_ASYNC_ERROR

ResolvedObjectName , yerel kuyruk yöneticisi adı çözdükten 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 çözdükten sonra hedef kuyruk yöneticisinin adı.

ResolvedQMGrName ' ın yorumu, MQSTAT Type parametresinin değerine bağlıdır.

MQSTAT_TYPE_ASYNC_ERROR

ResolvedQMGrName , yerel kuyruk yöneticisinin adı çözümlendikten sonra hedef kuyruk yöneticisinin adıdır. Döndürülen ad, *ResolvedObjectName* ile tanıtilan 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.

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_ID değeridir.

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_1 değeridir.

MQSTS için ilk değerler ve dil bildirimleri

Çizelge 91. MQSTS 'deki alanların ilk değerleri

Alan adı	Değişmezin adı	Değişmezin değeri
StrucId	MQSTS_STRUC_ID	'STAT↵'
Version	MQSTS_VERSION_1	1
CompCode	MQCC_OK	0
Reason	MQRC_NONE	0
PutSuccessCount	Yok	0
PutWarningCount	Yok	0
PutFailureCount	Yok	0
ObjectType	MQOT_Q	1
ObjectName	Yok	Dizgi ya da boşluk boş değeri
ObjectQMgrName	Yok	Dizgi ya da boşluk boş değeri
ResolvedObjectName	Yok	Dizgi ya da boşluk boş değeri
ResolvedQMgrName	Yok	Dizgi ya da boşluk boş değeri
ObjectString	MQCHARV_DEFAULT	{NULL,0,0,0,-3}
SubName	MQCHARV_DEFAULT	{NULL,0,0,0,-3}
OpenOptions	Yok	0
SubOptions	Yok	0

Notlar:

1. ↵ simgesi tek bir boş karakteri temsil eder.
2. Boş değeri dizgi ya da boşluklar, C içindeki boş değeri dizeyi ve diğeri programlama dillerindeki boş karakterleri belirtir.
3. C programlama dilinde, MQSTS_default makro değişkeninde çizelgede listelenen değeri bulunur. Yapıdaki alanlara ilk değeri sağlamak için aşağıdaki şekilde kullanılabilir:

```
MQSTS MySTS = {MQSTS_DEFAULT};
```

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 */
};

```

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 **

```

PL/I bildirimi

```

dcl
  1 MQSTS based,
    3 StrucId          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 */

```

High Level Assembler 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_PUTSUCCESSCOUNT              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
MQSTS_OBJECTSTRING_AREA            DS    ORG    MQSTS_OBJECTSTRING
                                     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               EQ    *-MQSTS_SUBNAME
MQSTS_SUBNAME_AREA                 DS    ORG    MQSTS_SUBNAME
                                     CL(MQSTS_SUBNAME_LENGTH)
*
MQSTS_OPENOPTIONS                  DS    F      Failing open options
MQSTS_SUBOPTIONS                   DS    F      Failing subscription option
MQSTS_LENGTH                       EQU    *-MQSTS
MQSTS_AREA                         DS    ORG    MQSTS
                                     CL(MQSTS_LENGTH)

```

MQTM-Tetikleme iletisi

Aşağıdaki çizelge yapıdaki alanları özetlemektedir.

Çizelge 92. MQTM 'deki Alanlar		
Alan	Tanım	Konu
StrucId	Yapı tanıttıcısı	StrucId
Version	Yapı sürüm numarası	S\u00fcr\u00fcm

Çizelge 92. MQTM ' deki Alanlar (devamı var)		
Alan	Tanım	Konu
<i>QName</i>	Tetiklenen kuyruğun adı	<u>QName</u>
<i>ProcessName</i>	Süreç nesnesinin adı	<u>ProcessName</u>
<i>TriggerData</i>	Verileri tetikle	<u>TriggerData</u>
<i>ApplType</i>	Uygulama tipi	<u>ApplType</u>
<i>ApplId</i>	Uygulama tanıtıcısı	<u>ApplId</u>
<i>EnvData</i>	Ortam verileri	<u>EnvData</u>
<i>UserData</i>	Kullanıcı verileri	<u>UserData</u>

MQTM ' ye Genel Bakış

Amaç: MQTM yapısı, bir kuyruk için tetikleme olayı ortaya çıktığında kuyruk yöneticisi tarafından bir tetikleme izleme uygulamasına gönderilen tetikleyici iletilerinde verileri tanımlar.

This structure is part of the IBM MQ Trigger Monitor Interface (TMI), which is one of the IBM MQ framework interfaces.

Biçim adı: MQFMT_TRIGGER.

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 *CodedCharSetId* and *Encoding* 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 iletilerinde bulunan bilgilerin bazılarını ya da tümünü, tetikleme izleme uygulamasının başladığı uygulamaya geçirmesi gerekebilir. Başlatılan uygulama tarafından gerekebilecek bilgiler, *QName*, *TriggerData* ve *UserData* içindir. 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. “MQTMC2 -Tetikleyici ileti 2 (karakter biçimi)” sayfa 586.

- On z/OS, for an MQAT_CICS application that is started using the CKTI transaction, the entire trigger message structure MQTM is made available to the started transaction; the information can be retrieved by using the EXEC CICS RETRIEVE command.
- IBM üzerinde, IBM MQ ile verilen tetikleme izleme uygulaması, başlatılan uygulamaya bir MQTMC2 yapısı iletir.

Tetikleyicileri kullanma hakkında bilgi için [Tetikleyicileri kullanarak IBM MQ uygulamalarının başlatılması](#) başlıklı konuya bakın.

Tetikleme ileti için MQMD: Kuyruk yöneticisi tarafından oluşturulan bir tetikleme iletilerinin MQMD ' deki alanları aşağıdaki gibi ayarlanır:

MQMD ' de 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>	MQE_UNSENSIZ

MQMD ' de alan	Kullanılan değer
<i>Feedback</i>	MQFB_YOK
<i>Encoding</i>	MQENC_NATIVE
<i>CodedCharSetId</i>	Kuyruk yöneticisinin <i>CodedCharSetId</i> özneliği
<i>Format</i>	MQFMT_TETIKLEYICISI
<i>Priority</i>	Başlatma kuyruğunun <i>DefPriority</i> özneliği
<i>Persistence</i>	MQPER_NOT_PERSISTENT
<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 aracısının uygun olduğu şekilde
<i>PutApplName</i>	Kuyruğun ilk 28 baytı-yönetici adı
<i>PutDate</i>	Tetikleme iletisinin gönderildiği tarih
<i>PutTime</i>	Tetikleme iletisinin gönderildiği saat
<i>ApplOriginData</i>	Boşluklar

Aşağıdaki durumlar dışında, benzer değerleri ayarlamak için bir tetikleme iletisi oluşturan bir uygulama önerilir:

- *Priority* alanı, MQPRI_PRIORITY_AS_Q_DEF olarak ayarlanabilir (kuyruk yöneticisi, ileti bulunduğu, başlatma kuyruğu için bu değeri varsayılan önceliğe değiştirecektir).
- *ReplyToQMgr* alanı boşluklara ayarlanabilir (kuyruk yöneticisi, iletiyi koyduğunda bu değeri yerel kuyruk yöneticisinin adıyla değiştirecektir).
- Bağlam alanlarını uygulama için uygun olarak ayarlayın.

MQTM için alanlar

MQTM yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

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* özneliğinin değeriyle birlikte başlatır; bu özneliğe ilişkin ayrıntılar için [“Süreç tanımlamalarına ilişkin öznelilikler” sayfa 844 ' 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.
- UNIX sistemleri 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.

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* özniteliğinin değeriyle birlikte başlatır; bu özniteliğe ilişkin ayrıntılar için [“Süreç tanımlamalarına ilişkin öznitelikler” sayfa 844](#) ' e bakın. Bu verilerin içeriği kuyruk yöneticisi için bir önem göstermiyor.

ApplType , 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_KöPRÜSÜ

CICS bridge uygulaması.

MQAT_CICS_VSE

CICS/VSE işlemi.

MQAT_DOS

PC DOS üzerinde IBM 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_NSK

HP Integrity NonStop Server 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.

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 844](#) ' e bakın. Bu verilerin içeriği kuyruk yöneticisi için bir önem göstermiyor.

z/OSüzerinde, 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.

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* özniteliğinin değeriyle birlikte başlatır; bu özniteliğin ayrıntıları için [“Kuyruklara ilişkin öznitelikler” sayfa 811](#) ' 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.

QName (MQCHAR48)

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 811](#) ' 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.

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.

TriggerData (MQCHAR64)

Bu, tetikleme iletisini alan, tetikleme izleme uygulaması tarafından kullanılacak serbest biçimli verilerdir. Kuyruk yöneticisi, bu alanı, *QName* alanı tarafından tanımlanan kuyruğun *TriggerData* özniteliğinin değeriyle birlikte başlatır; bu özniteliğin ayrıntıları için “[Kuyruklara ilişkin öznitelikler](#)” sayfa 811 ' e bakın. Bu verilerin içeriği kuyruk yöneticisi için bir önem göstermiyor.

z/OSüzerinde, 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.

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 844 ' 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 **runmqtm**.

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.

Sürüm (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.

MQTM için ilk değerler ve dil bildirimleri

Çizelge 93. MQTM için MQTM ' deki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
<i>StrucId</i>	MQTM_STRUC_ID	' TM-- '
<i>Version</i>	MQTM_VERSION_1	1

Çizelge 93. MQTM için MQTM 'deki alanların ilk değerleri (devamı var)

Alan adı	Değişmezin adı	Değişmezin değeri
QName	Yok	Dizgi ya da boşluk boş değerli
ProcessName	Yok	Dizgi ya da boşluk boş değerli
TriggerData	Yok	Dizgi ya da boşluk boş değerli
ApplType	Yok	0
ApplId	Yok	Dizgi ya da boşluk boş değerli
EnvData	Yok	Dizgi ya da boşluk boş değerli
UserData	Yok	Dizgi ya da boşluk boş değerli

Notlar:

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.
3. C programlama dilinde, makro değişkeniMQTM_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlara ilişkin ilk değerleri sağlamak için bunu aşağıdaki şekilde kullanın:

```
MQTM MyTM = {MQTM_DEFAULT};
```

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 */
};
```

COBOL bildirimi

```
**      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).
```

PL/I bildirimi

```

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 */

```

High Level Assembler bildirimi

```

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)

```

Visual Basic bildirimi

```

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

```

MQTMC2 -Tetikleyici iletisi 2 (karakter biçimi)

Aşağıdaki çizelge yapıdaki alanları özetlemektedir.

Çizelge 94. MQTMC2içindeki alanlar		
Alan	Tanım	Konu
<i>StrucId</i>	Yapı tanıtıcısı	StrucId
<i>Version</i>	Yapı sürüm numarası	S\u00f0fcr\u00f0cm
<i>QName</i>	Tetiklenen kuyruğun adı	QName
<i>ProcessName</i>	Süreç nesnesinin adı	ProcessName
<i>TriggerData</i>	Verileri tetikle	TriggerData
<i>ApplType</i>	Uygulama tipi	ApplType
<i>ApplId</i>	Uygulama tanıtıcısı	ApplId
<i>EnvData</i>	Ortam verileri	EnvData
<i>UserData</i>	Kullanıcı verileri	UserData

Çizelge 94. MQTMC2 içindeki alanlar (devamı var)		
Alan	Tanım	Konu
QMgrName	Kuyruk yöneticisi adı	QMgrName

MQTMC2'ye Genel Bakış

Amaç: Bir tetikleme izleme programı bir başlatma kuyruğundan bir tetikleme iletisi (MQTM) aldığı anda, tetikleme izleyicisinin tetikleyici iletisinde bulunan bilgilerin bir kısmını ya da tümünü tetikleme izleyicisinin başlattığı uygulamaya geçirmesi gerekebilir.

Information that the started application might need includes *QName*, *TriggerData*, and *UserData*. 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 özniteliği tarafından verilen karakter kümesidir.

Kullanım: MQTMC2 yapısı, MQTM yapısının biçimine çok benzerdir. Bu fark, MQTM 'deki karakter olmayan alanların MQTMC2 'de aynı uzunluktaki karakter alanlarına çevrilir ve yapının sonuna kuyruk yöneticisi adı eklenmektedir.

- z/OS üzerinde, CSQQTRMN uygulaması kullanılarak başlatılan bir MQAT_IMS uygulaması için, başlatılan uygulama için bir MQTMC2 yapısı kullanılabilir.
- IBM üzerinde, IBM MQ ile verilen tetikleyici izleyicisi uygulaması, başlatılan uygulamaya bir MQTMC2 yapısı iletir.

MQTMC2 alanları

MQTMC2 yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

ApplId (MQCHAR256)

Uygulama tanıtıcısı.

MQTM yapısındaki *ApplId* alanına bakın.

ApplType (MQCHAR4)

Uygulama tipi.

Bu alanda her zaman boşluk, özgün tetikleyici iletisinin MQTM yapısındaki *ApplType* alanındaki değer ne olursa olsun bulunur.

EnvData (MQCHAR128)

Ortam verileri.

MQTM yapısındaki *EnvData* alanına bakın.

ProcessName (MQCHAR48)

Süreç nesnesinin adı.

MQTM yapısındaki *ProcessName* alanına bakın.

QMgrName (MQCHAR48)

Kuyruk yöneticisi adı.

Bu, tetikleme olayının ortaya çıktığı kuyruk yöneticisinin adıdır.

QName (MQCHAR48)

Tetiklenen kuyruğun adı.

MQTM yapısındaki *QName* alanına bakın.

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.

TriggerData (MQCHAR64)

Verileri tetikler.

MQTM yapısındaki *TriggerData* alanına bakın.

UserData (MQCHAR128)

Kullanıcı verileri.

MQTM yapısındaki *UserData* alanına bakın.

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.

MQTMC2 için ilk değerler ve dil bildirimleri

Çizelge 95. Initial values of fields in MQTMC2 for MQTMC2		
Alan adı	Değişmezin adı	Değişmezin değeri
<i>StrucId</i>	MQTMC_STRUC_ID	'TMC↵'
<i>Version</i>	MQTMC_VERSION_2	'↵↵↵2'
<i>QName</i>	Yok	Dizgi ya da boşluk boş değerli
<i>ProcessName</i>	Yok	Dizgi ya da boşluk boş değerli
<i>TriggerData</i>	Yok	Dizgi ya da boşluk boş değerli
<i>ApplType</i>	Yok	Boşluklar
<i>ApplId</i>	Yok	Dizgi ya da boşluk boş değerli
<i>EnvData</i>	Yok	Dizgi ya da boşluk boş değerli
<i>UserData</i>	Yok	Dizgi ya da boşluk boş değerli

Çizelge 95. Initial values of fields in MQTMC2 for MQTMC2 (devamı var)

Alan adı	Değişmezin adı	Değişmezin değeri
QMgrName	Yok	Dizgi ya da boşluk boş değeri
Notlar: ↪ simgesi tek bir boş karakteri temsil eder. - Boş değeri dizgi ya da boşluklar, C içindeki boş değeri dizeyi ve diğer programlama dillerindeki boş karakterleri belirtir. - C programlama dilinde, makro değişkeniMQTMC2_DEFAULT , yukarıda listelenen değerleri içerir. Yapıdaki alanlara ilişkin ilk değerleri sağlamak için bunu aşağıdaki şekilde kullanın: <pre>MQTMC2 MyTMC = {MQTMC2_DEFAULT};</pre>		

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 */
};
```

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).
```

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 */

```

High Level Assembler bildirimi

```

MQTMC          DSECT
MQTMC_STRUCID  DS    CL4      Structure identifier
MQTMC_VERSION  DS    CL4      Structure version number
MQTMC_QNAME    DS    CL48     Name of triggered queue
MQTMC_PROCESSNAME DS    CL48   Name of process object
MQTMC_TRIGGERDATA DS    CL64   Trigger data
MQTMC_APPLTYPE DS    CL4      Application type
MQTMC_APPLID   DS    CL256    Application identifier
MQTMC_ENVDATA  DS    CL128    Environment data
MQTMC_USERDATA DS    CL128    User data
MQTMC_QMGRNAME DS    CL48     Queue manager name
*
MQTMC_LENGTH   EQU    *-MQTMC
                ORG    MQTMC
MQTMC_AREA     DS    CL(MQTMC_LENGTH)

```

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

```

MQWIH-İş bilgisi üstbilgisi

Aşağıdaki çizelge yapıdaki alanları özetlemektedir.

Çizelge 96. MQWIH içindeki alanlar		
Alan	Tanım	Konu
<i>StrucId</i>	Yapı tanıtıcısı	StrucId
<i>Version</i>	Yapı sürüm numarası	S\u00fcr\u00fcm
<i>StrucLength</i>	MQWIH yapısının uzunluğu	StrucLength
<i>Encoding</i>	MQWIH ' u izleyen verilerin sayısal kodlaması	Kodlama
<i>CodedCharSetId</i>	MQWIH ' u izleyen verilerin karakter kümesi tanıtıcısı	CodedCharSetId
<i>Format</i>	MQWIH ' u izleyen verilerin biçimi	Biçim
<i>Flags</i>	İşaretler	İşaretler
<i>ServiceName</i>	Hizmet adı	ServiceName
<i>ServiceStep</i>	Hizmet adımları adı	ServiceStep
<i>MsgToken</i>	İleti simgesi	MsgToken
<i>Reserved</i>	Ayrıldı	Ayrıldı

MQWIIH için genel bakış

MQWIIH yapısı, z/OS iş yükü yönetimine (WLM) gönderilen bir iletiye ilişkin üstbilgi bilgilerini açıklar. Desteklenen herhangi bir IBM MQ platformu için, MQWIIH yapısını içeren bir ileti yaratabilir ve bu iletiyi iletebilirsiniz; ancak, yalnızca bir IBM MQ for z/OS kuyruk yöneticisi WLM ile etkileşimde bulunabilir. Bu nedenle, iletininiz/OS olmayan bir kuyruk yöneticisinden WLM 'ye geçebilmesi için, kuyruk yöneticisi ağınızın, iletinin yönlendirilebileceği en az bir z/OS kuyruk yöneticisi içermesi gerekir.

Kullanılabilirlik: Tüm IBM MQ sistemleri, ayrıca bu sistemlere bağlı IBM MQ istemcileri.

Amaç: MQWIIH yapısı, z/OS iş yükü yöneticisi tarafından işlenecek bir iletinin başlangıcında var olması gereken bilgileri açıklar.

Biçim adı: MQFMT_WORK_INFO_HEADER.

Karakter kümesi ve kodlama: MQWIIH yapısındaki alanlar, MQWIIH 'den önce gelen üstbilgi yapısındaki *CodedCharSetId* ve *Encoding* alanları tarafından verilen karakter kümesi ve kodlamadır ve MQWIIH uygulama iletisi verilerinin başlangıcındaki bu alanlar tarafından MQWIIH 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 MQWIIH structure.

MQWIIH alanları

MQWIIH yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

CodedCharSetId (MQHOMER)

Bu, MQWIIH yapısını izleyen verilerin karakter kümesi tanıtıcısını belirtir; bu, MQWIIH 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

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.

Kodlama (MQUZE)

Bu, MQWIIH yapısını izleyen verilerin sayısal kodlamasını belirtir; bu, MQWIIH 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.

İşaretler (MQUZE)

Değer şu olmalıdır:

MQWIIH_NONE

Bayrak yok.

Bu alanın ilk değeri MQWIIH_NONE olur.

Biçim (MQCHAR8)

Bu, MQWIIH yapısından sonra gelen verilerin biçim adını belirler.

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 uzunluğu MQ_FORMAT_LENGTH tarafından verilir. Bu alanın ilk değeri MQFMT_NONE olur.

MsgToken (MQBYTE16)

Bu, iletiyi benzersiz bir şekilde tanımlayan bir ileti belirtecidir.

MQPUT 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.

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.

StrucId (MQCHAR4)

Bu, yapı tanıtıcısıdır. Değer şu olmalıdır:

MQWIIH_STRUC_ID

İş bilgileri üstbilgi yapısına ilişkin tanıtıcı.

C programlama dili için, MQWIIH_STRUC_ID_ARRAY değişmezi de tanımlanır; bu, MQWIIH_STRUC_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisidir.

Bu alanın ilk değeri MQWIIH_STRUC_ID ' dir.

StrucLength (MQUZE)

Bu, MQWIIH yapısının uzunluğudur. Değer şu olmalıdır:

MQWIIH_LENGTH_1

version-1 iş bilgileri üstbilgisi yapısı uzunluğu.

Aşağıdaki değişmez, yürürlükteki sürümün uzunluğunu belirtir:

MQWIIH_CURRENT_LENGTH

İş bilgisi üstbilgisi yapısının geçerli sürümü uzunluğu.

Bu alanın ilk değeri MQWIIH_LENGTH_1 ' dir.

Sürüm (MQUZE)

Bu, yapı sürüm numarasıdır. Değer şu olmalıdır:

MQWIIH_VERSION_1

Version-1 iş bilgileri üstbilgi yapısı.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

MQWIIH_CURRENT_VERSION

İş bilgisi üstbilgi yapısının yürürlükteki sürümü.

Bu alanın ilk değeri MQWIIH_VERSION_1 ' dir.

MQWIH için ilk değerler ve dil bildirimleri

Çizelge 97. MQWIH için MQWIH içindeki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
StrucId	MQWIH_STRUC_ID	'WIH¬'
Version	MQWIH_VERSION_1	1
StrucLength	MQWIH_LENGTH_1	120
Encoding	Yok	0
CodedCharSetId	MQCCSI_UNDEDINED	0
Format	MQFMT_NONE	Boşluklar
Flags	MQWIH_NONE	0
ServiceName	Yok	Boşluklar
ServiceStep	Yok	Boşluklar
MsgToken	MQMTOK_NONE	Boş Değerler
Reserved	Yok	Boşluklar
Notlar:		
1. ¬ simgesi tek bir boş karakteri temsil eder.		
2. C programlama dilinde, makro değişkeniMQWHI_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlara ilişkin ilk değerleri sağlamak için bunu aşağıdaki şekilde kullanın:		
<pre>MQWIH MyWIH = {MQWIH_DEFAULT};</pre>		

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 */
};
```

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).

```

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 */

```

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)

```

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'
              'MQWIH'
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

```

MQXP-Çıkış deęiřtirgesi öbeęi

Ařaęıdaki çizelge yapıdaki alanları özetlemektedir.

Çizelge 98. MQXP ' deki alanlar		
Alan	Tanım	Konu
<i>StrucId</i>	Yapı tanıtıcısı	StrucId
<i>Version</i>	Yapı sürüm numarası	S\u00fcr\u00fcm
<i>ExitId</i>	Çıkış tanıtıcısı	ExitId
<i>ExitReason</i>	Çıkışa çağırma nedeni	ExitReason
<i>ExitResponse</i>	Çıkıştan yanıt	ExitResponse
<i>ExitCommand</i>	API çağrı kodu	ExitCommand
<i>ExitParmCount</i>	Parametre sayısı	ExitParmSayı
<i>ExitUserArea</i>	Kullanıcı alanı	ExitUserAlanı

MQXP ' ye Genel Bakış

Kullanılabilirlik: z/OS.

Amaç: MQXP yapısı, API geçiř çıkışına giriř/çıkış parametresi olarak kullanılır. Bu çıkışa iliřkin ek bilgi için bkz. [API geçidi çıkışı](#).

Karakter kümesi ve kodlama: MQXP ' deki karakter verileri, yerel kuyruk yöneticisinin karakter takımında yer alıyor; bu, *CodedCharSetId* kuyruk yöneticisi öznitelięi tarafından verilir. MQXP ' deki sayısal veriler yerel makine kodlamasıdır; bu, MQENC_NATIVE tarafından verilir.

MQXP için alanlar

MQXP yapısı ařaęıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

ExitCommand (MQUZA)

Bu alan, çıkış yordamlarına giriř olarak ayarlanır. Çıkışa çağırılmasına 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_MQAC

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.

ExitId (MQUZE)

Bu, çıkış yordamlarına giriş olarak ayarlanır ve çıkış tipini belirtir:

MQXT_API_CROSSING_EXIT

CICS'in API geçiş çıkışı.

Bu, çıkışa giriş alanıdır.

ExitParmCount (MQLONG)

Bu alan, çıkış yordamlarına giriş olarak ayarlanır. Bu, MQ çağrısının aldığı parametrelerin sayısını içerir. Bu bilgiler şunlardır:

Arama adı	Parametre sayısı
MQBACK	3
MQCLOSE	5
MQCMIT	3
MQGet	9
MQINQ	10
MQOPEN	6
MQPUT	8
MQPUT1	8
MQSET	10

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, çağırın ile iletişim kurmak 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.

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.

Ayrılmış (MQUZE)

Bu ayrılmış bir alandır. Çıkış için değeri önemli değil.

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.

Dil bildirimleri

Bu yapı, aşağıdaki programlama dillerinde desteklenir.

C bildirimi

```
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 */
};
```

COBOL bildirimi

```
** 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).
```

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 */
```

High Level Assembler 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
```

MQXQH-İletim kuyruğu üstbilgisi

Aşağıdaki çizelge yapıdaki alanları özetlemektedir.

Çizelge 99. MQXQH içindeki alanlar		
Alan	Tanım	Konu
<i>StrucId</i>	Yapı tanıttıcısı	StrucId
<i>Version</i>	Yapı sürüm numarası	S\u00fcr\u00fcm
<i>RemoteQName</i>	Hedef kuyruğun adı	RemoteQName
<i>RemoteQMgrName</i>	Hedef kuyruk yöneticisinin adı	RemoteQMgrAd
<i>MsgDesc</i>	Özgün ileti tanımlayıcısı	MsgDesc

MQXQH 'ye Genel Bakış

Kullanılabilirlik: Tüm IBM MQ sistemleri ve IBM MQ istemcileri.

Amaç: MQXQH yapısı, ileti kuyruklarında yer alan iletilerin uygulama iletileri verilerine örnek olarak eklenmiş bilgileri açıklar. İletim kuyruğu, uzak kuyruklara ilişkin iletileri geçici olarak tutan (yerel kuyruk yöneticisine ait olmayan kuyruklar için yazılmış olan) özel bir yerel kuyruktur. A transmission queue is denoted by the *Usage* queue attribute having the value MQUS_TRANSMISSION.

Biçim adı: MQFMT_XMIT_Q_HEADER.

Karakter kümesi ve kodlama: MQXQH içindeki veriler, *CodedCharSetId* kuyruk yöneticisi özniteliği tarafından verilen karakter kümesinde ve MQENC_NATIVE tarafından verilen yerel kuyruk yöneticisinin kodlamasıyla olmalıdır.

MQXQH 'nin karakter kümesini ve kodlamasını *CodedCharSetId* ve *Encoding* alanlarına aşağıdaki şekilde kodlayın:

- 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 *MsgDesc* 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 iletileri verileri (varsa) sırası gelir. MQMDE aşağıdakilerden biri olabilir:

- Kuyruk yöneticisi tarafından üretilir (uygulama iletiyi koymak için bir version-2 MQMD kullanıyorsa) ya da
- Uygulama iletileri verilerinin başında zaten var (uygulama iletiyi koymak için bir version-1 MQMD kullanıyorsa).

Katıştırılmış ileti tanımlayıcısı, ileti son hedef kuyruğundan kaldırıldığında, MQGET çağrısının *MsgDesc* parametresindeki uygulamaya döndürülen bir tanımlayıcıdır.

Ayrı ileti tanımlayıcısındaki alanlar: Ayrı ileti tanımlayıcısındaki alanlar, kuyruk yöneticisi tarafından gösterildiği gibi ayarlanır. Kuyruk yöneticisi version-2 MQMD 'yi desteklemiyorsa, işlev kaybı olmadan bir version-1 MQMD kullanılır.

Ayrı MQMD ' de alan	Kullanılan değer
<i>StrucId</i>	MQMD_STRUC_ID
<i>Version</i>	MQMD_VERSION_2
<i>Report</i>	Gömülü ileti tanımlayıcısından kopyalandı, ancak MQRO_ACCEPT_UNSUP_IF_XMIT_MASK tarafından tanımlanan bitlerle sıfır olarak ayarlandı. (Bu işlem, bir ileti yerleştirildiğinde ya da iletim kuyruğundan kaldırıldığında, COA ya da COD (COD) rapor iletisinin oluşturulmasını önler.)
<i>MsgType</i>	Katıştırılmış ileti tanımlayıcısından kopyalandı.
<i>Expiry</i>	Katıştırılmış ileti tanımlayıcısından kopyalandı.
<i>Feedback</i>	Katıştırılmış ileti tanımlayıcısından kopyalandı.
<i>Encoding</i>	MQENC_NATIVE (nota bakın)
<i>CodedCharSetId</i>	Kuyruk yöneticisinin <i>CodedCharSetId</i> özneliği.
<i>Format</i>	MQFMT_XMIT_Q_HEADER
<i>Priority</i>	Katıştırılmış ileti tanımlayıcısından kopyalandı.
<i>Persistence</i>	Katıştırılmış ileti tanımlayıcısından kopyalandı.
<i>MsgId</i>	Kuyruk yöneticisi yeni bir değer üretti. Bu ileti tanıtıcısı, kuyruk yöneticisinin önceden açıklanan yerleşik ileti tanımlayıcısı için oluşturduğu <i>MsgId</i> ' den 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>	Katıştırılmış ileti tanımlayıcısından kopyalandı.
<i>ReplyToQMgr</i>	Katıştırılmış ileti tanımlayıcısından kopyalandı.
<i>UserIdentifier</i>	Katıştırılmış ileti tanımlayıcısından kopyalandı.
<i>AccountingToken</i>	Katıştırılmış ileti tanımlayıcısından kopyalandı. SYSTEM.CLUSTER.TRANSMIT.QUEUE, <i>AccountingToken</i> iç kullanım için ayrılmıştır.
<i>ApplIdentityData</i>	Katıştırılmış ileti tanımlayıcısından kopyalandı.
<i>PutApplType</i>	MQAT_QMGR
<i>PutApplName</i>	Kuyruk yöneticisi adının ilk 28 byte 'ı.
<i>PutDate</i>	İletin iletim kuyruğuna konduğu tarih.
<i>PutTime</i>	İletin iletim kuyruğuna konduğu saat.
<i>ApplOriginData</i>	Boşluklar
<i>GroupId</i>	MQGI_NONE
<i>MsgSeqNumber</i>	1
<i>Offset</i>	0

Ayrı MQMD ' de alan	Kullanılan değer
<i>MsgFlags</i>	MQMF_NONE
<i>OriginalLength</i>	MQOL_TANIMSIZ

- Windowsüzerinde, Micro Focus COBOL için MQENC_NATIVE değeri C değerinin değerinden farklıdır. Ayrı ileti tanımlayıcısındaki *Encoding* alanındaki değer, bu ortamlardaki C için her zaman değer olur; bu değer ondalık olarak 546 değeridir. Ayrıca, MQXQH yapısındaki tamsayı alanları, bu değere karşılık gelen kodlamadır (yerel Intel kodlaması).

Yerleşik ileti tanımlayıcısındaki alanlar: Gömülü ileti tanımlayıcısındaki alanların değerleri, aşağıdaki durumlar dışında, MQPUT ya da MQPUT1 çağrısının *MsgDesc* değiştirgesindeki değerlerle aynı olur:

- *Version* alanı her zaman MQMD_VERSION_1 değerine sahiptir.
- *Priority* alanında MQPRI_PRIORITY_AS_Q_DEF değeri varsa, bu değer, kuyruğun *DefPriority* özniteliğinin değeriyle değiştirilir.
- *Persistence* alanında MQPER_PERSISTENCE_AS_Q_DEF değeri varsa, bu değer, kuyruğun *DefPersistence* özniteliğinin değeriyle değiştirilir.
- *MsgId* alanında MQMI_NONE değeri varsa ya da MQPMO_NEW_MSG_ID seçeneği belirtildiyse ya da ileti bir dağıtım listesi iletiyse, *MsgId* , 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 *MsgId* alanı, özgün dağıtım listesi iletisinde olduğu gibi aynıdır.

- MQPMO_NEW_CORREL_ID seçeneği belirtilmişse, *CorrelId* , kuyruk yöneticisi tarafından oluşturulan yeni bir ilinti tanıtıcısıyla değiştirilir.
- Bağlam alanları, *PutMsgOpts* değiştirgesinde belirtilen MQPMO_*_CONTEXT seçenekleri tarafından belirtildiği gibi ayarlanır; bağlam alanları şunlardır:

- *AccountingToken*
- *ApplIdentityData*
- *ApplOriginData*
- *PutApplName*
- *PutApplType*
- *PutDate*
- *PutTime*
- *UserIdentifier*

- 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 yerleştirme: 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

İletileri doğrudan iletim kuyruklarına yerleştirme: Bir uygulama, doğrudan iletim kuyruğuna bir ileti de yerleştirebilir. Bu durumda, uygulama, uygulama iletisi verilerinin bir MQXQH yapısıyla önekinde ve alanları uygun değerlerle kullanıma hazırlamalıdır. Buna ek olarak, MQPUT ya da MQPUT1 çağrısının *MsgDesc* parametresindeki *Format* alanı MQFMT_XMIT_Q_HEADER değerine sahip olmalıdır.

Uygulama tarafından yaratılan MQXQH yapısındaki karakter verileri, yerel kuyruk yöneticisinin karakter kümesinde (*CodedCharSetId* kuyruk yöneticisi öznitelemesiyle tanımlanır) ve tamsayı verilerinin yerel makine kodlamasında olması gerekir. 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 yöneticisi, bir MQXQH yapısının var olduğunu ya da alanlar için geçerli değerlerin belirtildiğini denetmez.

Uygulamalar, iletilerini doğrudan SYSTEM.CLUSTER.TRANSMIT.QUEUE.

İletim kuyruklarından ileti alınması: İletim kuyruğundan ileti alan uygulamalar, MQXQH yapısındaki bilgileri uygun bir şekilde işlemelidir. Uygulama iletileri verilerinin başında MQXQH yapısının bulunması, MQGET çağrısının *MsgDesc* parametresindeki *Format* alanında MQFMT_XMIT_Q_HEADER değerinin döndürülmesiyle belirtilir. *MsgDesc* parametresindeki *CodedCharSetId* ve *Encoding* alanlarında döndürülen değerler, MQXQH yapısındaki karakter ve tamsayı verisinin karakter kümesini ve kodlamasını gösterir. Uygulama iletileri verilerinin karakter takımı ve kodlaması, yerleşik ileti tanımlayıcısında *CodedCharSetId* ve *Encoding* alanları tarafından tanımlanır.

MQXQH alanları

MQXQH yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

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.

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.

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 iletisiyse (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.

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.

MQXQH için ilk değerler ve dil bildirimleri

Çizelge 100. MQXQH için MQXQH içindeki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
StrucId	MQXQH_STRUC_ID	'XQH↵'
Version	MQXQH_VERSION_1	1
RemoteQName	Yok	Dizgi ya da boşluk boş değerli
RemoteQMgrName	Yok	Dizgi ya da boşluk boş değerli
MsgDesc	MQMD ile aynı adlar ve değerler; bkz. Çizelge 55 sayfa 449	-
Notlar: 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. 3. C programlama dilinde, makro değişkeniMQXQH_DEFAULT, çizelgede listelenen değerleri içerir. Yapıdaki alanlara ilişkin ilk değerleri sağlamak için bunu aşağıdaki şekilde kullanın: <pre>MQXQH MyXQH = {MQXQH_DEFAULT};</pre>		

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 */  
};
```

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).

```

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 */

```

High Level Assembler bildirimi

```

MQXQH                                DSECT
MQXQH_STRUCID                        DS CL4 Structure identifier
MQXQH_VERSION                        DS F Structure version number
MQXQH_REMOTEQNAME                    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
MQXQH_MSGDESC_AREA                    DS CL(MQXQH_MSGDESC_LENGTH)
*
MQXQH_LENGTH                          EQU *-MQXQH
MQXQH_AREA                            DS CL(MQXQH_LENGTH)

```

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

```

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 çağrılarının her biri için verilmiştir.

Arama açıklamaları

Bu bölümde, MQI çağrılarını açıklanır.

- [“MQBACK-Değişiklikleri geri al” sayfa 608](#)
- [“MQBEGIN-İş birimini başlat” sayfa 612](#)
- [“MQBUFMH-Arabelleği ileti tutamaçına dönüştür” sayfa 615](#)
- [“MQCB-Geri çağırmaya yönet” sayfa 619](#)
- [“MQCB_FUNCTION-Callback işlevi” sayfa 628](#)
- [“MQCLOSE-Nesneyi kapat” sayfa 630](#)
- [“MQCMIT-Kesinleştirme değişiklikleri” sayfa 637](#)
- [“MQCONN-Bağlantı kuyruğu yöneticisi” sayfa 641](#)
- [“MQCONN-Connect kuyruk yöneticisi \(genişletilmiş\)” sayfa 649](#)
- [“MQCRTMH-İleti tanıtıcısı yarat” sayfa 655](#)
- [“MQCTL-Denetim çağrılarını” sayfa 658](#)
- [“MQDISC-Bağlantı kesme kuyruğu yöneticisi” sayfa 665](#)
- [“MQDLTMH-İleti tanıtıcısını sil” sayfa 668](#)
- [“MQDLTMP-İleti özelliğini sil” sayfa 671](#)
- [“MQGET-İletiyi al” sayfa 673](#)
- [“MQINQ-Nesne özniteliklerinin sorgulaması” sayfa 686](#)
- [“MQINQMP-Questery ileti özelliği” sayfa 703](#)
- [“MQMHBUFF-İleti tanıtıcısını arabelleğe dönüştür” sayfa 708](#)
- [“MQOpen-Open nesnesi” sayfa 712](#)
- [“MQPUT-İleti koy” sayfa 730](#)
- [“MQPUT1 -Bir ileti girin” sayfa 743](#)
- [“MQSET-Nesne özniteliklerinin ayarlanması” sayfa 753](#)
- [“MQSETMP-İleti özelliğini ayarla” sayfa 760](#)
- [“MQSTAT-Durum bilgilerini al” sayfa 764](#)
- [“MQMHBUFF-İleti tanıtıcısını arabelleğe dönüştür” sayfa 708](#)
- [“MQSUB-Abonelikleri kaydet” sayfa 768](#)
- [“MQSUBRQ-Abonelik isteği” sayfa 775](#)

Bu çağrılar için, *kişi* sayfaları biçiminde UNIX platformları' de çevrimiçi yardım olanağı sağlanır.

Not: Veri dönüştürme, MQXCNCV ve MQ_DATA_CONV_EXIT ile ilişkili çağrılar [“Veri dönüştürme” sayfa 879](#) 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 236](#) içindeki veri türü ya da yapısıyla ilgili daha fazla bilgi 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. Her bir tamamlanma ve neden koduyla ilgili daha fazla bilgi için bkz. [“Dönüş kodları” sayfa 847](#).

Kullanım notları

Çağrıyla 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 bildirimi, 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 bildirimi 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 adresi değeri değer olarak geçirilir.

Bir işlevi her çağırdığınızda adresle geçirilen tüm parametreleri belirtmenize gerek yoktur. Belirli bir değiştirmeye gerek olmadığı durumlarda, parametre verilerinin adresi yerine, işlev çağırımında değiştirge olarak boş değerli bir gösterge belirtin. Bunun olası olduğu parametreler, çağrı açıklamalarında tanımlanır.

Çağrının değeri olarak bir parametre döndürülmez; C terminolojisinde, bu, tüm çağrılarının void döndürdüğü anlamına gelir.

Arabellek parametresinin bildirilmesi

MQGET , MQPUT ve MQPUT1 çağrılar, her birinin tanımlı olmayan bir veri tipine sahip bir parametresi vardır: *Buffer* parametresi. Uygulamanın ileti verilerini göndermek ve almak için bu değıştirgeyi kullanın.

Bu sıralamayı içeren değıştirgeler, 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.
- IBM işletim sistemi, uyumluluk kipinde çalışan uygulamalar için bu çağrıyı desteklemez.

Önemli:  Uyumluluk kipini kullanmaya devam etseniz, uygulamanızı yeniden derlemeyi ve bunun yerine statik bağlı çağrılar kullanmayı düşünün. Bkz. [IBM üzerindeki RPG uygulamaları için uyumluluk kipi](#)

Sözdizimi

MQBACK (*Hconn*, *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.

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_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 çağırısı.

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 çağırıcı, 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 612.](#)

2. Kuyruk yöneticisinin iş birimini koordine etmediği ortamlarda, MQBACK yerine uygun geri çağırma çağırısını kullanın. Ortam, uygulamanın olağan dışı bir şekilde sonlandırılmasına neden olan örtük bir geri dönüş özelliğini de destekleyebilir.
- z/OS' ta aşağıdaki çağırıcıları 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 Kurtarılabilir 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 665](#) 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 bir iletiyi göz atlayan son başarılı MQGET çağrısı (bu *olamaz* bir iş biriminin parçası olabilir).
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 yüklemeyiz.
- 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 seçenekleri” sayfa 482](#) içinde açıklanan MQPMO_LOGICAL_ORDER seçeneğine ve [“MQGMO-Get-message seçenekleri” sayfa 358](#) 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 641](#) 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, yöneticinin *MaxUncommittedMsgs* queue-manager özneliğini, kuyrukları dolduran kaçak uygulamaları önleyecek kadar düşük bir değere ayarlaması gerekir, ancak beklenen ileti sistemi uygulamalarının doğru şekilde çalışmasına izin verecek kadar yüksek.

C çağırma

```
MQBACK (Hconn, &CompCode, &Reason);
```

Değiştirelileri 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.
```

Değiştirelileri 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);
```

Değiştirelileri 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)
```

Değiştirelileri aşağıdaki gibi bildirin:

```
HCONN      DS F Connection handle
COMPCODE   DS F Completion code
REASON     DS F Reason code qualifying COMPCODE
```

Visual Basic çağrısı

```
MQBACK 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'
```

MQBEGIN-İş birimini başlat

MQBEGIN çağrısı, 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 tanıttıcı, kuyruk yöneticisiyle olan bağlantıyı gösterir. The value of *Hconn* was returned by a previous MQCONN or MQCONNX call.

Hconn , paylaşılmayan bir bağlantı tanıttıcısı olmalıdır. Paylaşılan bir bağlantı tanıttıcısı 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-Başlat seçenekleri” sayfa 277’inde 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 neden kodlarına 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 bir 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 MQRC_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 MQRC_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 665](#) içindeki kullanım notlarına bakın.
- 3. Bir uygulama, aynı anda yalnızca bir iş birimine katılabilir. Uygulamaya ilişkin var olan iş birimi tipine bakılmaksızın, MQRC_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 MQRC_ENVIRONMENT_ERROR.
- 5. Kuyruk yöneticisi, genel iş birimleri için iş birimi eşgüdümçüsü olarak işlev gördüğünde, iş birimine katılabilen kaynak yöneticileri kuyruk yöneticisi yapıları 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 yöneticisiyle 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.

Hmsg

Tip: MQHMSG-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-Arabelleğiyle ileti tanıtıcısı seçenekleri” sayfa 275.](#)

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-giriş

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 612](#) içinde 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 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_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 ayrıştırılamadı.

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 çağrıları API çıkışlarıyla algılanamaz; arabellek, uygulama alanındaki bir ileti tanıtıcısı içine dönüştürülür; çağrı kuyruk yöneticisine ulaşmaz.

C çağırma

```
MQBUFMMH (Hconn, Hmsg, &BufMsgHOpts, &MsgDesc, BufferLength, Buffer,  
           &DataLength, &CompCode, &Reason);
```

Değiştirgeleri 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 çağırısı, 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, and on IBM i for applications running in compatibility mode, you can specify the following special value for *MQHC_DEF_HCONN* to use the connection handle associated with this execution unit.

Önemli:  Uyumluluk kipini kullanmaya devam etseniz, uygulamanızı yeniden derlemeyi ve bunun yerine statik bağlı çağrılarını kullanmayı düşünün. Bkz. [IBM üzerindeki RPG uygulamaları için uyumluluk kipi](#)

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 değişmezi 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 değiştirilir. Geri çağırma değiştirilirken bir hata saptanırsa, işlev kayıttan kaldırılır.

Bir geri çağrı, daha önce kaydı kaldırıldığı aynı geri çağırma işlevinde kaydedilirse, bu bir değiştirme 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 tanıtıcıyı 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 seçimlerle aynı kuyruğa kayıt yaptırsa, her ileti için seçilen tüketici tanımsız olduğunu unutmayın.

GetMsgSeçenekleri

Tip: MQGMO-input

GetMsgOpts parametresi, ileti tüketicisinin iletileri nasıl alacağını denetler. Bu değiştirgenin tüm seçenekleri, bir MQGET çağrısında kullanıldığında, “MQGMO-Get-message seçenekleri” sayfa 358’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 yoklaşmasını önler.

MQGMO_WAIT ve artı WaitInterval

Tüketici, belirtilen bekleme aralığından sonra çağrılır; neden kodu MQRC_NO_MSG_AVAILABLE. Bu arama, herhangi bir iletinin tüketiciye teslim edilip edilmediği dikkate alınmaksızın yapılır. Bu, kullanıcının sağlıklı işletim bildirimi 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 değiştirgenin 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.

MQRC_CALLBACK_NOT_REGISTERED

(2448, X' 990 ') Kayıtlı bir geri çağırım olmadığı için kaydı kaldırılamıyor, askıya alınamıyor ya da devam ettirilemiyor.

MQRC_CALLBACK_ROUTINE_ERROR

(2486, X'9B6') *CallbackFunction* ya da *CallbackName* belirtilmeli, ancak her ikisi belirtilmemelidir.

MQRC_CALLBACK_TYPE_ERROR

(2483, X'9B3') Yanlış geri çağırma tipi alanı.

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') İlintilendirme tanıtıcısı hatası.

MQRC_DATA_LENGTH_ERROR

(2010, X'7DA') Veri uzunluğu parametresi 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'9B4') Yanlış *MaxMsgLength* 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 değil ya da geçerli bir dinamik bağlantı kitaplığı.

MQRC_MODULE_NOT_FOUND

(2495, X'9BF') Modül, arama yolunda bulunamadı ya da yükleme yetkisine sahip değil.

MQRC_MSG_SEQ_NUMBER_ERROR

(2250, X'8CA') İleti sıra numarası geçerli değil.

MQRC_MSG_TOKEN_HATASI

(2331, X'91B') İleti simgesinin kullanımı geçerli değil.

MQRC_NO_MSG_AVAIABLE

(2033, X'7F1') İleti yok.

MQRC_NO_MSG_UNDER_CURSOR

(2034, X'7F2') Browse imleci iletide konumlandırılmamış.

MQRC_NOT_OPEN_FOR_BROWSE

(2036, X'7F4') Kuyruk göz atma için açık değil.

MQRC_NOT_OPEN_FOR_INPUT

(2037, X'7F5') Kuyruk giriş için açık değil.

MQRC_OBJECT_CHANGED

(2041, X'7F9') Nesne tanımlaması açıldığından beri değiştirildi.

MQRC_OBJECT_ZARAR

(2101, X'835 ') Nesne zarar gördü.

MQRC_OPERATION_HATASI

(2206, X'89E') API Çağrısında yanlış işlem kodu.

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_Q_DELETED

(2052, X'804 ') Kuyruk silinmiştir.

MQRC_Q_INDEX_TYPE_ERROR

(2394, X'95A') Kuyruk yanlış dizin tipi 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_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 kullanılabilecek 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 çağırmak 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 101. MQCTL yüklemi tanımlamaları	
Komut	Anlamı
MQCTL (BAŞLAT)	MQOP_START İşlemi kullanılarak MQCTL çağırısı

Çizelge 101. MQCTL yüklemi tanımlamaları (devamı var)	
Komut	Anlamı
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 çağırılır.

START

Her zaman MQCTL (START) komutu ile zamanuyumlu olarak çağırılır.

- 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.

İleti tüketici bağlantısı kullanımı

Tüketiciyi, tüketicinin yaşam çevrimi boyunca anahtar noktalarında geri çağırmaya çağırmak 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 102. MQCTL yüklemi tanımlamaları	
Komut	Anlamı
MQCTL (BAŞLAT)	MQOP_START İşlemi kullanılarak MQCTL çağırısı
MQCTL (DURDUR)	MQOP_STOP İşlemi kullanılarak MQCTL çağırısı

Çizelge 102. MQCTL yüklemi tanımlamaları (devamı var)	
Komut	Anlamı
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 parçacığında 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);
```

Değiştirelileri 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 Descriptor  
01 CBDESC.  
COPY CMQCBDV.  
01 HOBJ PIC S9(9) BINARY.  
** Message Descriptor  
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, and on IBM i for applications running in compatibility mode, the MQCONN call can be omitted, and the following value specified for *Hconn*:

MQHC_DEF_CONN

Varsayılan bağlantı tanıtıcısı.

Önemli:  Uyumluluk kipini kullanmaya devam etmeniz, uygulamanızı yeniden derlemeyi ve bunun yerine statik bağlı çağrılar kullanmayı düşünün. Bkz. [IBM üzerindeki RPG uygulamaları için uyumluluk kipi](#)

MsgDesc

Tip: MQMD-input

Bu yapı, alınan iletinin özniteliklerini tanımlar.

Ayrıntılar için bkz. “MQMD-İleti tanımlayıcısı” sayfa 404.

İletilen MQMD sürümü, tüketici işlevini tanımlayan MQCB çağrısına iletildiği 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: MQBYTEBufferUzunluğu-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-Callback bağlamı” sayfa 279.

Kullanım notları

1. Geri çağırım yordamlarınız iş parçacığının geciktirilebilecek ya da engellenecek hizmetleri kullanırsa, örneğin, MQGET ile bekleme süresi, diğer geri çağrıların dağıtımını 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 yayınlayabilir 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 nesneler 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 yayınlayabilir, 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 aldığıında, uygun işlemi yapması gerekir.
7. MQCB_FUNCTION, IMS bağdaştırıcısı içinde desteklenmiyor.

MQCLOSE-Nesneyi kapat

MQCLOSE çağrısı bir nesneye yeniden bağlantı verir ve MQOPED ve MQSUB çağrılarının tersi olur.

Sözdizimi

MQCLOSE (*Hconn*, *Hobj*, *Options*, *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, and on IBM i for applications running in compatibility mode, you can omit the MQCONN call, and specify the following value for *Hconn*:

MQHC_DEF_HCONN

Varsayılan bağlantı tanıtıcısı.

Hobj

Tip: MQHOBJ-giriş/çıkış

Bu tanıtıcı, kapatılmakta olan nesneyi gösterir. Nesne herhangi bir tipte olabilir. The value of *Hobj* 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:

MQHO_UNUSABLE_HOBJ

Kullanılamaz nesne tanıtıcısı.

z/OSüzerinde, *Hobj* tanımsız bir değere ayarlıdır.

Seçenekler

Tip: MQHOME-girişi

Bu parametre, nesnenin kapatılma şeklini denetler.

Yalnızca kalıcı dinamik kuyruklar ve abonelikler birden çok yolla kapatılabilir; bunlar alıkonmalı ya da silinmelidir; bunlar, MQQDT_PERMANENT_DYNAMIC değerine sahip *DefinitionType* özniteliğe sahip kuyruklar (“Kuyruklara ilişkin öznitelikler” sayfa 811 içinde açıklanan *DefinitionType* özniteliğine bakın). Kapatma seçenekleri bu konuda özetlenmiştir.

Sürekli abonelikler alıkonabilir ya da kaldırılabilir; bunlar MQSO_DAYİD seçeneği ile MQSUB çağrısı kullanılarak yaratılır.

Tanıtıcısı yönetilen bir hedefe kapatırken (MQSO_Managed seçeneğini kullanan bir MQSUB çağrısına döndürülen *Hobj* parametresidir) kuyruk yöneticisi, ilişkili abonelik de kaldırıldığında alınmamış tüm yayınları temizler. The subscription is removed using the MQCO_REMOVE_SUB option on the *Hsub* parameter returned on an MQSUB call. Not MQCO_REMOVE_SUB, kalıcı olmayan bir abonelik için MQCLOSE ' de varsayılan davranışın olduğunu unutmayın.

Yönetilmeyen bir hedefe yönelik bir tutamacı kapattığınızda, yayınların gönderildiği kuyruğu temizlemekten sorumlu olduğunuz bir hedef vardır. Önce MQCO_REMOVE_SUB komutunu kullanarak aboneliği kapatın ve iletileri kuyruktan kaldırmak için hiçbir işlem kalmayıncaya kadar kuyruğun dışına çıkarın.

Yalnızca aşağıdaki seçeneklerden birini belirlemelisiniz:

Dinamik kuyruk seçenekleri: Bu seçenekler, kalıcı dinamik kuyrukların nasıl kapatılacağını denetler.

MQCO_DELETE

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, *Hobj*döndüren MQOPEN çağrısıyla yaratılan geçici dinamik kuyruktır. Bu durumda, kuyrukta bulunan tüm iletiler temizlenir.

In all other cases, including the case where the *Hobj* was returned on an MQSUB call, the call fails with reason code MQRC_OPTION_NOT_VALID_FOR_TYPE, and the object is not deleted.

z/OS' da, kuyruk mantıksal olarak silinmiş bir dinamik kuyruksa ve bu kuyruk için son tanıtıcı ise, kuyruk fiziksel olarak silinmiştir. Ek ayrıntılar için "[Kullanım notları](#)" sayfa 635 ' e bakın.

MQCO_DELETE_PURGE

Kuyruk silinir ve aşağıdakilerden biri doğruysa, temizlenen iletiler de temizlenir:

- 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, *Hobj*döndüren MQOPEN çağrısıyla yaratılan geçici dinamik kuyruktır.

In all other cases, including the case where the *Hobj* was returned on an MQSUB call, the call fails with reason code MQRC_OPTION_NOT_VALID_FOR_TYPE, and the object is not deleted.

Çizelge, hangi kapanış seçeneklerinin geçerli olduğunu ve nesnenin alıkonulup silinmediğini ve silinip silinmeyeceğini gösterir.			
Nesne ya da kuyruk tipi	MQCO_NONE	MQCO_DELETE	MQCO_DELETE_PURGE
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, tanıtıcı kapatıldığında dayanıklı aboneliklerin kaldırılıp kaldırılmayacağını ve uygulamanın hala uygulama tarafından okunmayı beklemesinin temizlenip temizlenmediğini denetler. Bu seçenekler, yalnızca bir MQSUB çağrısının *Hsub* parametresindeki bir nesne tanıtıcısı ile kullanılmak üzere geçerlidir.

MQCO_KEEP_SUB

Aboneliğin tanıtıcısı kapatıldı, ancak yapılan abonelik alıkonur. Yayınlar, abonelikte belirtilen hedefe gönderilmeye devam eder. Bu seçenek yalnızca, abonelik MQSO_MUKATIAL seçeneğiyle birlikte yapıldıysa geçerlidir.

Abonelikte dayanıklı ise, MQCO_KEEP_SUB varsayılan değerdir

MQCO_REMOVE_SUB

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.

Abonelikte dayanıklı değilse, MQCO_REMOVE_SUB varsayılan değerdir.

MQCO_REMOVE_SUB ' in başarıyla tamamlanması, işlemin tamamlandığı anlamına gelmez.

Bu arama işleminin tamamlandığını denetlemek için [Dağıtılmış ağlara ilişkin zamanuyumsuz komutların sona erdiği denetleniyor](#) içindeki SUB SIL adımına bakın.

Bu abonelik kapatma seçenekleri aşağıdaki tablolarda özetlenmiştir.

Dayanıklı bir abonelik tanıtıcısını kapatmak, ancak aboneliği alıkoymak için aşağıdaki abonelik kapatma seçeneklerini kullanın:

Görev	Abonelik kapatma seçeneği
Yayınları MQOPENed tanıtıcısında tut	MQCO_KEEP_SUB
Bir MQOPENed tanıtıcısındaki yayınları kaldır	Eyleme izin verilmiyor
Yayınları MQSO_YÖNETİLEN tanıtıcısında tut	MQCO_KEEP_SUB
Bir MQSO_YÖNETİCİLER tanıtıcısındaki yayınları kaldır	Eyleme izin verilmiyor

Bir kalıcı abonelik tanıtıcısını kapatarak ya da kalıcı olmayan bir abonelik tanıtıcısını kapatarak aboneliği kaldırmak için, aşağıdaki abonelik kapatma seçeneklerini kullanın:

Görev	Abonelik kapatma seçeneği
Yayınları MQOPENed tanıtıcısında tut	MQCO_REMOVE_SUB
Bir MQOPENed tanıtıcısındaki yayınları kaldır	Eyleme izin verilmiyor
Yayınları MQSO_YÖNETİLEN tanıtıcısında tut	MQCO_REMOVE_SUB

İleriye okuma seçenekleri: Aşağıdaki seçenekler, bir uygulama tarafından istekte bulunmadan 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 MQCLOSE işlemi tamamlanmadan önce atılabilir ya da kuyruktan tüketilebilir.

MQCO_IMMEDIATE

Nesne hemen kapatılır ve bir uygulama istenmeden önce istemciye gönderilen iletiler atılır ve herhangi bir uygulama tarafından tüketilemez. Bu varsayılan değerdir.

MQCO_QUIESCE

Nesneyi kapatma isteği yapıldı, ancak bir uygulama tarafından istekte bulunmadan önce istemciye gönderilen herhangi bir ileti, istemcinin önünde okuma arabelleğinde bulunmaya devam ederse, MQCLOSE çağrısı MQRC_READ_AHEAD_MSGS uyarısıyla 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. İstemciye, istekte bulunan bir uygulamadan daha fazla ileti gönderilmez, önceden okuma işlemi artık kapatılır.

İstemcide daha fazla ileti olmadığı bir noktaya ulaşmaya çalışmak yerine, MQCO_QUIESCE ' yi kullanmanız önerilir; MQCO_IMMEDIATE kullanıldıysa, son MQGET çağrısıyla ve izleyen MQCLOSE arasında bir ileti varolabilir.

Zamanuyumsuz bir geri bildirme işlevinden MQCO_QUIESCE ile MQCLOSE komutu verilirse, ileriki iletileri okuma davranışı da geçerli olur. MQRC_READ_AHEAD_MSGS uyarısı döndürülürse, geri bildirme işlevi en az bir kez daha çağrılır. Önceden okunan son ileti geri bildirme işlevine geçirildiğinde, MQCBC ConsumerFlags alanı MQCBCF_READA_BUFFER_BOPTY olarak ayarlanır.

Varsayılan seçenek: Önceden açıklanan seçeneklerden hiçbirine gerek duymuyorsanız, aşağıdaki seçeneği kullanabilirsiniz:

MQCO_NONE

İsteğe bağlı kapatma işlemi gerekli değil.

Bu *gerekir* için şu öge belirtilmeli:

- Kuyruktan başka nesneler
- Önceden Tanımlı Kuyruk
- Geçici dinamik kuyruklar (ancak, yalnızca *Hobj* , kuyruğu yaratan MQOPER çağrısının döndürdüğü tanıtıcıyı *değil*).
- Dağıtım listeleri

Yukarıdaki durumların tümünde, nesne korunur ve silinmez.

Bu seçenek geçici bir dinamik kuyruk için belirtilirse:

- The queue is deleted, if it was created by the MQOPEN 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.

z/OS' da, kuyruk mantıksal olarak silinmiş bir dinamik kuyruksa ve bu kuyruk için son tanıtıcı ise, kuyruk fiziksel olarak silinmiştir. Ek ayrıntılar için [“Kullanım notları” sayfa 635](#) ' e 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

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_INCOMPLE_GROUP

(2241, X'8C1') İleti grubu tamamlanmadı.

MQRC_INCOMPLE_MSG

(2242, X'8C2') Mantıksal ileti tamamlanmadı.

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_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_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 değil.

MQRC_HOBJ_ERROR

(2019, X'7E3') Nesne tanıtıcısı geçerli değil.

MQRC_NOT_YETKILI

(2035, X'7F3') Erişim için yetkili değil.

MQRC_OBJECT_ZARAR

(2101, X'835 ') Nesne zarar gördü.

MQRC_OPTION_NOT_VALID_FOR_TYPE

(2045, X'7FD') Bir MQOL ya da MQCLOSE çağrısında: 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_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_Q_NOT_EMPTY

(2055, X'807 ') Kuyruk, bir ya da daha fazla ileti içeriyor ya da kesinleştirilmemiş bir put ya da alma isteği içeriyor.

MQRC_READ_AHEAD_MSGS

(nnnn, X'xxx ') İstemci, uygulama tarafından henüz tüketilmemiş iletileri okumuştur.

MQRC_RESOURCE_SORUNU

(2102, X'836 ') Yetersiz sistem kaynakları var.

MQRC_SECURITY_ERROR

(2063, X'80F') Güvenlik hatası oluştu.

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.

Bu kodlarla ilgili ayrıntılı bilgi için [İletiler ve neden kodları](#) başlıklı konuya bakın.

Kullanım notları

1. Bir uygulama MQDISC çağrısını yayınlarken ya da olağan dışı ya da olağan dışı sona erdirildiğinde, uygulama tarafından açılmış olan ve hala açık olan nesneler MQCO_NONE seçeneğiyle otomatik olarak kapatılır.
2. Kapatılmakta olan nesne bir *kuyruksaise* aşağıdaki noktalar geçerlidir:
 - Kuyruktaki işlemler bir iş biriminin bir parçası olarak gerçekleştirilirse, eşitleme noktasının, eşitleme noktasının sonucunu etkilemeden önce ya da sonra kapatılabilir olması gerekir. Kuyruk tetiklenirse, kuyruğu kapatmadan önce geriye işleme işlemi gerçekleştirmek, tetikleme iletilerinin yayınlanmasına neden olabilir. Tetikleme iletilerine ilişkin ek bilgi için [Tetikleme iletilerinin özellikleri](#) başlıklı konuya bakın.
 - Kuyruk MQOO_BROWSE seçeneğiyle açıldıysa, göz atma imleçle yok edilir. Kuyruk MQOO_BROWSE seçeneğiyle yeniden açılırsa, yeni bir imleç yaratılır (bkz. [MQOO_BROWSE](#)).
 - MQCLOSE çağrısı sırasında bu tanıtıcı için şu anda bir ileti kilitliyse, kilit serbest bırakılır (bkz. [MQGMO_LOCK](#)).
 - z/OS üzerinde, MQGMO_SET_SIGNAL seçeneğiyle MQGET isteği kapatılmakta olan kapatılmakta olan bir MQGET isteği varsa, istek iptal edilir (bkz. [MQGMO_SET_SIGNAL](#)). Aynı kuyruk için sinyal istekleri, ancak farklı kulvarlara (*Hobj*) takılmış durumda (dinamik bir kuyruk silinmedikçe, bu durumda da iptal edilir) etkilenmez.
3. Kapatılan nesne bir *dinamik kuyruksa* (kalıcı ya da geçici) ise, aşağıdaki noktalar geçerlidir:
 - Dinamik bir kuyruk için, ilgili MQOPER çağrısında belirlenen seçeneklere bakılmaksızın, MQCO_DELETE ve MQCO_DELETE_PURGE seçeneklerini belirtebilirsiniz.
 - Dinamik bir kuyruk silindiğinde, kuyruğa karşı bekleyen MQGMO_WEKE seçeneği ile tüm MQGET çağrıları iptal edilir ve MQRC_Q_DELETED neden kodu döndürülür. Bkz. [MQGMO_WAIT](#).

Uygulamalar silinmiş bir kuyruğa erişememesine rağmen, kuyruk sistemden kaldırılmaz ve ilişkili kaynaklar serbest bırakılmaz; kuyruğa gönderme yapan tüm işler kapatılıncaya kadar, kuyruğu etkileyen tüm iş birimleri kesinleştirilmiştir ya da yedeklenmiştir.

z/OS işletim sisteminde, mantıksal olarak silinmiş ancak henüz sistemden kaldırılmamış bir kuyruk, silinen kuyrukla aynı adı taşıyan yeni bir kuyruk yaratılmasını önler; bu durumda MQOPED çağrısı, MQRC_NAME_IN_USE neden koduyla başarısız olur. Ayrıca, uygulamalar tarafından erişilememesine rağmen, bu tür bir kuyruk MQSC komutları kullanılarak görüntülenmeye devam edebilir.

- Kalıcı bir dinamik kuyruk silindiğinde, MQCLOSE çağrısında belirtilen *Hobj* tanıtıcısı, kuyruğu yaratan MQOPER çağrısının döndürdüğü *değil*, bir denetim, MQOPER çağrısının geçerliliğini denetlemek için kullanılan kullanıcı kimliğinin kuyruğu silme yetkisi olduğunu doğrulayın. If the MQOO_ALTERNATE_USER_AUTHORITY option was specified on the MQOPEN call, the user identifier checked is the *AlternateUserId*.

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 *Hobj* 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 bir iş birimi varsa, kuyruk ve iletileri silinmeye devam eder, ancak iş birimleri başarısız 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:

- Bir dağıtım listesi için geçerli tek kapatma seçeneği, MQCO_NONE olur; çağrı, MQRC_OPTIONS_ERROR ya da MQRC_OPTION_NOT_VALID_FOR_TYPE neden kodlarıyla başarısız olur; başka seçenekler belirtilirse.
- Bir dağıtım listesi kapatıldığında, listedeki kuyruklar için tek tek tamamlanma kodları ve neden kodları döndürülmez; tanılama amaçlı olarak yalnızca çağrıya ilişkin *CompCode* ve *Reason* parametreleri kullanılabilir.

Kuyruklardan biri kapatılırsa, kuyruk yöneticisi işleme devam eder ve dağıtım listesindeki kalan kuyrukları kapatmayı dener. Çağrıya ilişkin *CompCode* ve *Reason* parametreleri, başarısızlığı açıklayan bilgileri döndürmek için ayarlanır. Kuyrukların çoğu başarıyla kapatılmış olsa da, tamamlanma kodunun MQCC_FAILED olması mümkündür. Hatayla karşılaştıran kuyruk saptanmaz.

Birden çok kuyruktaki hata varsa, *CompCode* ve *Reason* parametrelerinde hangi hatanın raporlanmadığı tanımlanmaz.

5. IBM üzerinde, uygulama ilk MQOP çağrısı yayınlandığında örtük olarak bağlandıysa, son MQCLOSE komutu verildiğinde örtük bir MQDISC oluşur.

Yalnızca uyumluluk kipinde çalışan uygulamalar örtük olarak bağlanabilir; diğer uygulamalar da, kuyruk yöneticisine belirttik olarak bağlanmak için MQCONN ya da MQCONNEX çağrısını yayınlamalıdır.

Önemli:  Uyumluluk kipini kullanmaya devam etseniz, uygulamanızı yeniden derlemeyi ve bunun yerine statik bağlı çağrıları kullanmayı düşünün. Bkz. [IBM üzerindeki RPG uygulamaları için uyumluluk kipi](#)

C çağırma

```
MQCLOSE (Hconn, &Hobj, Options, &CompCode, &Reason);
```

Değiştirelileri 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 çağırma

```
CALL 'MQCLOSE' USING HCONN, HOBJ, OPTIONS, COMPCODE, REASON.
```

Değiştirelileri 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 çağırımı

```
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 çağırısı

```
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 çağırısı

```
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 çağırısı, 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.
- **IBM i** IBM işletim sistemi, uyumluluk kipinde çalışan uygulamalar için bu çağırışı desteklemez.

Önemli:  Uyumluluk kipini kullanmaya devam etmeniz, uygulamanızı yeniden derlemeyi ve bunun yerine statik bağlı çağrılar kullanmayı düşünün. Bkz. [IBM üzerindeki RPG uygulamaları için uyumluluk kipi](#)

Sözdizimi

MQCMIT (*Hconn*, *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 MQCONNEX 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 612.

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 Kurtarılabılır 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 etkilenen, 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 kuyrukları doldurabilir. Buna karşı koruma sağlamak için, yöneticinin *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.
8. UNIX ve Windows sistemlerinde, *Reason* değiştirgesi MQRC_CONNECTION_BROKEN (*CompCode* MQCC_FAILED ile) ya da MQRC_UNEXPECTED_ERROR ise, iş biriminin başarıyla kesinleştirilmesinin olanaklı olduğunu sağlar.

C çağırma

```
MQCMIT (Hconn, &CompCode, &Reason);
```

Değiştirgeleri 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.
```

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 MQCMIT (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 MQCMIT,(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

```
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 çağırısı bir uygulama programını kuyruk yöneticisine bağlar.

Bu, uygulamanın sonraki ileti kuyruklama çağrılarında kullandığı bir kuyruk yöneticisi bağlantı tanıtıcısı sağlar.

- z/OSüzerinde, CICS uygulamalarının bu çağrıyı vermek zorunda kalmaması gerekir. Bu uygulamalar, CICS sisteminin bağlı olduğu kuyruk yöneticisine otomatik olarak bağlanır. Ancak, MQCONN ve MQDISC çağrıları hala CICS uygulamalarından kabul edilir.

- IBM işletim sistemi, uyumluluk kipinde çalışan uygulamaların bu çağrışı yayınlamaması gerekmez. Bu uygulamalar, ilk MQOPEN çağrışını yayınlarken kuyruk yöneticisine otomatik olarak bağlanır. Ancak, MQCONN ve MQDISC çağrışları hala IBM i uygulamalarından kabul edilir.

Önemli:  Uyumluluk kipini kullanmaya devam etseniz, uygulamanızı yeniden derlemeyi ve bunun yerine statik bağlı çağrışları kullanmayı düşünün. Bkz. [IBM üzerindeki RPG uygulamaları için uyumluluk kipi](#)

Diğer uygulamalar (yani, uyumluluk kipinde çalışmayan uygulamalar), kuyruk yöneticisine bağlanmak için MQCONN ya da MQCONNX çağrışını ve kuyruk yöneticisinden bağlantıyı kesmek için MQDISC çağrışını kullanmalıdır. Bu, önerilen programlama stilidir.

Yalnızca bir sunucu kuruluşunda istemci bağlantısı kurulamaz ve yalnızca istemci kuruluşunda yerel bir bağlantı kurulamaz.

Sözdizimi

MQCONN (*QMGrName*, *Hconn*, *CompCode*, *Reason*)

Parametreler

QMGrName

Tip: MQCHAR48 -giriş

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 (/), 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:

- EBCDIC Katakana kullanan sistemlerde küçük harfli karakterler kullanılamaz.
- z/OS işletim sistemi üzerinde, alt çizgi ile başlayan ya da biten adlar, işlemler ve denetim panoları tarafından işlenemez. Bu nedenle, bu isimleri önlemekten kaçının.
- IBM i' ta, komutlarda belirtildiğinde, küçük harf karakterleri, eğik çizgi ya da yüzde imi tırnak işaretleri içine alın. *QMGrName* parametresindeki bu tırnak işaretlerini belirtmeyin.

Ad tümüyle boşluklardan oluşuyorsa, *varsayılan* kuyruk yöneticisinin adı kullanılır. However, note the use of blank queue manager names described in the section on IBM MQ MQI client applications.

QMGrName için belirtilen ad, bir *bağlanabilir* kuyruk yöneticisinin adı olmalı ya da kuyruk yöneticisi grupları kullanılıyorsa, kuyruk yöneticisi grubunun adı ..

z/OS' ta, bağlanmanın mümkün olduğu kuyruk yöneticileri ortam tarafından belirlenir:

- CICS için, yalnızca CICS sisteminin bağlı olduğu kuyruk yöneticisini kullanabilirsiniz. *QMGrName* parametresi yine de belirtilmeli, ancak değeri yoksayılr; boş karakterler uygun bir seçenektir.
- For IMS, only queue managers that are listed in the subsystem definition table (CSQQDEFV), ve listed in the SSM table in IMS, are connectable (see usage note 6).
- z/OS toplu işi ve TSO için, yalnızca uygulamayla aynı sistemde bulunan kuyruk yöneticileri bağlanabilir (bkz. kullanım notu 6).

Kuyruk paylaşımı grupları: Birkaç kuyruk yöneticisinin var olduğu ve bir kuyruk paylaşım grubu oluşturmak üzere yapılandırıldığı sistemlerde, kuyruk paylaşım grubunun adı, kuyruk yöneticisi adının yerine *QMGrName* için belirtilebilir. Bu, uygulamanın, kuyruk paylaşımı grubunda bulunan ve uygulamayla aynı z/OS görüntüsünde bulunan *any* kuyruk yöneticisine bağlanmasını sağlar. Sistem,

boş bir *QMgrName* bağlantısı kullanılarak varsayılan kuyruk yöneticisi yerine kuyruk paylaşım grubuna bağlanacak şekilde de yapılandırılabilir.

QMgrName , kuyruk paylaşım grubunun adını belirtiyorsa, ancak sistemde o adı taşıyan bir kuyruk yöneticisi de varsa, ikinciye tercihte eski olan bir kuyruk yöneticisi de vardır. Yalnızca bu bağlantı başarısız olursa, kuyruk paylaşım grubundaki kuyruk yöneticilerinden biriyle bağlantı girişiminde bulunmaya çalışılır.

If the connection is successful, you can use the handle returned by the MQCONN or MQCONNEX call to access *Tümü* the resources (both shared and nonshared) that belong to the 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 MQCONNEX çağrısını yayınlarsa ve bir ya da her ikisi de kuyruk paylaşım grubunun adını belirtiyorsa, ikinci çağrı, ilk çağrıyla aynı kuyruk yöneticisine bağlandığında, MQCC_UYARI tamamlanma kodunu döndürür ve neden kodu MQRC_ALREADY_CONNECTED neden kodunu döndürür.

Kuyruk paylaşım grupları yalnızca z/OS üzerinde desteklenir. Bir kuyruk paylaşım grubuyla bağlantı yalnızca toplu, RRS toplu işi, CICS ve TSO ortamlarında desteklenir. CICS için, yalnızca CICS sisteminin bağlı olduğu kuyruk paylaşım grubunu kullanabilirsiniz. Yine de **QMgrName** parametresini belirtmeniz gerekir, ancak değeri yoksayılr; boş karakterler uygun bir seçenektir.



Uyarı: IMS , bir kuyruk paylaşım grubuna bağlanamıyor.

IBM MQ MQI client 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 bir boş kuyruk yöneticisi adına sahip bir kanal yöneticisi adı başarılı oluncaya kadar denir; bu durumda kuyruk yöneticisinin gerçek adına ilişkin herhangi bir denetim yoktur.

IBM MQ istemci uygulamaları z/OS' de desteklenmez, ancak z/OS , IBM MQ istemci uygulamalarının bağlanabileceği bir IBM MQ sunucusu olarak işlev görebilirler.

IBM MQ MQI client queue-manager grupları: Belirtilen ad bir yıldız işaretiyle (*) başlıyorsa, bağlantının yapıldığı kuyruk yöneticisinin uygulamanın belirlediği farklı bir adı olabilir. Belirtilen ad (yıldız işareti olmadan), bağlantı için uygun olan kuyruk yöneticilerine ilişkin bir *grup* tanımlar. Uygulama, bir bağlantının yapılabileceği bulunana kadar her birini sırayla deneyerek gruptan bir tane seçer. Bağlantıların denendiği sıra, aday kanalların istemci kanal ağırlığı ve bağlantı benzeşimi değerlerinden etkilenir. Gruptaki kuyruk yöneticilerinden hiçbirisi 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 bir kuyruk yöneticisi adını belirtiyorsa, çağrı başarısız olur. Bir grup, gruptaki kuyruk yöneticilerinin her biriyle 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 bir kuyruk yöneticisi adına sahip olan bir ya da daha çok istemci bağlantısı kanalı tanımlaması sağlayarak tanımlanır (bu nedenle, bir istemci uygulaması adı için tek bir yıldız işareti belirtilmesiyle aynı etkiye sahiptir).

Bir grubun tek bir kuyruk yöneticisine bağlandıktan sonra, bir uygulama, ileti ve nesne tanımlayıcılarındaki kuyruk yöneticisi adı alanlarının tipik şekilde, uygulamanın bağlı olduğu kuyruk yöneticisinin adını (*yerel kuyruk yöneticisi*) belirtmek için boşluk belirleyebilir. Uygulamanın bu adı bilmesi gerekiyorsa, *QMgrName* queue-manager özniteliğini sorgulamak için MQINQ çağrısını kullanın.

Bağlantı adına bir yıldız imi önizlenmek, uygulamanın gruptaki belirli bir kuyruk yöneticisine bağlanmaya bağlı olmadığını gösterir. Uygun uygulamalar şunlardır:

- İleti içeren, ancak ileti alamayan uygulamalar.

- İstek iletilerini yerleştiren ve daha sonra, yanıt iletilerini bir *geçici dinamik* kuyruktan alan uygulamalar.

Belirli bir kuyruk yöneticisinde belirli bir kuyruktan ileti almak için uygun olmayan uygulamalar vardır; bu tür uygulamaların adı yıldız işaretiyle önlememelidir.

Yıldız imi belirtirseniz, adın geri kalanının uzunluk üst sınırı 47 karakterdir.

Bu parametrenin uzunluğu MQ_Q_MGR_NAME_LENGTH tarafından verilir.

Hconn

Tip: MQHCONN-output

Bu tanıtıcı, kuyruk yöneticisiyle olan bağlantıyı gösterir. Uygulama tarafından yayınlanan sonraki tüm ileti kuyruklama çağrılarında bunu belirtin. MQDISC çağrısı yayınlandığında ya da tutamaç işleminin kapsamını tanımlayan işlem birimi sonlandığında geçerli olmak üzere sona erer.

IBM MQ şimdi, sunucu paketlerinin yanı sıra istemci paketleriyle birlikte mqm kitaplığını da sağlar. Bu, mqm kitaplığında bulunan bir MQI çağrısı yapıldığında, bir istemci ya da sunucu bağlantısı olup olmadığını görmek için bağlantı tipinin denetlendiğini ve daha sonra doğru temeldeki arama yapıldığını belirtir. Therefore an exit which is passed an *Hconn* can now be linked against the mqm library, but used on a client installation.

Tanıtıcı kapsamı: Döndürülen tanıtıcı kapsamı, kuyruk yöneticisine bağlanmak için kullanılan çağrıya bağlıdır (MQCONN ya da MQCONNX). Kullanılan çağrı MQCONNX ise, tutamaç kapsamı da MQCNO yapısının *Options* alanında belirtilen MQCNO_HANDLE_SHARE_* seçeneğine de bağlıdır.

- Çağrı MQCONN ise ya da MQCNO_HANDLE_SHARE_NONE seçeneği belirtilirse, döndürülen tanıtıcı bir *paylaşılmayan* tanıtıcı olur.

Paylaşılmayan bir tanıtıcı kapsamı, uygulamanın çalıştığı altyapının desteklediği en küçük koşut işleme birimidir (ayrıntılar için [Çizelge 103 sayfa 644](#) 'a bakın); tanıtıcı, çağrılarının yayınlandığı koşut işleme birimi dışında geçerli değildir.

- MQCNO_HANDLE_SHARE_BLOCK ya da MQCNO_HANDLE_SHARE_NO_BLOCK seçeneğini belirtirseniz, döndürülen tanıtıcı bir *paylaşılan* tanıtıcı olur.

Paylaşılan bir tanıtıcı kapsamı, çağrının verildiği iş parçasının iyesi olan işlemdir; tanıtıcı, o sürece ait olan herhangi bir iş parçasığından kullanılabilir. Tüm altyapılar iş parçacıklarını desteklemez.

- MQCONN ya da MQCONNX çağrısının tamamlanma kodu MQCC_FAILED ile başarısız olursa, Hconn değeri tanımsız olur.

Çizelge 103. Çeşitli platformlarda paylaşılmayan tutamaçların kapsamı	
Altyapı	Paylaşılmayan tutamaç kapsamı
z/OS	• CICS: CICS görevi • IMS: görev, bir sonraki eşitleme noktasına kadar (görevin alt görevleri hariç olmak üzere) • z/OS toplu iş ve TSO: görev (görevin alt görevleri hariç olmak üzere)
IBM i	İş
UNIX sistemleri	Başlık
16 bit Windows uygulamaları	Süreç
32 bit Windows uygulamaları	Başlık

On z/OS for CICS applications, and on IBM i for applications running in compatibility mode, the value returned is:

MQHC_DEF_HCONN

Varsayılan bağlantı tanıtıcısı.

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_ALREADY_CONNECTED

(2002, X'7D2') Uygulaması zaten bağlı.

MQRC_CLUSTER_EXIT_LOAD_ERROR

(2267, X'8DB') Küme iş yükü çıkışı yüklenemiyor.

MQRC_SSL_ALREADY_INITIALIZED

(2391, X' 957 ') SSL önceden kullanıma hazırlandı.

CompCode MQCC_FAILED ise:

MQRC_ADAPTER_CONN_LOAD_ERROR

(2129, X'851 ') Bağdaştırıcı bağlantısı modülü yüklenemiyor.

MQRC_ADAPTER_DEFS_ERROR

(2131, X'853 ') Bağdaştırıcı altsistem tanımlaması modülü geçerli değil.

MQRC_ADAPTER_DEFS_LOAD_ERROR

(2132, X'854 ') Bağdaştırıcı altsistemi tanımlama modülü yüklenemiyor.

MQRC_ADAPTER_NOT_AVAM

(2204, X'89C') Bağdaştırıcı kullanılmıyor.

MQRC_ADAPTER_SERV_LOAD_ERROR

(2130, X'852 ') Bağdaştırıcı hizmeti modülü yüklenemiyor.

MQRC_ADAPTER_STORAGE_EKSIKLİK

(2127, X'84F') Bağdaştırıcı için bellek yetersiz.

MQRC_ANOTHER_Q_MGR_CONNECTED

(2103, X'837 ') Başka bir kuyruk yöneticisi zaten bağlı.

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_BUFFER_LENGTH_ERROR

(2005, X'7D5') Arabellek uzunluğu parametresi geçerli değil.

MQRC_CALL_IN_PROGRESS

(2219, X'8AB') Önceki arama tamamlanmadan önce girilen MQI çağırısı.

MQRC_CONN_ID_IN_USE

(2160, X'870 ') Bağlantı tanıtıcısı zaten kullanılıyor.

MQRC_CONNECTION_BROKEN

(2009, X'7D9') Kuyruk yöneticisine yönelik bağlantı kaybedildi.

MQRC_CONNECTION_ERROR

(2273, X'8E1') MQCONN çağrısı işlenirken hata oluştu.

MQRC_CONNECTION_NOT_AVAM

(2568, X'A08') Bir MQCONN ya da MQCONNX çağrısında, kuyruk yöneticisi yürürlükteki kuruluştaki istenen bağlantı tipi için bağlantı sağlayamadığında ortaya çıkar. Yalnızca sunucu kuruluşunda istemci bağlantısı kurulamaz. Yalnızca istemci kuruluşunda yerel bir bağlantı kurulamaz.

MQRC_CONNECTION QUIESCING

(2202, X'89A') Bağlantı susturulmuş durumda.

MQRC_CONNECTION_DURATION

(2203, X'89B') Bağlantı sona erdiriliyor.

MQRC_CRYPTO_HARDWARE_ERROR

(2382, X'94E') Şifreleme donanımı yapılandırma hatası.

MQRC_DUPLICATE_RECOV_COORD

(2163, X'873 ') Kurtarma koordinatörü var.

MQRC_ENVIRONMENT_ERROR

(2012, X'7DC') Ortamda arama geçerli değil.

Ayrıca, MQCONNX çağrısında, [“MQCSP-Güvenlik değiştirgeleri” sayfa 328](#) denetim öbeğini bir CICS ya da IMS uygulamasından geçirmektedir.

MQRC_HCONN_ERROR

(2018, X'7E2') Bağlantı tanıtıcısı geçerli değil.

MQRC_HOST_NOT_AVAM

(2538, X'9EA') Bir MQCONN çağrısı bir istemciden bir kuyruk yöneticisine bağlanamadı, ancak uzak sisteme etkileşim ayırma girişimi başarısız oldu.

MQRC_INSTALLATION_MISMATCH

(2583, X'A17') Kuyruk yöneticisi kuruluşu ile seçilen kitaplık arasındaki uyumsuzluk.

MQRC_KEY_REPOSITORY_ERROR

(2381, X'94D') Anahtar havuzu geçerli değil.

MQRC_MAX_CONNS_IMlim_REACHED

(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_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_RESOURCE_SORUNU

(2102, X'836 ') Yetersiz sistem kaynakları var.

MQRC_SECURITY_ERROR

(2063, X'80F') Güvenlik hatası oluştu.

MQRC_SSL_INITIALIZATION_ERROR

(2393, X' 959 ') SSL başlatma hatası.

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. 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 almamak mümkündür.

3. Bir uygulama çalışırken kuyruk yöneticisi başarısız olursa, uygulama sonraki IBM MQ çağrılarında kullanılmak üzere yeni bir bağlantı tanıtıcısı edinmek için MQCONN çağrısını yeniden yayınlamalıdır. Uygulama, çağrı başarılı oluncaya kadar MQCONN çağrısını düzenli olarak yayınlatabilir.

Bir uygulama kuyruk yöneticisine bağlı olup olmadığından emin değilse, uygulama bir bağlantı tanıtıcısı almak için bir MQCONN çağrısını güvenli bir şekilde verebilir. Uygulama önceden bağlandıysa, döndürülen tanıtıcı, önceki MQCONN çağrısının döndürdüğü, ancak tamamlanma kodu MQCC_UYARY ve neden kodu MQRC_ALREADY_CONNECTED ile aynı olur.

4. Uygulama IBM MQ çağrılarını kullanmayı bitirdiğinde, kuyruk yöneticisinden bağlantıyı kesmek için uygulamanın MQDISC çağrısını kullanması gerekir.
5. MQCONN çağrısı, tamamlanma kodu MQCC_FAILED ile başarısız olursa, Hconn değeri tanımsız olur.
6. Açıkz/OS:

- Toplu iş, TSO ve IMS uygulamaları, diğer IBM MQ çağrılarını kullanmak için MQCONN çağrısını yayınlamalıdır. Bu uygulamalar, koşut zamanlı olarak birden çok kuyruk yöneticisine bağlanabilirler.

Kuyruk yöneticisi başarısız olursa, kuyruk yöneticisi yeni bir bağlantı tanıtıcısı elde etmek için yeniden başlatıldıktan sonra uygulama çağrısı yeniden yayınlamalıdır.

IMS uygulamaları MQCONN çağrısını sürekli olarak yayınlayabilse de, önceden bağlanmış olsa bile, çevrimiçi ileti işleme programları (MPP ' ler) için önerilmez.

- CICS uygulamalarının, diğer IBM MQ çağrılarını kullanmak için MQCONN çağrısını yayınlamaması gerekmez; ancak, bu çağrıların hem MQCONN çağrısının hem de MQDISC çağrısının kabul edilmesi durumunda da kabul edilmesi gerekir. Ancak, koşut zamanlı olarak birden çok kuyruk yöneticisine bağlanmak olanaklı değildir.

Kuyruk yöneticisi başarısız olursa, kuyruk yöneticisi yeniden başlatıldığında bu uygulamalar otomatik olarak yeniden bağlanır ve bu nedenle MQCONN çağrısını yayınlamamaya gerek yoktur.

7. On z/OS, to define the available queue managers:

- Toplu iş uygulamaları için, sistem programcıları, varsayılan kuyruk yöneticisi adını ya da kuyruk paylaşım grubu adını tanımlayan bir modül (CSQBDEFV) yaratmak için CSQBDEF makrosunu kullanabilir.
- For IMS applications, system programmers can use the CSQQDEFX macro to create a module (CSQQDEFV) that defines the names of the available queue managers and specifies the default queue manager.

Buna ek olarak, her kuyruk yöneticisinin IMS denetim bölgesi ve o kuyruk yöneticisine erişen her bir bağımlı bölge için tanımlanması gerekir. Bunu yapmak için, IMSiçinde bir altsistem üyesi oluşturmanız gerekir. PROCLIB kitaplığı ve ilgili IMS bölgelerinin altsistem üyesini tanımlayın. If an application attempts to connect to a queue manager that is not defined in the subsystem member for its IMS region, the application abends.

z/OS Bu makroları kullanma hakkında daha fazla bilgi için bkz. [Müşteri kullanımı için tasarlanan makrolar](#).

8. IBM üzerinde, kuyruk yöneticisinin önceki yayınları için yazılan uygulamalar yeniden derlemeye gerek kalmadan çalıştırılabilir. Buna denir. *uyumluluk kipi*. Bu işlem kipi uygulamalar için uyumlu bir yürütme ortamı ortamı sağlar. Bu, aşağıdaki özellikleri içerir:

- QMQM kitaplığında bulunan AMQZSTUB hizmet programı.

AMQZSTUB, önceki yayın düzeyleriyle aynı genel arabirimi sağlar ve aynı imzayı alır. Bağlı yordam çağrılarına ilişkin MQI ' ye erişmek için bu hizmet programını kullanın.

- QMQM, QMQM kitaplığında bulunan QMQM programı.

QMQM, devingen program çağrılar yoluyla MQI ' ye erişilmesini sağlar.

- MQCLOSE, MQCONN, MQDISC, MQGET, MQINQ, MQOPEN, MQPUT, MQPUT1, ve QMQM kitaplığında bulunan MQSET programları programlar.

Bu programlar, dinamik program çağrılarını, ancak IBM MQ çağrılarının standart açıklamalarına karşılık gelen bir parametre listesiyle birlikte MQI ' ye erişim olanağı da sağlar.

Bu üç arabirim, IBM MQ Sürüm 5.1içinde kullanıma sunulan yetenekleri içermez. Örneğin, MQBACK, MQCMIT ve MQCONNX çağrıları desteklenmez. Bu arabirimler tarafından sağlanan destek, yalnızca tek iş parçacıklı uygulamalar içindir.

Tek iş parçacıklı uygulamalarda yeni IBM MQ çağrıları ve çok iş parçacıklı uygulamalarda tüm IBM MQ çağrıları için destek, LIBMQM ve LIBMQM_R hizmet programları aracılığıyla sağlanır.

Önemli: **IBM i** Uyumluluk kipini kullanmaya devam etseniz, uygulamanızı yeniden derlemeyi ve bunun yerine statik bağlı çağrıları kullanmayı düşünün. Bkz. [IBM üzerindeki RPG uygulamaları için uyumluluk kipi](#)

9. IBM i' ta, olağan dışı sona erdirilen programlarda kuyruk yöneticisinden otomatik olarak bağlantı kesilmez. MQCONN ya da MQCONNX çağrısının tamamlanma kodu MQCC_UYARY ve neden kodu MQRC_ALREADY_CONNECTED olasılığına izin vermek için uygulamaları yazın. Bu durumda döndürülen bağlantı tanıtıcısını olağan biçimde kullanın.

C çağırma

```
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 çağırma

```
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 çağırımı

```
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 çağırısı

```
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 çağırısı

```
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'

```

MQCONN-Connect kuyruk yöneticisi (genişletilmiş)

MQCONN çağırısı bir uygulama programını kuyruk yöneticisine bağlar. Bu, sonraki IBM MQ çağrılarında uygulama tarafından kullanılan bir kuyruk yöneticisi bağlantı tanıtıcısı sağlar.

MQCONN çağırısı, MQCONN çağırısının, çağrılarının çalışma şeklini denetlemek için belirtilmesine izin vermesi dışında, MQCONN çağırısına benzer.

- Bu çağrı, tüm IBM MQ sistemlerinde ve bu sistemlere bağlı IBM MQ istemcilerinde desteklenir.
- IBM işletim sistemi, uyumluluk kipinde çalışan uygulamalar için bu çağrıyı desteklemez.

Yalnızca bir sunucu kuruluşunda istemci bağlantısı kurulamaz ve yalnızca istemci kuruluşunda yerel bir bağlantı kurulamaz.

Sözdizimi

MQCONN (QMGrName, ConnectOpts, Hconn, CompCode, Reason)

Parametreler

QMGrName

Tip: MQCHAR48 -giriş

Ayrıntılar için “MQCONN-Bağlantı kuyruğu yöneticisi” sayfa 641 içinde açıklanan QMGrName parametresine bakın.

ConnectOpts

Tip: MQCNO-giriş/çıkış

Ayrıntılar için bkz. “MQCNO-Bağlantı seçenekleri” sayfa 314.

Hconn

Tip: MQHCONN-output

Bu tanıtıcı, kuyruk yöneticisiyle olan bağlantıyı gösterir. Uygulama tarafından yayınlanan sonraki tüm ileti kuyruklama çağrılarında bunu belirtin. MQDISC çağrısı yayınlandığında ya da tutamaç işleminin kapsamını tanımlayan işlem birimi sonlandığında geçerli olmak üzere sona erer.

IBM MQ şimdi, sunucu paketlerinin yanı sıra istemci paketleriyle birlikte mqm kitaplığını da sağlar. Bu, mqm kitaplığında bulunan bir MQI çağrısı yapıldığında, bir istemci ya da sunucu bağlantısı olup olmadığını görmek için bağlantı tipinin denetlendiğini ve daha sonra doğru temeldeki arama yapıldığını belirtir. Therefore an exit which is passed an Hconn can now be linked against the mqm library, but used on a client installation.

Tanıtıcı kapsamı: Döndürülen tanıtıcı kapsamı, kuyruk yöneticisine bağlanmak için kullanılan çağrıya bağlıdır (MQCONN ya da MQCONNX). Kullanılan çağrı MQCONNX ise, tutamaç kapsamı da MQCNO yapısının Options alanında belirtilen MQCNO_HANDLE_SHARE_* seçeneğine de bağlıdır.

- Çağrı MQCONN ise ya da MQCNO_HANDLE_SHARE_NONE seçeneği belirtilirse, döndürülen tanıtıcı bir paylaşılmayan tanıtıcı olur.

Paylaşılmayan bir tanıtıcı kapsamı, uygulamanın çalıştığı altyapının desteklediği en küçük koşt işleme birimidir (ayrıntılar için Çizelge 104 sayfa 650 'a bakın); tanıtıcı, çağrılarının yayınlandığı koşt işleme birimi dışında geçerli değildir.

- MQCNO_HANDLE_SHARE_BLOCK ya da MQCNO_HANDLE_SHARE_NO_BLOCK seçeneğini belirtirseniz, döndürülen tanıtıcı bir paylaşılan tanıtıcı olur.

Paylaşılan bir tanıtıcı kapsamı, çağrının verildiği iş parçacığının iyesi olan işlemdir; tanıtıcı, o sürece ait olan herhangi bir iş parçacığından kullanılabilir. Tüm altyapılar iş parçacıklarını desteklemez.

- MQCONN ya da MQCONNX çağrısının tamamlanma kodu MQCC_FAILED ile başarısız olursa, Hconn değeri tanımsız olur.

Çizelge 104. Çeşitli platformlarda paylaşılmayan tutamaçların kapsamı	
Altyapı	Paylaşılmayan tutamaç kapsamı
z/OS	• CICS: CICS görevi • IMS: görev, bir sonraki eşitleme noktasına kadar (görevin alt görevleri hariç olmak üzere) • z/OS toplu iş ve TSO: görev (görevin alt görevleri hariç olmak üzere)
IBM i	İş
UNIX sistemleri	Başlık
16 bit Windows uygulamaları	Süreç

Çizelge 104. Çeşitli platformlarda paylaşılmayan tutamaçların kapsamı (devamı var)	
Altyapı	Paylaşılmayan tutamaç kapsamı
32 bit Windows uygulamaları	Başlık

On z/OS for CICS applications, and on IBM i for applications running in compatibility mode, the value returned is:

MQHC_DEF_HCONN

Varsayılan bağlantı tanıtıcısı.

CompCode

Tip: MQXX_ENCODE_CASE_ONE long-output

Ayrıntılar için “MQCONN-Bağlantı kuyruğu yöneticisi” sayfa 641 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_UYARI ise:

MQRC_ALREADY_CONNECTED

(2002, X'7D2') Uygulaması zaten bağlı.

MQRC_CLUSTER_EXIT_LOAD_ERROR

(2267, X'8DB') Küme iş yükü çıkışı yüklenemiyor.

MQRC_SSL_ALREADY_INITIALIZED

(2391, X' 957 ') SSL önceden kullanıma hazırlandı.

CompCode MQCC_FAILED ise:

MQRC_ADAPTER_CONN_LOAD_ERROR

(2129, X'851 ') Bağdaştırıcı bağlantısı modülü yüklenemiyor.

MQRC_ADAPTER_DEFS_ERROR

(2131, X'853 ') Bağdaştırıcı altsistem tanımlaması modülü geçerli değil.

MQRC_ADAPTER_DEFS_LOAD_ERROR

(2132, X'854 ') Bağdaştırıcı altsistemi tanımlama modülü yüklenemiyor.

MQRC_ADAPTER_NOT_AVAM

(2204, X'89C') Bağdaştırıcı kullanılamıyor.

MQRC_ADAPTER_SERV_LOAD_ERROR

(2130, X'852 ') Bağdaştırıcı hizmeti modülü yüklenemiyor.

MQRC_ADAPTER_STORAGE_EKSIKLİK

(2127, X'84F') Bağdaştırıcı için bellek yetersiz.

MQRC_ANOTHER_Q_MGR_CONNECTED

(2103, X'837 ') Başka bir kuyruk yöneticisi zaten bağlı.

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_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_CONN_ID_IN_USE

(2160, X'870 ') Bağlantı tanıtıcısı zaten kullanılıyor.

MQRC_CONNECTION_BROKEN

(2009, X'7D9') Kuyruk yöneticisine yönelik bağlantı kaybedildi.

MQRC_CONNECTION_ERROR

(2273, X'8E1') MQCONN çağrısı işlenirken hata oluştu.

MQRC_CONNECTION_NOT_AVAM

(2568, X'A08') Bir MQCONN ya da MQCONNX çağrısında, kuyruk yöneticisi yürürlükteki kuruluştaki istenen bağlantı tipi için bağlantı sağlayamadığında ortaya çıkar. Yalnızca sunucu kuruluşunda istemci bağlantısı kurulamaz. Yalnızca istemci kuruluşunda yerel bir bağlantı kurulamaz.

MQRC_CONNECTION QUIESCING

(2202, X'89A') Bağlantı susturulmuş durumda.

MQRC_CONNECTION_DURATION

(2203, X'89B') Bağlantı sona erdiriliyor.

MQRC_CRYPTO_HARDWARE_ERROR

(2382, X'94E') Şifreleme donanımı yapılandırma hatası.

MQRC_DUPLICATE_RECOV_COORD

(2163, X'873 ') Kurtarma koordinatörü var.

MQRC_ENVIRONMENT_ERROR

(2012, X'7DC') Ortamda arama geçerli değil.

Ayrıca, MQCONNX çağrısında, [“MQCSP-Güvenlik değiştirgeleri” sayfa 328](#) denetim öbeğini bir CICS ya da IMS uygulamasından geçirmektedir.

MQRC_HCONN_ERROR

(2018, X'7E2') Bağlantı tanıtıcısı geçerli değil.

MQRC_HOST_NOT_AVAM

(2538, X'9EA') Bir MQCONN çağrısı bir istemciden bir kuyruk yöneticisine bağlanamadı, ancak uzak sisteme etkileşim ayırma girişimi başarısız oldu.

MQRC_INSTALLATION_MISMATCH

(2583, X'A17') Kuyruk yöneticisi kuruluşu ile seçilen kitaplık arasındaki uyumsuzluk.

MQRC_KEY_REPOSITORY_ERROR

(2381, X'94D') Anahtar havuzu geçerli değil.

MQRC_MAX_CONNS_IMlim_REACHED

(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_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_RESOURCE_SORUNU

(2102, X'836 ') Yetersiz sistem kaynakları var.

MQRC_SECURITY_ERROR

(2063, X'80F') Güvenlik hatası oluştu.

MQRC_SSL_INITIALIZATION_ERROR

(2393, X' 959 ') SSL başlatma hatası.

MQRC_STORAGE_NOT_AVAM

(2071, X'817 ') Kullanılabilir bellek yetersiz.

MQRC_UNEXPECTED_ERROR

(2195, X'893 ') Beklenmeyen bir hata oluştu.

İzleyen ek neden kodları, MQCONNX çağrısı tarafından döndürülebiliyor:

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);
```

Değiştirelileri 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.
```

Değiştirelileri aşağıdaki gibi bildirin:

```
** Name of queue manager
01 QMGRNAME      PIC X(48).
** Options that control the action of MQCONN
01 CONNECTOPTS.
   COPY CMQCNV.
** 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);
```

Değiştirelileri 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	CMQCN0A	,	Options that control the action of MQCONN
HCONN	DS	F	Connection handle
COMP CODE	DS	F	Completion code
REASON	DS	F	Reason code qualifying COMP CODE

Visual Basic çağrısı

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 çağrısını 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, konulmakta olan iletiyle ilişkilendirmek için çağrılar. Benzer şekilde, MQGET çağrısında bir ileti tanıtıcısı belirleyerek, MQGET çağrısı tamamlandığında ileti tanıtıcısı kullanılarak, alınmakta olan iletinin özelliklerine erişilir.

İleti tanıtıcısını silmek için [MQDLTMH](#) ögesini kullanın.

Sözdizimi

MQCRTMH (*Hconn*, *CrtMsgHOpts*, *Hmsg*, *CompCode*, *Reason*)

Parametreler

Hconn

Tip: MQHCONN-input

Bu tanıtıcı, kuyruk yöneticiyle 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:

MQHC_UNASSOCIATED_HCONN

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 parçacığında kurulmuş bir kuyruk yöneticisine 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 kullanım için sınırlanmıştır. İleti tanıtıcısı farklı bir kuruluma sağlanırsa, MQRC_HMSG_NOT_AVAVAL neden kodu döndürülür.

On z/OS for CICS applications, and on IBM i for applications running in compatibility mode, the MQCONN call can be omitted, and you can specify the following value for *Hconn* :

MQHC_DEF_CONN

Varsayılan bağlantı tanıtıcısı

Önemli:  Uyumluluk kipini kullanmaya devam etseniz, uygulamanızı yeniden derlemeyi ve bunun yerine statik bağlı çağrılarını kullanmayı düşünün. Bkz. [IBM üzerindeki RPG uygulamaları için uyumluluk kipi](#)

CrtMsgHOpts

Tip: MQCMHO-input

MQCRTMH işleminin denetlenmesini denetleyen seçenekler. Ayrıntılar için bkz. [MQCMHO](#) .

Hmsg

Tip: MQHMSG-output

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 değer varsayılan değerleri içerir. İlişkili ileti tanımlayıcısı alanlarının değerleri, MQSETMP ve MQINQMP çağrıları kullanılarak ayarlanabilir ve sorgulanabilir. MQDLTMP çağrısı, ileti tanımlayıcısının bir alanını varsayılan değerine geri döndürür.

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 aynı anda yalnızca bir IBM MQ çağrısı tarafından kullanılabilir. Tanıtıcı, ikinci bir IBM MQ çağrısı aynı ileti tanıtıcısını kullanma girişiminde bulunursa, ikinci IBM MQ çağrısı MQRC_MSG_HANDLE_IN_USE neden kodlarıyla başarısız olur.

Hconn değiştirgesi MQHC_UNASSOCIATED_HCONN değilse, döndürülen ileti tanıtıcısı yalnızca belirtilen bağlantıda kullanılabilir.

Aynı *Hconn* parametre değerinin, bu ileti tanıtıcısı kullanıldığı sonraki MQI çağrılarında kullanılması gerekir:

- MQDLTMH
- MQSETMP
- MQINQMP
- MQDLTMP
- MQMHBUFF
- 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.

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 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_CALL_IN_PROGRESS

(2219, X'08AB') Önceki arama tamamlanmadan önce girilen MQI çağırısı.

MQRC_CMHO_ERROR

(2461, X'099D') İleti tanıtıcısı yaratma seçenekleri yapısı geçerli değil.

MQRC_CONNECTION_BROKEN

(2273, X'7D9') Kuyruk yöneticisine yönelik bağlantı kaybedildi.

MQRC_HANDLE_NOT_VAR

(2017, X'07E1') Kullanılabilir başka tanıtıcı yok.

MQRC_HCONN_ERROR

(2018, X'7E2') Bağlantı tanıtıcısı geçerli değil.

MQRC_HMSG_ERROR

(2460, X'099C') İleti tanıtıcısı göstergesi geçerli değil.

MQRC_OPTIONS_ERROR

(2046, X'07FE') Seçenekler geçerli değil ya da tutarlı 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

```
MQCRTMH (Hconn, &CrtMsgHOpts, &Hmsg, &CompCode, &Reason);
```

Değiştirelileri 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, CRTMSGOPTS, HMSG, COMPCODE, REASON.
```

Değiştirelileri 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 çağrıları

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, and on IBM i for applications running in compatibility mode, the MQCONN call can be omitted, and you can specify the following special value for *Hconn* :

MQHC_DEF_HCONN

Varsayılan bağlantı tanıtıcısı.

Önemli:  Uyumluluk kipini kullanmaya devam etmeniz, uygulamanızı yeniden derlemeyi ve bunun yerine statik bağlı çağrılar kullanmayı düşünün. Bkz. [IBM üzerindeki RPG uygulamaları için uyumluluk kipi](#)

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 çağrıları 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 işleminde yayınlanabilir. Geri çağırma yordamıyla aynı bağlantı için, tek amacı daha önce verilen bir MQOP_STOP işlemini iptal etmek.

Bu seçenek aşağıdaki ortamlarda desteklenmez: z/OS üzerinde CICS ya da uygulama iş parçacıklı bir IBM MQ kitaplığından bağliysa.

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 alınırsa, 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 çağrılmaz.

Bir geri çağrı yordamı dışında yayınlanırsa, denetim, önceden okunan iletiler için tüketici rutinleri tamamlanıncaya 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 tamamlanıncaya kadar arayanın çağrısına dönmez ve daha fazla çağrılmaz.

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 işlemini 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 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_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 CICS tarafından 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') İlintilendirme tanıtıcısı hatası.

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') Tutarlı göz atma belirtimi.

MQRC_INCONSISTENT_UOW

(2244, X'8C5') Tutarlı 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 işlem 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.

MQRC_MODULE_NOT_FOUND

(2495, X'9BF') Modül, arama yolunda bulunamadı ya da yükleme yetkisine sahip değil.

MQRC_MSG_ID_ERROR

(2206, X'89E') İleti-tanıtıcısı hatası.

MQRC_MSG_SEQ_NUMBER_ERROR

(2250, X'8CA') İleti sıra numarası geçerli değil.

MQRC_MSG_TOKEN_HATASI

(2331, X'91B') İleti simgesinin kullanımı geçerli değil.

MQRC_NOT_OPEN_FOR_BROWSE

(2036, X'7F4') Kuyruk göz atma için açık değil.

MQRC_NOT_OPEN_FOR_INPUT

(2037, X'7F5') Kuyruk giriş için açık değil.

MQRC_OBJECT_CHANGED

(2041, X'7F9') Nesne tanımlaması açıldığından beri değiştirildi.

MQRC_OBJECT_ZARAR

(2101, X'835 ') Nesne zarar gördü.

MQRC_OPERATION_HATASI

(2488, X'9B8') API Çağrısında Hatalı İşlem kodu

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_Q_DELETED

(2052, X'804 ') Kuyruk silinmiştir.

MQRC_Q_INDEX_TYPE_ERROR

(2394, X'95A') Kuyruk yanlış dizin tipi 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_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_NOT_AVAM

(2072, X'818 ') Syncpoint 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 kullanılabilecek 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 parçacığının 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 (USS) olanağını kullanma yetkisi olması gerekir.
- 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, USS 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)
```

Değiştirelileri aşağıdaki gibi bildirin:

```
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 CompCode */
```

Cobol çağırma

```
CALL 'MQCTL' USING HCONN, OPERATION, CTLOPTS, COMPCODE, REASON.
```

Değiştirelileri 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)
```

Değiştirelileri 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 çağrısı, kuyruk yöneticisi ile uygulama programı arasındaki bağlantıyı keser ve MQCONN ya da MQCONNX çağrısının 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 çağrısı yayınlaması gerekir. Daha fazla ayrıntı için bkz. [Zamanuyumsuz tüketim IBM MQ iletileri](#) .
- z/OSüzerinde, CICS uygulamalarının kuyruk yöneticisinden bağlantıyı kesmek için bu çağrıyı yayınlamaya gerek yoktur.

Bir CICS uygulaması bu çağrıyı gerçekleştirirse, aşağıdaki belirtilerden birini belirterek, daha önceki bir MQCONNX çağrısı 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.

- IBM işletim sistemi, uyumluluk kipinde çalışan uygulamaların bu çağrıyı yayınlamaya gerek yoktur. Ek bilgi için [“MQCONN-Bağlantı kuyruğu yöneticisi” sayfa 641](#) başlıklı konuya bakın.

Önemli:  Uyumluluk kipini kullanmaya devam etseniz, uygulamanızı yeniden derlemeyi ve bunun yerine statik bağlı çağrıları kullanmayı düşünün. Bkz. [IBM üzerindeki RPG uygulamaları için uyumluluk kipi](#)

Sözdizimi

MQDISC (*Hconn*, *CompCode*, *Reason*)

Parametreler

Hconn

Tip: MQHCONN-giriş/çıkış

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, and on IBM i for applications running in compatibility mode, you can omit the MQCONN call, and specify the following value for *Hconn*:

MQHC_DEF_HCONN

Varsayılan bağlantı tanıtıcısı.

Çağrı başarıyla tamamlandığında, kuyruk yöneticisi *Hconn* değerini, ortam için geçerli bir tanıtıcı olmayan bir değere ayarlar. Bu değer:

MQHC_UNUSABLE_HCONN

Kullanılamayan bağlantı tanıtıcısı.

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 beklemeye.

CompCode MQCC_FAILED ise:

MQRC_ADAPTER_DISC_LOAD_ERROR

(2138, X'85A') Bağdaştırıcı bağlantı kesme modülü yüklenmiyor.

MQRC_ADAPTER_NOT_AVAM

(2204, X'89C') Bağdaştırıcı kullanılamıyor.

MQRC_ADAPTER_SERV_LOAD_ERROR

(2130, X'852 ') Bağdaştırıcı hizmeti modülü yüklenmiyor.

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

1. Bağlantı o bağlantı altında açık olan nesneler varsa, MQDISC çağrısı yayınlanırsa, kuyruk yöneticisi bu nesneleri kapatır ve kapama seçenekleri MQCO_NONE olarak ayarlanmış olur.
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 bir 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.
 - b. 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 eşgüdümcüsü 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 MQCONN 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 eşgüdümcüsü iş birimini çözünceye kadar bağlantı etiketi yeniden kullanılmak üzere kullanılamaz.
4. IBM işletim sistemi, uyumluluk kipinde çalışan uygulamaların bu çağrıyı yayınlamaması gerekmez; daha fazla ayrıntı için MQCONN çağrısına bakın.

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 tanıtıcısını sil

MQDLTMH çağırısı bir ileti tanıtıcısını siler ve MQCRTMH çağırısının tersi olur.

Sözdizimi

MQDLTMH (*Hconn*, *Hmsg*, *DltMsgHOpts*, *CompCode*, *Reason*)

Parametreler

Hconn

Tip: MQHCONN-input

Bu tanıttıcı, kuyruk yöneticiyle olan bağlantıyı gösterir.

Değerin, *Hmsg* parametresinde belirtilen ileti tanıttıcısını yaratmak için kullanılan bağlantı tanıttıcısı ile eşleşmesi gerekir.

İleti tanıttıcısı MQHC_UNASSOCIATED_HCONN kullanılarak yaratıldıysa, ileti tanıttıcısını silmeye ilişkin iş parçasısında geçerli bir bağlantı oluşturulmalıdır; tersi durumda, çağrı MQRC_CONNECTION_BROKEN ile başarısız olur.

Hmsg

Tip: MQHMSG-giriş/çıkış

Bu, silinmek üzere ileti tanıttıcısıdır. Değer önceki bir MQCRTMH çağrısı tarafından döndürüldü.

Çağrıyı başarıyla tamamlamaya ilişkin tanıttıcı, ortam için geçersiz bir değere ayarlıdır. Bu değer:

MQHM_UNUSABLE_HMSG

Kullanılamayan ileti tanıttıcısı.

The message handle cannot be deleted if another IBM MQ call is in progress that was passed the same message handle.

DltMsgHOpts

Tip: MQDMHO-input

Ayrıntılar için bkz. [MQDMHO](#) .

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 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_CALL_IN_PROGRESS

(2219, X'08AB') Önceki arama tamamlanmadan önce girilen MQI çağrısı.

MQRC_CONNECTION_BROKEN

(2009, X'07D9') Kuyruk yöneticisine yönelik bağlantı kaybedildi.

MQRC_DMHO_ERROR

(2462, X'099E') İleti tanıtıcısı seçeneklerini silme yapısı geçerli değil.

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_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

```
MQDLTMH (Hconn, &Hmsg, &DltMsgHOpts, &CompCode, &Reason);
```

Değiştirelileri 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 çağırma

```
CALL 'MQDLTMH' USING HCONN, HMSG, DLTMSGOPTS, COMPCODE, REASON.
```

Değiştirelileri aşağıdaki gibi bildirin:

```
** Connection handle
01  HCONN    PIC S9(9) BINARY.

** Options that control the action of MQDLTMH
01  DLTMSGOPTS.
COPY CMQDLMHOV.

** Completion code
01  COMPCODE  PIC S9(9) BINARY.

** Reason code qualifying COMPCODE
01  REASON    PIC S9(9) BINARY.
```

PL/I çağırımı

```
call MQDLTMH (Hconn, Hmsg, DltMsgHOpts, CompCode, Reason);
```

Değiştirelileri 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 çağrısı

```
CALL MQDLTMH, (HCONN,HMSG,DLTMSGHOPTS,COMP CODE,REASON)
```

Değiştirgeleri aşağıdaki gibi bildirin:

HCONN	DS	F	Connection handle
HMSG	DS	D	Message handle
DLTMSGHOPTS	CMQDMHOA	,	Options that control the action of MQDLTMH
COMP CODE	DS	F	Completion code
REASON	DS	F	Reason code qualifying COMP CODE

MQDLTMP-İleti özelliğini sil

MQDLTMP çağrısı, bir özelliği ileti tanıtıcısından siler ve MQSETMP çağrısının tersi olur.

Sözdizimi

MQDLTMP (*Hconn, Hmsg, DltPropOpts, Ad, 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 tanıtıcısını silmeye ilişkin iş parçasısında geçerli bir bağlantı kurulmalıdır; tersi durumda, MQRC_CONNECTION_BROKEN çağrısı başarısız olur.

Hmsg

Tip: MQHMSG-giriş

Bu, silinecek özelliği içeren ileti tanıtıcısıdır. Değer önceki bir MQCRTMH çağrısı tarafından döndürüldü.

DltPropOpts

Tip: MQDMPO-input

Ayrıntılı bilgi için [MQDMPO](#) veri tipine bakın.

NAME

Tip: MQCHARV-input

Silinecek özelliğin adı. Özellik adlarıyla ilgili ek bilgi için [Özellik adları](#) konusuna bakın.

Özellik adında genel arama karakterlerine izin verilmez.

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_PROPERTY_NOT_VAR

(2471, X'09A7') Özellik yok.

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'0852 ') Bağdaştırıcı hizmeti modülü yüklenemiyor.

MQRC_ASID_MISSMATCH

(2157, X'086D') Birincil ve ana ASID değerleri farklı.

MQRC_CALL_IN_PROGRESS

(2219, X'08AB') Önceki arama tamamlanmadan önce girilen MQI çağrısı.

MQRC_CONNECTION_BROKEN

(2009, X'07D9') Kuyruk yöneticisine yönelik bağlantı kaybedildi.

MQRC_DMPO_ERROR

(2481, X'09B1') İleti silme 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'07FE') Seçenekler geçerli değil ya da tutarlı değil.

MQRC_ÖZELLİĞİ _

(2442, X'098A') Özellik adı geçersiz.

MQRC_SOURCE_CCSID_ERROR

(2111, X'083F') Özellik adı kodlanmış karakter takımı tanıtıcısı geçerli değil.

MQRC_UNEXPECTED_ERROR

(2195, X'0893 ') Beklenmeyen bir hata oluştu.

Bu kodlarla ilgili ayrıntılı bilgi için aşağıdaki başlara bakın:

- IBM MQ for z/OS için [İletiler ve neden kodları](#)
- Diğer IBM MQ platformları için [API tamamlama ve neden kodları](#)

.

C çağırma

```
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 çağırma

```
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     PIC S9(9) BINARY.
   COPY CMQCHRVV.
** Completion code
01 COMPCODE PIC S9(9) BINARY.
** Reason code qualifying COMPCODE
01 REASON   PIC S9(9) BINARY.

```

PL/I çağırımı

```
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 */
dcl Name       like MQCHARV;  /* Property name */
dcl CompCode   fixed bin(31); /* Completion code */
dcl Reason     fixed bin(31); /* Reason code qualifying CompCode */

```

High Level Assembler çağırısı

```
CALL MQDLTMP, (HCONN,HMSG,DLTPROPOPTS,NAME,COMPCODE,REASON)
```

Değiştirgeleri aşağıdaki gibi bildirin:

HCONN	DS	F	Connection handle
HMSG	DS	D	Message handle
DLTPROPOPTS	CMQDMPOA	,	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 çağırısı, yerel bir kuyruktan gelen bir iletiyi, MQOPEN çağırısıyla birlikte alır.

Sözdizimi

MQGET (*Hconn, Hobj, MsgDesc, GetMsgOpts, BufferLength, Buffer, DataLength, 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, and on IBM i for applications running in compatibility mode, the MQCONN call can be omitted, and the following value specified for *Hconn* :

MQHC_DEF_HCONN

Varsayılan bağlantı tanıttıcısı.

Önemli:  Uyumluluk kipini kullanmaya devam etseniz, uygulamanızı yeniden derlemeyi ve bunun yerine statik bağlı çağrılar kullanmayı düşünün. Bkz. [IBM üzerindeki RPG uygulamaları için uyumluluk kipi](#)

Hobj

Tip: MQHOBJ-input

Bu tanıttıcı, 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 712](#) 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 tanımlayıcısı” sayfa 404](#).

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 eklenmiş bir MQMDE öneki vardır; ancak, MQMDE içindeki bir ya da daha çok alanın varsayılan değeri olmayan bir değeri varsa, *yalnızca* . MQMDE ' deki tüm alanlarda varsayılan değerler varsa, MQMDE atlanır. MQMD 'deki *Biçim* alanında MQFMT_MD_EXTENSION 'in biçim adı, bir MQMDE' nin var olduğunu gösterir.

MsgHandle alanında geçerli bir ileti tanıttıcısı sağlanırsa, uygulamanın bir MQMD yapısı sağlaması gerekmez. Bu alanda hiçbir şey sağlanmıyorsa, iletinin tanımlayıcı, ileti tanıttıcılarıyla ilişkili tanımlayıcıdan alınır.

Uygulama bir MQMD yapısı yerine bir ileti tanıttıcısı 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 özniteliği MQPROP_FORCE_MQRFH2 ise, çağrı da başarısız olur. Neden kodu: MQRC_MD_ERROR.

Eşleşme seçenekleri belirlendiyse ve ileti tanıttıcısı ile ilişkilendirilen ileti tanımlayıcısı kullanılıyorsa, ileti tanıttıcısından eşleşen giriş alanları için kullanılır.

GetMsgSeçenekleri

Tip: MQGMO-giriş/çıkış

Ayrıntılar için bkz. [“MQGMO-Get-message seçenekleri” sayfa 358](#).

BufferLength

Tip: MQHOME-girişi

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 özniteliği tarafından verilir; bkz. “Kuyruklara ilişkin öznitelikler” sayfa 811.

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; *GetMsgOpts* parametresinde MQGMO_ACCEPT_TRUNCATED_MSG belirtildiğinde bu durum oluşur (ek bilgi için MQGMO-Options field 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 358 .

Not: MQGET çağrısındaki diğer tüm parametreler, yerel kuyruk yöneticisinin karakter kümesinde ve kodlamasında (*CodedCharSetId* kuyruk yöneticisi özniteliği 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 sıfırsa, 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 *Bufferi*çine 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_INCONTINUED_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 kilitlenmedi.

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_CCSID_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_CCSID_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 kullanılamıyor.

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_BACKED_OUT

(2003, X'7D3') İş birimi yedeklendi.

MQRC_BUFFER_ERROR

(2004, X'7D4') Arabellek parametresi geçerli değil.

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 çağırısı.

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_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') İlintilendirme tanıtıcısı hatası.

MQRC_DATA_LENGTH_ERROR
(2010, X'7DA') Veri uzunluğu parametresi geçerli değil.

MQRC_DB2_NOT_AVAILABLE
(2342, X' 926 ') Db2 altsistemi kullanılamıyor.

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_MD_ERROR
(2026, X'7EA') İleti tanımlayıcısı geçerli değil.

MQRC_MSG_ID_ERROR
(2206, X'89E') İleti-tanıtıcısı hatası.

MQRC_MSG_SEQ_NUMBER_ERROR
(2250, X'8CA') İleti sıra numarası geçerli değil.

MQRC_MSG_TOKEN_HATASI
(2331, X'91B') İleti simgesinin kullanımı geçerli değil.

MQRC_NO_MSG_AVAIABLE
(2033, X'7F1') İleti yok.

MQRC_NO_MSG_UNDER_CURSOR
(2034, X'7F2') Browse imleci iletide konumlandırılmamış.

MQRC_NOT_OPEN_FOR_BROWSE
(2036, X'7F4') Kuyruk göz atma için açık değil.

MQRC_NOT_OPEN_FOR_INPUT
(2037, X'7F5') Kuyruk giriş için açık değil.

MQRC_OBJECT_CHANGED
(2041, X'7F9') Nesne tanımlaması açıldığından beri değiştirildi.

MQRC_OBJECT_ZARAR
(2101, X'835 ') Nesne zarar gördü.

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_Q_DELETED
(2052, X'804 ') Kuyruk silinmiştir.

MQRC_Q_INDEX_TYPE_ERROR
(2394, X'95A') Kuyruk yanlış dizin tipi 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_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 çağrılarını karışımı desteklenmiyor.

MQRC_UOW_NOT_VAR

(2255, X'8CF') Kuyruk yöneticisinin kullanması için kullanılabilecek 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 iletisi yalnızca bu tutamaçla görünür olacak şekilde kilitlenir.

MQGMO_UNLOCK seçeneği belirtilirse, önceden kilitlenmiş bir iletinin kilidi açılır. Bu durumda hiçbir ileti alınmaz; *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 taşıyıcı, vekil, iletiyi uygulamaya döndürmek üzere oluncaya kadar uygulamanın kaybını saptayamadığı için, *bundan sonra* ileti kuyruktan kaldırıldığını saptamıştır. 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](#).

This situation can arise with IBM MQ clients as well as with applications that are running on the same platform as the queue-manager.
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* bir kuyruksa (yani, yerel bir kuyruksa), iş birimi içindeki tüm iletiler aynı anda kullanılabilir olur.
 - Kuyruk bir *paylaşılan* kuyruksa, iş birimi içindeki iletiler 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 *CorrelId* belirtilerek, 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 başlara 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 tanıtilan 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 iletileri 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 niteliğın tanımsız olduğunu gösteren özel bir değerdır. Sonuç olarak, kuyruk yönetıcısı ileti kuyruktan alındığında dönüştürme girişiminde bulunmaz.

Not: MQGET çağırısında MQFMT_NONE biçim adı bulunan bir ileti için MQGMO_CONVERT belirtilirse ve iletinin karakter kümesi ya da kodlaması *MsgDesc* değıştirgesinde belirtilenden farklıysa, ileti *Buffer* parametresine döndürölür (başka hata varsayılarak), ancak çağrı tamamlanma kodu MQCC_UYANCE ve neden kodu MQRC_FORMAT_ERROR ile tamamlanır.

MQFMT_NONE ' u ileti verilerinin niteliğı, dönüştürme gerektirmedię anlamına geldiğinde ya da gönderme ve alma uygulamaları, kendileriyle ileti verilerinin gönderileceğı form arasında anlaştığında, 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” sayfa 879 .

İletideki kullanıcı verileri, desteklenen karakter kümeleri ve kodlamalar arasında dönüştürölülebiliyor. 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 dönüştürölümez. Bu denenirse neden kodu MQRC_SOURCE_CCSD_ERROR ya da MQRC_TARGET_CCSD_ERROR sonucu, bu denenirse ve ileti dönüştürölmemiş olarak döndürölür. Unicode karakter kümesi UCS-2 , bu tür bir karakter kümesine bir örnektir.

MQGET ' den dönüşte, aşağıdaki neden kodu iletinin başarıyla dönüştürölmüş olduğunu gösterir:

- MQRC_NONE

The following reason code indicates that the message *Olabilir* 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 dönüştürölmediğini gösterir.

Not: The interpretation of this reason code is true for conversions performed by a user-written exit *yalnızca* if the exit conforms to the processing guidelines described in “Veri dönüştürme” sayfa 879.

9. İletileri almak için nesne yönelimli arabirimi kullanırken, bir MQGET çağırısına ilişkin ileti verilerini tutmak için bir arabellek belirtmemeyi seçebilirsiniz. Ancak, sürüm 7 öncesi IBM MQsürümlerinde, bir arabellek belirtilmediğinde bile MQGET ' in neden kodu MQRC_CONVERTED_MSG_TO_BIG ile başarısız olması mümkün oldu. IBM MQ Sürüm 7 'den, ileti alma arabelleğinin büyüklüğünü kısıtlamadan nesne yönelimli bir uygulama kullanarak bir ileti aldığınızda, uygulama MQRC_CONVERTED_MSG_TOO_BIG ile başarısız olmaz ve dönüştürölün 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 Sürüm 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 dönüştürülmesi yalnızca *Tümü* aşağıdaki 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 105. İletiyi 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 ^A	Önceden okuma özelliği etkinleştirildiğinde izin verilmeyen MQGET seçenekleri ^B
MQGET MD değerleri	MsgId ^C CorrelId ^C	Kodlama CodedCharSetId	
MQGET MQGMO seçenekleri	MQGMO_BEKLE MQGMO_NO_BEKLEME MQGMO_FAIL_IF_QUIESCING MQGMO_BROWSE_FIRST ^D MQGMO_BROWSE_NEXT ^D MQGMO_BROWSE_MESSAGE_UNDER_CURSOR ^D	MQGMO_SYNCPOINT_IF_PERSISTENT MQGMO_NO_SYNCPOINT MQGMO_ACCEPT_TRUNCATED_MSG MQGMO_CONVERT MQGMO_LOGICAL_ORDER MQGMO_COMPLE_MSG MQGMO_ALL_MSGS_AVALABİLİR MQGMO_ALL_SEGMENTS_AVALABİLİR 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 ^D MQGMO_LOCK MQGMO_UNLOCK

Çizelge 105. İleriye okurken izin verilen MQGET seçenekleri geçerli kılındığında (devamı var)			
	Önceden okuma etkinleştirildiğinde ve MQGET çağrıları arasında değiştirilebilir olduğunda izin verilir	Okuma önden okuma etkinleştirildiğinde izin verilir, ancak MQGET çağrıları arasında değiştirilemez ^A	Önceden okuma özelliği etkinleştirildiğinde izin verilmeyen MQGET seçenekleri ^B
MQGMO değerleri		MsgHandle	

- Bu seçenekler MQGET çağrıları arasında değiştirilirse, bir MQRC_OPTIONS_CHANGED neden kodu döndürülür.
 - Bu seçenekler ilk MQGET çağrısında belirtilirse, önceden okuma geçersiz kılınır. İzleyen bir MQGET çağrısında bu seçenekler belirtilirse, MQRC_OPTIONS_ERROR neden kodunda bir neden kodu döndürülür.
 - The client applications need to be aware that if the MsgId and CorrelId values are altered between MQGET calls messages with the previous values might have already been sent to the client and remain in the client read ahead buffer until consumed (or automatically purged).
 - İlk MQGET çağrısı, önceden okuma etkinleştirildiğinde iletilerin bir kuyruktan mı göz atılacağını, yoksa kuyruktan mı gönderileceğini belirler. Uygulama, göz atma ve bir MQRC_OPTIONS_CHANGED neden kodu alma girişiminde bulunursa, bu neden kodu döndürülür.
 - MQGMO_MSG_UNDER_CURSOR önden okuma ile olanaklı değil. İleriye okuma etkinleştirildiğinde, iletiler göz atılabilir ya da var olabilir, ancak her ikisinin birleşimi etkinleştirilmez.
- 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.
 - 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);
```

Değiştirelileri 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.
```

Değiştirelileri 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

```

Değiştirgeleri 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 çağrısı, 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 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, and on IBM i for applications running in compatibility mode, the MQCONN call can be omitted, and the following value specified for *Hconn* :

MQHC_DEF_HCONN

Varsayılan bağlantı tanıtıcısı.

Önemli:  Uyumluluk kipini kullanmaya devam etseniz, uygulamanızı yeniden derlemeyi ve bunun yerine statik bağlı çağrıları kullanmayı düşünün. Bkz. [IBM üzerindeki RPG uygulamaları için uyumluluk kipi](#)

Hobj

Tip: MQHOBJ -giriş

Bu tanıtıcı, gerekli özniteliklere sahip nesneyi (herhangi bir tipte) gösterir. Tanıtıcı, MQ00_SORGULAMA seçeneğini belirten önceki bir MQOPEN çağrısı 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 700 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 öge numaraları kullanılabilir.
3. *SelectorCount* parametresi sıfırsa, *Selectors* ifadesine başvurulmaz. Bu durumda, C ya da S/390 çevirici programlarında yazılan programlarla geçirilen parametre adresi boş değeri 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 106 sayfa 687 MQINQ öznitelik seçicileri
- Ad listeleri için Çizelge 107 sayfa 690 MQINQ öznitelik seçicileri
- Süreç tanımlamaları için Çizelge 108 sayfa 690 MQINQ öznitelik seçicileri
- Kuyruk yöneticisi için Çizelge 109 sayfa 691 MQINQ öznitelik seçicileri

Tüm seçiciler, **Not** sütununda belirtildiği şekilde, tüm IBM MQ platformlarında aşağıdaki gibi desteklenir:

DEĞİL z/OS

hariç z/OS tüm platformlarda desteklenir

z/OS

z/OS üzerinde desteklenen **yalnızca**

Çizelge 106. Kuyruklar için MQINQ 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_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-tesis 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	

Çizelge 106. Kuyruklar içinMQINQ öznelik seçicileri (devamı var)			
Seçici	Alan uzunluğu	Tanım	Not
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	
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	

Çizelge 106. Kuyruklar içinMQINQ öznitelik seçicileri (devamı var)

Seçici	Alan uzunluğu	Tanım	Not
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_SEQUEN CE	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	
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 paylaşılp paylaşılamayacağı	
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	

Çizelge 106. Kuyruklar içinMQINQ öznitelik seçicileri (devamı var)

Seçici	Alan uzunluğu	Tanım	Not
MQIA_TRIGGER_MSG_PRIORITY	MQLONG	Tetikleyiciler için eşik iletisi önceliği	
MQIA_TRIGGER_TYPE	MQLONG	Tetikleme Tipi	
MQIA_USAGE	MQLONG	Kullanım	

Çizelge 107. 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 108. 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	
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

Çizelge 109. Kuyruk yöneticisi 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_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 öznelik	
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	Katılmak 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
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 dinleyicisi için genel LU adı	z/OS

Çizelge 109. Kuyruk yöneticisi içinMQINQ öznelik seçicileri (devamı var)

Seçici	Alan uzunluğu	Tanım	Not
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
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

Çizelge 109. Kuyruk yöneticisi içinMQINQ öznitelik seçicileri (devamı var)

Seçici	Alan uzunluğu	Tanım	Not
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 öznitelięi	DEĞİ Lz/O S
MQIA_BRIDGE_EVENT	MQLONG	IMS köprüsü olaylarına ilişkin denetim öznitelięi	z/OS
MQIA_CHANNEL_AUTO_DEF	MQLONG	Otomatik kanal tanımlamasına ilişkin denetim öznitelięi	DEĞİ Lz/O S
MQIA_CHANNEL_AUTO_DEF_EVENT	MQLONG	Otomatik kanal tanımlaması olaylarına ilişkin denetim öznitelięi	DEĞİ Lz/O S
MQIA_CHANNEL_EVENT	MQLONG	Kanal olaylarına ilişkin denetim öznitelięi	
MQIA_CHINIT_ADAPTERS	MQLONG	IBM MQ çağrılarını işlemek için kullanılacak bağdařtırıcı alt görevlerinin sayısı	z/OS
MQIA_CHINIT_DISPATCHERS	MQLONG	Kanal başlatıcısı için kullanılacak daęıtıcıların 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.	
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 öznitelięi	
MQIA_COMMAND_LEVEL	MQLONG	Kuyruk yöneticisi tarafından desteklenen komut düzeyi	
MQIA_CONFIGURATION_EVENT	MQLONG	Yapılandırma olaylarına ilişkin denetim öznitelięi	DEĞİ Lz/O S

Çizelge 109. Kuyruk yöneticisi içinMQINQ öznitelik seçicileri (devamı var)			
Seçici	Alan uzunluğu	Tanım	Not
MQIA_DEF_CLUSTER_XMIT_Q_TYPE	MQLONG	Küme gönderici kanalları 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 Yüğü Yöneticisi ile kaydolup olmadığı	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 özniteliği. GROUP birimi kurtarma (recovery) yok etme birimi 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 özniteliği	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 özniteliği	DEĞİ Lz/O S
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).  Uyarı: Bu değeri varsayılan 5000 değerinin altında ayarlamamalısınız.	

Çizelge 109. Kuyruk yöneticisi içinMQINQ öznitelik seçicileri (devamı var)

Seçici	Alan uzunluğu	Tanım	Not
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 IBM MQ 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	
MQIA_PUBSUB_MODE	MQLONG	Yayınlama/abone olma motorunun ve kuyruğa alınan yayınlama/abone olma arabiriminin çalışıp çalışmadığı. Uygulama programlama arabirimi 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 arabirimi 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	

Çizelge 109. Kuyruk yöneticisi içinMQINQ öznelik seçicileri (devamı var)			
Seçici	Alan uzunluğu	Tanım	Not
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	
MQIA_SSL_FIPS_REQUIRED	MQLONG	Kriptografi için yalnızca FIPS onaylı algoritmalar kullan	
MQIA_SSL_RESET_COUNT	MQLONG	SSL anahtarı sıfırlama sayısı	
MQIA_START_STOP_EVENT	MQLONG	Durdurma olaylarının başlatılması için denetim özneliği	DEĞİ Lz/O S
MQIA_STATISTICS_AUTO_CLUSTER	MQLONG	Küme gönderen kanallarına ilişkin istatistik izleme bilgilerinin derlemine denetler	
MQIA_STATISTICS_CHANNEL	MQLONG	Kanallar için istatistik verilerinin toplanmasını denetler	
MQIA_STATISTICS_INTERVAL	MQLONG	İstatistik izleme verileri yazma sıklığı	DEĞİ Lz/O S

Çizelge 109. Kuyruk yöneticisi için MQINQ öznitelik seçicileri (devamı var)

Seçici	Alan uzunluğu	Tanım	Not
MQIA_STATISTICS_MQI	MQLONG	Kuyruk yöneticisine ilişkin istatistik izleme bilgilerinin derlemine 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, *IntAttr*s dizideki öğelerin sayısıdır. Sıfır geçerli bir değerdir.

IntAttrCount , en az *Selectors* parametresindeki MQIA_* seçicilerinin sayısını gösteriyorsa, istenen tüm tamsayı öznitelikleri döndürülür.

IntAttr

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. *IntAttr*s dizisine karşılık gelen öğe için döndürülür.

IntAttrCount ya da *SelectorCount* parametresi sıfırsa, *IntAttr*s 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, *CharAttr*s 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 özniteliğin 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: `MLONG` -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:

MQRC_CHAR_ATTRS_TOO_SHORT

(2008, X'7D8') Karakter öznitelikleri için yeterli yer yok.

MQRC_INT_ATTR_COUNT_TOO_SMALL

(2022, X'7E6') Tamsayı öznitelikleri için yeterli yer izni yok.

MQRC_SELECTOR_NOT_FOR_TYPE

(2068, X'814') Seçici, kuyruk tipi için geçerli değil.

`CompCode` `MQCC_FAILED` ise:

MQRC_ADAPTER_NOT_AVAILABLE

(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_MISMATCH

(2157, X'86D') Birincil ve ana ASID ' ler farklı.

MQRC_CALL_IN_PROGRESS

(2219, X'8AB') Önceki arama tamamlanmadan önce girilen MQI çağırısı.

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_CHAR_ATTR_LENGTH_ERROR

(2006, X'7D6') Karakter özniteliklerinin uzunluğu geçerli değil.

MQRC_CHAR_ATTRS_ERROR

(2007, X'7D7') Karakter öznitelikleri dizgisi geçerli değil.

MQRC_CICS_WAIT_FAILED

(2140, X'85C') Bekleme isteği CICStarafından reddedildi.

MQRC_CONNECTION_BROKEN

(2009, X'7D9') Kuyruk yöneticisine bağlantı kaybedildi.

MQRC_CONNECTION_NOT_AUTHORIZED

(2217, X'8A9') Bağlantı için yetkili değil.

MQRC_CONNECTION_STOPPING

(2203, X'89B') Bağlantı sona erdiriliyor.

MQRC_HCONN_ERROR

(2018, X'7E2') Bağlantı tanıtıcısı geçerli değil.

MQRC_HOBJ_ERROR

(2019, X'7E3') Nesne tanıtıcısı geçerli değil.

MQRC_INT_ATTR_COUNT_ERROR

(2021, X'7E5') Tamsayı öznitelikleri sayısı geçerli değil.

MQRC_INT_ATTRS_ARRAY_ERROR

(2023, X'7E7') Tamsayı öznitelikleri dizisi geçerli değil.

MQRC_NOT_OPEN_FOR_INQUIRE

(2038, X'7F6') Kuyruk sorgulamak için açık değil.

MQRC_OBJECT_CHANGED

(2041, X'7F9') Nesne tanımlaması açıldığından bu yana değiştirildi.

MQRC_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 bkz. [İletiler ve neden kodları](#)

Kullanım notları

1. Döndürülen değerler, seçilen özniteliklerin bir anlık görünüşüdür. Uygulamanın döndürülen değerler üzerinde işlem yapılabilmesi 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 112 sayfa 813](#) .

3. Sorgulanmakta olan nesne bir diğer ad kuyruğuna, 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 kuyruğusa, 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 MQRC_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 çıktıysa, yalnızca listelenen öznitelikler sorgulanabilir. **QType** özniteliği, 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 deęişmez bir baę tanımlı olmadan açabilirsiniz. Bunu MQOPEN çağrısında belirtilen MQOO_BIND_NOT_FIXED ile açabilirsiniz. Dięer bir seçenek olarak, MQOO_BIND_AS_Q_DEFbelirtisini belirtin ve kuyruğun **DefBind** özniteliğini MQBND_BIND_NOT_FIXEDolarak ayarlayın. Deęişmez baę tanımlı olmayan bir küme kuyruğunu açsanız, kuyruk için art arda yapılan MQINQ çağrıları, küme kuyruğunun farklı eşgörünümlerini sorgulayabilir. Ancak, tipik olarak, tüm eşgörünümler için aynı öznitelik deęerleri vardır.

- Bir küme için dięer ad kuyruęu nesnesi tanımlanabilir. TARGTYPE ve TARGET küme öznitelikleri olmadıęından, dięer ad kuyruęunda bir MQOPEN işlemi gerçekleştiren işlem, dięer adın çözölen nesneden farkında deęil.

İlk MQOPENsırasında, dięer ad kuyruęu, bir kuyruk yöneticisine ve kümedeki bir kuyruęa çözömlenir. Ad çözme işlemi, uzak kuyruk yöneticisinde yeniden gerçekleşir ve dięer ad kuyruęunun TARGTPE 'nin çözöldüęü yerdir.

Dięer ad kuyruęu bir konu dięer adına çözöülürse, dięer ad kuyruęuna gönderilen iletilerin yayınlanması, bu uzak kuyruk yöneticisinde yer alır.

Bkz. [Küme kuyrukları](#)

5. Bir dizi öznitelięi sorgulamak ve daha sonra, bazılarını MQSET çağrısını kullanarak ayarlamak isteyebilirsiniz. Sorgulamak ve verimli bir şekilde ayarlamak için, ayırıcı dizilerinin başında ayarlanacak öznitelikler 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), döndürölen neden kodu, aşıęıdaki 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. Aşıęıdaki konu nesne öznitelikleriyle ilgili bilgi içerir:
 - [“Kuyruklara ilişkin öznitelikler” sayfa 811](#)
 - [“Ad listelerine ilişkin öznitelikler” sayfa 842](#)
 - [“Süreç tanımlamalarına ilişkin öznitelikler” sayfa 844](#)
 - [“Kuyruk yöneticisine ilişkin öznitelikler” sayfa 778](#)

C çağırma

```
MQINQ (Hconn, Hobj, SelectorCount, Selectors, IntAttrCount, IntAttrs,
      CharAttrLength, CharAttrs, &CompCode, &Reason);
```

Deęiştirgeleri aşıęıdaki gibi bildirin:

```
MQHCONN  Hconn;          /* Connection handle */
MQHOBJ    Hobj;          /* Object handle */
MQQLONG   SelectorCount; /* Count of selectors */
MQQLONG   Selectors[n];  /* Array of attribute selectors */
MQQLONG   IntAttrCount;  /* Count of integer attributes */
MQQLONG   IntAttrs[n];   /* Array of integer attributes */
MQQLONG   CharAttrLength; /* Length of character attributes buffer */
MQCHAR    CharAttrs[n];  /* Character attributes */
MQQLONG   CompCode;      /* Completion code */
MQQLONG   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:

```
dc1 Hconn          fixed bin(31); /* Connection handle */
dc1 Hobj           fixed bin(31); /* Object handle */
dc1 SelectorCount  fixed bin(31); /* Count of selectors */
dc1 Selectors(n)   fixed bin(31); /* Array of attribute selectors */
dc1 IntAttrCount   fixed bin(31); /* Count of integer attributes */
dc1 IntAttrs(n)    fixed bin(31); /* Array of integer attributes */
dc1 CharAttrLength fixed bin(31); /* Length of character attributes
                                buffer */
dc1 CharAttrs      char(n);       /* Character attributes */
dc1 CompCode       fixed bin(31); /* Completion code */
dc1 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 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 tutamacındaki bir özelliği geciktiren iş parçacığında geçerli bir bağlantı oluşturulmalıdır; tersi durumda, çağrı MQRC_CONNECTION_BROKEN ile başarısız olur.

Hmsg

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.

NAME

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_INQUIRE_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 değiştirgeyi MQINQMP çağrısından dönüşte, *Değer* veri tipine ayarlanır. Veri tipi aşağıdakilerden biri olabilir:

MQTYPE_BOOLEAN

Bir boole.

MQTYPE_BYTE_STRING

Bayt dizilimi.

MQTYPE_INT8

8 bitlik işaretli tamsayı.

MQTYPE_INT16

16 bitlik bir işaretli tamsayı.

MQTYPE_INT32

32 bitlik işaretli 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 dizisi.

MQTYPE_NULL

Özellik var, ancak boş değere sahip.

Özellik değerinin veri tipi tanınmadıysa, MQTYPE_STRING değeri döndürülür ve değer dizgi gösterimi *Değer* alanına yerleştirilir. 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 seçeneği belirtilirse, özellik değerinin dönüştürülmesi istenilir. Özelliğin döndürülmesini istediğiniz veri tipini belirtmek için, giriş olarak *Type* (Tip) kullanın. Veri tipi dönüştürmesine ilişkin ayrıntılar için MQIMPO yapısının MQIMPO_CONVERT_TYPE seçeneğinin açıklamasına bakın.

Tip dönüştürme isteğinde bulunmuyorsanız, giriş sırasında aşağıdaki değeri kullanabilirsiniz:

MQTYPE_AS_SET

Özelliğin değeri, veri tipini dönüştürmeden döndürülür.

ValueLength

Tip: MQHOME-girişi

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, MQIMPO_QUERY_LENGTH seçeneği belirtilmişse, sıfır değerini de belirtin; bu durumda değer döndürülmez.

Değer

Tip: MQBYTEEx *ValueLength* -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 hatayla sonuçlanabilir.

ValueLength özelliği, özellik değerinin uzunluğundan küçükse, olası değer *Değer* 'e taşınır ve çağrı, MQCC_FAILED tamamlanma kodu ile başarısız olur ve MQRC_PROPERTY_VALUE_TOO_BIG neden olur.

Değer alanındaki verilerin karakter takımı, *InqPropOpts* parametresindeki *ReturnedCCSID* alanı tarafından verilir. The encoding of the data in *Değer* 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 ayrıştırılamadı.

CompCode MQCC_FAILED ise:

MQRC_ADAPTER_NOT_AVAM

(2204, X'089C') Bağdaştırıcı kullanılamıyor.

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, PROPDSC, TYPE,
VALUELENGTH, VALUE, DATALENGTH, COMPCODE, REASON.
```

Değiştirelileri 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 CMQCHRVV.
** Property descriptor
01 PROPDSC.
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,PROPDSC,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
PROPDSC	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 çağırısı, bir ileti tanıtıcısını arabellek içine dönüştürür ve MQBUFMH çağırısının tersi olur.

Sözdizimi

MQMHBUF (*Hconn*, *Hmsg*, *MsgHBufOpts*, *Ad*, *MsgDesc*, *BufferLength*, *Buffer*, *DataLength*, *CompCode*, *Reason*)

Parametreler

Hconn

Tip: MQHCONN-input

Bu tanıttıcı, kuyruk yöneticiyle olan bağlantıyı gösterir. *Hconn* değeri, *Hmsg* değiştirgesinde belirtilen ileti tanıttıcısını yaratmak için kullanılan bağlantı tanıttıcısı ile eşleşmelidir.

İleti tanıttıcısı MQHC_UNASSOCIATED_HCONN kullanılarak yaratıldıysa, ileti tanıttıcısını silmeye ilişkin iş parçasısında geçerli bir bağlantı kurulmalıdır. Geçerli bir bağlantı kurulmamışsa, çağrı MQRC_CONNECTION_BROKEN ile başarısız olur.

Hmsg

Tip: MQHMSG-giriş

Bu, arabelleğin doldurulması gereken ileti tanıttıcısıdır. Değer önceki bir MQCRTMH çağrısı tarafından döndürüldü.

MsgHBufOpts

Tip: MQMHBO-input

MQMHBO yapısı, uygulamaların, ileti tanıttıcılarından arabelleklerin nasıl üretildiğini denetleyen seçenekleri belirlemesine olanak sağlar.

Ayrıntılar için bkz. [“MQMHBO-İleti tanıttıcısı-arabellek seçenekleri” sayfa 460.](#)

NAME

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

İletin tüm özelliklerini arabelleğe koy

MQPROP_INQUIRE_ALL_USR

İletin 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 tanıttıcısını 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_MISMATCH

(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 çağrısı.

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.
```

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 MQMHBUF
01 MSGHBUFOPTS.
   COPY CMQMHBV.
** Property name
01 NAME          PIC S9(9) BINARY.
   COPY CMQCHRVV.
** Message descriptor
01 MSGDESC       PIC S9(9) BINARY.
   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);
```

Değiştirgeleri aşağıdaki gibi bildirin:

```

dcl Hconn          fixed bin(31); /* Connection handle */
dcl Hmsg           fixed bin(63); /* Message handle */
dcl MsgHBufOpts    like MQMHBV;  /* 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)
```

Değiştirgeleri aşağıdaki gibi bildirin:

HCONN	DS	F	Connection handle
HMSG	DS	D	Message handle
MSGHBUFOPTS	CMQMHBV	,	Options that control the action of MQMHBUF
NAME	CMQCHRVV	,	Property name
MSGDESC	CMQMDV	,	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 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, and on IBM i for applications running in compatibility mode, the MQCONN call can be omitted, and the following value specified for *Hconn* :

MQHC_DEF_HCONN

Varsayılan bağlantı tanıtıcısı.

Önemli:  Uyumluluk kipini kullanmaya devam etseniz, uygulamanızı yeniden derlemeyi ve bunun yerine statik bağlı çağrılar kullanmayı düşünün. Bkz. [IBM üzerindeki RPG uygulamaları için uyumluluk kipi](#)

ObjDesc

Tip: MQOD-giriş/çıkış

Bu yapı, açılacak nesneyi tanımlayan bir yapıdır; ayrıntılar için [“MQOD-Nesne tanımlayıcısı” sayfa 462](#) '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 çağrısı tarafından döndürülen *Hobj* ile sonraki işlemler, model kuyruğunda değil, yeni dinamik kuyruğun üzerinde gerçekleştirilir. Bu, MQINQ ve MQSET çağrılarının 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 811](#)). Dinamik kuyruklar için geçerli olan kapatma seçeneklerine ilişkin bilgi için, MQCLOSE çağrısının açıklamasına bakın.

Seçenekler

Tip: MQHOME-girişi

Aşağıdaki seçeneklerden en az birini belirlemelisiniz:

- MQOO_BROWSE
- MQOO_INPUT_ * (bu öğelerden yalnızca biri)
- MQOO_SORGULAMA
- MQOO_OUTPUT
- MQOO_SET
- MQOO_BIND_ * (bu öğelerden yalnızca biri)

Bu seçeneklerin ayrıntıları için aşağıdaki tabloya bakın; diğer seçenekler zorunlu olarak belirtilebilir. 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 destekliyse). Bileşimleri not edilir; diğer tüm birleşimler geçerlidir. Yalnızca, *ObjDesc* ile belirtilen nesne tipi için geçerli olan seçeneklere izin verilir. Aşağıdaki çizelge, sorgular ve konular için geçerli MQOPER seçeneklerini göstermektedir.

Seenek	Diğler Ad ¹	Yerel ve Model	Uzak	Yerel Olmayan Kme	Dağıtım listesi	Konu
MQOO_INPUT_AS_Q_DEF	Evet	Evet	Hayır	Hayır	Hayır	Hayır
MQOO_INPUT_SHARED	Evet	Evet	Hayır	Hayır	Hayır	Hayır
MQOO_INPUT_EXCLUSIVE	Evet	Evet	Hayır	Hayır	Hayır	Hayır
MQOO_OUTPUT	Evet	Evet	Evet	Evet	Evet	Evet
MQOO_BROWSE	Evet	Evet	Hayır	Hayır	Hayır	Hayır
MQOO_CO_OP	Evet	Evet	Hayır	Hayır	Hayır	Hayır
MQOO_SORGULAMA	Evet	Evet	²	Evet	Hayır	Hayır
MQOO_SET	Evet	Evet	²	Hayır	Hayır	Hayır
MQOO_BIND_ON_Aç ³	Evet	Evet	Evet	Evet	Evet	Hayır
MQOO_BIND_NOT_FIXED ³	Evet	Evet	Evet	Evet	Evet	Hayır
MQOO_BIND_ON_GROUP ³	Evet	Evet	Evet	Evet	Evet	Hayır
MQOO_BIND_AS_Q_DEF ³	Evet	Evet	Evet	Evet	Evet	Hayır
MQOO_SAVE_AL_CONTEXT	Evet	Evet	Hayır	Hayır	Hayır	Hayır
MQOO_PASS_IDENTITY_CONTEXT	Evet	Evet	Evet	Evet	Evet	⁴
MQOO_PASS_ALL_CONTEXT	Evet	Evet	Evet	Evet	Evet	Evet
MQOO_SET_IDENTITY_CONTEXT	Evet	Evet	Evet	Evet	Evet	⁴
MQOO_SET_AL_CONTEXT	Evet	Evet	Evet	Evet	Evet	Evet
MQOO_NO_READ_AHEAD	Evet	Evet	Hayır	Hayır	Hayır	Hayır
MQOO_READ_AHEAD	Evet	Evet	Hayır	Hayır	Hayır	Hayır
MQOO_READ_AHEAD_AS_Q_DEF	Evet	Evet	Hayır	Hayır	Hayır	Hayır
MQOO_ALTERNATE_USER_AUTHORITY	Evet	Evet	Evet	Evet	Evet	Evet
MQOO_FAIL_IF QUIESCING	Evet	Evet	Evet	Evet	Evet	Evet
MQOO_RESOLVE_LOCAL_Q	Evet	Evet	Evet	Evet	Hayır	Hayır
MQOO_RESOLVE_LOCAL_TOPIC	Hayır	Hayır	Hayır	Hayır	Hayır	Evet
MQOO_NO_MULTICAST	Hayır	Hayır	Hayır	Hayır	Hayır	Evet

Notlar:

1. Diğler adlara ilişkin seeneklerin geerliliğı, diğler adın özdüğü kuyruğına ilişkin seeneğın geerliliğine baėlıdır.
2. Bu seenek yalnızca uzak kuyruğun yerel tanımlaması için geerlidir.
3. Bu seenek herhangi bir kuyruk tipi için belirlenebilir; ancak, kuyruk bir küme kuyruğı deėilse yoksayılır. Ancak, *DefBind* kuyruk özniteliğı, diğler ad kuyruğı bir kümede yer almıyorsa bile, temel kuyruğı geersiz kılar.
4. Bu öznitelikler bir konu ile birlikte kullanılabilir; ancak, herhangi bir aboneye gönderilen baėlam alanları deėil, yalnızca alıkonan ileti için ayarlanan baėlamı etkiler.

Eriřim seenekleri: Ařağıdaki seenekler, nesne üzerinde geekleřtirilebilecek iřlemlerin tipini denetler:

MQOO_INPUT_AS_Q_DEF

Kuyruk tanımlı varsayılan iletileri kullanarak iletileri almak için kuyruğı açın.

Kuyruk sonraki MQGET aėrılarıyla kullanılmak üzere açılmıştır. Eriřim tipi, *DefInputOpenOption* kuyruk özniteliğinin deėerine baėlı olarak paylaşılan ya da dışlayıcı bir tiptir; ayrıntılar için “Kuyruklara ilişkin öznitelikler” sayfa 811 ' e bakın.

Bu seçenek yalnızca yerel, diğer ad ve model kuyrukları için geçerlidir; uzak kuyruklar, dağıtım listeleri ve kuyruklar olmayan nesneler için geçerli değildir.

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 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 nesneler için geçerli değildir.

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.

Bu seçenek yalnızca yerel, diğer ad ve model kuyrukları için geçerlidir; uzak kuyruklar, dağıtım listeleri ve kuyruklar olmayan nesneler 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 çağrılarıyla kullanılmak üzere açılır.

InhibitPut kuyruk özniteliği 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 özniteliği MQQA_GET_INHIBITED olarak ayarlansa bile, bu seçeneklerden biriyle MQOPER çağrısı başarılı olabilir (ancak, öznitelik bu değere ayarlandığında sonraki MQGET çağrıları başarısız olur).
- Kuyruk paylaşılamaz olarak tanımlandıysa (yani, *Shareability* kuyruk özniteliğinin 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 çözümleyeceği temel kuyruğa karşıdır.
- *ObjectQMgrName* bir kuyruk yöneticisi diğer adının adıysa, bu seçenekler geçerli değildir; bu, kuyruk yöneticisi aliting için kullanılan bir uzak kuyruğun yerel tanımlamasındaki *RemoteQMgrName* özniteliğinin değeri yerel kuyruk yöneticisinin adıdır.

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 iletiyle

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 nesneler 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_UNMARKET_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 nesneler için geçerli değildir.

MQOO_SORGULAMA

Öznitelikleri sorgulamak için nesneyi açın.

İzleyen MQINQ çağrısıyla 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 geçerli değildir; bu, kuyruk yöneticisi aliting için kullanılan bir uzak kuyruğun yerel tanımlamasındaki *RemoteQMGrName* özniteliğinin değeri yerel kuyruk yöneticisinin adıdır.

MQOO_SET

Öznitelikleri ayarlamak için kuyruğu açın.

Kuyruk, sonraki MQSET çağrısıyla kullanılmak üzere açılmıştır.

Bu seçenek, dağıtım listesi dışındaki tüm kuyruk tipleri için geçerlidir. Uzak bir kuyruğun yerel tanımlamasının adı *ObjectQMGrName* ise geçerli değildir; bu, kuyruk yöneticisi aliting için kullanılan bir uzak kuyruğun yerel tanımlamasındaki *RemoteQMGrName* özniteliğinin değeri yerel kuyruk yöneticisinin adıdır; bu durumda da geçerlidir.

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_Açık

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 yoksayılır.

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 eşgörünümlerine 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 aracı (MCA) tarafından değiştirilmesini de sağlar.

Not: Dizideki ardışık iletiler sunucu uygulamasının farklı örneklerine gönderilebileceği için, bir hareketin tamamlanması için bir *serisi* ileti değiştirilmesi gereken istemci ve sunucu uygulamaları MQOO_BIND_NOT_FIXED (ya da *DefBind* değeri MQBND_BIND_NOT_FIXED değeri olduğunda MQOO_BIND_AS_Q_DEF) kullanılmamalıdır.

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 çağrılar küme kuyruğunun farklı eşgörünümlerini sorgulayabilir, ancak tipik olarak tüm eşgörünümler 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 yoksayılır.

MQOO_BIND_ON_GROUP

Bir uygulamanın, bir ileti grubunun tümünün aynı hedef eşgörünüme 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 yoksayılır.

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 çağrılarını kullanan bir kuyruğa aktarılabilecek bir iletiye aktarılabilir. “MQPMO-put-message seçenekleri” sayfa 482’inde açıklanan MQPMO_PASS_IDENTITY_CONTEXT ve MQPMO_PASS_ALL_CONTEXT seçeneklerine bakın.

Bir ileti başarıyla alınıncaya kadar, bir kuyrukte 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 nesneler 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* değiştirgesinde 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_AHEAD ile MQOP ' u çağırdığınızda, IBM MQ istemcisi yalnızca belirli koşullar karşılandığında önden okuma seçeneğini etkinleştirir. Bu koşullar şunları içerir:

- İstemci ve uzak kuyruk yöneticisi, WebSphere MQ Sürüm 7 ya da sonraki bir sürümde olmalıdır.
- İstemci uygulaması, iş parçacıklı IBM MQ MQI istemci kitaplıklarıyla derlenmeli ve ilişkilendirilmelidir.
- İstemci kanalının TCP/IP protokolünü kullanması gerekir
- Kanal, hem istemci hem de sunucu kanalı tanımlamalarında sıfır dışında bir SharingConversations (SHARECNV) ayarına sahip olmalıdır.

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 öznitemesiyle 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 yoksayılr.

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 kuyruğıdır. 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 kullanılamıyor.

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 CICS tarafından 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 tanıttıcı yok.

MQRC_HCONN_ERROR

(2018, X'7E2') Bağlantı tanıttıcısı geçerli değil.

MQRC_HOBJ_ERROR

(2019, X'7E3') Nesne tanıttıcısı 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-yönetici 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_DÜZEYİ

(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 aşağıdaki başlıklara bakın:

-  IBM MQ for z/OS için IBM MQ for z/OS iletileri, tamamlama ve neden kodları .
- z/OS dışındaki tüm diğer IBM MQ platformları için [İletiler ve neden kodları](#) .

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 kuyruğsa, dinamik bir yerel kuyruk yaratılır. “MQOpen-Open nesnesi” sayfa 712’inde 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 çağrılılarıyla kullanılamayacak 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 yayınlatabilmek 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 only) IGQ aracı tarafından kullanılan paylaşılan iletim kuyruğunun adına, uzak kuyruk yöneticisi adının çözülmesi (yalnızca yerel ve uzak kuyruk yöneticileri aynı kuyruk paylaşım grubuna ait olduğunda geçerlidir).
 - 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 kuyruğunsa, 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_Açık
- 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 çağrılarıyla 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 yayınlarak, 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* değiştirgesinde belirtilebilir. Ek ayrıntılar için [“MQTMC2 -Tetikleyici iletisi 2 \(karakter biçimi\)” sayfa 586](#) ' e bakın.
6. IBM işletim sistemi üzerinde, uyumluluk kipinde çalışan uygulamalar, uygulama tarafından yayınlanan ilk MQOPEN çağrısıyla otomatik olarak kuyruk yöneticisine bağlanır (uygulama, MQCONN çağrısını kullanarak kuyruk yöneticisine önceden bağlanmadysa).

Uyumluluk kipinde çalışmayan uygulamalar, bir nesneyi açmak için MQOPED çağrısını kullanmadan önce, kuyruk yöneticisine belirtik olarak bağlanmak için MQCONN ya da MQCONNx çağrısını yayınlamalıdır.

Önemli:  Uyumluluk kipini kullanmaya devam etseniz, uygulamanızı yeniden derlemeyi ve bunun yerine statik bağlı çağrılar kullanmayı düşünün. Bkz. [IBM üzerindeki RPG uygulamaları için uyumluluk kipi](#)

Önden okuma seçenekleri

MQOO_READ_AHEAD ile MQOP ' u çağırdığınızda, IBM MQ istemcisi yalnızca belirli koşullar karşılandığında önden okuma seçeneğini etkinleştirir. Bu koşullar şunları içerir:

- İstemci ve uzak kuyruk yöneticisi, WebSphere MQ Sürüm 7 ya da sonraki bir sürümde olmalıdır.
- İstemci uygulaması, iş parçacıklı IBM MQ MQI istemci kitaplıklarıyla derlenmeli ve ilişkilendirilmelidir.
- İstemci kanalının TCP/IP protokolünü kullanması gerekir
- Kanal, hem istemci hem de sunucu kanalı tanımlamalarında sıfır dışında bir SharingConversations (SHARECNV) ayarına sahip olmalıdır.

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 kılınır.
3. Uygulama bir IBM MQ istemcisi olarak çalışmıyorsa, ileriye doğru okuma seçenekleri yoksayılır.

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 şu ortamlarda desteklenir: AIX, HP-UX, IBM i, Solaris, Linux, Windows, artı IBM MQ MQI clients bu sistemlere bağlı olarak.

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 doğrulayarak, dağıtım listesinde olmayan tek bir kuyruğu açmak için de kullanılabilir.

2. *Options* değiştirgesinde 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 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, 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 ayarlandı.
 - Yanıt kayıtları (uygulama tarafından sağlandıysa), dağıtım listesindeki kuyruklara ilişkin tek tek tamamlama kodlarına ve neden kodlarına ayarlanmıştır.

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 özniteliğine bakın) kontrol ederken, dağıtım listesindeki her başarıyla açılan hedef ayrı bir tanıtıcı olarak sayılıyor. 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 çözümlerse, 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 tanımlayıcısı” sayfa 462 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 gerçekleştirmeden 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 üzerinde, kuyruk yöneticisi yalnızca güvenlik etkinleştirilmişse güvenlik denetimlerini gerçekleştirir. Güvenlik denetiyle 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* özniteliğe yapılan bir değişiklik.
 - Açık olan bir diğer ad kuyruğunun *TargetType* 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 that resolves through this definition as a queue-manager alias.
 - Uzak kuyruğun şu anda açık olan bir uzak kuyruğun farklı bir iletim kuyruğuna çözülmesine 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* özniteliğe ilişkin bir değişiklik.
 - (Yalnızca z/OS) *IntraGroupQueueing* 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* queue-manager 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 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 tutamaçları 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);
```

Değiştirelileri 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
```

Değiştirelileri 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);
```

Değiştirelileri 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
COMP CODE	DS	F	Completion code
REASON	DS	F	Reason code qualifying COMP CODE

Visual Basic çağırısı

```
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 tanıtıcı, kuyruk yöneticisiyle olan bağlantıyı gösterir. The value of *Hconn* was returned by a previous MQCONN or MQCONNEX call.

On z/OS for CICS applications, and on IBM i for applications running in compatibility mode, the MQCONN call can be omitted, and the following value specified for *Hconn* :

MQHC_DEF_HCONN

Varsayılan bağlantı tanıtıcısı.

Önemli:  Uyumluluk kipini kullanmaya devam etmeniz, uygulamanızı yeniden derlemeyi ve bunun yerine statik bağlı çağrılar kullanmayı düşünün. Bkz. [IBM üzerindeki RPG uygulamaları için uyumluluk kipi](#)

Hobj

Tip: MQHOBJ-input

Bu tanıtıcı, 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 tanımlayıcısı” sayfa 404](#).

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, MQMMDE' nin var olduğunu göstermek için MQFM_MD_UNUTPT olarak ayarlanmalıdır. Daha ayrıntılı bilgi için bkz. “MQMDE-İleti tanımlayıcı uzantısı” sayfa 453 .

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 kesinlemeyecek şekilde ifade eder.

PutMsgOpts

Tip: MQPMO-giriş/çıkış

Ayrıntılar için bkz. “MQPMO-put-message seçenekleri” sayfa 482.

BufferLength

Tip: MQHOME-giriş

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* özniteliğinin ve kuyruk yöneticisinin *MaxMsgLength* özniteliğinin daha küçük değerini geçemez.

- Hedef uzak bir kuyruksa ya da uzak bir kuyruğa çözümlerse, 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, *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: MQBYTExBufferUzunluğu-giriş

Bu, gönderilecek uygulama verilerini içeren bir arabelleğidir. Arabellek, iletteki verilerin niteine uygun bir sınırla hizalanmalıdır. 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: MQPUT çağrısındaki diğer tüm parametreler, yerel kuyruk yöneticisinin karakter kümesinde ve kodlamasında (*CodedCharSetId* kuyruk yöneticisi özniteliği 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 birliği-iş belirtimi.

MQRC_MULTIPLE_REASONS

(2136, X'858 ') Birden çok neden kodu döndürüldü.

MQRC_PRIORITY_EXCEEDS_MAXIMUM

(2049, X'801 ') İleti Önceliği desteklenen üst sınır değerini aşıyor.

MQRC_UNKNOWN_REPORT_OPTION

(2104, X'838 ') İleti tanımlayıcısında rapor seçenek (ler) tanınmıyor.

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_ALIAS_TARGTYPE_CHANGED

(2480, X'09B0') Abonelik hedef tipi, kuyruktan konu nesnesine çevrim yaptı.

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_BACKED_OUT

(2003, X'7D3') İş birimi yedeklendi.

MQRC_BUFFER_ERROR

(2004, X'7D4') Arabellek parametresi geçerli değil.

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 çağırısı.

MQRC_CALL_INTERRUPT

(2549, X'9F5') MQPUT ya da MQCMIT kesintiye uğradı ve yeniden bağlanma işlemi kesin bir sonucu yeniden oluşturamıyor.

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_CFGR_ERROR

(2416, X' 970 ') İleti verilerinde PCF grup değiştirgesi yapısı MQCFGR geçerli değil.

MQRC_CFH_ERROR

(2235, X'8BB') PCF üstbilgi yapısı geçerli değil.

MQRC_CFIF_ERROR

(2414, X'96E') İleti verilerinde PCF tamsayı süzgeç parametresi yapısı geçerli değil.

MQRC_CFIL_ERROR

(2236, X'8BC') PCF tamsayı listesi parametre yapısı ya da PCIF*64 tamsayı listesi parametre yapısı geçerli değil.

MQRC_CFIN_ERROR

(2237, X'8BD') PCF tamsayı değiştirgesi yapısı ya da PCIF*64 tamsayı değiştirgesi yapısı geçerli değil.

MQRC_CFSF_ERROR

(2415, X'96F') İleti verilerinde PCF dizgi süzgeci değiştirgesi yapısı geçerli değil.

MQRC_CFSL_ERROR

(2238, X'8BE') PCF dizgi listesi değiştirgesi yapısı geçerli değil.

MQRC_CFST_ERROR

(2239, X'8BF') PCF dizgi değiştirgesi yapısı geçerli değil.

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_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_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_HEADER_ERROR

(2142, X'85E') MQ üstbilgi yapısı geçerli değil.

MQRC_HOBJ_ERROR

(2019, X'7E3') Nesne tanıtıcısı geçerli değil.

MQRC_IH_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 ') Tutarlı kalıcılık belirtimi.

MQRC_INCONSISTENT_UOW

(2244, X'8C5') Tutarlı 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 kullanılabilecek 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 NPMMSGDLV 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 hiçbirisi teslim edilmezse, MQPUT, MQRC_PUBLICATION_FAILURE ile geri döner. Bu durum aşağıdaki durumlarda oluşabilir:

- Bir ileti, PMSGDLV ya da NPMMSGDLV (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 NPMMSGDLV (iletinin kalıcısına bağlı olarak) değeri ALLUR ile bir TOPIC 'e yayınlanır ve kalıcı bir abonelik, yayını alamayacak bir kuyruğa sahiptir.

Yayınların bazı abonelere aşağıdaki durumlarda teslim edilememesine rağmen, MQPUT MQRC_NONE ile geri dönebilir:

- Bir ileti, PMSGDLV ya da NPMMSGDLV (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 NPMMSGDLV (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 USEDQ konu özniteliğini kullanabilirsiniz. USEDQ 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 saklanmasıyla 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 yerleştirmek için MQPUT1 çağrısını kullanın.

Bu çağrı, MQOPER, MQPUT ve MQCLOSE çağrılarını tek bir çağrıya sarkıyor ve verilmesi gereken çağrılarının 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 hiçbirisi 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 çağrılarını, aynı nesne tanıtıcısı *Hobj* kullanılarak yapılır.

Bazı ortamlarda, çağrılar aynı uygulamadan yapılmışsa, farklı nesne tanıtıcıları kullanıldığında ileti sırası da korunur. *Aynı uygulama* 'ın anlamı, ortam tarafından belirlenir:

- z/OS üzerinde uygulama:
 - CICS için, CICS görevi
 - IMS için görev
 - z/OS toplu işi için, görev
- IBM üzerinde, uygulama işidir.
- Windows ve UNIX sistemleri üzerinde uygulama iş parçasıdır.

- 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 MQNPMS_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 göreceli kalıcı iletilerin sırası ve birbiriyle göreceli 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 şu ortamlarda desteklenir: AIX, HP-UX, IBM i, Solaris, Linux, Windows, artı IBM MQ MQI clients bu sistemlere bağlı olarak.

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_2 olarak 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 değıştirmeleri ařağıdaki gibi ayarlanır:

- Dağıtım listesindeki kuyruklara başarılı olan ya da aynı şekilde başarısız olursa, tamamlanma kodu ve neden kodu değıştirmeleri 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 değetine ayarlanır ve MQCC_FAILED değetine 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 olduğı 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 çözüldürse, 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 kullanılmaz ve kuyruk yöneticisi, bu iletim kuyruğunu kullanan her hedef için iletinin olağan biçimde bir kopyasını oluşturur.

Farklı hedeflerde farklı ileti önceliğı 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 ařağıdakiyle 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 ařağıdakilerden hangisinin gerçekleştirilmediğıne bağlıdır:

- Listedeki hedefler yerel, uzak ya da bir karışımdır.
- Hedefler aynı ileti önceliğıne ve ileti kalıcılarına sahiptir.
- İletim kuyrukları, dağıtım listesi iletilerini tutabilirler.
- İletim kuyruklarının üst sınır ileti uzunluğı, 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 iletilisinin ortaya konması), aşağıdaki durumlarda yalnızca *bir* ileti olarak sayılır:

- Uygulamanın bir iş birimindeki izin verilen ileti sayısı üst sınırını aşıp aşmadığını kontrol etmek (*MaxUncommittedMsgs* kuyruk yöneticisi özneliğine bakın).
 - Tetikleme koşullarının yerine getirilip karşılanmadığı denetleniyor.
 - Kuyruk derinliklerinin artırılması ve kuyrukların kuyruk derinliği üst sınırının aşıp 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 iletilisi 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_VARIFY, 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 iletilisinde yapıların uzunluklarının toplamı, MQPUT ya da MQPUT1 çağrısında *BufferLength* parametresiyle belirtilen uzunluğa eşit olmalıdır. Bir PCF iletilisi, 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. (Yapılar *takip edilen* ilk MQDLH kesilebilir; MQDLH ' den 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* değiştirgesi Anytipinde olduğu için, kuyrukta herhangi bir veri tipinin yerleştirilmesine olanak tanımaktadır. Ancak bu, *Buffer* 'in büyüklük olarak en az *BufferLength* byte olduğundan emin olmak için denetleyemediği anlamına gelir.

C çağırma

```
MQPUT (Hconn, Hobj, &MsgDesc, &PutMsgOpts, BufferLength, Buffer,  
       &CompCode, &Reason);
```

Değiştirgeleri 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.
```

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 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

```

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 tanırtıcı, kuyruk yöneticisiyle olan bağlantıyı gösterir. The value of *Hconn* was returned by a previous MQCONN or MQCONNEX call.

On z/OS for CICS applications, and on IBM i for applications running in compatibility mode, the MQCONN call can be omitted, and the following value specified for *Hconn* :

MQHC_DEF_HCONN

Varsayılan bağlantı tanıtıcısı.

Önemli:  Uyumluluk kipini kullanmaya devam etseniz, uygulamanızı yeniden derlemeyi ve bunun yerine statik bağlı çağrılar kullanmayı düşünün. Bkz. [IBM üzerindeki RPG uygulamaları için uyumluluk kipi](#)

ObjDesc

Tip: MQOD-giriş/çıkış

Bu yapı, iletinin eklendiği kuyruğu ya da iletinin yayınlandığı konuyu tanımlayan bir yapıdır. Ayrıntılar için bkz. [“MQOD-Nesne tanımlayıcısı” sayfa 462.](#)

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 tanımlayıcısı” sayfa 404.](#)

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* alanını MQMDE' nin var olduğunu göstermek için MQFM_MD_EXTNUF olarak ayarlayın. Daha ayrıntılı bilgi için bkz. [“MQMDE-İleti tanımlayıcı uzantısı” sayfa 453](#).

MQGMO yapısının *MsgHandle* alanında ya da MQPMO yapısının *OriginalMsgHandle* ya da *NewMsgHandle* alanlarında geçerli bir ileti tanıtıcısı sağlandıysa, uygulamanın MQMD yapısı sağlaması gerekmez. Bu alanlardan birinde hiçbir şey sağlanmıyorsa, iletinin tanımlayıcı, ileti tanıtıcılarıyla ilişkili tanımlayıcıdan alınır.

PutMsgOpts

Tip: MQPMO-giriş/çıkış

Ayrıntılar için bkz. [“MQPMO-put-message seçenekleri” sayfa 482.](#)

BufferLength

Tip: MQHOME-girişi

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 730](#).

Arabellek

Tip: MQBYTExBufferUzunluğu-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 özniteliği 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 kullanılamıyor.

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_BACKED_OUT

(2003, X'7D3') İş birimi yedeklendi.

MQRC_BUFFER_ERROR

(2004, X'7D4') Arabellek parametresi geçerli değil.

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_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 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_CFGR_ERROR

(2416, X' 970 ') İleti verilerinde PCF grup değıştirgesi yapısı MQCFGR geçerli değil.

MQRC_CFH_ERROR

(2235, X'8BB') PCF üstbilgi yapısı geçerli değil.

MQRC_CFIF_ERROR

(2414, X'96E') İleti verilerinde PCF tamsayı süzgeç parametresi yapısı geçerli değil.

MQRC_CFIL_ERROR

(2236, X'8BC') PCF tamsayı listesi parametre yapısı ya da PCIF*64 tamsayı listesi parametre yapısı geçerli değil.

MQRC_CFIN_ERROR

(2237, X'8BD') PCF tamsayı değıştirgesi yapısı ya da PCIF*64 tamsayı değıştirgesi yapısı geçerli değil.

MQRC_CFSF_ERROR

(2415, X'96F') İleti verilerinde PCF dizgi süzgeci değıştirgesi yapısı geçerli değil.

MQRC_CFSL_ERROR

(2238, X'8BE') PCF dizgi listesi değıştirgesi yapısı geçerli değil.

MQRC_CFST_ERROR

(2239, X'8BF') PCF dizgi değıştirgesi yapısı geçerli değil.

MQRC_CICS_WAIT_FAILED

(2140, X'85C') Bekleme isteğı 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 seçeneğı, XCF kuyruğı 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 edilip edilemeyeceğini belirlemek için ayrıştırılamadı.

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 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_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_AVALABILIR

(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-yönetici 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 kullanılabilecek iş birimi.

MQRC_WIH_ERROR
(2333, X'91D') MQWIH yapısı geçerli değil.

MQRC_HATA_CF_DÜZEYİ
(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 şu ortamlarda desteklenir: AIX, HP-UX, IBM i, Solaris, Linux, Windows, ayrıca bu sistemlere bağlı IBM MQ istemcileri.

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* bir tanedir:
 - 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 çağırısını 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 MQCONNEX call.

On z/OS for CICS applications, and on IBM i for applications running in compatibility mode, the MQCONN call can be omitted, and the following value specified for *Hconn* :

MQHC_DEF_HCONN

Varsayılan bağlantı tanıtıcısı.

Önemli:  Uyumluluk kipini kullanmaya devam etseniz, uygulamanızı yeniden derlemeyi ve bunun yerine statik bağlı çağrılarını kullanmayı düşünün. Bkz. [IBM üzerindeki RPG uygulamaları için uyumluluk kipi](#)

Hobj

Tip: MQHOBj-input

Bu tanıtıcı, ayarlanacak özniteliklere sahip kuyruk nesnesini gösterir. Tanıtıcı, MQOO_SET seçeneğini belirten önceki bir MQOPER çağırısı 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 geçerli olabilir.

Ayarlanabilen öznitelikler aşağıdaki çizelgede listelenir. Bu çağrıyı kullanarak başka hiçbir öznitelik ayarlanmaz. MQCA_* özniteliği seçicileri için, *CharAttrs* içinde gerekli olan dizinin bayt cinsinden uzunluğunu tanımlayan sabit parantez içinde sağlanır.

Çizelge 110. 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.	

Çizelge 110. Kuyruklar için MQSET öznitelik seçicileri (devamı var)

Seçici	Tanım	Not
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. Bu sistemlere bağlı olarak yalnızca AIX, HP-UX, IBM i, Solaris, Linux, Windows, artı IBM MQ MQI clients sistemlerinde desteklenir.

IntAttrSayı

Tip: MQHOME-girişi

Bu, *IntAttrs* dizideki öğelerin sayısıdır ve *Selectors* parametresindeki MQIA_ * seçicilerinin en az sayıda olması gerekir. Sıfır (sıfır) değeri yoksa, geçerli bir değerdir.

IntAttrs

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, *IntAttrs* 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, *CharAttrs* parametresinin bayt cinsinden uzunluğudur ve en az *Selectors* dizisinde belirtilen karakter özniteliklerinin uzunluklarının toplamını olmalıdır. *Selectors* içinde MQCA_ * seçicileri yoksa, sıfır geçerli bir değerdir.

CharAttrs

Tip: MQCHARxCharAttrLength -giriş

Bu, karakter öznitelik değerlerini içeren arabelleğidir ve birlikte bitleştirilir. Arabellek uzunluğu *CharAttrLength* parametresiyle verilir.

Karakter öznitelikleri, *Selectors* dizisindeki 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 isteği CICS tarafından 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_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 değil.

MQRC_HOBJ_ERROR

(2019, X'7E3') Nesne tanıtıcısı geçerli değil.

MQRC_INHIBIT_VALUE_ERROR

(2020, X'7E4') Inhibit-get ya da inhibit-put queue özniteliğinin değeri geçerli değil.

MQRC_INT_ATTR_COUNT_ERROR

(2021, X'7E5') Tamsayı öznitelikleri sayısı geçerli değil.

MQRC_INT_ATTRS_ARRAY_ERROR

(2023, X'7E7') Tamsayı öznitelikleri dizisi geçerli değil.

MQRC_NOT_OPEN_FOR_SET

(2040, X'7F8') Kuyruk, küme için açık değil.

MQRC_OBJECT_CHANGED

(2041, X'7F9') Nesne tanımlaması açıldığından beri değiştirildi.

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 değil ya da bilinmiyor.

MQRC_Q_MGR_NOT_VAR

(2059, X'80B') Kuyruk yöneticisi bağlantı için kullanılabılır değil.

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 değil.

MQRC_SELECTOR_ERROR

(2067, X'813 ') Öznitelik seçicisi geçerli değil.

MQRC_SELECTOR_LIMIT_EXACID

(2066, X'812 ') Seçicilerin sayısı çok büyük.

MQRC_STORAGE_NOT_AVAM

(2071, X'817 ') Kullanılabılır bellek yetersiz.

MQRC_SUPPRESSED_BY_EXIT

(2109, X'83D') Çıkış programı tarafından çağrı engellendi.

MQRC_TRIGGER_CONTROL_ERROR

(2075, X'81B') Tetikleme denetimi özniteliği için değer geçerli değil.

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 dizgileri 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 686 ' 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 811](#)
- [“Ad listelerine ilişkin öznitelikler” sayfa 842](#)
- [“Süreç tanımlamalarına ilişkin öznitelikler” sayfa 844](#)
- [“Kuyruk yöneticisine ilişkin öznitelikler” sayfa 778](#)

C çağırma

```
MQSET (Hconn, Hobj, SelectorCount, Selectors, IntAttrCount, IntAttrs,  
CharAttrLength, CharAttrs, &CompCode, &Reason);
```

Değiştirelileri 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.
```

Değiştirelileri 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 çağrısını kullanın.

Sözdizimi

MQSETMP (*Hconn*, *Hmsg*, *SetPropOpts*, *Name*, *PropDesc*, *Type*, *ValueLength*, *Value*, *Compcode*, *Reason*)

Parametreler

Hconn

Tip: MQHCONN-input

Bu tanıttıcı, 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 parçasında geçerli bir bağlantı oluşturulmalıdır; tersi durumda çağrı, MQRC_CONNECTION_BROKEN neden koduyla başarısız olur.

Hmsg

Tip: MQHMSG-giriş

Bu, değiştirilecek ileti işidir. Değer önceki bir MQCRTMH çağrısı 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 çağrısında bir giriş değiştirgişidir. Ek bilgi için [MQSMPO](#) başlıklı konuya bakın.

NAME

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 bağlamı
- 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: MQBYTEValueUzunluk-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 çağrısı.

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);
```

Değiştirelileri aşağıdaki gibi bildirin:

```

MQHCONN Hconn;      /* Connection handle */
MQHMSG Hmsg;        /* Message handle */
MQSMP0 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 CMQCHARV.
** 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 MQPD;     /* 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	CMQSMP0A	,	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 çağrısını kullanın. Döndürülen durum bilgisi tipi, çağrıda belirtilen Tip değeri tarafından belirlenir.

Sözdizimi

MQSTAT (*Hconn*, *Tip*, *Stat*, *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 MQCONNEX call.

On z/OS for CICS applications, and on IBM i for applications running in compatibility mode, the MQCONN call can be omitted, and the following value specified for *Hconn* :

MQHC_DEF_HCONN

Varsayılan bağlantı tanıttıcısı.

Önemli:  Uyumluluk kipini kullanmaya devam etseniz, uygulamanızı yeniden derlemeyi ve bunun yerine statik bağlı çağrılarını kullanmayı düşünün. Bkz. [IBM üzerindeki RPG uygulamaları için uyumluluk kipi](#)

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 571.

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_AVAILABLE

(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 zaman uyumsuz 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);
```

Değiştirgeleri 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.
```

Değiştirgeleri 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 çağrısını 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 MQCONNEX call.

On z/OS for CICS applications, and on IBM i for applications running in compatibility mode, the MQCONN call can be omitted, and the following value specified for *Hconn* :

MQHC_DEF_HCONN

Varsayılan bağlantı tanıtıcısı.

Önemli:  Uyumluluk kipini kullanmaya devam etseniz, uygulamanızı yeniden derlemeyi ve bunun yerine statik bağlı çağrılarını kullanmayı düşünün. Bkz. [IBM üzerindeki RPG uygulamaları için uyumluluk kipi](#)

SubDesc

Tip: MQSD-giriş/çıkış

Bu yapı, uygulama tarafından kaydedilmekte olan kullanıdaki nesneyi tanımlayan bir yapıdır. Ek bilgi için [“MQSD-Abonelik tanımlayıcısı”](#) sayfa 546 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 belirlendiye, 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, bu değiştirgenin çeşitli abonelik seçenekleri ile kullanımını özetler:

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.
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.

Seenekler	Hobj	Tanım
MQSO_ALTER	Geerli bir nesne tanıtıcısı	Var olan bir abonelięi, ynetilen olsun ya da olmasın, artık belirli bir kuyruęu kullansın diye deęiřtirir. MQSO_MANAGED seeneęi kullanılmadıęında, saęlanan kuyruk deęiřtirilebilir, ancak hedef sınıfı (ynetilen ya da deęil) deęiřtirilemez.

Saęlanan ya da iade edilen *Hobj* , bu abonelięe gnderilen yayın iletilerini almak isteyen sonraki MQGET ya da MQCB aęrılarında 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). Dndrlen nesne tanıtıcısı kapsamı, aęrıda belirtilen baęlantı tanıtıcısı ile aynı. Tutama kapsamına iliřkin bilgi iin Hconn (MQHCONN)-output bařlıklı konuya bakın. *Hobj* tutamacından bir MQCLOSE, *Hsub* tanıtıcısını etkilemez.

Halt

Tip: MQHOBJ-ıkıř

Bu tanıtıcı, yapılan abonelięi temsil eder. Dięer iki iřlem iin de kullanılabilir:

- Sonraki bir MQSUBRQ aęrısında, abonelik yapılırken MQSO_PUBLICATIONS_ON_REQUEST seeneęi kullanıldıęında yayınların gnderilmesini istemek iin kullanılabilir.
- Bu, yapılan bir MQCLOSE abonelięinde, yapılan abonelięi kaldırmak iin 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. Dndrlen nesne tanıtıcısı kapsamı, aęrıda belirtilen baęlantı tanıtıcısı ile aynı. *Hsub* tutamacından bir MQCLOSE, *Hobj* tanıtıcısını etkilemez.

Bu tanıtıcı bir MQGET ya da MQCB aęrısına geirilemez. *Hobj* parametresini kullanmanız gerekir. Bu tanıtıcıyı, MQCLOSE ya da MQSUBRQ dıřında herhangi bir IBM MQ aęrısı zerinde kullanamazsınız. Bu tanıtıcıyı dięer herhangi bir IBM MQ aęrısı sonularıyla geirirken MQRC_HOBJ_ERROR iinde yer alan sonular.

CompCode

Tip: MQXX_ENCODE_CASE_ONE long-output

Tamamlanma kodu; bu kod ařaęıdakilerden biridir:

MQCC_OK

Bařarıyla tamamlandı

MQCC_UYARI

Uyarı (kısmi tamamlama)

MQCC_FAILED

Arama bařarsız oldu

Neden

Tip: MQXX_ENCODE_CASE_ONE long-output

Neden kodu niteleyici *CompCode*.

CompCode MQCC_OK ise, neden kodu řu řekildir:

MQRC_NONE

(0, X'000 ') Raporlamak iin bir neden yok.

CompCode MQCC_FAILED ise, neden kodu ařaęıdakilerden biri olabilir:

MQRC_CLUSTER_RESOLUTION_ERROR

(2189, X'88D') Kme adının zlmesi bařarsız oldu.

MQRC_DURABILITY_NOT_ALLOWALIZE

2436 (X'0984 ') MQSO_DAYİD seçeneğini kullanan bir MQSUB çağırısı 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 çağırısı 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.

MQRC_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.

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 kısımin bitiştirilmesiyle oluşturulmuş bir konuya eklenir. “MQSD-Abonelik tanımlayıcısı” sayfa 546’deki *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_PROMESD seçeneği kullanılmıyorsa, çıkış için yetki denetimi yapıp 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.

MQSUB ' dan MQSD çıkışı	
MQSD ' de alan adı	Özel noktalar
Erişim ya da yaratma seçenekleri	Bazı seçenekler MQSUB çağrısından döndürülebilecek şekilde ilk durumuna 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.

MQSUB ' dan MQSD çıkışı (devamı var)	
MQSD ' de alan adı	Özel noktalar
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.
SubUserVerileri ve SelectionString	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 ' ın <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. 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.

MQSUB ' dan MQSD çıkışı (devamı var)	
MQSD ' de alan adı	Özel noktalar
SubCorrelTanıtıcısı ve PubAccountingSimgesi	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. 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.
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 çağrısı

```
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 yayınlı ilgili bir istek yapmak için MQSUBRQ çağrısını kullanın.

Sözdizimi

MQSUBRQ (*Hconn*, *Hsub*, *Action*, *SubRqOpts*, *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 MQCONNEX call.

On z/OS for CICS applications, and on IBM i for applications running in compatibility mode, the MQCONN call can be omitted, and the following value specified for *Hconn* :

MQHC_DEF_HCONN

Varsayılan bağlantı tanıttıcısı.

Önemli:  Uyumluluk kipini kullanmaya devam etseniz, uygulamanızı yeniden derlemeyi ve bunun yerine statik bağlı çağrılarını kullanmayı düşünün. Bkz. [IBM üzerindeki RPG uygulamaları için uyumluluk kipi](#)

Halt

Tip: MQHOBJ-input

Bu tanıttıcı, bir güncellemenin isteneceği aboneliği temsil eder. *Hsub* değeri önceki bir MQSUB çağrısından döndürüldü.

Action

Tip: MQHOME-girişi

Bu değiştirge, 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 işlemini denetler, ayrıntılar için [“MQSRO-Abonelik isteği seçenekleri”](#) sayfa 569 ' 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_SUPPORT

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 */
MQHOBJS 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 CMQSR0A , 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

Bazı kuyruk yöneticisi öznitelikleri belirli somutlamalar için sabitlenir; diğer kullanıcılar da ALTER QMGRMQSC komutu kullanılarak değiştirilebilir.

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. Öznitelikleri tanımlamak, değiştirmek ya da görüntülemek için MQSC komutları kullanıldığında, diğer kısa adlar kullanılır; ek bilgi için [Script \(MQSC\) Commands](#) (Komut Dosyası (MQSC) Komutları) konusuna bakın.

Çizelge 111. Kuyruk yöneticisine ilişkin öznitelikler	
Öznitelik	Tanım
AccountingConnGeçersiz Kılma	Hesap ayarlarını geçersiz kıl.
AccountingInterval	Ara muhasebe kayıtlarının ne sıklıkta yazılabileceği.
ActivityConnGeçersiz Kılma	Etkinlik ayarlarını geçersiz kıl.
ActivityTrace	IBM MQ MQI uygulaması etkinlik izleme derlemini denetler.
AdoptNewMCACheck	Yeni MCA ' yı kabul edip etmeyeceğini belirlemek için kullanılan öğeler imlendi.
AdoptNewMCATYPE	Belirli bir kanal tipindeki bir MCA ' nın (MCA) otomatik olarak başlatılıp başlatılmayacağı.
AlterationDate	Tanımlamanın son olarak değiştirildiği tarih
AlterationTime	Tanımlamanın son olarak değiştirildiği saat
AuthorityEvent	Yetkilendirme (Yetkili Değil) olaylarının oluşturulup oluşturulmayacağını denetler
BridgeEvent	Köprü olaylarına ilişkin denetim özniteliği.
ChannelAutoDef	Otomatik kanal tanımlamasının kullanılmasına izin verilip verilmediğini denetler
ChannelAutoDefEvent	Kanal otomatik tanımlama olaylarının oluşturulup oluşturulmayacağını denetler
ChannelAutoDefExit	Otomatik kanal tanımlaması için kullanıcı çıkışının adı

Çizelge 111. Kuyruk yöneticisine ilişkin öznitelikler (devamı var)

Öznitelik	Tanım
ChannelEvent	Kanal olaylarına ilişkin denetim özniteliği.
ChannelInitiatorDenetimi	Kanal başlatıcısı için denetim özniteliği
ChannelMonitoring	Kanallar için çevrimiçi izleme verileri
ChannelStatistics	Kanallar için istatistik verilerinin toplanmasını denetler.
ChinitAdapters	IBM MQ çağrılarının işlenmesine ilişkin bağdaştırıcı alt görevi sayısı.
ChinitDispatchers	Kanal başlatıcısı için kullanılacak dağıtıcıların sayısı.
	IBM kullanımı için ayrılmıştır.
ChinitTraceAutoStart	Kanal başlatıcı izlemesinin otomatik olarak başlatılıp başlatılmayacağını.
ChinitTraceTableSize	Kanal başlatıcısının izleme verileri alanı büyüklüğü.
ClusterSenderMonitoringDefault	Küme gönderen kanalları için çevrimiçi izleme verileri varsayılanı
ClusterSenderİstatistikleri	Küme gönderen kanallarına ilişkin istatistik izleme bilgileri derlemine denetler.
ClusterWorkloadVerileri	Küme iş yükü çıkışa ilişkin kullanıcı verileri
ClusterWorkloadÇıkış	Küme iş yükü yönetimi için kullanıcı çıkışının adı
ClusterWorkloadUzunluğu	Küme iş yükü çıkışa aktarılan ileti verisi uzunluğu üst sınırı
CLWLMRUChannels	Küme iş yükü dengelemesi için en son kullanılan kanalların sayısı
CLWLUseQ	Küme iş yükü, uzak kuyruğu kullanır.
CodedCharSetId	Kodlanmış karakter takımı tanıtıcısı
CommandEvent	Komut olaylarına ilişkin denetim özniteliği.
CommandInputQName özniteliği	Komut giriş kuyruğu adı
CommandLevel	Komut düzeyi
CommandServerControl özniteliği	Komut sunucusuna ilişkin denetim özniteliği.
Yapılandırma Olayı özniteliği	Yapılandırma olaylarına ilişkin denetim özniteliği.
DeadLetterQName	Ölü-mektup kuyruğunun adı
DEFCLXQ	Varsayılan küme iletim kuyruğu tipi
DefXmitQName	Varsayılan iletim kuyruğu adı
DistLists	Dağıtım listesi desteği
DNSGroup	İş Yüğü Yöneticisi Dinamik Etki Alanı Adı Hizmetleri desteği kullanılırken TCP dinleyicisine ilişkin grubun adı.
DNSWLM	TCP dinleyicisinin, Dinamik Etki Alanı Adı Hizmetleri için İş Yüğü Yöneticisi ile birlikte kaydolup olmadığı.
ExpiryInterval	Süresi dolan iletiler için taramalar arasındaki aralık
IGQPutAuthority	Grup içi kuyruğa alma yetkisi
IGQUserId	Grup içi kuyruğa alma kullanıcı kimliği
InhibitEvent	Engelleyici (Inhibit Get ve Inhibit Sput) olaylarının oluşturulup oluşturulmayacağını denetler
IPAddressVersion	Internet Protocol adresinin sürümü
IntraGroupqueuing	Grup içi kuyruğa alma desteği
ListenerTimer	APPC ya da TCP/IP hatasından sonra dinleyiciyi yeniden başlatma girişimleri arasındaki zaman aralığı.
LocalEvent	Yerel hata olaylarının oluşturulup oluşturulmadığını denetler
LoggerEvent	Günlüğe kaydedici olaylarının oluşturulup oluşturulmayacağını denetler
LUGroupName	Kuyruk paylaşım grubu için gelen iletimleri işleyen LU 6.2 dinleyicisi için genel LU adı.
LUAdı	Giden LU 6.2 iletimleri için kullanılacak LU adı.
LU62ARMSuffix	SYS1.PARMLIB üyesi APPCPMxx, bu kanal başlatıcısı için LUADD ' yi aday göstermektedir.
LU62Channels	Maximum number of current channels or connected clients that use LU 6.2.
MaxActiveKanalları	Herhangi bir zamanda etkin olabilecek kanal sayısı üst sınırı.
MaxChannels	Geçerli kanal sayısı üst sınırı.

Çizelge 111. Kuyruk yöneticisine ilişkin öznelilikler (devamı var)

Öznelilik	Tanım
MaxHandles	Tanıtıcı sayısı üst sınırı
MaxMsgUzunluğu	İleti uzunluğu üst sınırı (byte)
MaxPriority özneliliği	En yüksek öncelik
MaxPropertiesUzunluğu	Özellik verileri uzunluğu üst sınırı (byte)
MaxUncommittedMsgs	Bir iş birimi içinde kesinleştirilmemiş ileti sayısı üst sınırı
MQIAccounting	MQI verilerine ilişkin muhasebe bilgilerinin toplanmasını denetler.
MQIStatistics	Kuyruk yöneticisine ilişkin istatistik izleme bilgilerinin derlemi denetler.
MsgMarkBrowseInterval	Kuyruk yöneticisinin işaretlenen iletilerden işareti kaldırabileceği aralık.
OutboundPortAlt Sınır	<i>OutboundPortMinile</i> , giden kanalları bağlarken kullanılacak kapı numaralarının aralığını tanımlar.
OutboundPortAlt Sınır	<i>OutboundPortMaxile</i> , giden kanalları bağlarken kullanılacak kapı numaralarının aralığını tanımlar.
PerformanceEvent	Başarımla ilgili olayların oluşturulup oluşturulmayacağını denetler
Altyapı	Kuyruk yöneticisinin çalışmakta olduğu altyapı
PubSubNPInputMsg	Teslim edilmemiş bir giriş iletilerinin atılıp atılmayacağını (ya da alıkoymak)
PubSubNPresponse	Teslim edilmemesinin davranışını denetler
PubSubMaxMsgRetryCount	Başarısız bir komut iletili işlenirken (syncpoint altında) işlem sırasında yeniden deneme sayısı
PubSubSyncPoint	Yalnızca kalıcı (ya da tüm) iletilerin uyumluluk noktası altında işlenip işlenmeyeceğini
PubSubKipi	Kuyruğa alınan yayınlama/abone olma arabiriminin çalıştırılıp çalıştırılmayacağı
QMGrDesc	Kuyruk yöneticisi tanımı
QMGrIdentifier	Kuyruk yöneticisinin iç sistem tarafından üretilen benzersiz tanıtıcısı
QMGrName	Kuyruk yöneticisi adı
QSGName	Kuyruk paylaşım grubunun adı
QueueAccounting	Kuyruklara ilişkin muhasebe bilgilerinin toplanmasını denetler.
QueueMonitoring	Kuyruklar için çevrimiçi izleme verileri
QueueStatistics	Kuyruklar için istatistik verilerinin toplanmasını denetler.
ReceiveTimeout	TCP/IP kanalı, etkin olmayan duruma dönmeden önce verilerin ne kadar süreyle bekleyeceğini belirler.
ReceiveTimeoutAlt Sınır	<i>ReceiveTimeoutniteleyicisi</i> .
ReceiveTimeoutTipi	TCP/IP kanalı etkin olmayan duruma dönmeden önce veri için bekleme süresi alt sınırı.
RemoteEvent	Uzak hata olaylarının oluşturulup oluşturulmadığını denetler
RepositoryName	Bu kuyruk yöneticisinin havuz hizmetleri sağladığı kümenin adı
RepositoryNamelist	Bu kuyruk yöneticisinin havuz hizmetleri sağladığı kümelerin adlarını içeren ad listesi nesnesinin adı
ScyCase	Güvenlik profilleri vakası
SharedQMGrAd	Paylaşılan kuyruk kuyruğu-yönetici adı
“KYR” sayfa 807	IBM MQ Advanced Message Security protection for a queue Manager açık ya da kapalı.
SSLCRLNamelist 1	Kimlik doğrulama bilgileri nesnelerinin adlarını içeren ad listesi nesnesinin adı.
SSLCryptoHardware 1	Şifreleme donanımı yapılandırma dizesi.
SSLEC	SSL olaylarına ilişkin denetim özneliliği.
SSLFIPSTRequired	Kriptografi için yalnızca FIPS onaylı algoritmalar kullanın.
SSLKeyRepository 1	SSL anahtar havuzunun yeri.
SSLKeyResetSayı	SSL anahtarı sıfırlama sayısı.
SSLTasks 1	SSL çağrılarının işlenmesine ilişkin sunucu alt görevlerinin sayısı.
StatisticsInterval	İstatistik izleme verileri yazma sıklığı.
StartStopOlayı	Başlatma ve durdurma olaylarının oluşturulup oluşturulmayacağını denetler
SyncPoint	Eşitleme noktası kullanılabilirliği

Çizelge 111. Kuyruk yöneticisine ilişkin öznitelikler (devamı var)

Öznitelik	Tanım
TCPKanal	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 özniteliği	İzleme rotası bilgilerinin kaydedilmesini denetler.
TriggerInterval	Tetikleme-ileti aralığı
S\u00f0fcr\u00f0fcm	S\u00f0fcr\u00f0fcm
XrCapability	Telemetry komutlarının desteklenip desteklenmediğini belirtir.
Notlar: 1. Bu öznitelik, MQINQ çağrısını kullanarak sorgulanamaz ve bu bölümde anlatılmaz. Bu özniteliğe ilişkin ayrıntılar için Kuyruk Yöneticisinin Değiştirilmesi başlıklı konuya bakın.	

İlgili bilgiler

MQI istemcisinde çalıştırma sırasında yalnızca FIPS onaylı CipherSpecs ' in kullanıldığını belirtme [Federal Information Processing Standards \(FIPS\) for UNIX, Linux and Windows](#)

AccountingConnOverride (MQUZE)

Bu, uygulamaların, Qmgr öznitelideki ACCTMQI ve ACCTQDATA değerlerinin ayarını geçersiz kılmasına olanak sağlar.

Değer aşağıdakilerden biridir:

MQMON_DEVRE Dışı

Uygulamalar, MQCONNX çağrısındaki MQCNO yapısındaki Seçenekler alanını kullanarak, ACCTMQI ve ACCTQ Qmgr özniteliklerinin ayarını geçersiz kılamaz. Bu varsayılan değerdir.

MQMON_ENABLED

Uygulamalar, MQCNO yapısındaki Options alanını kullanarak, ACCTQ ve ACCTMQI Qmgr özniteliklerini geçersiz kılabilir.

Bu değerdeki değişiklikler yalnızca, öznitelik üzerindeki değişiklikten sonra kuyruk yöneticisine yönelik bağlantılar için geçerlidir.

Bu öznitelik yalnızca IBM i, Unix sistemlerinde ve Windowssistemlerinde desteklenir.

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 IBM i, Windows, UNIXve Linux sistemlerinde desteklenir.

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 Dışı

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 IBM i, Unix sistemlerinde ve Windowssistemlerinde 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 IBM i, Unix sistemlerinde ve Windowssistemlerinde desteklenir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA_ACTIVITY_TRACE seçiciyi 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/OSüzerinde 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 özniteliğin 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 öznitelik yalnızca z/OS üzerinde desteklenir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA_BRIDGE_EVENT seçiciyi kullanın.

ChannelAutoDef (MQUZE)

Bu öznitelik, 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 öznitelik yalnızca AIX, HP-UX, IBM i, Linux, Solaris ve Pencereleüzerinde desteklenir.

Bu özniteliğın deęerini saptamak için, MQINQ çağırısıyla 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. 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 öznitelik yalnızca AIX, HP-UX, IBM i, Linux, Solaris ve Pencereleüzerinde desteklenir.

Bu özniteliğın deęerini saptamak için, MQINQ çağırısıyla MQIA_CHANNEL_AUTO_DEF_EVENT seçiciyi kullanın.

ChannelAutoDefExit (MQCHARn)

Bu, otomatik kanal tanımlamasına ilişkin kullanıcı çıkışıdır. Bu ad boş değilse ve *ChannelAutoDef* MQCHAD_ENABLED deęeri varsa, kuyruk yöneticisinin kanal tanımlaması yaratmak üzere olduęu her saat çıkışı çağırılır. Bu, MQCHT_RECEIVER, MQCHT_SVRCONN ve MQCHT_CLUSSDR tipi kanallara uygulanır. Bundan sonra, çıkış aşağıdakilerden birini yapabilir:

- Kanal tanımlamasını deęiřmeden yaratın.
- Oluřturulan kanal tanımlamasının özniteliklerini deęiřtirin.
- Kanalı oluşturulmasını tamamen engelle.

Not: Bu özniteliğın hem uzunluk hem de deęeri, ortama özgü bir deęerdir. Çeřitli ortamlardaki bu özniteliğın deęerinin ayrıntıları için, “MQCD-Kanal tanımlaması” sayfa 1496 içindeki MQCD yapısına giriş yapın.

Bu öznitelik yalnızca AIX, HP-UX, IBM i, Linux, Solaris, Windowsve z/OSüzerinde desteklenir. z/OSüzerinde, yalnızca küme-gönderici ve küme alıcı kanalları için geçerlidir.

Bu özniteliğın deęerini saptamak için, MQINQ çağırısıyla MQCA_CHANNEL_AUTO_DEF_EXIT seçiciyi kullanın. Bu özniteliğın uzunluęu, MQ_EXIT_NAME_LENGTH tarafından verilir.

ChannelEvent (MQUZE)

Bu, kanal olaylarının oluşturulup oluşturulmayacağını belirtir.

Bu deęer aşağıdaki deęerlerden biridir:

MQEVR_EXCEPTION

Yalnızca aşağıdaki kanal olaylarını oluřturun:

- MQRC_CHANNEL_ACTIVATED
- MQRC_CHANNEL_CONV_ERROR
- MQRC_CHANNEL_NOT_ETKINED
- MQRC_CHANNEL_DURDURDU, řu ReasonQualifiers:
 - MQRQ_CHANNEL_STOPPED_ERROR
 - MQRQ_CHANNEL_STOPPED_RETRY
 - MQRQ_CHANNEL_STOPPED_DISABET
 - MQRC_CHANNEL_STOPPED_BY_USER

MQEVRENABLED

Tüm kanal olaylarını oluştur. Kural Dışı Durum (EXCEPTION) tarafından oluşturulanların yanı sıra, aşağıdaki kanal olaylarını oluşturun:

- MQRC_CHANNEL_STARTED
- MQRC_CHANNEL_DURG, şu ReasonQualifier ile durduruldu:

MQRC_CHANNEL_STOPPED_OK

MQEVDR_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_ORTAMI

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' ta bu parametre, seçtiğiniz değer ne olursa olsun, istatistik verileri derlemine açar. LOW, MEDIUM ya da HIGH belirtildiğinde, sonuçlarınız için fark yoktur.

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 derlemi kapatın.

MQMON_LOW

İstatistik verileri derlemi, STATCHL kanal öznitelideki QMGR belirten kanallar için düşük bir veri toplama oranına sahip olarak döndürün.

MQMON_ORTAMı

İstatistik verileri derlemi, STATCHL kanal öznitelideki QMGR belirten kanallar için ılımlı bir veri toplama oranına sahip olarak döndürün.

MQMON_YÜKSEK

İstatistik verileri derlemi, 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 (Yükün) seçeneğini belirlemek isteyebilirsiniz.

z/OS' ta bu parametre, seçtiğiniz değer ne olursa olsun, istatistik verileri derlemi açar. LOW, MEDIUM ya da HIGH belirtildiğinde, sonuçlarınız için fark yoktur.

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 özneliğin değeri 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 özneliğin değeri saptamak için, MQINQ çağrısıyla MQIA_CHINIT_TRACE_TABLE_SIZE seçiciyi kullanın.

ClusterSenderMonitoringDefault (MQLONG)

Bu, otomatik olarak tanımlanmış küme gönderen kanallarının ChannelMonitoring özneliği için yerine konabilecek değeri belirtir.

Değer aşağıdakilerden biridir:

MQMON_Q_MGR

Çevrimiçi izleme verilerinin toplanması, kuyruk yöneticisi *ChannelMonitoring* özneliğinin ayarından devralınır. Bu varsayılan değerdir.

MQMON_OFF

Kanala ilişkin izleme kapalı

MQMON_LOW

ChannelMonitoring MQMON_NONE ise, izleme, düşük oranda veri toplama işlemi uygulanmıyorsa, sistem başarımıyla ilgili en düşük düzeyde bir etki sağlar. Toplanan veriler büyük olasılıkla en güncel veri kaynağı değil.

MQMON_ORTAMı

ChannelMonitoring MQMON_NONE ise, izleme, sistem başarımıyla ilgili sınırlı etkiyle, orta düzeyde veri toplama işlemi ile açık olmalıdır.

MQMON_YÜKSEK

ChannelMonitoring MQMON_NONE ise, izleme, yüksek oranda veri toplama işlemi uygulanmıyorsa, sistem başarımı üzerinde büyük bir etkisi olur. Toplanan veriler, en güncel kullanılabilir verilerdir.

Bu özneliğin değeri 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 derlemi kapatın.

MQMON_LOW

Veri toplama oranı düşük olan otomatik tanımlı küme gönderen kanallarına ilişkin istatistik verileri derlemine geçin.

MQMON_ORTAMI

Otomatik olarak tanımlanmış küme gönderen kanallarına ilişkin istatistik verileri derlemine, veri toplama oranı ılımlı bir oran ile geçiş yapabilirsiniz.

MQMON_YÜKSEK

Veri toplama oranı yüksek olan otomatik tanımlı küme gönderen kanallarına ilişkin istatistik verileri derlemine geçin.

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' ta bu parametre, seçtiğiniz değer ne olursa olsun, istatistik verileri derlemine açar. LOW, MEDIUM ya da HIGH belirtildiğinde, sonuçlarınız için fark yoktur.

Bu öznenin değerini saptamak için, MQINQ çağrısıyla MQIA_STATISTICS_AUTO_CLUSTER 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 öznenin yalnızca AIX, HP-UX, IBM i, Linux, Solaris, Windows ve z/OS üzerinde desteklenir.

Bu öznenin değerini saptamak için, MQINQ çağrısıyla MQCA_CLUSTER_WORKLOAD_DATA seçiciyi 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 öznenin hem uzunluk hem de değeri, ortama özgü bir değerdir.

Bu öznenin yalnızca AIX, HP-UX, IBM i, Linux, Solaris, Windows ve z/OS üzerinde desteklenir.

Bu öznenin değerini saptamak için, MQINQ çağrısıyla MQCA_CLUSTER_WORKLOAD_EXIT seçiciyi kullanın. Bu öznenin 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:

- İletin uzunluğu.
- Kuyruk yöneticisinin *MaxMsgLength* öznenin.
- *ClusterWorkloadLength* öznenin.

Bu öznenin yalnızca AIX, HP-UX, IBM i, Linux, Solaris, Windows ve z/OS üzerinde desteklenir.

Bu öznenin 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/OSüzerinde, kuyruk yöneticisi başlatıldığında değer sistem değiştirtgelerinden ayarlanır; varsayılan değer 500 'dür.
- Windowsüzerinde değer, kuyruk yöneticisini yaratan kullanıcının birincil CODEPAGE değeridir.
- IBM üzerinde, kuyruk yöneticisi ilk yaratıldığında, bu değer, ortamdaki ayarlandır.
- UNIX sistemleri' 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)

Bu, kuyruk yöneticisi tarafından desteklenen sistem denetimi komutlarının düzeyini gösterir. Bu, aşağıdaki değerlerden biri olabilir:

MQCMDL_LEVEL_600

Sistem denetimi komutlarının 600. düzeyi.

Bu değer aşağıdaki sürümler tarafından döndürülür:

- AIX Sürüm 6.0 için IBM WebSphere MQ
- HP-UX Sürüm 6.0 için IBM WebSphere MQ
- iSeries Sürüm 6.0 için IBM WebSphere MQ
- Linux Sürüm 6.0 için IBM WebSphere MQ
- Solaris Sürüm 6.0 için IBM WebSphere MQ
- Windows Sürüm 6.0 için IBM WebSphere MQ
- z/OS Sürüm 6.0 için IBM WebSphere MQ

MQCMDL_LEVEL_700

Sistem denetimi komutlarının 700. düzeyi.

Bu değer aşağıdaki sürümler tarafından döndürülür:

- AIX Sürüm 7.0 için IBM WebSphere MQ
- HP-UX Sürüm 7.0 için IBM WebSphere MQ
- IBM i Sürüm 7.0 için IBM WebSphere MQ
- Linux Sürüm 7.0 için IBM WebSphere MQ
- Solaris Sürüm 7.0 için IBM WebSphere MQ
- Windows Sürüm 7.0 için IBM WebSphere MQ
- z/OS Sürüm 7.0 için IBM WebSphere MQ

MQCMDL_LEVEL_701

Sistem denetimi komutlarının 701. düzeyi.

Bu değer aşağıdaki sürümler tarafından döndürülür:

- AIX Sürüm 7.0.1 için IBM WebSphere MQ
- HP-UX Sürüm 7.0.1 için IBM WebSphere MQ
- IBM i Sürüm 7.0.1 için IBM WebSphere MQ
- Linux Sürüm 7.0.1 için IBM WebSphere MQ
- Solaris Sürüm 7.0.1 için IBM WebSphere MQ
- Windows Sürüm 7.0.1 için IBM WebSphere MQ
- z/OS Sürüm 7.0.1 için IBM WebSphere MQ

MQCMDL_LEVEL_710

Sistem denetimi komutlarının 710 düzeyi.

Bu değer aşağıdaki sürümler tarafından döndürülür:

- AIX Sürüm 7.1 için IBM WebSphere MQ
- HP-UX Sürüm 7.1 için IBM WebSphere MQ
- IBM i Sürüm 7.1 için IBM WebSphere MQ
- Linux Sürüm 7.1 için IBM WebSphere MQ
- Solaris Sürüm 7.1 için IBM WebSphere MQ
- Windows Sürüm 7.1 için IBM WebSphere MQ
- z/OS Sürüm 7.1 için IBM WebSphere MQ

MQCMDL_LEVEL_750

Sistem denetimi komutlarının 750. düzeyi.

Bu değer aşağıdaki sürümler tarafından döndürülür:

- AIX Sürüm 7.5 için IBM WebSphere MQ
- HP-UX Sürüm 7.5 için IBM WebSphere MQ
- IBM i Sürüm 7.5 için IBM WebSphere MQ
- Linux Sürüm 7.5 için IBM WebSphere MQ
- Solaris Sürüm 7.5 için IBM WebSphere MQ
- Windows Sürüm 7.5 için IBM WebSphere MQ

MQCMDL_LEVEL_800

Sistem denetimi komutlarının 800. düzeyi.

Bu değer aşağıdaki sürümler tarafından döndürülür:

- AIX Sürüm 8.0 için IBM WebSphere MQ
- HP-UX Sürüm 8.0 için IBM WebSphere MQ
- IBM i Sürüm 8.0 için IBM WebSphere MQ
- Linux Sürüm 8.0 için IBM WebSphere MQ
- Solaris Sürüm 8.0 için IBM WebSphere MQ
- Windows Sürüm 8.0 için IBM WebSphere MQ
- z/OS Sürüm 8.0 için IBM WebSphere MQ

V 8.0.0.2 MQCMDL_LEVEL_801

Sistem denetimi komutlarının 801 düzeyi.

Bu değer aşağıdaki sürümler tarafından döndürülür:

- IBM MQ for AIX Sürüm 8.0.0.2
- IBM MQ for HP-UX Sürüm 8.0.0.2
- IBM MQ for IBM i Sürüm 8.0.0.2
- IBM MQ for Linux Sürüm 8.0.0.2
- IBM MQ for Solaris Sürüm 8.0.0.2

V 8.0.0.3 MQCMDL_LEVEL_802

Sistem denetimi komutlarının 802 düzeyi.

Bu değer aşağıdaki sürümler tarafından döndürülür:

- IBM MQ for AIX Sürüm 8.0.0.3
- IBM MQ for HP-UX Sürüm 8.0.0.3
- IBM MQ for IBM i Sürüm 8.0.0.3
- Linux Sürüm 8.0 için IBM MQ .0.3
- IBM MQ for Solaris Sürüm 8.0.0.3
- IBM MQ for Windows Sürüm 8.0.0.3

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/OSüzerinde desteklenmez.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQIA_CMD_SERVER_CONTROL seçiciyi kullanın.

ConfigurationEvent (MQUWN)

Yapılandırma olaylarının oluşturulup oluşturulmadığını denetler.

Bu özneliğin 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.

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 iletisi hedef kuyruğuna teslim edilemiyorsa (genellikle özgün iletinin ileti tanımlayıcısındaki *ReplyToQ* alanı tarafından belirlenen kuyruk), rapor iletisi 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 iletisi (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 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. [“MQDLH-Dead-letter üstbilgisi” sayfa 341](#) .

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.

DefClusterXmitQueueTipi (MQUBE)

DefClusterXmitQueueType özniteliği, ileti almak için varsayılan olarak küme gönderen kanalları tarafından hangi iletim kuyruğunun seçildiğini, iletileri küme alıcılı kanallara göndermesini sağlar.

DefClusterXmitQueueType değeri, MQCLXQ_SCTQ ya da MQCLXQ_CHANNEL' dir.

MQCLXQ_SCTQ

All cluster-sender channels send messages from SYSTEM.CLUSTER.TRANSMIT.QUEUE. İletim kuyruğuna yerleştirilen iletilerin correlID , iletinin hangi küme gönderici kanalına gönderildiğini belirler.

SCTQ is set when a queue manager is defined. Bu davranış, Sürüm 7.5sürümünden önceki IBM WebSphere MQsürümlerinde örtük bir davranışa neden olur. Önceki sürümlerde, kuyruk yöneticisi özniteliği DefClusterXmitQueueTürü mevcut değildi.

MQCLXQ_CHANNEL

Her bir küme gönderici kanalı, iletileri farklı bir iletim kuyruğundan gönderir. Each transmission queue is created as a permanent dynamic queue from the model queue SYSTEM.CLUSTER.TRANSMIT.MODEL.QUEUE.

If the queue manager attribute, DefClusterXmitQueueTürü, is set to KANAL, varsayılan yapılandırma, tek tek küme iletim kuyruklarıyla ilişkilendirilmekte olan küme gönderici kanallarına çevrilir. The transmission queues are permanent-dynamic queues created from the model queue SYSTEM.CLUSTER.TRANSMIT.MODEL.QUEUE. Her iletim kuyruğu bir küme gönderici kanalı ile ilişkilendirilir. Bir küme gönderici kanal hizmetleri bir küme iletim kuyruğu olarak, iletim kuyruğu tek bir kümede yalnızca tek bir kuyruk yöneticisine ilişkin iletileri içerir. Kümeleri yapılandırabilir; böylece, bir kümedeki her bir kuyruk yöneticisinin tek bir küme kuyruğu içermesi gerekir. Bu durumda, kuyruk yöneticisinden her küme kuyruğuna gelen ileti trafiği, iletilerden diğer kuyruklara ayrı ayrı 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

“MQINQ-Nesne özniteliklerinin sorgulması” sayfa 686

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

İlgili bilgiler

[Kuyruk Yöneticisini Değiştir](#)

[Sorgu Kuyruğu Yöneticisi](#)

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_DESTEKLLENEN

Dağıtım listeleri desteklenir.

MQDL_NOT_SUPPORTANT

Dağıtım listeleri desteklenmiyor.

Bu özneliğin değerini saptamak için, MQINQ çağrısına sahip MQIA_DIST_LISTS seçiciyi kullanın.

DNSGroup (MQCHAR18)

Bu parametre artık kullanılmıyor. [WLM/DNS artık desteklenmiyor](#) konusuna bakın.

Bu öznelik yalnızca z/OS üzerinde desteklenir.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQCA_DNS_GROUP seçiciyi kullanın. Bu özneliğin uzunluğu MQ_DNS_GROUP_NAME_LENGTH tarafından verilir.

DNSWLM (MQUZA)

Bu parametre artık kullanılmıyor. [WLM/DNS artık desteklenmiyor](#) konusuna bakın.

Değer aşağıdakilerden biridir:

MQDNSWLM_YES

Bu değer, daha önceki bir yayından geçirilen bir kuyruk yöneticisinden görülebilir. Değer yoksayılır.

MQDNSWLM_NO

Bu, kuyruk yöneticisi tarafından desteklenen tek değerdir.

Bu öznelik yalnızca z/OS üzerinde desteklenir.

Bu özneliğin 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 özneliğin değerini saptamak için, MQINQ çağrısıyla MQIA_EXPIRY_INTERVAL seçiciyi kullanın.

Bu öznelik yalnızca z/OS üzerinde desteklenir.

IGQPutAuthority (MQHOT)

Bu öznelik 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 özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA_IGQ_PUT_AUTHORITY seçicisini kullanın.

Bu öznitelik yalnızca z/OSüzerinde desteklenir.

IGQUserId (MQUZN)

Bu öznitelik, yerel kuyruk yöneticisi bir kuyruk paylaşım grubunun bir üyesiyse geçerlidir. Yerel grup içi kuyruğa alma aracısıyla (IGQ aracısı) ilişkilendirilmiş olan kullanıcı kimliğini belirtir. Bu tanıtıcı, IGQ aracısı yerel kuyruklara ileti yerleştirdiğinde yetki için denetlenebilecek kullanıcı tanıtıcılarından biridir. Denetlenen gerçek kullanıcı tanıtıcıları, *IGQPutAuthority* özniteliğinin 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 özniteliğin değerini saptamak için, MQINQ çağrısıyla MQCA_IGQ_USER_ID seçiciyi kullanın. Bu özniteliğin uzunluğu MQ_USER_ID_LENGTH tarafından verilir.

Bu öznitelik yalnızca z/OSüzerinde 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 özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA_INHIBIT_EVENT seçiciyi kullanın.

z/OS' ta, bu özniteliğin değerini saptamak için MQINQ çağrısını kullanamazsınız.

IntraGroupqueuing (MQUZE)

Bu öznitelik yalnızca, yerel kuyruk yöneticisi bir kuyruk paylaşım grubunun bir üyesiyse geçerlidir. Bu, 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 değişmezi 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 görelî olarak korunmaz.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA_INTRA_GROUP_queueing seçiciyi kullanın.

Bu öznitelik yalnızca z/OS üzerinde 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 özniteliğin 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 özniteliğin 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 özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA_LOCAL_EVENT seçiciyi kullanın.

z/OS' ta, bu özniteliğin 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 özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA_LOGGER_EVENT seçiciyi kullanın.

Bu öznitelik yalnızca AIX, HP-UX, IBM i, Linux, Solaris ve Pencereleler üzerinde desteklenir.

LUGroupName (MQCHAR8)

Bu, 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 'nün adıdır. Bunu dinleyici tarafından gelen iletimler için kullandığı LU 'ya ayarlayın. Bu adı boş bırakırsanız, APPC/MVS varsayılan LU kullanılır; bu bir değişikliktir; 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 sınısıdır.

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 sınısıdır.

Varsayılan değer, MaxChannels özniteliği için belirtilen değerdir.

z/OS için, değer 1-9 999 aralığında olmalıdır.

Diğer tüm platformlar için, değer 1 ile 65 535 aralığında olmalıdır.

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 bilgiler

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.

Diğer tüm platformlar için değer, 1 ile 65 535 arasında bir değer olmalıdır, varsayılan değer 100 olmalıdır.

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.

MaxChannels parametresi, yalnızca z/OS üzerinde bir kuyruk yöneticisi öznesidir. Diğer platformlarda **MaxChannels**, qm.ini dosyasındaki bir öznedir. Diğer platformlarda **MaxChannels** özneliğini ayarlamaya ilişkin bilgi için bkz. [Distributed queuing için yapılandırma dosyası stanzas for distributed queuing](#).

Bu özneliğin değerini saptamak için, **MQINQ** çağrısıyla MQIA_MAX_CHANGES seçiciyi kullanın.

İlgili bilgiler

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 özneliğin değerini saptamak için, MQINQ çağrısıyla MQIA_MAX_HANDHANS seçiciyi kullanın.

MaxMsgUzunluğu (MQUZE)

Bu, kuyruk yöneticisinin işleyebileceği en uzun *fiziksel* iletinin uzunluğudur. Ancak, *MaxMsgLength* kuyruk yöneticisi özneliği *MaxMsgLength* kuyruk özneliden 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* öznelikten daha uzun olan bir *mantıksal* iletiyi, ancak uygulama MQMD ' de MQMF_SEGMENTATION_ALLOWLY işaretini belirtirse, bu iletiyi 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 özneliği için alt sınır 32 KB ' dir (32 768 bayt). Üst sınır 100 MB ' dir (104 857 600 bayt).

Bu özneliğin 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 değışiyor.

Bu özniteliğın değeri 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 değeri için bayt cinsinden boyutunu içerir.

Bu özniteliğın değeri saptamak için, MQINQ çağrısıyla MQIA_MAX_PROPERTIES_LENGTH seçiciyi 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 seçeneğı ile gönderilen iletiler
- Uygulama tarafından MQGMO_SYNCPOINT seçeneğı ile alınan iletiler
- MQPMO_SYNCPOINT seçeneğı ile gönderilen iletiler için kuyruk yöneticisi tarafından oluşturulan iletileri ve COA rapor iletilerini tetikler
- MQGMO_SYNCPOINT seçeneğıyle alınan iletiler için kuyruk yöneticisi tarafından oluşturulan COD raporu iletileri

Aşağıdakiler, kesinleştirilmemiş iletiler olarak sayılan *değil* ' dir:

- 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 seçeneğı 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 seçeneğı 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 değeri 10000 'dir.

Bu özniteliğın değeri saptamak için, MQINQ çağrısıyla MQIA_MAX_UNCOMMITTED_MSGS seçicisini kullanın.

MQIAccounting (MQUZE)

Bu, MQI verilerine ilişkin muhasebe bilgilerinin toplanmasını denetler.

Değeri aşağıdakilerden biridir:

MQMON_ON

API muhasebe verilerini toplar.

MQMON_OFF

API muhasebe verilerini toplamaz. Bu varsayılan değerdir.

ACCTCONO kuyruk yöneticisi özniteliğini ENABLE değerine ayarladıysanız, bu değer MQCNO yapısındaki Seçenekler alanını kullanan tek tek bağlantılar için geçersiz kılınabilir. Bu değerdeki değişiklikler yalnızca, öznitelige yapılan değişiklikten sonra oluşan kuyruk yöneticisine yönelik bağlantılar için geçerlidir.

Bu öznitelik yalnızca IBM i, UNIX sistemler ve Windows üzerinde desteklenir.

Bu özniteliğin 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 topla.

MQMON_OFF

MQI istatistikleri toplanmasın. Bu varsayılan değerdir.

Bu öznitelik yalnızca IBM i, UNIX ve Linux sistemleri ve Windows sistemlerinde desteklenir.

Bu özniteliğin 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 özniteliğin 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 öznitelik yalnızca z/OS üzerinde desteklenir.

Bu özniteliğin 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 öznitelik yalnızca z/OS üzerinde desteklenir.

Bu özneliğin değeri 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 değeri saptamak için, MQINQ çağrısıyla MQIA_PERFORMANCE_EVENT seçiciyi kullanın.

Altyapı (MQPLONG)

Kuyruk yöneticisinin çalışmakta olduğu işletim sistemini gösterir:

MQPL_AIX

AIX (MQPL_UNIX ile aynı değer).

V 8.0.0.2 MQP_APPLIANCE

IBM MQ Appliance

MQPL_MVS

z/OS (MQPL_ZOS ile aynı değer).

MQPL_NSK

HP Integrity NonStop Server.

MQPL_OS390

z/OS (MQPL_ZOS ile aynı değer).

MQPL_OS400

IBM i.

MQPL_UNIX

UNIX sistemleri.

MQPL_WINDOWS_NT

Windows sistemleri.

MQPL_ZOS

z/OS.

Bu özneliğin değeri saptamak için, MQINQ çağrısıyla MQIA_PLATFORM seçiciyi kullanın.

PubSubNPInputMsg (MQHOD)

Teslim edilmemiş bir giriş iletilisinin atılıp atılmayacağını ya da saklanmayacağını.

Değer aşağıdakilerden biridir:

MQUNDELIVERED_ATIN

İşlenemezse, kalıcı olmayan giriş iletileri atılabilir.

Bu varsayılan değerdir.

MQUNDELIVERED_KEEP

İşlenemezse, 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 özneliğin değeri 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 özniteliğin 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:

MQSYNCPPOINT_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.

MQSYNCPPOINT_YES

Bu, kuyruğa alınan yayınlama/abone olma arabiriminin tüm iletileri eşitleme noktası altında almasına neden olur.

Bu özniteliğin 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 şu ortamlarda desteklenir: AIX, HP-UX, z/OS, IBM i, Solaris, Linux, Windows, artı IBM MQ istemcileri bu sistemlere bağlı olarak.

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 tanıtıcıları, kuyruk yöneticisi ağında benzersiz olması için ilk 12 karakterde farklı olan adlara sahip 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/OSüzerinde desteklenir.

QueueAccounting (MQHOMER)

Bu, kuyruklar için muhasebe bilgilerinin toplanmasını denetler.

Değer aşağıdakilerden biridir:

MQMON_NONE

Kuyruk hesap özniteliği ACCTQ ayarından bağımsız olarak, kuyruklar için muhasebe verilerini toplamaz. Bu varsayılan değerdir.

MQMON_OFF

ACCTQ kuyruk öznitelide QMGR belirten kuyruklar için muhasebe verileri toplanmayın.

MQMON_ON

ACCTQ kuyruk öznitelide QMGR belirten kuyruklar için muhasebe verilerini toplayın.

Bu değerdeki değişiklikler yalnızca, özniteliğe yapılan değişiklikten sonra oluşan kuyruk yöneticisine yönelik bağlantılar için geçerlidir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA_ACCOUNTING_Q seçiciyi kullanın.

QueueMonitoring (MQUZE)

Bu, kuyrukların çevrimiçi izlenmesine ilişkin varsayılan ayarı belirtir.

QueueMonitoring kuyruk özniteliği MQMON_Q_MGR olarak ayarlandıysa, bu öznitelik, 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* özniteliğinin 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 özniteliğin 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 özniteliğinin ayarından bağımsız olarak kuyruklar için kuyruk istatistiği toplanmayın. Bu varsayılan değerdir.

MQMON_OFF

QueueStatistics kuyruk öznitelide Kuyruk Yöneticisi belirten kuyruklar için istatistik verileri toplanmayın.

MQMON_ON

QueueStatistics kuyruk öznitelide Kuyruk Yöneticisi belirten kuyruklar için istatistik verileri toplayın.

Bu özniteliğin 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:

- MQRCVTIME_EQUAL-Bu değer, kanala ilişkin bekleyeceği saniye cinsinden sayıdır. 0-999999 aralığında bir değer belirtin.
- MQRCVTIME_ADD-bu değer, kararlaştırılan HBINT ' ye eklenecek saniye cinsinden sayıdır ve bir kanal ne kadar süreyle bekleyeceğini belirler. 1-999999 aralığında bir değer belirtin.
- MQRCVTIME_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 MQRCVTIME_MULTIPLE or MQRCVTIME_EQUAL, and ReceiveTimeout to 0, to stop a channel from timing out its wait to receive data from its partner.

Bu öznitelik yalnızca z/OS üzerinde desteklenir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA_RECEVE_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 görelilik olarak hesaplanacağını belirtmek için ReceiveTimeouttipini kullanırsanız ve sonuçtaki değer bu parametrenin değerinden küçükse, bunun yerine bu değer kullanılır.

Bu öznitelik yalnızca z/OS üzerinde desteklenir.

Bu özniteliğin 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:

MQRCVTIME_ÇARP

ReceiveTimeout , bir kanalın ne kadar bekleyeceğini belirlemek üzere anlaşmalı HBINT değerine uygulanacak bir çarpıcıdır. Bu varsayılan değerdir.

MQRCVTIME_ADD

ReceiveTimeout is a value, in seconds, to add to the negotiated HBINT value to determine how long a channel waits.

MQRCVTIME_EQUAL

ReceiveTimeout , kanal bekleyeceği saniye cinsinden bir değer.

Bir kanal zamanlamasını, ortasından veri almak için beklemesini durdurmak için, ReceiveTimeouttipini MQRCVTIME_ÇOĞAL ya da MQRCVTIME_EQUAL olarak ayarlayın ve ReceiveTimeout değerini 0 olarak ayarlayın.

Bu öznitelik yalnızca z/OS üzerinde desteklenir.

Bu özniteliğin 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:

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_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 öznitelik yalnızca AIX, HP-UX, IBM i, Linux, Solaris, Windowsve z/OSüzerinde desteklenir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısına sahip MQCA_REPOSITORY_NAME seçiciyi kullanın. Bu özniteliğin 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 öznitelik yalnızca AIX, HP-UX, IBM i, Linux, Solaris, Windowsve z/OSüzerinde desteklenir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQCA_REPOSITORY_NAMELIST seçiciyi kullanın. Bu özniteliğin 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 öznitelik yalnızca z/OSüzerinde desteklenir.

Bu özneliğin değeri 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 paylaşıyorsa ve *ObjectQmgrName* , aynı kuyruk paylaşım grubundaki bir kuyruk yöneticisine, açık yerel olarak gerçekleştirilir.

Bu öznitelik yalnızca z/OSüzerinde geçerlidir.

Bu özneliğin değeri saptamak için, MQINQ çağrısıyla MQIA_SHARED_Q_Q_MGR_NAME seçicisini kullanın.

KYR

Bir kuyruk yöneticisi için IBM MQ Advanced Message Security güvenlik yeteneklerinin kullanılabilir olup olmadığını belirtir.

MQCAP_DESTEKLENEN

Kuyruk yöneticisinin çalışmakta olduğu kuruluş için IBM MQ AMS bileşeni kuruluysa, bu varsayılan değerdir.

MQCAP_NOT_SUPPORTANT

SSLEVENT (MQUZA)

Bu, SSL olaylarının oluşturulup oluşturulmayacağını belirtir.

Bu değer aşağıdaki değerlerden biridir:

MQEVRENABLED

SSL olaylarını şu şekilde oluşturun:

MQRC_CHANNEL_SSL_ERROR

MQEVR_DISABLE

SSL olayları oluşturmayın; bu varsayılan değerdir.

Bu özneliğin değeri saptamak için, MQINQ çağrısıyla MQIA_SSL_OLAY seçiciyi kullanın.

SSLFIPSRequired (MQHOT)

Bu, şifreleme donanımında değil, IBM MQ içinde yürütülürse, yalnızca FIPS onaylı algoritmaların kullanılacağını belirtmenize olanak tanır. Şifreleme donanımı yapılandırılmışsa, kullanılan şifreleme modülleri, donanım ürünü tarafından sağlanan modüllerdir; bu modüller, kullanılmakta olan donanım ürününe bağlı olarak belirli bir düzey için FIPS onaylı olabilir ya da olmayabilir.

Değer, aşağıdaki değerlerden biridir:

MQSSL_FIPS_NO

Kullanımda platformda desteklenen herhangi bir CipherSpec ' i kullanın. Bu değer, varsayılan değerdir.

MQSSL_FIPS_YES

Bu kuyruk yöneticisinden ve bu kuyruk yöneticisinden tüm SSL bağlantılarında izin verilen CipherSpecs içinde yalnızca FIPS onaylı şifreleme algoritmaları kullanın.

Bu parametre yalnızca UNIX, Linux, Windows ve z/OS platformlarında geçerlidir.

Bu özneliğin değerini saptamak için, MQINQ çağrısına sahip MQIA_SSL_FIPS_REQUIRED seçiciyi kullanın.

İlgili bilgiler

MQI istemcisinde çalıştırma sırasında yalnızca FIPS onaylı CipherSpecs ' in kullanıldığını belirtme [Federal Information Processing Standards \(FIPS\) for UNIX, Linux and Windows](#)

SSLKeyResetSayı (MQUZE)

Bu, iletişim başlatan SSL kanal 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 arasında bir SSL/TLS gizli anahtarı sıfırlama sayısı belirtirseniz, SSL/TLS kanalları 32 KB gizli anahtar sıfırlama sayısını kullanır. Bu, küçük SSL/TLS gizli anahtar ilk duruma getirme değerleri için oluşacağı aşırı anahtar sıfırlamalarının işlem maliyetinden kaçınmak içindir.

Başlangıç kanalı MCA tarafından gönderilen ve alınan toplam şifrelenmemiş bayt sayısı belirtilen değeri aştığında ya da kanal kalp atışları, bir kanal kalp atımı sonrasında gönderilmeden ya da alınmadan önce kanal kalp atışları etkinleştirilmişse, gizli anahtar 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 anlaşılamadığını belirtmek için 0 değerini kullanın.

Bu özneliğin 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 özneliğin 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 özneliğin değerini saptamak için, MQINQ çağrısıyla MQIA_STATISTICS_INTERVAL seçiciyi kullanın.

SyncPoint (MQUZE)

Bu durum, yerel kuyruk yöneticisinin, MQGET, MQPUT ve MQPUT1 çağrılarına 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 (KAINT) değerini belirlerseniz, bu özniteliğin ayarının kullanılacağı değer kullanılır.

MQTCPKEEP_NO

TCP KEPAVE kullanılmayı kullanmaniz. 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, TCPStackTypedegerine 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 başlatıcısının yalnızca TCPName 'de belirtilen TCP/IP yığınını kullanıp kullanamayacağını ya da isteğe bağlı olarak seçilen herhangi bir TCP/IP yığınınına 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 boşdur. Ancak, belirli koşullar altında, kuyruk boş olmasa da, MQTT_FIRST tetikleyicisiyle ek bir tetikleyici iletileri 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 özniteliğin 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 boşdur. Ancak, belirli koşullar altında, kuyruk boş olmasa da, MQTT_FIRST tetikleyicisiyle ek bir tetikleyici iletileri 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 özniteliğin değerini saptamak için MQINQ çağrısıyla MQIA_TRIGGER_INTERVAL seçiciyi kullanın.

Sürüm (MQCFST)

Bu, VVRRMFF biçiminde IBM MQ kodunun sürüşidir. Burada:

VV-Sürüm

RR-Serbest Bırak

MM-Bakım düzeyi

FF-Düzeltilme düzeyi

XrCapability(MQUZE)

This controls whether IBM MQ Telemetry commands are supported by the queue manager.

Değer aşağıdakilerden biridir:

MQCAP_DESTEKLENEN

IBM MQ Telemetry bileşeni kurulu ve Telemetry komutları desteklenmektedir.

MQCAP_NOT_SUPPORTANT

IBM MQ Telemetry bileşeni kurulu değil.

Bu öznitelik yalnızca IBM i, Unix sistemlerinde ve Windowssistemlerinde desteklenir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA_XR_CAPABILITY seçiciyi kullanın.

Kuyruklara ilişkin öznitelikler

Beş tip kuyruk tanımlaması vardır. Bazı kuyruk öznitelikleri tüm kuyruk tipleri için geçerlidir; diğer kuyruk öznitelikleri 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 z/OS üzerinde, bunu paylaşılan ya da özel bir kuyruk yapabilirsiniz.

Bir kuyruk, programın bağlı olduğu kuyruk yöneticisine aitse, *yerel* olarak bilinen bir programı tanımaktadır. Yerel kuyruklardan iletiler alabilir ve iletileri, yerel kuyruklara yerleştirebilirsiniz.

Kuyruk tanımlaması nesnesi, kuyruğa ilişkin fiziksel iletilerin yanı sıra, kuyruğun tanımlama bilgilerini de içerir.

Yerel kuyruk yöneticisi kuyruğu

Kuyruk yerel kuyruk yöneticisinde var.

z/OS Kuyruk, z/OS üzerinde özel bir kuyruk olarak bilinir.

z/OS Paylaşılan kuyruk (yalnızca z/OS)

Kuyruk, paylaşılan havuzun sahibi olan kuyruk paylaşım grubuna ait olan tüm kuyruk yöneticilerinin erişebildiği paylaşılan bir havuzda var olur.

Kuyruk paylaşımı 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, küme kuyruk yöneticisinin barındırdığı ve kümedeki diğer kuyruk yöneticilerine sunulan bir kuyruktur. **QType** kuyruğu özniteliğinin değeri MQQT_CLUSTER' dir.

Küme kuyruğu tanımlaması, kümedeki diğer kuyruk yöneticilerine duyurulmaktadır. Kümedeki diğer kuyruk yöneticileri, karşılık gelen bir uzak kuyruk tanımlamasına gerek duymadan, iletileri bir küme kuyruğuna yerleştirebilir. Küme kuyruğu, bir küme adı listesi kullanılarak birden çok kümede tanımlanabilir.

Bir kuyruk duyurulduğunda, kümedeki herhangi bir kuyruk yöneticisi iletiye ileti yerleştirebilir. Bir iletiyi koymak için, kuyruk yöneticisinin kuyruğun barındırıldığı tüm havuzlardan 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. Birden çok küme iletim kuyruğundaki iletileri iki farklı şekilde saklamak için bir kuyruk yöneticisi yapılandırabilirsiniz. If you set the queue manager attribute **DEFCLXQ** to KANAL, a different cluster transmission queue is created automatically from **SYSTEM.CLUSTER.TRANSMIT.MODEL.QUEUE** for each cluster-sender channel. **CLCHNAME** iletim kuyruğu seçeneğini bir ya da daha çok küme gönderici kanalına eşleştirmek için ayarlıyorsanız, kuyruk yöneticisi eşleşen kanallara ilişkin iletileri bu iletim kuyruğunda saklayabilir.



Uyarı: If you are using dedicated **SYSTEM.CLUSTER.TRANSMIT.QUEUES** with a queue manager that was upgraded from a version of the product earlier than IBM WebSphere MQ 7.5, ensure that the **SYSTEM.CLUSTER.TRANSMIT.MODEL.QUEUE** has the **SHARE/NOSHARE** option set to **SHARE**.



Küme kuyruğu, IBM MQ for z/OS' ta 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 diğer adı

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 addır. 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

V8.0.0.15

Sürüm 8.0.0, Düzeltme Paketi 15'tan, kuyruk yöneticisi kuyruk dosyası büyüklüğü üst sınırını varsayılan olarak 2 TB' ye sınırlar.

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 112 sayfa 813](#) içinde ve sonraki tablolarda gösterilir.

[Çizelge 112 sayfa 813](#) , 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. Öznitelikleri tanımlamak, değiştirmek ya da görüntülemek için MQSC komutları kullanıldığında, diğer kısa adlar kullanılır; ayrıntılar için [Script \(MQSC\) komutları](#) konusuna bakın.

Çizelge 112. Kuyruklara ilişkin öznitelikler. Kolonlar 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.

Öznitelik	Tanım	Yerel	Model	Diğer Ad	Uzak	Küme
<u>AlterationDate</u>	Tanımlamanın son olarak değiştirildiği tarih	✓		✓	✓	
<u>AlterationTime</u>	Tanımlamanın son olarak değiştirildiği saat	✓		✓	✓	
<u>BackoutRequeueQName</u>	Aşırı geriletme yeniden kuyruğa alma kuyruğu adı	✓	✓			
<u>BackoutThreshold</u>	Geriletme eşiği	✓	✓			
<u>BaseQName</u>	Diğer adın çözümleyicileceği kuyruk adı			✓		
<u>CFStrucName</u>	Bağlaşım-tesis yapısı adı	✓	✓			
<u>CLCHNAME</u>	Küme gönderici kanal adları	✓	✓			
<u>ClusterName</u>	Kuyruğun ait olduğu kümenin adı	✓		✓	✓	✓
<u>ClusterNameList</u>	Kuyruğun ait olduğu kümelerin adlarını içeren ad listesi nesnesinin adı	✓		✓	✓	
<u>CLWLQueuePriority</u>	Küme iş yükü kuyruğu önceliği	✓		✓	✓	✓
<u>CLWLQueueRank</u>	Küme iş yükü kuyruğu sırası	✓		✓	✓	✓
<u>CLWLUseQ</u>	Uzak kuyruğu kullan	✓				
<u>CreationDate</u>	Kuyruğun yaratıldığı tarih	✓				
<u>CreationTime</u>	Kuyruğun yaratıldığı saat	✓				
<u>CurrentQDepth</u>	Yürürlükteki kuyruk derinliği	✓				
<u>DefaultPutResponse</u>	Varsayılan yanıt koyma yanıtı	✓	✓	✓	✓	
<u>DefBind</u>	Varsayılan bağ tanımı	✓		✓	✓	✓
<u>DefinitionType attribute</u>	Kuyruk tanımlaması tipi	✓	✓			
<u>DefInputOpenOption</u>	Varsayılan giriş açma seçeneği	✓	✓			
<u>DefPersistence</u>	Varsayılan ileti kalıcılığı	✓	✓	✓	✓	✓
<u>DefPriority</u>	Varsayılan ileti önceliği	✓	✓	✓	✓	✓
<u>DefReadAhead</u>	Varsayılan okuma önden okuma	✓	✓	✓		
<u>DistLists</u>	Dağıtım listesi desteği	✓	✓			
<u>HardenGetBackout</u>	Doğru bir geriletme sayısının sağlanıp korunmayacağı	✓	✓			
<u>IndexType</u>	Dizin tipi	✓	✓			
<u>InhibitGet</u>	Kuyruğa ilişkin alma işlemlerine izin verilip verilmeyeceği	✓	✓	✓		

Çizelge 112. Kuyruklara ilişkin öznitelikler. Kolonlar 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.

(devamı var)

Öznitelik	Tanım	Yerel	Model	Diğer Ad	Uzak	Küme
InhibitPut	Kuyruğa ilişkin koyma işlemlerine izin verilip verilmeyeceği	✓	✓	✓	✓	✓
InitiationQName	Başlatma kuyruğunun adı	✓	✓			
MaxMsgLength	İleti uzunluğu üst sınırı (byte)	✓	✓			
MaxQDepth	Kuyruk derinliği üst sınırı	✓	✓			
MsgDeliverySequence attribute	İleti teslim sırası	✓	✓			
NonPersistentMessage Class	Kalıcı olmayan iletiler için güvenilirlik hedefi	✓	✓			
OpenInputCount	Giriş için açma sayısı	✓				
OpenOutputCount	Çıkış için açma sayısı	✓				
PropertyControl	Özellik denetimi	✓	✓	✓		
ProcessName	İşlem adı	✓	✓			
QDepthHighEvent attribute	Kuyruk Derinliği Yüksek olaylarının oluşturulup oluşturulmayacağı	✓	✓			
QDepthHighLimit	Kuyruk derinliği üst sınırı	✓	✓			
QDepthLowEvent attribute	Kuyruk Derinliği Düşük olaylarının oluşturulup oluşturulmayacağı	✓	✓			
QDepthLowLimit attribute	Kuyruk derinliği için alt sınır	✓	✓			
QDepthMaxEvent	Kuyruk Dolu olaylarının oluşturulup oluşturulmayacağı	✓	✓			
QDesc	Kuyruk tanımlaması	✓	✓	✓	✓	✓
QName	Kuyruk adı	✓		✓	✓	✓
QServiceInterval	Kuyruk hizmet aralığı hedefi	✓	✓			
QServiceIntervalEvent attribute	Hizmet Aralığı Yüksek ya da Hizmet Aralığı Tamam olaylarının oluşturulup oluşturulmayacağı	✓	✓			
QSGDisp attribute	Kuyruk paylaşımı grubu yok etme	✓		✓	✓	
QueueAccounting	Kuyruk muhasebesi veri toplama	✓	✓	✓	✓	✓
QueueMonitoring	Kuyruklar için çevrimiçi izleme verileri	✓	✓			
QueueStatistics	Kuyruk istatistikleri veri toplama	✓	✓	✓	✓	✓
QType	Kuyruk tipi	✓		✓	✓	✓
RemoteQMGrName	Uzak kuyruk yöneticisinin adı				✓	

Çizelge 112. Kuyruklara ilişkin öznitelikler. Kolonlar 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.

(devamı var)

Öznitelik	Tanım	Yerel	Model	Diğer Ad	Uzak	Küme
<u>RemoteQName</u>	Uzak kuyruğun adı				✓	
<u>RetentionInterval</u>	Alıkoyma aralığı	✓	✓			
<u>Scope</u>	Kuyruğa ilişkin bir girişin, bir hücre dizininde de bulunup bulunmayacağını	✓		✓	✓	
<u>Shareability</u>	Kuyruk paylaşılabilişliği	✓	✓			
<u>StorageClass</u>	Kuyruğa ilişkin depolama sınıfı	✓	✓			
<u>TriggerControl</u>	Tetikleyici denetimi	✓	✓			
<u>TriggerData</u>	Verileri tetikle	✓	✓			
<u>TriggerDepth</u>	Tetikleyici derinliği	✓	✓			
<u>TriggerMsgPriority</u>	Tetikleyiciler için eşik iletili önceliği	✓	✓			
<u>TriggerType</u>	Tetikleme Tipi	✓	✓			
<u>Usage attribute</u>	Kuyruk kullanımı	✓	✓			
<u>XmitQName</u>	İletim kuyruğu adı				✓	

İlgili bilgiler

Küme kuyrukları

Yerel kuyruklar

AlterationDate (MQCHAR12)

Tanımlamanın en son değiştirildiği tarih.

Yerel	Model	Diğer Ad	Uzak	Küme
X		X	X	

Bu, tanımın en son değiştirildiği tarihtir. Tarihin biçimi şöyledir: YYYY-MM-DD, uzunluğun 12 baytı (örneğin, 1992-09-23--, burada -- iki boş karakteri temsil eder) içeren iki boşlukla doldurulabilir.

Kuyruk yöneticisi olarak, belirli özniteliklerin değerleri (örneğin, *CurrentQDepth*) değişir. Bu özniteliklerde yapılan değişiklikler *AlterationDate* işlemini etkilemez.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQCA_ALTERATION_DATE seçiciyi kullanın. Bu özniteliğin uzunluğu MQ_DATE_LENGTH tarafından verilir.

AlterationTime (MQCHAR8)

Tanımlamanın en son değiştirildiği saat.

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 'dır.

- 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 özniteliklerin değerleri (örneğin, *CurrentQDepth*) değişir. Bu özniteliklerde yapılan değişiklikler *AlterationTime* işlemini etkilemez.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQCA_ALTERATION_TIME seçiciyi kullanın. Bu özniteliğin uzunluğu MQ_TIME_LENGTH tarafından verilir.

BackoutRequeueQName (MQCHAR48)

Bu, aşırı geriletme yeniden kuyruğa alma kuyruğu adı. Değerin sorgulanmasına izin verilmesinin dışında, kuyruk yöneticisi bu özniteliğin değerini temel alarak bir işlem gerçekleştirmez.

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, kuyruk yöneticisi özniteliğin 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 değerini saptamak için, MQINQ çağrısıyla MQCA_BACKUUT_REQ_Q_NAME seçicisini kullanın. Bu özniteliğin uzunluğu MQ_Q_NAME_LENGTH tarafından verilir.

BackoutThreshold (MQLONG)

Bu geriletme eşiği. Değerin sorgulanmasına izin verilmesinin dışında, kuyruk yöneticisi bu özniteliğin değerini temel alarak bir işlem gerçekleştirmez.

Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

WebSphere Application Server içinde çalışan uygulamalar ve IBM MQ Application Server Faciliti'lerini kullanan uygulamalar, bir iletinin geriletilmesi gerekip gerekmediğini belirlemek için bu özniteliği kullanır. Diğer tüm uygulamalar için, kuyruk yöneticisi özniteliğin değerini temel alarak herhangi bir işlem gerektirir.

IBM MQ classes for JMS , iletiyi *BackoutRequeueQName* özniteliği tarafından belirtilen kuyruğa aktarılmadan önce bir iletinin kaç kez geriletileceğini belirlemek için bu özniteliği kullanır.

Bu özniteliğin değerini saptamak için MQINQ çağrısıyla MQIA_BACKOUT_THRESHOLD seçiciyi kullanın.

BaseQName (MQCHAR48)

Bu ad, yerel kuyruk yöneticisinde tanımlı olan bir kuyruğun adıdır.

Yerel	Model	Diğer Ad	Uzak	Küme
		X		

(Kuyruk adlarıyla ilgili daha fazla bilgi için bkz. [MQOD- ObjectName alanı](#).) Kuyruk, aşağıdaki tiplerden biridir:

MQQT_LOCAL

Yerel kuyruk.

MQQT_REMOTE

Uzak kuyruğun yerel tanımlaması.

MQQT_CLUSTER

Küme kuyruğu.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQCA_BASE_QNAME seçiciyi kullanın. Bu özniteliğin uzunluğu MQ_Q_NAME_LENGTH tarafından verilir.

BaseType (MQCFIN)

Diğer adın çözdüğü nesne tipi.

Yerel	Model	Diğer Ad	Uzak	Küme
		X		

Bu değer aşağıdaki değerlerden 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.

Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Bağlaşım tesisinde yapının tam adını almak için, QSGName queue-manager özniteliğinin değerini CFStrucName kuyruk özniteliğinin değeriyle sonek yazın.

Bu öznitelik yalnızca paylaşılan kuyruklar için geçerlidir; QSGDisp ' un MQQSGD_SHARED değeri yoksa, yoksayılr.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQCA_CF_STRUC_NAME seçicisini kullanın. Bu özniteliğin uzunluğu, MQ_CF_STRUC_NAME_LENGTH tarafından verilir.

Bu öznitelik yalnızca z/OSüzerinde desteklenir.

ClusterChannelAd (MQCHAR20)

ClusterChannelName , bu kuyruğu bir iletim kuyruğu olarak kullanan, kümenin gönderici kanallarının genel adıdır. Öznitelik, hangi küme gönderen kanallarının bu küme iletim kuyruğundan bir küme alıcı kanalına ileti gönderdiğini belirtir.

Yerel	Model	Diğer Ad	Uzak	Küme
✓	✓			

Varsayılan kuyruk yöneticisi yapılandırması, tek bir iletim kuyruğundan ileti göndermek için tüm küme gönderici kanallarının SYSTEM.CLUSTER.TRANSMIT.QUEUE' dir. Varsayılan yapılanış, kuyruk yöneticisi özniteliği (DefClusterXmitQueueType) değiştirilerek değiştirilebilir. Özniteliğin varsayılan değeri SCTQ' dur. Değeri KANALolarak değiştirebilirsiniz. DefClusterXmitQueueType özniteliğini KANALolarak ayarlıyorsanız, her bir küme gönderen kanalı varsayılan olarak belirli bir küme iletim kuyruğunu kullanacak şekilde varsayılan değer olarak SYSTEM.CLUSTER.TRANSMIT.ChannelNamekullanılır.

Ayrıca, iletim kuyruğu özniteliği `ClusterChannelName` özniteliğini bir küme gönderen kanala el ile de ayarlayabilirsiniz. Küme gönderen kanalının bağlı olduğu kuyruk yöneticisine gönderilen iletiler, küme gönderici kanalını tanımlayan iletim kuyruğunda saklanır. Bunlar varsayılan küme iletim kuyruğunda saklanmaz. `ClusterChannelName` özniteliğini boşluklara ayarlıyorsanız, kanal yeniden başlatıldığında, kanal varsayılan küme iletim kuyruğuna geçer. Varsayılan kuyruk, kuyruk yöneticisi `DefClusterXmitQueueType` özniteliğinin değerine bağlı olarak `SYSTEM.CLUSTER.TRANSMIT.ChannelName` ya da `SYSTEM.CLUSTER.TRANSMIT.QUEUE` olur.

Yıldız işaretlerini ("*"), **`ClusterChannelName`**) belirterek, bir iletim kuyruğunu bir küme gönderici kanalları kümesiyle ilişkilendirebilirsiniz. Yıldız imleri, kanal adı dizgisinin başında, başında ya da ortasındaki herhangi bir sayıda yere olabilir. **`ClusterChannelName`**, 20 karakterden oluşan bir uzunlula sınırlıdır: `MQ_CHANNEL_NAME_LENGTH`.

ClusterName (MQCHAR48)

Bu, kuyruğun ait olduğu kümenin adıdır.

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.

Yerel	Model	Diğer Ad	Uzak	Küme
X		X	X	

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 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.

Yerel	Model	Diğer Ad	Uzak	Küme
X		X	X	X

Ek bilgi için [Küme kuyrukları](#) 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.

Yerel	Model	Diğer Ad	Uzak	Küme
X		X	X	X

Ek bilgi için [Küme kuyrukları](#) başlıklı konuya bakın.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA_CLWL_Q_RANK seçiciyi kullanın.

CLWLUseQ (MQUSEL)

Bu, hedef kuyrukta 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.

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 kuyrukları](#) başlıklı konuya bakın.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQCA_CLWL_USEQ seçiciyi kullanın. Bu özniteliğin uzunluğu MQ_CLWL_USEQ_LENGTH tarafından verilir.

CreationDate (MQCHAR12)

Bu, kuyruğun oluşturulduğu tarihtir.

Yerel	Model	Diğer Ad	Uzak	Küme
X				

Tarihin biçimi şöyledir: YYYY-MM-DD; uzunluğun 12 baytı (örneğin, 2013-09-23--), burada -- 2 boş karakteri simgeler) içeren iki boşlukla doldurulabilir.

- 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.

Yerel	Model	Diğer Ad	Uzak	Küme
X				

Saatin biçimi, saatin 10 'dan küçük olması (örneğin, 09.10.20gibi), 24 saatlik zaman biçimini kullanarak HH.MM.SS ' dır.

- 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.

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.

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 tanıımıdır.

Yerel	Model	Diğer Ad	Uzak	Küme
X		X	X	X

Değer aşağıdakilerden biridir:

MQBND_BIND_ON_Açık

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 bağlıyorsa, bu değer kullanılmamalıdır.

Bu özniteliğin değerini saptamak için, MQINQ çağrısına sahip MQIA_DEF_BIND seçiciyi kullanın.

DefinitionType (MQUZE)

Bu, kuyruğun nasıl tanımlandığını gösterir.

Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Değer aşağıdakilerden biridir:

MQQDT_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).

MQQDT_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 MQQDT_PERMANENT_DYNAMIC değerini aldı.

Bu kuyruk tipi, MQCLOSE çağrısını kullanarak silinebilir. Daha ayrıntılı bilgi için bkz. [“MQCLOSE- Nesneyi kapat” sayfa 630](#).

Kalıcı dinamik kuyruğa ilişkin *QSGDISP* özniteliğinin değeri MQQSGD_Q_MGR ' dir.

MQQDT_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 MQQDT_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.

MQQDT_SHARED_DYNAMIC

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 tanımı, *DefinitionType* özniteliği için MQQDT_SHARED_DYNAMIC değerini aldı.

Bu kuyruk tipi, MQCLOSE çağrısını kullanarak silinebilir. Daha ayrıntılı bilgi için bkz. [“MQCLOSE- Nesneyi kapat” sayfa 630](#).

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.

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 açıksa 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.

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 bulunsan bile geçerlidir.

Paylaşılan kuyruklar durumunda, kalıcı olmayan iletiler *do* kuyruk-paylaşım grubundaki kuyruk yöneticilerinin yeniden başlatılmasını sağlar, ancak paylaşılan kuyruklarda iletileri saklamak için kullanılan bağlaşımlı tesisinin başarısızlıklarını atlamaz.

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.

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.

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.

Yerel	Model	Diğer Ad	Uzak	Küme
Evet	Evet	Evet	Evet	Evet

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 özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA_DEF_PUT_RESPONSE_TYPE seçiciyi kullanın.

DistLists (MQUZE)

Bu, dağıtım listesi iletilerinin kuyruğa yerleştirilip yerleştirilemeyeceğini gösterir.

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 özniteliği 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 özniteliği ayarlar. Bu şekilde, MCA 'nın gönderilmesi, yerel kuyruk yöneticisinin yalnızca iş ortağı kuyruk yöneticisinin doğru işleyebileceği iletileri ileti kuyruğuna yerleştirmesine neden olabilir.

Bu öznitelik öncelikle iletim kuyruklarıyla kullanım içindir, ancak tanımlanan işleme, kuyruk için tanımlanan kullanımdan bağımsız olarak gerçekleştirilir (bkz. [Kullanım özniteliği](#)).

Değer aşağıdakilerden biridir:

MQDL_DESTEKLENEN

Dağıtım listesi iletileri kuyrukta saklanabilir ve bu formdaki iş ortaklığı kuyruk yöneticisine iletebilir. 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 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 çağrısına sahip MQIA_DIST_LISTS seçiciyi kullanın. Bu özniteliğin değerini değiştirmek için MQSET çağrısını kullanın.

Bu öznitelik 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.

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.

IBM i, UNIX sistemlerive Windowsüzerinde, bu öz niteliğin ayarından bağımsız olarak, ileti arka çıkış 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 öz niteliğin değerini saptamak için, MQINQ çağrısıyla MQIA_HARDEN_GET_BACKUP seçiciyi kullanın.

IndexType (MQUZE)

Kuyruk yöneticisinin kuyruklardaki iletiler için sakladığı dizin tipini belirtir.

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 öz niteliği](#)). *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 *gerekir*, uygulamanın MQGET çağrısındaki MQGMO_LOGICAL_ORDER seçeneğini kullanarak iletileri aldığı kuyruklar için kullanılır.

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:

1. 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.
2. 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 113 sayfa 826](#) ve [Çizelge 114 sayfa 827](#) içinde gösterilir.

<i>Çizelge 113. MQGMO_LOGICAL_ORDER belirtilmediğinde, kuyruk dizini tipi için önerilen ya da gerekli değerler</i>		
MQGET çağrısındaki seçim ölçütleri	Paylaşılmayan kuyruk için dizin tipi	Paylaşılan kuyruğa ilişkin dizin tipi
Yok	Herhangi	Herhangi
Bir tanıtıcı kullanarak seçim:		
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
İki tanıtıcı kullanarak seçim:		
İleti tanıtıcısı artı 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 anahtarlara 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
Üç tanıtıcı kullanarak seçim:		
İleti tanıtıcısı artı ilinti tanıtıcısı artı grup tanıtıcısı	MQIT_MSG_ID ya da MQIT_COREL_ID ya da MQIT_GROUP_ID önerilendi	Desteklenmiyor
Grup ile ilgili ölçütler kullanarak seçim:		
Grup tanıtıcısı ve ileti sıra numarası	MQIT_GROUP_ID gerekli	MQIT_GROUP_ID gerekli

Çizelge 113. MQGMO_LOGICAL_ORDER belirtilmediğinde, kuyruk dizini tipi için önerilen ya da gerekli değerler (devamı var)

MQGET çağrısındaki seçim ölçütleri	Paylaşılmayan kuyruk için dizin tipi	Paylaşılan kuyruğa ilişkin dizin tipi
İleti sıra numarası (1 olmalıdır)	MQIT_GROUP_ID gerekli	MQIT_GROUP_ID gerekli
İleti simgesi kullanılarak seçim:		
Uygulama kullanımı için ileti simgesi	MQIT_MSG_TOKEN kullanımını kullanmayın	
WLM kullanımı için ileti simgesi	MQIT_MSG_TOKEN gerekli	Desteklenmiyor

Çizelge 114. MQGMO_LOGICAL_ORDER belirlendiğinde, kuyruk dizin tipi için önerilen ya da gereken değerler

MQGET çağrısındaki seçim ölçütleri	Paylaşılmayan kuyruk için dizin tipi	Paylaşılan kuyruğa ilişkin dizin tipi
Yok	MQIT_GROUP_ID gerekli	MQIT_GROUP_ID gerekli
Bir tanıtıcı kullanarak seçim:		
İleti Tanıtıcısı	MQIT_GROUP_ID gerekli	Desteklenmiyor
İlinti tanıtıcısı	MQIT_GROUP_ID gerekli	Desteklenmiyor
Grup tanıtıcısı	MQIT_GROUP_ID gerekli	MQIT_GROUP_ID gerekli
İki tanıtıcı kullanarak seçim:		
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
Üç tanıtıcı kullanarak seçim:		
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/OSüzerinde desteklenir.

InhibitGet (MQUZE)

Bu, kuyruğa ilişkin alma işlemlerinin izin verilip verilmemesini denetler.

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 engellemektedir.

MQQA_GET_ALLOWD

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.

Yerel	Model	Diğer Ad	Uzak	Küme
X	X	X	X	X

Kuyruk adı çözme yolunda birden fazla tanımlama varsa, koyma işlemi sırasında, MQPUT ya da MQPUT1 çağrısının başarılı olması için yoldaki *her* tanımlaması (herhangi bir 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 engellemektedir.

MQQA_PUT_ALLOWD

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.

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.

Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Ancak, *MaxMsgLength* kuyruk özniteliği *MaxMsgLength* queue-manager özniteinden bağımsız olarak ayarlanabileceğinden, kuyruğa yerleştirilebilecek en uzun fiziksel iletinin uzunluğuna ilişkin gerçek üst sınır, bu iki değerden daha küçük olan bir değer olur.

Kuyruk yöneticisi kesimlere ayırma özelliğini destekliyorsa, uygulama, iki *MaxMsgLength* özneliğinin 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 özneliği için alt sınır sıfır; üst sınır 100 MB ' dir (104 857 600 bayt).

Daha fazla bilgi için bakınız: [MQPUT- BufferLength](#) parameter.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQIA_MAX_MSG_LENGTH seçiciyi kullanın.

MaxQDepth (MQUZE)

Bu, herhangi bir zamanda kuyrukta bulunabilecek fiziksel ileti sayısı için tanımlanan üst sınırdır.

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 bölümlenmesinin her ikisi de kuyruktaki gerçek fiziksel ileti sayısının *MaxQDepth* ' u aşmasına neden olabilir. Ancak bu, iletilerin alınabilirliğini etkilemez; MQGET çağrısı kullanılarak kuyruklardaki tüm iletiler alınabilir.

Bu özneliğin değeri sıfır ya da daha büyük. Üst sınır, ortam tarafından belirlenir:

- AIX, HP-UX, z/OS, Solaris, Linuxve Windowsüzerinde değer 999 999 999 'u aşamaz.
- IBM üzerinde değer 640 000 'i aşamaz.

Not: Kuyruktaki *MaxQDepth* iletisinden daha az ileti olsa da, kuyruğun kullanabileceği saklama alanı çok yorulmuş olabilir.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQIA_MAX_Q_DEPTH seçiciyi kullanın.

MsgDeliverySırası (MQUBE)

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 iletilisini 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/OSüzerinde, *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.

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 başlatılınca atılır. 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 özniteliğin 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.

Yerel	Model	Diğer Ad	Uzak	Küme
X				

Bu, yerel kuyruk yöneticisi tarafından bilinen tanıtıcı sayıların toplam sayısıdır. Kuyruk paylaşılan bir kuyruksa, kuyruk için, yerel kuyruk yöneticisinin ait olduğu kuyruk paylaşım grubundaki diğer kuyruk yöneticilerindeki kuyruk için gerçekleştirilen giriş için sayı 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 özniteliğin değeri kuyruk yöneticisi olarak işlev görmektedir.

Bu özniteliğin 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.

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, kuyruk için, 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 özniteliğin değeri kuyruk yöneticisi olarak işlev görmektedir.

Bu özniteliğin 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.

Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQCA_PROCESS_NAME seçiciyi kullanın. Bu özniteliğin uzunluğu MQ_PROCESS_NAME_LENGTH tarafından verilir.

PropertyControl (MQUZE)

MQGMO_PROPERTIES_AS_Q_DEF seçeneğiyle MQGET çağrısını kullanan kuyruklardan alınan iletiler için ileti özelliklerinin nasıl işleneceğini belirtir.

Yerel	Model	Diğer Ad	Uzak	Küme
X	X	X		

Değer 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. önekine 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, iletin tüm özellikleri atılır ve artık bu uygulamanın erişilemeyecek şekilde bulunur. Bu varsayılan değerdir; JMS ile ilgili özelliklerin, değiştirilmemiş 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. İletinin 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, Diğer Ad ve Model kuyrukları için geçerlidir. Değerini saptamak için, MQINQ çağrısıyla MQIA_PROPERTY_CONTROL seçiciyi kullanın.

QDepthHighOlayı (MQUZE)

Kuyruk Derinliği Yüksek olaylarının oluşturulup oluşturulmayacağını denetler.

Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Kuyruk Derinliği Yüksek olayı, bir uygulamanın bir kuyruğa ileti yerleştirdiğini ve bu, kuyruktaki ileti sayısının kuyruk derinliği üst eşiğinden büyük ya da ona eşit olmasına neden olduğunu gösterir (*QDepthHighLimit* özneliğine bakın).

Not: Bu özneliğin değeri dinamik olarak değişebilir.

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_Q_DEPTH_HIGH_EVENT seçiciyi kullanın.

Bu öznelik z/OS üzerinde desteklenir, ancak bu özneliğin değerini belirlemek için MQINQ çağrısı kullanılamaz.

QDepthHighSınırı (MQUZE)

Kuyruk Derinliği Yüksek bir olay oluşturmak için kuyruk derinliğinin karşılaştırıldığı eşik değeri budur.

Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Bu olay, bir uygulamanın bir kuyruğa ileti koyduğunu ve kuyruğun kuyruk derinliği üst eşiğinden büyük ya da bu değere eşit olacak ileti sayısına neden olduğunu gösterir. Bkz. [QDepthHighOlay özneliği](#).

Değer, kuyruk derinliği üst sınırı (*MaxQDepth* özneliği) 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 değer. Varsayılan değer 80 'dir.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQIA_Q_DEPTH_HIGH_LIMIT seçicisini kullanın.

Bu öznelik z/OS üzerinde desteklenir, ancak bu özneliğin değerini belirlemek için MQINQ çağrısı kullanılamaz.

QDepthLowOlayı (MQUZE)

Bu, Kuyruk Derinliği Düşük olaylarının oluşturulup oluşturulmayacağını denetler.

Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Kuyruk Derinliği Düşük olayı, bir uygulamanın bir kuyruktan ileti alındığını ve kuyruktaki ileti sayısının, kuyruk derinliği düşük eşiğinden küçük ya da bu değere eşit olmasına neden olduğunu gösterir (bkz. [QDepthLowSınır](#) özneliği).

Not: Bu özneliğin değeri dinamik olarak değişebilir.

Değer aşağıdakilerden biridir:

MQEVN_DISABLE

Olay raporlama devre dışı bırakıldı.

MQEVN_ENABLED

Olay raporlaması etkinleştirildi.

Olaylarla ilgili daha fazla bilgi için [Olay izleme](#) başlıklı konuya bakın.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQIA_Q_DEPTH_LOW_EVENT seçiciyi kullanın.

Bu öznelik z/OS üzerinde desteklenir, ancak bu özneliğin değerini belirlemek için MQINQ çağrısı kullanılamaz.

QDepthLowSınırı (MQUZE)

Kuyruk Derinliği Düşük bir olay oluşturmak için kuyruk derinliğinin karşılaştırıldığı eşik budur.

Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Bu olay, bir uygulamanın kuyruktan ileti aldığı ve kuyruğun kuyruk derinliği düşük eşiğinden küçük ya da bu değere eşit olmasına neden olduğunu gösterir. Bkz. [QDepthLowOlay](#) özneliği.

Değer, kuyruk derinliği üst sınırı (*MaxQDepth* özneliği) 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 değer. Varsayılan değer 20'dir.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQIA_Q_DEPTH_LOW_LIMIT seçicisini kullanın.

Bu öznelik z/OS üzerinde desteklenir, ancak bu özneliğin değerini belirlemek için MQINQ çağrısı kullanılamaz.

QDepthMaxOlayı (MQUZE)

Bu, Kuyruk Tam olaylarının oluşturulup oluşturulmayacağını denetler. Kuyruk Dolu olayı, kuyruğun dolu olduğu için, kuyruğun bir kuyruğa koyma değerinin reddedildiğini, yani kuyruk derinliğinin zaten üst sınır değerine ulaştığını gösterir.

Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Not: Bu özneliğin değeri dinamik olarak değişebilir.

Değer aşağıdakilerden biridir:

MQEVN_DISABLE

Olay raporlama devre dışı bırakıldı.

MQEVN_ENABLED

Olay raporlaması etkinleştirildi.

Olaylarla ilgili daha fazla bilgi için [Olay izleme](#) başlıklı konuya bakın.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla MQIA_Q_DEPTH_MAX_EVENT seçiciyi kullanın.

Bu öznitelik z/OSüzerinde desteklenir, ancak bu özniteliğin değerini belirlemek için MQINQ çağrısı kullanılamaz.

QDesc (MQCHAR64)

Açıklayıcı açıklamalar için bu alanı kullanın.

Yerel	Model	Diğer Ad	Uzak	Küme
X	X	X	X	X

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_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.

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.

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.

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**.

MQQSI_OK

Kuyruk Hizmet Aralığı Tamam olayları etkinleştirildi.

- Kuyruk Hizmet Aralığı Yüksek olayları **geçersiz kılındı** ve
- Kuyruk Hizmet Aralığı Tamam olayları **etkindir**.

MQQSI_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 yoksayılr; MQQSI_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.

Yerel	Model	Diğer Ad	Uzak	Küme
X		X	X	

Değer aşağıdakilerden biridir:

MQQSGD_Q_MGR

Nesnenin kuyruk yöneticisi yok etme değeri var. Bu, nesne tanımlamasının yalnızca yerel kuyruk yöneticisinde bulunduğu 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 nesneldir 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, kuyruk-paylaşım grubundaki tüm kuyruk yöneticileri tarafından bilinen nesnenin tek bir somut örneğinin bulunduğu anlamına gelir. Gruptaki bir kuyruk yöneticisi nesneye eriştiğinde, nesneye ilişkin tek paylaşılan eşgörünümüne erişir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA_QSG_DISP seçicisini kullanın.

Bu öznitelik yalnızca z/OSüzerinde desteklenir.

QueueAccounting (MQHOMER)

Yerel	Model	Diğer Ad	Uzak	Küme
X	X	X	X	

Bu, kuyruğa ilişkin muhasebe verilerinin toplanmasını denetler. Bu kuyruk için toplanacak hesap verileri için, MQCONNX çağrısındaki MQCNO yapısındaki QMGR özniteliği ACCTQ ya da Options alanı kullanılarak, bu bağlantıya ilişkin hesap verileri de etkinleştirilmelidir.

Bu öznitelik aşağıdaki değerlerden birine sahiptir:

MQMON_Q_MGR

Bu kuyruğa ilişkin muhasebe verileri, ACCTQ QMGR özniteliği ayarına dayalı olarak toplanır. Bu varsayılan ayardır.

MQMON_OFF

Bu kuyruk için hesap verileri toplanmasın.

MQMON_ON

Bu kuyruk için muhasebe verilerini topla.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA_ACCOUNTING_Q seçiciyi kullanın.

QueueMonitoring (MQUZE)

Kuyruklar için çevrimiçi izleme verilerinin toplanmasını denetler.

Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Değer aşağıdakilerden biridir:

MQMON_Q_MGR

İzleme verilerini *QueueMonitoring* kuyruk yöneticisi özniteliğinin ayarına göre toplayın. Bu varsayılan değerdir.

MQMON_OFF

Bu kuyruk için çevrimiçi izleme verileri toplama kapatılır.

MQMON_LOW

QueueMonitoring kuyruk yöneticisi özniteliğinin değeri MQMON_NONE değilse, çevrimiçi izleme verileri derlemi açık olur ve bu kuyruk için düşük veri toplama işlemi olur.

MQMON_ORTAMı

QueueMonitoring kuyruk yöneticisi özniteliğinin değeri MQMON_NONE değilse, çevrimiçi izleme verileri derlemi açık olur ve bu kuyruk için orta düzeyde veri toplama işlemi vardır.

MQMON_YÜKSEK

QueueMonitoring kuyruk yöneticisi özniteliğinin değeri MQMON_NONE değilse, çevrimiçi izleme verileri derlemi açık olur ve bu kuyruk için yüksek oranda veri toplama işlemi vardır.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA_MONITORING_Q seçiciyi kullanın.

QueueStatistics (MQCHAR12)

Yerel	Model	Diğer 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 toplamasına geçiş yapabilirsiniz.

QType (MQUZE)

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 çağrısıyla MQIA_Q_TYPE seçiciyi kullanın.

RemoteQMGrAd (MQCHAR48)

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. Otherwise, if *XmitQName* is blank when the open occurs, there must be a local queue with a name that is the same as *RemoteQMGrName* ; this queue is used as the transmission queue.

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 değerini saptamak için, MQINQ çağrısıyla MQCA_REMOTE_Q_MGR_NAME seçicisini kullanın. Bu özniteliğin uzunluğu, MQ_Q_MGR_NAME_LENGTH tarafından verilir.

RemoteQName (MQCHAR48)

Yerel	Model	Diğer Ad	Uzak	Küme
			X	

Bu, uzak kuyruk yöneticisinin 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, *ReplyToQ* 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 MQCA_REMOTE_Q_NAME seçicisini kullanın. Bu özniteliğin uzunluğu MQ_Q_NAME_LENGTH tarafından verilir.

RetentionInterval (MQUZE)

Bu, kuyruğun alıkonacağı süre (time). Bu süre geçtikten sonra kuyruk silinmeye uygun olur.

Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Saat, kuyruğun oluşturulduğu tarih ve saatten başlayarak saat olarak ölçülür. Kuyruğun 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 sağlanmıştır.

Not: Kuyruk yöneticisi, bu özniteliğe dayalı olarak kuyrukları silmek ya da süresi dolmayan bir alıkoyma aralığıyla kuyrukların silinmesini önlemek için hiçbir zaman herhangi bir işlem kullanmaz; bu, gereken işlemi yapmak kullanıcının sorumluluğundadır.

Kalıcı dinamik kuyrukların birikmesini önlemek için gerçekçi bir alıkoyma aralığı kullanın (bkz. *DefinitionType* özniteliği). Ancak, bu öznitelik önceden tanımlanmış kuyruklarla da kullanılabilir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA_RETENTION_INTERVAL seçiciyi kullanın.

Kapsam (MQLONG)

Bu, bu kuyruğa ilişkin bir girişin bir hücre dizininde de olup olmadığını denetler.

Yerel	Model	Diğer Ad	Uzak	Küme
X		X	X	

Kurulabilir bir Ad hizmeti tarafından bir hücre dizini sağlanır. Değer aşağıdakilerden biridir:

MQSCO_Q_MGR

Kuyruk tanımlamasının kuyruk yöneticisi kapsamı var: Kuyruğun tanımlaması, sahip olduğu kuyruk yöneticisinin ötesine geçmiyor. Kuyruğu 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.

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.

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.

Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Bu, aşağıdakilerden biridir:

MQTC_OFF

Bu kuyruk için herhangi bir tetikleme iletileri 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 iletinin başlatma kuyruğuna yazılmasına neden olduğunda, tetikleme iletinin eklediği serbest biçimli verilerdir.

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 dizgisi 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)

Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Bu, bir tetikleme ileti 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 ileti oluşturulup oluşturulmamaya karar verirken kuyruk yöneticisi bu iletileri yoksayar) ileti önceliğidir.

Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

TriggerMsgPriority, sıfır (en düşük) ile *MaxPriority* (en düşük) aralığında olabilir (highest; bkz. *MaxPriority* özniteliği); sıfır değeri, tüm iletilerin, tetikleyici iletileri oluşturulmasına katkıda bulunmasına neden olur.

Bu özniteliğin değerini saptamak için MQINQ çağrısıyla MQIA_TRIGGER_MSG_PRIORITY seçiciyi kullanın. Bu özniteliğin değerini değiştirmek için MQSET çağrısını kullanın.

TriggerType (MQUZE)

Bu, kuyruğa gelen iletilerin bir sonucu olarak tetikleme iletilerinin yazıldığı koşulları denetler.

Yerel	Model	Diğer Ad	Uzak	Küme
X	X			

Aşağıdaki değerlerden birine sahiptir:

MQTT_NONE

Bu kuyruktaki iletilerin sonucu olarak hiçbir tetikleme ileti yazılmadı. Bu, *TriggerControl* 'un MQTT_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 ileti 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 MQTT_OFF olarak ayarlanır.

Bu özniteliğin değerini saptamak için, MQINQ çağrısına sahip MQIA_TRIGGER_TYPE seçiciyi kullanın. Bu özniteliğin değerini değiştirmek için MQSET çağrısını kullanın.

Kullanım (MQUZE)

Bu, kuyruğun ne için kullanıldığını gösterir.

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 aracılığıyla ileti, iletim kuyruğundan okur ve iletiyi uzak kuyruk yöneticisine aktarır. İletim kuyruklarıyla ilgili ek bilgi için İletim kuyruğunun tanımlanması 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 599) ya da iletim işlemi sırasında hata oluşabilir. Bağlam, MQPMO_ *_CONTEXT bağlam seçeneklerinden biri belirlenmedikçe geçirilmez ya da ayarlanmaz.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA_USAGE seçiciyi kullanın.

XmitQName (MQCHAR48)

Bu, iletim kuyruğu adıdır. Bir uzak kuyruk ya da bir kuyruk yöneticisi diğer adı tanımlaması için açık bir durum olduğunda bu öznitelik boş değilse, iletiyi iletmek için kullanılacak yerel iletim kuyruğunun adını belirler.

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 özniteliği tarafından tanımlanan kuyruk kullanılır.

Tanımlama bir kuyruk yöneticisi diğer adı olarak kullanılıyorsa ve *RemoteQMGrName* yerel kuyruk yöneticisinin adıyla bu öznitelik yoksayılar. Tanımlama, bir yanıt kuyruğu diğer adı tanımı olarak kullanılırsa da yoksayılar.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQCA_XMIT_QNAME seçicisini kullanın. Bu özniteliğin uzunluğu MQ_Q_NAME_LENGTH tarafından verilir.

Ad listelerine ilişkin öznitelikler

Aşağıdaki çizelge, ad listelerine özgü öznitelikleri özetlemektedir. Öznitelikler alfabetik sırayla açıklanır.

Ad listeleri tüm IBM MQ sistemlerinde desteklenir, artı IBM MQ MQI clients bu sistemlere bağlı olarak desteklenir.

Not: Bu kısımda gösterilen özniteliklerin adları, MQINQ ve MQSET çağrılarıyla kullanılan açıklayıcı adlardır; bu adlar, PCF komutlarıyla aynı olur. Öznitelikleri tanımlamak, değiştirmek ya da görüntülemek için MQSC komutları kullanıldığında, diğer kısa adlar kullanılır; ek bilgi için [Script \(MQSC\) Commands](#) (Komut Dosyası (MQSC) Komutları) konusuna bakın.

Çizelge 115. Ad listelerine ilişkin öznitelikler	
Öznitelik	Tanım
AlterationDate	Tanımlamanın son olarak değiştirildiği tarih
AlterationTime	Tanımlamanın son olarak değiştirildiği saat
NameCount	Ad listesindeki ad sayısı
NamelistDesc	Ad listesi açıklaması
NamelistName	Ad listesi adı
Adlar	<i>NameCount</i> adlarının listesi
NamelistType	Ad listesi tipi
QSGDisp	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 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.

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 nesnelerinin 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 nesnelerinin adlarını içeren ad listesi.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQIA_NAMELIST_TYPE seçiciyi kullanın.

Bu öznitelik yalnızca z/OSüzerinde 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 nesnelerinin 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 nesneldir 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/OSüzerinde desteklenir.

Süreç tanımlamalarına ilişkin öznitelikler

Aşağıdaki çizelge, süreç tanımlamalarına özgü öznitelikleri özetlemektedir. Öznitelikler alfabetik sırayla açıklanır.

Not: Bu kısımdaki özniteliklerin adları, MQINQ ve MQSET çağrılarıyla kullanılan açıklayıcı adlardır; bu adlar PCF komutlarıyla aynıdır. Öznitelikleri tanımlamak, değiştirmek ya da görüntülemek için MQSC komutları kullanıldığında, diğer kısa adlar kullanılır; ek bilgi için [Script \(MQSC\) Commands](#) (Komut Dosyası (MQSC) Komutları) konusuna bakın.

Çizelge 116. Süreç tanımlamalarına ilişkin öznitelikler	
Öznitelik	Tanım
AlterationDate	Tanımlamanın son olarak değiştirildiği tarih
AlterationTime	Tanımlamanın son olarak değiştirildiği saat
ApplId	Uygulama tanıtıcısı
ApplType	Uygulama tipi
EnvData	Ortam verileri
ProcessDesc	Süreç Tanımı
ProcessName	İşlem adı
QSGDisp	Kuyruk paylaşımı grubu yok etme
UserData	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.

ApplId (MQCHAR256)

Bu, başlatılacak uygulamayı tanımlayan bir karakter dizilimidir. Bu bilgiler, başlatma kuyruğunda iletileri işleyen bir tetikleme izleme programı uygulaması tarafından kullanılır; bilgiler, tetikleme iletinin bir parçası olarak başlatıcı kuyruğuna gönderilir.

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* 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 sistemlerinde program adının başına bir sürücü ve izin yolu eklenerek önek olarak eklenebilirsiniz.
- UNIX sistemleri üzerinde, program adının başına bir izin yolu önek olarak eklenebilir.

Karakter dizisi boş boş (nulls) içeremez. Gerekirse boşluklarla doldurulabilir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla MQCA_APPL_ID seçiciyi kullanın. Bu özniteliğin uzunluğu, MQ_PROCESS_APPL_ID_LENGTH tarafından verilir.

ApplType (MQUBE)

Bu, bir tetikleme iletişimin alınına yanıt olarak başlatılacak programın niteliyi tanımlar. Bu bilgiler, başlatma kuyruğunda iletileri işleyen bir tetikleme izleme programı uygulaması tarafından kullanılır; bilgiler, tetikleme iletişimin bir parçası olarak başlatıcı kuyruğuna gönderilir.

ApplType herhangi bir değere sahip olabilir, ancak standart tipler için aşağıdaki değerler önerilir; kullanıcı tanımlı uygulama tiplerini MQAT_USER_FIRST ile MQAT_USER_SON aralığında değerlerle sınırlayın.

MQAT_AIX

AIX uygulaması (MQAT_UNIX ile aynı değer).

MQAT_BATCH

Toplu iş uygulaması

MQAT_BROKER

Aracı uygulaması

MQAT_CICS

CICS işlemi.

MQAT_CICS_KöPRÜSÜ

CICS bridge uygulaması.

MQAT_CICS_VSE

CICS/VSE işlemi.

MQAT_DOS

PC DOS üzerinde IBM 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_NSK

HP Integrity NonStop Server 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 uygulaması.

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 özniteliğin değerini saptamak için, MQINQ çağırısına sahip MQIA_APPL_TYPE seçiciyi 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 iletinin 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:

- Açıkz/OS:
 - *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.
- UNIX sistemleri 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 çağırısıyla 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 çağırısıyla 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 nesneldir 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 dizilimidir. Bu bilgiler, başlatma kuyruğunda iletileri işleyen bir tetikleme izleme uygulaması ya da tetikleme izleme programı tarafından başlatılan uygulama tarafından kullanılmak içindir. Bilgiler, tetikleme iletinin 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 dizgisi 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 ¹
 - MQOO_INPUT_SHARED ¹
 - MQOO_INPUT_AS_Q_DEF ¹
 - MQOO_SORGULAMA
 - MQOO_OUTPUT

- MQOO_SET
- MQOO_BIND_ON_Açık 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_Açık
- 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 *değil* ' a izin veriliyor.

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 aşağıdakilerin 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 852](#)
- [“Abone İletisi Deregister İletisi” sayfa 853](#)
- [“İletiyi yayınlama” sayfa 857](#)
- [“Abone iletisini kaydet” sayfa 859](#)
- [“Güncelleme İletisi İste” sayfa 863](#)

Kuyruğa alınmış yayınlama/abone olma uygulamaları yazıyorsanız, bu iletileri, kuyruk yöneticisi yanıt iletilisini ve ileti tanımlayıcısını (MQMD) anlamanız gerekir; aşağıdaki bilgilere bakın:

- “Kuyruk Yöneticisi Yanıt iletilisi” sayfa 865
- “Kuyruk yöneticisi tarafından iletilen yayınlara ilişkin MQMD ayarları” sayfa 871
- “Kuyruk yöneticisi yanıt iletilerinde MQMD ayarları” sayfa 871
- “Yayınlama/abone olma neden kodları” sayfa 867

Komutlar, MQRFH2 üstbilgisinin **NameValueData** alanındaki bir <psc> klasöründe yer alır. Bir komut iletilisine 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 iletilisini sil

Kuyruk yöneticisine, belirtilen konulara ilişkin alıkonan yayınları silmesini sağlamak için, **Delete Publication** komut iletilisi 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 iletilisinde belirtilen konuların tümünü değil, ancak tümü için yetkiniz varsa, yalnızca bu konular silinir. Bir **Broker Response** iletilisi 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 iletilisi gönderirken gerekli olan ileti tanımlayıcı (MQMD) parametrelerinin ayrıntıları için “Kuyruk yöneticisi tarafından iletilen yayınlara ilişkin MQMD ayarları” sayfa 871 başlıklı konuya bakın.

Özellikler

< Komut> (MQPSC_COMMAND)

Değer: DeletePub (MQPSC_DELETE_YAYINI).

Bu özellik belirtilmeli.

< Topic> (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 yinelenabilir.

<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 iletinin gönderildiği kuyruk yöneticisi) silinir; bunlar Yerel seçeneğiyle yayınlanıp yayınlanmayacaklarını belirtmez.

Diğer kuyruk yöneticilerindeki yayınlar etkilenmez.

Yok (MQPSC_NONE)

Tüm seçenekler varsayılan değerlerini alır. Bu, DelOpt 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>
  <DelOpt>Local</DelOpt>
</psc>
```

Abone İletisi Deregister İletisi

Deregister Subscriber komut iletisi, bir abone tarafından ya da bir abone adına başka bir uygulama tarafından, verili değişirgelerle eşleşen iletileri almak istemediğini belirtmek için bir kuyruk yöneticisine gönderilir.

Bu ileti SYSTEM.BROKER.CONTROL.QUEUE(kuyruk), kuyruk yöneticisinin denetim kuyruğu. Kullanıcının bu kuyruğa 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 aboneliğin ilgili konusu, abonelik noktası ve süzgeç değerleri belirtilerek, bireysel abonelikte kayıt dışı olabilir. Değerlerden herhangi biri belirtilmediyse (varsayılan değerleri almışlardı), özgün abonelikte, abonelik kayıttan kaldırıldığında atlanmalırlar.

Bir aboneye ya da bir aboneye ilişkin tüm abonelikler, DeregAll seçeneği kullanılarak kayıttan kaldırılabilir. Örneğin, 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 değil, kuyrukla ilişkilendirilir. Kuyruk, belirtik olarak ya da kuyruk yöneticisinden gelen uygulama bağlantısını keserek silinirse, artık o kuyruğa ilişkin abonelikleri kayıttan kaldırmak için **Deregister Subscriber** komutunu kullanmanız artık mümkün değildir. Abonelikler, geliştirici çalışma ortamı kullanılarak kayıttan kaldırılabilir ve kuyruk yöneticisi tarafından bir sonraki yayınlara ya da kuyruk yöneticisi yeniden başlatıldığında otomatik olarak kuyruk yöneticisi tarafından kaldırılabilir. Normal koşullar altında, uygulamaların kuyruğu silmeden önce aboneliklerini silmesi ya da kuyruk yöneticisinden bağlantıyı kesmeleri gerekir.

Bir abone, bir aboneliği kayıttan kaldırmak için bir ileti gönderirse ve bu iletinin başarıyla işlendiğini belirten bir yanıt iletisi alırsa, abonelik kayıttan kaldırıldığı için aynı zamanda kuyruk yöneticisi tarafından işlenmekte olan bazı yayınlar abone kuyruğuna erişmeye devam edebilir. İletiler kuyruktan kaldırılmamışsa, abone kuyruğunda 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.

< Topic> (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 belirleyemeyeniniz 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 *MQRCCF_SUB_IDENTITY_ERROR*.

< Süzgeç > (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 yinelenabilir. 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 > yinelenabilir.
- <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 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 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 eşiyle 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, abonelik 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.

< Topic> (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 V6' da, <SubPoint> özelliğinin değeri, yayınlamayı işleyen Publication düğümünün Abonelik Noktası özneteliğinin değeridir.

IBM MQ V7.0.1' de, <SubPoint> özelliğinin değeri bir abonelik noktasının adıyla eşleşmelidir. Bkz. [Abonelik noktası eklenmesi](#).

<PubOpt> (MQPSC_PUBLICTION_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 Register Subscriber ya da Request Update komut iletisi yanıtında 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) ReplyToQMgr 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, alıkonan 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 kaçının.

İleti, SYSTEM.BROKER.CONTROL.QUEUE, kuyruk yöneticisinin denetim kuyruğudur. Bu kuyruğa bir ileti koyma yetkisi (kuyruk yöneticisinin sistem denetimsi 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 iletlerinde kuyruk yöneticisine MQMD ayarları” sayfa 870](#) başlıklı konuya bakın.

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.

< Topic> (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 V6' da, <SubPoint> özelliğinin değeri, abone olunan Publication düğümlerinin Abonelik Noktası özneliğinin değeriyle eşleşmelidir.

IBM MQ V7.0.1' de, <SubPoint> özelliğinin değeri bir abonelik noktasının adıyla eşleşmelidir. Bkz. [Abonelik noktası eklenmesi](#).

< Süzgeç > (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 iletilinde, aboneliğin tüm öznitelikleri yanıt iletilinde döndürüldüğünde, komut başarısız olur.

FullResp , yalnızca komut iletili 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 iletilisine yanıt olarak bir Yayınlama iletili gönderdiğinde bir yayının alıkonması 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 kilitte değiştirilir. Tersisi 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 iletilisiyle aynı kimlik olduğu sürece, komut MQRCCF_SUBSCRIPTION_LOCKED dönüş koduyla başarısız olur. Bu durumda, kilit otomatik olarak paylaşılan bir kilitte değiştirilir.

Yerel

(MQPSC_LOCAL)

Abonelik yereldir ve ağdaki 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 öznelikleri değiştirilmez.

Bir abonelik oluşturulduğunda, bu seçenek yoksayılr. Belirtilen diğer tüm seçenekler yeni abonelik için geçerli olur.

Bir SubIdentity , birleştirme seçeneklerinden birine de sahipse (JoinExc1 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 (bunun *kayıt seçenekleri özelliğini atlayarak aynı şekilde affectolmadığını* unutmayın) ve abonelik süre bitimi, **Register Subscriber** iletilisinin 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 iletilisine 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 iletilisinin 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` ' dır 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`), yeni **Register Subscriber** 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 **Register Subscriber** 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 Register Subscriber komutunun sonucu olarak, aboneliğin özniteliklerinin değiştirilmesi, yalnızca bu abonelik için belirlenen kimlikler 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 dizgisi 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 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 PubOnReqOnly 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.

< Topic> (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 > (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 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. 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 iletisinin kullanıcı kimliği aboneliğin eşyle eşleşiyorsa başarılı olur. Komut iletisinin kullanıcı kimliği, aboneliğin tanıttıcısı ile eşleşmiyorsa, komut *MQRCCF_DUPLICATE_IDENTITY* dönüş koduyla başarısız olur.

VariableUserId değeri ayarlandıysa ve kullanıcı kimliği, abonelikten farklıysa, yeni komut iletisi kullanıcı kimliğinin akış kuyruğuna göz atma ve aboneliğin abone kuyruğuna konması için yetkisi varsa, komut başarılı olur. Tersisi durumda, komut *MQRCCF_NOT_YETKILİ* 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_error* dönüş koduyla başarısız olur.

Geleneksel kimlikle eşleşen bir komut iletisi kullanarak SubName içeren bir abonelik için güncelleme isteğinde bulunmaya çalışılırsa, ancak SubName belirtilmez, komut başarılı olur.

Örnek

Burada, bir **Request Update** komut iletisi 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 iletisi

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 iletisi, 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 iletisinde yer almıyorsa. Bir hata durumunda, bir hataya neden olan iletinin hiçbirisi işlenmedi; bir uyarı durumunda, iletinin 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 iletisi normal olarak geri alınır. Bu durum, MQInput düğümünün nasıl yapılandırılıp yapılandırılmadığına bağlıdır.

Özellikler

< Completion> (MQPSSCR_COMPLETION)

Üç değerden birini alabilen tamamlama kodu:

tamam

Komut başarıyla tamamlandı

uyarı

Komut tamamlandı, ancak uyarı ile

hata

Komut başarısız oldu

< Response> (MQPSSCR_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> (MQPSSCR_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>
  </Reponse>
</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>
  </Reponse>
  <Response>
    <Reason>3081</Reason>
    <Topic>topic2</Topic>
  </Reponse>
</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 kullanılabilecek sabitler de listelenir.

MQRC_sabitleri, IBM MQ cmqc.h üstbilgi dosyasını gerektirir. The MQRCCF_constants require the IBM MQ cmqcfc.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 MQRC_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 MQRC_RFH_PARM_ERROR	<psc> ve <mcd> klasörlerinde her ikisinin de içinde belirtilebilecek geçerli bir parametre kümesi vardır. Bu klasörlerin tanımlarını denetleyin ve yanlış değiştiğerler belirtmediğinizden emin olun.	Herhangi bir komut
2338 MQRC_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 MQRC_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 MQRC_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 kullanılabilecek genişletilmiş ileti seçimi sağlayıcısı yok.	Aboneyi ve İstek Güncellemesini Kaydet
2554 MQRC_İÇERİĞİ	Genişletilmiş ileti seçimi sağlayıcısı, geçerli ya da alıkonan yayında bir hata buldu.	Yayınlama ve Güncelleme İste

Neden kodu ve metin	Açıklama	Sertifikayı Veren
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

Neden kodu ve metin	Açıklama	Sertifikayı Veren
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 yayımla 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 değiştirgesi 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 değiştirilebilir 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 aşıyor 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 değiştirme 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 gerekirse, bir komut iletileri için MsgType , bir komut iletileri 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_CORREL_ID ile birlikte bu değeri belirtin.

ReplyToQ

Bu alan, yanıtların (varsa) gönderileceği kuyruğu tanımlar. Bu, gönderenin kuyruğu olabilir; bu, QName parametresinin iletiden atlanabileceği 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 Publish 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 parçası, 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 belirlediye çözülür.

Kalıcılık

Persistence (Kalıcılık) yayıncı tarafından ayarlanan değeri alır ya da yayıncı tarafından bu yayının gönderilmekte olduğu aboneye ilişkin Register Subscriber iletilisinde aksi belirtilmedikçe, MQPER_PERSISTENCE_AS_Q_DEF değerini belirttiye çözümlendi.

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 iletilisine 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 tanımlayıcısı” sayfa 404](#) .

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 edilir³Kayan noktalı biçim, aşağıda gösterildiği gibi, byte 'lar:

³ 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 117 sayfa 875’inde gösterilir.

Çizelge 117. Makine mimarilerine ilişkin kodlamaların özeti			
Makine mimarisi	İkili tamsayı kodlaması	Paketli-onlu tamsayı kodlaması	Kayan noktalı kodlama
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-işaretler alanı

MQMD ileti tanımlayıcısına ilişkin daha fazla bilgi için bkz. “MQMD-İleti tanımlayıcısı” sayfa 404.

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:

- 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 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 destekliorsa, 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/OSüzerinde, 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 sınaama 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 destekliorsa, 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 sınaadı:

- 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 kaplar.

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:

- 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 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

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, <https://www.ibm.com/support/docview.wss?uid=swg27005729> adresindeki *Data Conversion under IBM MQ* belgesine bakın.

Uygulama iletileri verilerini alan uygulamanın gerektirdiği gösteriye dönüştürmek için, MQGET çağrısının işlenmesinin bir parçası olarak, veri-dönüştürme çıkışı çağrılır. Uygulama iletileri 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 879](#).
- Yerleşik bir biçim işlenirken kuyruk yöneticisi tarafından kullanılan işleme kuralları; kullanıcı tarafından yazılan çıkışlar için de bu kurallar önerilir. Bkz. [“İşleme kuralları” sayfa 880](#).
- Rapor iletilerini dönüştürmeye ilişkin özel bilgiler; bkz. [“Rapor iletilerinin dönüştürülmesi” sayfa 884](#).
- Veri dönüştürme çıkışa aktarılan parametreler; bkz. [“MQ_DATA_CONV_EXIT-Veri dönüştürme çıkışı” sayfa 897](#).
- Karakter verilerini farklı gösterimler arasında dönüştürmek için çıkıştan kullanılabilecek bir çağrı; bkz. [“MQXCNVK-Karakterleri dönüştür” sayfa 891](#).
- Çıkışa özel veri yapısı parametresi; bkz. [“MQDXP-Data-conversion çıkış değiştirgesi” sayfa 885](#).

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:

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 yapılır:
 - MQGMO_ACCEPT_TRUNCATED_MSG seçeneği *değil* belirtildiyse, ileti dönüştürülmeden geri döndürülür; tamamlanma kodu MQCC_UYARI ve neden kodu MQRC_TRUNCATED_MSG_FAILED.
 - 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 yapılır:
 - 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:

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 *değil* belirtildiyse, ileti dönüştürülemez olarak döndürülür; tamamlanma kodu MQCC_UYARY ve neden kodu MQRC_CONVERTED_MSG_TOO_BÜYÜKÜ.
- 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 sırasında ya da dönüştürülürken), *Buffer* değiştirilmesinde döndürülen geçerli byte sayısı arabelleğin uzunluğu *küçüktür* 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. If truncation occurs (either before or during conversion), the length returned for the *DataLength* 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 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 tanılanırsa, 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, iletin kesilmesini de gerektirmediği sürece, dönüştürmenin neden gerçekleştirilemediğini gösteren bir kod olarak ayarlanır; kesme ile ilgili neden kodları, dönüştürme ile ilgili neden kodlarına göre önceliklidir. (Kesilen bir iletin 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_CCSID_ERROR ya da MQRC_TARGET_CCSID_ERROR.

Unicode karakter takımı UCS-2 , 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 *değil* 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 *gerekliyse*, kuyruk yöneticisi kullanıcı tarafından yazılan çıkışı, MQDLH yapısındaki *Format* alanı tarafından verilen adda çağırır ya da dönüştürmeyi gerçekleştir (*Format*, yerleşik bir biçimin adıdır).

MQGET çağırısı, 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 içindeki *içinde* bulunan 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 çağırısından geri dönün:

- The values of the *Format*, *CodedCharSetId*, and *Encoding* fields in the *MsgDesc* parameter describe the data in the MQXQH structure, and *değil* the application message data; the values are therefore *değil* the same as those specified by the application that issued the MQGET call.

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 çağırısını 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 çağırısına dönüştürülmez, ancak MQGMO_CONVERT seçeneği belirtilmişse bile bu işlem MQGET çağırısına dönüştürülmez.

Daha önce açıklanan MQXQH ve MQDH yapılarının işlenmesi öncelikli olarak ileti kanalı aracılığı 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 olasılıklar nedeniyle, veri dönüştürme çıkışı, aktarılan verilerin uzunluğunu bir şekilde ifade etmek için biçim adını *kullanmamalıdır* ; bunun yerine, çıkış, sağlanan verilerin uzunluğunu denetlememeli ve biçim adının belirttiği uzunluktan az veri dönüştürmeye hazırlıklı 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 özneteliği tarafından verilen karakter kümesidir. 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 yoksayılar. Ancak dönüştürülmekte olan ileti, mantıksal iletinin yalnızca bir kısmını içeren bir bölümse, *DataLength* alanı *olamaz* değiştirilir.

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 118. MQDXP ' deki alanlar		
Alan	Tanım	Konu
<i>StrucId</i>	Yapı tanıtıcısı	StrucId
<i>Version</i>	Yapı sürüm numarası	S\u00fcr\u00fcm
<i>AppOptions</i>	Uygulama seçenekleri	AppOptions
<i>Encoding</i>	Uygulama için sayısal kodlama gerekiyor	Kodlama
<i>CodedCharSetId</i>	Uygulama için karakter kümesi gerekli	CodedCharSetId
<i>DataLength</i>	İleti verilerinin bayt cinsinden uzunluğu	DataLength
<i>CompCode</i>	Tamamlanma kodu	CompCode
<i>Reason</i>	Neden kodu ön eleme <i>CompCode</i>	Neden
<i>ExitResponse</i>	Çıkıştan yanıt	ExitResponse
<i>Hconn</i>	Bağlantı tanıtıcı	Hconn
<i>pEntryPoints</i>	MQIEP yapısının adresi	pEntryNoktaları

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 kararlaştıracak 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 dönüştürülebilecek, 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 tanıttırıcısıdır. Bu tanıttırıcı değeri, MQGET çağrısını yayınlayan uygulama tarafından belirlenen tanıttırıcı 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 MQCC_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, MQRC_TRUNCATED_MSG_ACCEPTED döndürülmelidir.

• *Reason* ' un çıkışa giriş üzerinde başka bir değeri varsa:

- Dönüştürme başarılı olursa, *CompCode* MQCC_OK değerine ayarlanmalıdır ve *Reason* MQRC_NONE olarak ayarlanmalıdır.
- Dönüştürme başarısız olursa ya da iletinin genişlemesi ve arabelleğe sığması için kesilmesi gerekiyorsa, *CompCode* MQCC_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, MQGET çağrısını yayınlayan uygulama MQGMO_ACCEPT_TRUNCATED_MSG seçeneğini belirttiyse kesilmelidir:

- Bu seçenek belirtilirse, MQRC_TRUNCATED_MSG_ACCEPTED neden döndürülmedi.
- Bu seçenek belirlenmezse, ileti dönüştürülmeden geri döndürülür; neden kodu MQRC_CONVERTED_MSG_TOO_BÜYÜK değeri döndürülür.

Dönüştürmenin başarısız olmasının nedenini belirtmek için, listelenen neden kodları, çıkışta kullanılması önerilir; ancak, çıkış, uygun görüldüğünde MQRC_ * kodları kümesinden başka değerler döndürebilir. Ayrıca, çıkışta MQGET çağrısını veren uygulamayla iletişim kurmak istediği koşulları belirtmek için çıkış tarafından MQRC_APPL_FIRST ile MQRC_APPL_FIRST değerleri ayrılarak kullanılır.

Not: İleti başarıyla dönüştürülemezse, kuyruk yöneticisinin dönüştürülenmemiş iletiyi döndürmesi için,exit alanında MQXDR_CONVERSION_FAILED çıkışı *gerekir* çıkışı, *ExitResponse* alanında başarısız oldu. Bu, *Reason* alanında döndürülen neden kodundan bağımsız olarak geçerlidir.

MQRC_APPL_FIRST

(900, X'384 ') Uygulama tanımlı neden kodu için En Düşük değeri.

MQRC_APPL_SON

(999, X'3E7') Uygulama tanımlı neden kodlarına ilişkin en yüksek değer.

MQRC_CONVERTED_MSG_TOO_BÜYÜK

(2120, X'848 ') Dönüştürülen veriler arabellek için çok büyük.

MQRC_NOT_CONVERTED

(2119, X'847 ') İleti verileri dönüştürülmedi.

MQRC_SOURCE_CCSID_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_TARGET_CCSID_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).

Bu, çıkışa giriş/çıkış alanıdır.

StrucId

Tip: MQCHAR4

Yapı tanıttıcısı. Değer şu olmalıdır:

MQDXP_STRUC_ID

Veri dönüştürme çıkış parametresi yapısıyla ilgili tanıttıcı.

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 bildirimi

```
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 bildirimi (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
ORG MQDXP
MQDXP_AREA 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 tanıttıcı, 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, *Hconn* için aşağıdaki özel değer belirtilebilir:

MQHC_DEF_HCONN

Varsayılan bağlantı tanıttıcısı.

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 kaçınılabilir.

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 şu ortamlarda desteklenir: AIX, HP-UX, IBM i, Solaris, 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, *TargetCCSID* ve *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 takımı (örneğin, UCS-2gibi) olduğunda bu durum oluşabilir; ancak *TargetLength* , tek sayıda byte olan bir uzunluk belirtiyorsa.
- *TargetLength* , *SourceLength* değerinden küçük ya da daha büyük olabilir. MQXCNCV ' den dönüştürme *DataLength* , *TargetLength* 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 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_BIG ile tamamlanır. Bu durumda *TargetLength* byte 'tan az not döndürülebileceğini unutmayın.

- *TargetLength* , *SourceLength* değerinden küçük ya da daha büyük olabilir. MQXCNCV ' den dönüşte *DataLength* , *TargetLength* değerinden küçük ya da ona eşit.

Bu seçenek şu ortamlarda desteklenir: AIX, HP-UX, IBM i, Solaris, Linux, Windows.

Kodlama seçenekleri: Tanımlanan seçenekler, kaynak ve hedef dizgilerinin tamsayı kodlamasını belirtmek için kullanılabilir. The relevant encoding is used *yalnızca* when the corresponding character set identifier indicates that the representation of the character set in main storage is dependent on the encoding used for binary integers. Bu, yalnızca bazı çok baytlı karakter kümelerini (örneğin, UCS-2 karakter kümeleri) 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.

MQDCC_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 alınırsa, 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 874 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, karşılık gelen karakter kümesi, kodlamaya (örneğin, bir UCS-2 karakter kümesi gibi) bağlı bir çok baytlı karakter kümesine, çağrı neden kodu MQRC_SOURCE_INTEGER_ENC_ERROR ya da MQRC_TARGET_INTEGEREN_ENC_ERROR neden ile başarısız olur.

Kodlama seçenekleri şu ortamlarda desteklenmektedir: AIX, HP-UX, z/OS, IBM i, Solaris, Linux, Windows.

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: MQCHARxSourceUzunluk-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. *SourceLength* değerinden küçük ya da daha büyük olabilir.

TargetBuffer

Tip: MQCHARxTargetUzunluk-çıkış

Bu, *TargetCCSID* tarafından 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_CCSID_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_CCSID_HATASI

(2115, X'843 ') Hedef kodlanmış karakter takımı tanıtıcısı 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)

```
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 değıştirgeleri 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 değıştirgesi” sayfa 885](#).

MsgDesc

Tip: MQMD-giriş/çıkış

Çıkışa girişte, bu, *InBuffer* parametresindeki çıkışa aktarılan ileti verileriyle ilişkili ileti tanımlayıdır.

Not: Çıkışa aktarılan *MsgDesc* değıştirgesi her zaman, çıkış çağırın kuyruk yöneticisi tarafından desteklenen MQMD 'nin en son sürümü olur. 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, HP-UX, IBM i, Solaris, 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 değışiklikler uygulamaya geri yansıtılır. Çıkışta yapılan diğer değışiklikler yoksayılır; bu değışiklikler uygulamaya geri yansıtılmaz.

Çıkış MQDXP yapısının *ExitResponse* alanında MQXDR_OK değerini döndürürse, ancak ileti tanımlayıcısındaki *Encoding* ya da *CodedCharSetId* alanlarını değıştirmesse, kuyruk yöneticisi bu alanlar için, MQDXP yapısındaki ilgili alanların çıkışa giriş için sahip olduğu değerleri döndürür.

InBufferUzunluğu

Tip: MQHOME-giriş

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.

OutBufferUzunluğu

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: MQBYTEExOutBufferLength -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 yoksayılr.

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 işlev, çıkışa ilişkin sağlayıcı tarafından tanımlanır; ancak çıkış, burada açıklanan kurallara ve MQDXP ' nin ilişkili değiştirgeyle uyumlu olmalıdır.

Veri dönüştürme çıkışı için kullanılabilecek programlama dilleri ortam tarafından belirlenir.

2. Çıkış yalnızca aşağıdakilerin *tümü* true olduğunda 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 884).

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 uzunluğun *küçüktür* olduğu anlamına gelir. The exit should therefore check the value of *InBufferLength* 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.

Çıkışta kesilen iletileri dönüştürmek için *not* yazılmışsa ve *InBufferLength* beklenen değerden küçükse, çıkış *DataConvExitParms* parametresinin *ExitResponse* alanında MQXDR_CONVERSION_FAILED değerini döndürür; *CompCode* ve *Reason* alanları MQCC_UYARD ve MQRC_FORMAT_ERROR olarak ayarlanır.

If the exit *has* been written to convert truncated messages, the exit will 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 *InBuffer*. 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.

It is also possible for a message to expand *sırasında* conversion, to the point where it is bigger than *OutBuffer*. 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, arabellenin 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. 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 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 altsisteminin 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 *ainvolved* '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 aracısınan ileti dönüştürmesini desteklemeyen bir hedef kuyruk yöneticisine ileti göndermesine yol açabilirdi. Çıkış, kuyruk yöneticisinin ortamında çalıştırılmadığından, kuyruk yöneticisinin bütünlüğünden ödün veremiyor.

10. Çıkış tarafından kullanılabilir 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 sağlanıyor ve 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/OSüzerinde, 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)

```
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, (DATA CONVEXITPARMS, MSGDESC, INBUFFERLENGTH,      X
                INBUFFER, OUTBUFFERLENGTH, OUTBUFFER)

```

Çıkışa geçirilen parametreler aşağıdaki gibi bildirilir:

```

DATA CONVEXITPARMS  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 öğe 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 -Kural ve biçimlendirme üstbilgisi 2”](#) sayfa 508’inde açıklanan bilgileri tamamlar.

Özellik veri tiplerinin MQRFH2 veri tiplerine eşlenmesi

Bu konu, karşılık gelen MQRFH2 veri tipleriyle eşlenen ileti özelliği tiplerine ilişkin bilgi sağlar.

Çizelge 119. Desteklenen MQRFH2 veri tipleri	
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' öğe özniteyle gösterilir. Boş değer olmayan değerler için xsi:nil='false' özniteliğini 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 MQtanımlı 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.

- <mmps>

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 iletileri 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 iletileri 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 ileti 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>

Yayınla/abone olma komut iletilerini aracıya iletmek için IBM Integration Bus tarafından kullanılır.

- <pscr>

Yayınla/abone olma komut iletilerine yanıt olarak, aracıdan bilgi içermek için IBM Integration Bus tarafından kullanılır.

- IBM tarafından tanımlı olmayan, content= 'properties' özniteliğiyle işaretlenmemiş herhangi bir klasör.

<psc> ya da <pscr> 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, MQDLHya da MQRFH üstbilgisinde olmayan ilk üstbilginin hemen önüne yerleştirilir.

 z/OS sistemlerinde, oluşturulan MQRFH2 üstbilgisi, uygulamanın CCSIDiçinde 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 USS 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 USS threads, the value of THLICCSID can be changed using USS mapping macro **BPXYTHLI**.

Oluşturulan MQRFH2birleştirmeye ilişkin kurallar

Aşağıdaki kurallar, oluşturulmuş bir MQRFH2 ' ı var olan bir MQRFH2ile birleştirmek için geçerlidir. Oluşturulan MQRFH2 üstbilgisi, varsa, var olan bir MQRFH2 üstbilgisiyle birleştirilir:

1. Var olan MQRFH2 aynı konumda; IBM MQ , oluşturulmuş bir MQRFH2ya 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 MQRFH2içinde klasörleri birleştirmeye ilişkin kurallar

İleti özellikleri var olan bir MQRFH2ile 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 ögelerine 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 [Javakomutunu kullanma](#).

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:

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 türJMS 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 901 dosyasında belgelenmez; aşağıdakileri dikkate aldıysanız:

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 erişebilen 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 760 , 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 Sürüm 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 özniteliğine 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.

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 özniteliğini kullanın. Yalnızca bir değer kullanın. Bu öge özniteliği, özellikleri içeren herhangi bir klasördeki bir özellik üzerinde geçerlidir.

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 özniteliğini 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"**

CopyOption değeri	MQRFH2 öznitelik değeri
MQPD_COPY_FORWY	ileri
MQPD_COPY_REPLY	Yanıt
MQPD_COPY_REPORT	rapor
MQPD_COPY_YAYINLAMA	yayınlamak
MQPD_COPY_ALL	tümü Bunu başka bir değerle belirtmeyin. Başka bir değerle kullanıldığında, bu değer noned ışındaki 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 yoksayılır.
- 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 olmayanMQRFH2 üstbilgileri

Bir MQPUT, MQPUT1 ya da MQGET çağrı işlemleri sırasında, iletilerde 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, StrucLength alanı çok küçük olduğundan, aşağıdaki gibi:

- MQPUT ya da MQPUT1 çağrısı, MQRC_RFH_ERROR neden koduyla başarısız olur; uygulamanın bazı IBM MQ Sürüm 7 seçeneğini içerdiği belirlenirse, var olan uygulamaların başarısız olmamasını sağlar.
- MQGET çağrısı başarıyla geri döner ve hatayı içeren MQRFH2 , belirttiğiniz arabellekte döndürülür.

Kısmi 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 MQ Sürüm 7 seçeneğini içerdiğini saptaması durumunda, varolan uygulamaların başarısız olmamasını sağlar.
- 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
- İletin işlenmesinde kullanılır

Bu nedenle, algılama garanti edilmez.

Uygulamalarınızdan biri bir özelliğe erişmek için “MQSETMP-İleti özelliğini ayarla” sayfa 760ya 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 kullanılabilecek 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
- Desteklenen **olmayan** kod sayfası dönüştürmeleri

Bilgilerde aşağıdaki terimler kullanılmıştır:

-8

Indicates for HP-UX that the CCSID is for the HP-UX defined codeset *roman8*

AIX

IBM MQ for AIXdeğerini belirtir

HP-UX

IBM MQ for HP-UXdeğerini belirtir

Linux

Indicates IBM MQ for Linux for Intel and IBM MQ for Linux for zSeries

HP Integrity NonStop Server

IBM WebSphere MQ for HP Integrity NonStop Serverdeğerini belirtir

OS/400

IBM MQ for IBM ideğerini belirtir

Solaris

IBM MQ for Solarisdeğerini belirtir

Windows

IBM MQ for Windowsdeğerini belirtir

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 özneteliği CONVERT , kaynakta EVET olarak ayarlanarak değiş tokuş edilen bir kanal ayarlanabilir ve veri değiş tokacı olur.

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.

Bazı CCSID numaraları ve bazı sektör kod kümesi adları arasında çapraz başvuru için [Çizelge 120 sayfa 910](#) ' e bakın.

Kod kümesi adları ve CCSID ' ler

IBM MQ for z/OS , dile özgü tablolarda listelenenden daha fazla dönüştürme sağlar.

Çizelge 120. Kod kümesi adları ve CCSID ' ler	
Kod kümesi adları	CCSID ' ler
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

Çizelge 120. Kod kümesi adları ve CCSID ' ler (devamı var)

Kod kümesi adları	CCSID ' ler
big5	950
eucJP	954 5050 33722
eucKR	970
eucTW	964
eucCN	1383
PCK	943
GBK	1386
koi8-r	878

 Sağlanan dönüştürmelerin tam listesi Çizelge 121 sayfa 935’inde gösterilir.

Ulusal diller

Bu bilgiler, IBM MQ tarafından desteklenen dilleri içerir.

IBM MQ tarafından desteklenen diller şunlardır:

- US English - see topic [“ABD İngilizcesi” sayfa 912](#)
- Almanca-see topic [“Almanca” sayfa 912](#)
- Danca ve Norveççe-bkz. [“Danimarka ve Norveççe” sayfa 913](#)
- Fince ve İsveççe-bkz. [“Fince ve İsveççe” sayfa 914](#)
- İtalyanca-bkz. [“İtalyanca” sayfa 914](#)
- İspanyolca-bkz. [“İspanyolca” sayfa 915](#)
- İngiliz İngilizcesi/Gaelic-bkz. [“İngiliz İngilizcesi /Gaelc” sayfa 915](#)
- Fransızca-bkz. [“Fransızca” sayfa 916](#)
- Çok dilli-konuya bakın [“Çok dilli” sayfa 917](#)
- Portekizce-bkz. [“Portekizce” sayfa 917](#)
- İzlandaca-konuya bakın [“İzlandaca” sayfa 918](#)
- Doğu Avrupa dilleri-bkz. [“Doğu Avrupa dilleri” sayfa 919](#)
- Kiril-konuya bakın [“Kiril” sayfa 920](#)
- Estonca-bkz. [“Estonyaca” sayfa 920](#)
- Letonya ve Litvanca-see topic [“Letonca ve Litvanca” sayfa 921](#)
- Ukranian-konuya bakın [“Ukraynaca” sayfa 923](#)
- Yunanca-see topic [“Yunanca” sayfa 923](#)
- Türkçe-bkz. [“Türkçe” sayfa 924](#)
- İbranice-bkz. [“İbranice” sayfa 924](#)
- Farsça-konuya bakın [“Farsça” sayfa 926](#)
- Urdu - see topic [“Urduca” sayfa 926](#)
- Thai - see topic [“Tay dili” sayfa 927](#)
- Lao - see topic [“Lao” sayfa 927](#)
- Vietnam dili-konu başlığı [“Vietnam dili” sayfa 927](#)
- Japonca Latin SBCS-bkz. [“Japonca Latin SBCS” sayfa 927](#)

- Japonca Katakana SBCS-bkz. “Japonca Katakana SBCS” sayfa 929
- Japonca Kanji/Latin Karma-see topic “Japonca Kanji/Latin Karma” sayfa 930
- Japonca Kanji/Katakana Karma-see topic “Japonca Kanji/Katakana Karma” sayfa 932
- Kore dili-bkz. “Korece” sayfa 933
- Basitleştirilmiş Çince-see topic “Yalınlaştırılmış Çince” sayfa 933
- Geleneksel Çince-bkz. “Geleneksel Çince” sayfa 934

ABD İngilizcesi

ABD İngilizcesi için CCSID 'lerin ve CCSID 'lerin ayrıntıları.

Aşağıdaki çizelge, desteklenen platformlarda ABD İngilizcesi için yerel CCSID 'leri göstermektedir:

Altyapı	Yerel CCSID 'ler
IBM i, z/OS	37, 924, 1140
AIX	819, 923, 5348
HP-UX	819, 923, 1051
Windows	437, 850, 1252, 5348, 858
HP Integrity NonStop Server, Solaris, 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

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.

Aşağıdaki tabloda, desteklenen platformlarda Almanca için yerel CCSID 'ler gösterilmektedir:

Altyapı	Yerel CCSID 'ler
IBM i, z/OS	273, 924, 1141
AIX	819, 923, 5348
HP-UX	819, 923, 1051
Windows	437, 850, 858, 1252, 5348
HP Integrity NonStop Server, Solaris, 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:

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.

Aşağıdaki çizelge, desteklenen altyapılarda Danca ve Norveççe için yerel CCSID ' leri göstermektedir:

Altyapı	Yerel CCSID ' ler
IBM i, z/OS	277, 924, 1142
AIX	819, 923, 5348
HP-UX	819, 923, 1051
Windows	850, 858, 865, 1252, 5348
HP Integrity NonStop Server, Solaris, 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

HP-UX

Kod sayfası:

1051

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.

Aşağıdaki çizelge, desteklenen platformlarda Fince ve İsveççe için yerel CCSID ' leri göstermektedir:

Altyapı	Yerel CCSID ' ler
IBM i, z/OS	278, 924, 1143
AIX	819, 923, 5348
HP-UX	819, 923, 1051
Windows	437, 850, 858, 865, 1252, 5348
HP Integrity NonStop Server, Solaris, 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:

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

HP-UX

Kod sayfası:

1051

865 kod sayfasına dönüştürülmez

Windows

Kod sayfası:

865

1051, 1275 kod sayfalarına dönüştürülmez

İtalyanca

CCSID ' lerin Ayrıntıları ve İtalyanca için CCSID dönüştürmesi.

Aşağıdaki tabloda, desteklenen platformlarda İtalyanca için yerel CCSID ' ler gösterilmektedir:

Altyapı	Yerel CCSID ' ler
IBM i, z/OS	280, 924, 1144
AIX	819, 923, 5348
HP-UX	819, 923, 1051
Windows	437, 850, 858, 1252, 5348
HP Integrity NonStop Server, Solaris, 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:

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.

Aşağıdaki çizelge, desteklenen platformlarda İspanyolca için yerel CCSID ' leri göstermektedir:

Altyapı	Yerel CCSID ' ler
IBM i, z/OS	284, 924, 1145
AIX	819, 923, 5348
HP-UX	819, 923, 1051
Windows	437, 850, 858, 1252, 5348
HP Integrity NonStop Server, Solaris, 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:

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.

Aşağıdaki çizelge, desteklenen platformlarda İngiltere İngilizcesi/Gaelic için yerel CCSID ' leri göstermektedir:

Altyapı	Yerel CCSID ' ler
IBM i, z/OS	285, 924, 1146
AIX	819, 923, 5348
HP-UX	819, 923, 1051
Windows	437, 850, 858, 1252, 5348
HP Integrity NonStop Server, Solaris, 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.

Aşağıdaki çizelge, desteklenen altyapılarda Fransızca için yerel CCSID ' leri göstermektedir:

Altyapı	Yerel CCSID ' ler
IBM i, z/OS	297, 924, 1147
AIX	819, 923, 5348
HP-UX	819, 923, 1051
Windows	437, 850, 858, 1252, 5348
HP Integrity NonStop Server, Solaris, 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:

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.

Aşağıdaki tabloda, desteklenen platformlarda çok dilli dönüştürmeye ilişkin yerel CCSID ' ler gösterilmektedir:

Altyapı	Yerel CCSID ' ler
IBM i, z/OS	500, 924, 1148
AIX	819, 923, 5348
HP-UX	819, 923, 1051
Windows	437, 850, 858, 1252, 5348
HP Integrity NonStop Server, SINIX, Solaris, 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:

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.

Aşağıdaki çizelge, desteklenen altyapılarda Portekizce 'ye ilişkin yerel CCSID' leri göstermektedir:

Altyapı	Yerel CCSID ' ler
IBM i	37, 500, 924, 1140
z/OS	500, 924, 1140
AIX	819, 923, 5348
HP-UX	819, 923, 1051
Windows	850, 858, 860, 1252, 5348
HP Integrity NonStop Server, Solaris, 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

HP-UX

Kod sayfası:

1051

860 kod sayfasına dönüştürülmez

Windows

Kod sayfası:

860

1051, 1275 kod sayfalarına dönüştürülmez

İzlandaca

İzlandaca için CCSID ' lerin ve CCSID dönüştürmenin ayrıntıları.

Aşağıdaki çizelge, İzlandaca ilişkin yerel CCSID ' leri desteklenen platformlarda göstermektedir:

Altyapı	Yerel CCSID ' ler
IBM i, z/OS	871, 924, 1149
AIX	819, 923, 5348
HP-UX	819, 923, 1051
Windows	850, 858, 861, 1252, 5348
HP Integrity NonStop Server, Solaris, 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:

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

HP-UX

Kod sayfası:

1051

861 kod sayfasına dönüştürülmez

Windows

Kod sayfası:

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.

Aşağıdaki tabloda, desteklenen platformlarda Doğu Avrupa dillerine ilişkin yerel CCSID ' ler gösterilmektedir:

Altyapı	Yerel CCSID ' ler
IBM i, z/OS	870, 1153
Windows	852, 1250, 5346, 9044
AIX, HP-UX, HP Integrity NonStop Server, Solaris, Linux	912
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 dönüştürmeyi destekler.

z/OS

Kod sayfası:

870

1284, 1285 kod sayfalarına dönüştürülmez

1153

1250, 1284, 1285 kod sayfalarına dönüştürülmez

IBM i

Kod sayfası:

870

1284, 1285, 5346, 9044 kod sayfalarına dönüştürülmez

1153

1282, 1284, 1285, 5346, 9044 kod sayfalarına dönüştürülmez

HP-UX, Solaris, Linux

Kod sayfası:

912

1284, 1285 kod sayfalarına dönüştürülmez

HP Integrity NonStop Server

Kod sayfası:

912

1153, 1284, 1285, 9044 kod sayfalarına dönüştürülmez

Windows

Kod sayfası:

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.

Aşağıdaki çizelge, desteklenen altyapılarda Kiril CCSID için yerel CCSID ' leri göstermektedir:

Altyapı	Yerel CCSID ' ler
z/OS	1025
IBM i	880, 1025
Windows	855, 866, 1131, 1251, 5347
Solaris	878, 915
AIX, HP-UX, Linux , HP Integrity NonStop Server	915
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.

Aşağıdaki çizelge, desteklenen altyapılarda Estonya 'ya ilişkin yerel CCSID ' leri göstermektedir:

Altyapı	Yerel CCSID ' ler
IBM i, z/OS	1122, 1157

Altyapı	Yerel CCSID ' ler
Windows	902, 922, 1257, 5353, 9449
AIX, HP-UX, Solaris, Linux	902, 922
HP Integrity NonStop Server	922

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

HP-UX, Solaris, 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

HP Integrity NonStop Server

Kod sayfası:

922

902, 1157, 9449 kod sayfalarına dönüştürülmez

Letonca ve Litvanca

Letonya ve Litvanca için CCSID 'lerin ve CCSID' lerin ayrıntıları.

Aşağıdaki çizelge, desteklenen platformlarda Letonya ve Litvanca 'ya ilişkin yerel CCSID ' leri göstermektedir:

Altyapı	Yerel CCSID ' ler
IBM i, z/OS	1112, 1156
Windows	901, 921, 1257, 5353, 9449
AIX, HP-UX, Solaris, Linux	901, 921
HP Integrity NonStop Server,	921

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

HP-UX, Solaris, 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

HP Integrity NonStop Server

Kod sayfası:

921

901, 1156, 9449 kod sayfalarına dönüştürülmez

Ukraynaca

Ukraynaca için CCSID 'lerin ve CCSID' lerin ayrıntıları.

Aşağıdaki çizelge, desteklenen altyapılarda Ukraynca için yerel CCSID ' leri göstermektedir:

Altyapı	Yerel CCSID ' ler
IBM i, z/OS	1123
Windows	1124, 1125, 1251, 5347
AIX, HP-UX, HP Integrity NonStop Server, Solaris, 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

HP-UX

Kod sayfası:

1124

5347 kod sayfasına dönüştürülmez

Windows

Kod sayfası:

1125

1123 kod sayfasına dönüştürülmez

Yunanca

GCCSID ' lerin ayrıntıları ve Yunanca için CCSID dönüştürmesi.

Aşağıdaki tabloda, desteklenen platformlarda Yunanca için yerel CCSID ' ler gösterilmektedir:

Altyapı	Yerel CCSID ' ler
IBM i, z/OS	875
HP-UX	813 (nota bakın)
Windows	869, 1253, 5349
AIX, NCR, HP Integrity NonStop Server, Solaris, Linux	813
Apple istemcisi	1280
DOS istemcisi	737

Not: HP-UX üzerinde yalnızca ISO kod kümesi desteklenmektedir. HP-UX patentli greek8 kod kümesinin kayıtlı bir CCSID 'si yok ve bu kod desteklenmiyor.

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.

Aşağıdaki tabloda desteklenen platformlarda Türkçe için yerel CCSID ' ler gösterilmektedir:

Altyapı	Yerel CCSID ' ler
IBM i, z/OS	1026
HP-UX	920 (nota bakın)
Windows	857, 1254, 5350
AIX, HP Integrity NonStop Server, Solaris, Linux	920
Apple istemcisi	1281

Not: HP-UX üzerinde yalnızca ISO kod kümesi desteklenmektedir. HP-UX patentli turkish8 kod kümesinin kayıtlı bir CCSID ' si yok ve bu kod desteklenmiyor.

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.

Aşağıdaki çizelge, desteklenen altyapılarda İbranice için yerel CCSID ' leri göstermektedir:

Altyapı	Yerel CCSID ' ler
z/OS	424, 803, 4899, 12712
IBM i	424
AIX	916, 9048
HP-UX	916 (nota bakın)
Windows	1255, 5351
HP Integrity NonStop Server, Solaris, Linux	916

Not: HP-UX üzerinde yalnızca ISO kod kümesi desteklenmektedir. HP-UX patentli greek8 kod kümesinin kayıtlı bir CCSID ' si yok ve bu kod desteklenmiyor.

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

Aşağıdaki çizelge, desteklenen altyapılarda Arapça için yerel CCSID ' leri göstermektedir:

Altyapı	Yerel CCSID ' ler
IBM i, z/OS	420
AIX	1046, 1089
HP-UX	1089 (nota bakın)
Windows	720, 864, 1256, 5352
HP Integrity NonStop Server, Solaris, Linux	1089

Not: HP-UX üzerinde yalnızca ISO kod kümesi desteklenmektedir. HP-UX patentli arabic8 kod kümesinin kayıtlı bir CCSID 'si yok ve bu kod desteklenmiyor.

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

HP-UX, Solaris, Linux, HP Integrity NonStop Server, 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.

Aşağıdaki çizelge, desteklenen altyapılarda Farsça için yerel CCSID 'leri göstermektedir:

Altyapı	Yerel CCSID ' ler
IBM i, z/OS	1097
AIX, HP-UX, HP Integrity NonStop Server, Solaris, Linux Windows	1098 (nota bakın)

Not: Bu altyapılar için yerel CCSID değeri standartlaştırılmamış ve değiştirilebilir.

Tüm platformlar, yerel CCSID 'leri ile diğer platformların yerel CCSID' leri arasındaki dönüştürmeyi destekler.

Urduca

Urdu 'ya ilişkin CCSID 'lerin ve CCSID' lerin ayrıntıları.

Aşağıdaki çizelge, desteklenen altyapılarda Urduca için yerel CCSID 'leri göstermektedir:

Altyapı	Yerel CCSID ' ler
IBM i, z/OS	918
Windows	868
AIX, HP-UX, HP Integrity NonStop Server, Solaris, Linux	1006

Tüm platformlar, yerel CCSID 'leri ile diğer platformların yerel CCSID' leri arasında, aşağıdaki istisnalar dışında dönüştürmeyi destekler.

IBM i

Kod sayfası:

918

1006 kod sayfasına dönüştürülmez

Tay dili

Thai için CCSID ' lerin ve CCSID dönüştürmenin ayrıntıları.

Aşağıdaki çizelge, desteklenen platformlarda Thai dilinde yerel CCSID ' leri göstermektedir:

Altyapı	Yerel CCSID ' ler
IBM i, z/OS	838
AIX, HP-UX, HP Integrity NonStop Server, Solaris, Linux Windows	874 (nota bakın)

Not: Bu altyapılar için yerel CCSID değeri standartlaştırılmamış ve değiştirilebilir.

Tüm platformlar, yerel CCSID 'leri ile diğer platformların yerel CCSID' leri arasındaki dönüştürmeyi destekler.

Lao

Lao için CCSID 'lerin ve CCSID' lerin ayrıntıları.

Aşağıdaki çizelge, desteklenen altyapılarda Lao için yerel CCSID ' leri göstermektedir:

Altyapı	Yerel CCSID ' ler
IBM i, z/OS	1132
AIX, HP-UX, HP Integrity NonStop Server, Solaris, Linux Windows	1133

Tüm platformlar, yerel CCSID 'leri ile diğer platformların yerel CCSID' leri arasındaki dönüştürmeyi destekler.

Vietnam dili

Vietnamca için CCSID ' lerin ayrıntıları ve CCSID dönüştürmesi.

Aşağıdaki tabloda, desteklenen platformlarda Vietnamlılar için yerel CCSID ' ler gösterilmektedir:

Altyapı	Yerel CCSID ' ler
IBM i, z/OS	1130
Windows	1258, 5354
AIX, HP-UX, HP Integrity NonStop Server, Solaris, Linux	1129

Tüm platformlar, yerel CCSID 'leri ile diğer platformların yerel CCSID' leri arasında, aşağıdaki istisnalar dışında dönüştürmeyi destekler.

IBM i

Kod sayfası:

1130

1129, 5354 kod sayfalarına dönüştürülmez

Japonca Latin SBCS

Japonca Latin SBCS için CCSID ' lerin ayrıntıları ve CCSID dönüştürmesi.

Aşağıdaki çizelgede, desteklenen platformlarda Japonca Latin SBCS için yerel CCSID ' ler gösterilmektedir:

Altyapı	Yerel CCSID ' ler
IBM i, z/OS	1027
AIX	932, 5050, 33722 (bkz. Not 1)
Windows	932, 943 (bkz. Not 2 ve 3)
Linux, HP Integrity NonStop Server, Solaris	943, 5050
HP-UX	Bilinmiyor

Not:

1. 5050 ve 33722, AIXüzerinde 954 temel kod sayfasıyla ilgili CCSID'lerdir. İşletim sistemi tarafından bildirilen CCSID değeri 33722 'dir.
2. 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.

3. IBM MQ , JIS X 0213 (JIS2004) standardına dayalı kod sayfalarını desteklemez.

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

Solaris

Kod sayfası:

943

1027 kod sayfasına dönüştürülmez

5050

1027 kod sayfasına dönüştürülmez

HP Integrity NonStop Server

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.

Aşağıdaki çizelge, desteklenen altyapılarda Japonca (Katakana SBCS) yerel CCSID ' leri (CCSID) göstermektedir:

Altyapı	Yerel CCSID ' ler
IBM i, z/OS	290
HP-UX	897
AIX	932, 5050, 33722 (bkz. Not 1)
Windows	932, 943 (bkz. Not 2 ve 3)
Linux, HP Integrity NonStop Server, Solaris	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.
- IBM MQ , JIS X 0213 (JIS2004) standardına dayalı kod sayfalarını desteklemez.
- Önceki dönüştürmelerin yanı sıra, AIX, HP-UX, Solaris, Linux ve Tru64 üzerindeki IBM MQ ürünleri, CCSID 897 'den CCSID ' ye 37, 273, 277, 278, 280, 284, 285, 290, 297, 437, 500, 819, 850, 1027 ve 1252 'ye dönüştürülmesini destekler.

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

Kod sayfası:

290

932 kod sayfasına dönüştürülmez

AIX

Kod sayfası:

932

290, 897 kod sayfalarına dönüştürülmez

5050

290, 897 kod sayfalarına dönüştürülmez

33722

290, 897 kod sayfalarına dönüştürülmez

HP-UX

Kod sayfası:

897

932, 943, 954, 5050, 33722 kod sayfalarına dönüştürülmez

Linux

Kod sayfası:

943

290, 897 kod sayfalarına dönüştürülmez

5050

290, 897 kod sayfalarına dönüştürülmez

Solaris

Kod sayfası:

943

290, 897 kod sayfalarına dönüştürülmez

5050

290, 897 kod sayfalarına dönüştürülmez

HP Integrity NonStop Server

Kod sayfası:

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.

Aşağıdaki çizelgede, desteklenen platformlarda Japonca Kanji/Latin Karma için yerel CCSID ' ler gösterilmektedir:

Altyapı	Yerel CCSID ' ler
IBM i, z/OS	1399, 5035 (bkz. Not 1)

Altyapı	Yerel CCSID ' ler
AIX	932, 5050, 33722 (bkz. Not 2)
HP-UX	932, 954, 5039 (bkz. Not 3)
Windows	932, 943 (bkz. Not 4 ve 5)
Linux, HP Integrity NonStop Server, Solaris	943, 5050

Not:

1. 5035, 939 kod sayfasıyla ilgili bir CCSID ' dir
2. 5050 ve 33722, AIXüzerinde 954 temel kod sayfasıyla ilgili CCSID'lerdir. İşletim sistemi tarafından bildirilen CCSID değeri 33722 'dir.
3. Code sets japan15 and SJIS on HP-UX are represented by CCSID 932. Bu karakterler, HP-UX sisteminde dönüştürme gerçekleştirilmezse, SJIS 932 'de farklı gösterimler içeren birkaç DBCS karakterinin yanlış dönüştürülebileceği bir durum ortaya çıkabilir. IBM MQ for HP-UX , HP SJIS için doğru CCSID ' yi (5039) destekler. 932 'den 5039 'a kullanılan CCSID ' yi değiştirmek için /var/mqm/conv/ccsid.tb1 kütüğüne yapılan bir değişiklik yapılabilir.
4. 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.tb1 kütüğüne yapılan bir değişiklik yapılabilir.
5. IBM MQ , JIS X 0213 (JIS2004) standardına dayalı kod sayfalarını desteklemez.

Tüm platformlar, yerel CCSID 'leri ile diğer platformların yerel CCSID' leri arasında, aşağıdaki istisnalar dışında dönüştürmeyi destekler.

z/OS

Kod sayfası:

1399

954, 5035, 5050, 33722 kod sayfalarına dönüştürülmez

5035

954, 1399, 5050, 33722 kod sayfalarına dönüştürülmez

IBM i

Kod sayfası:

1399

5039 kod sayfasına dönüştürülmez

5035

5039 kod sayfasına dönüştürülmez

HP-UX

Kod sayfası:

932

942, 943, 1399 kod sayfalarına dönüştürülmez

954

942, 943, 1399 kod sayfalarına dönüştürülmez

5039

942, 943, 1399 kod sayfalarına dönüştürülmez

HP Integrity NonStop Server

Kod sayfası:

943

1399 kod sayfasına dönüştürülmez

5050

1399 kod sayfasına dönüştürülmez

Japonca Kanji/Katakana Karma

Japonca Kanji/Katakana Mixed için CCSID ' lerin ve CCSID dönüştürmenin ayrıntıları.

Aşağıdaki çizelgede, desteklenen platformlarda Japonca Kanji/Katakana Mixed için yerel CCSID ' ler gösterilmektedir:

Altyapı	Yerel CCSID ' ler
z/OS	1390, 5026 (bkz. Not 1)
IBM i	5026 (bkz. Not 1)
AIX	932, 5050, 33722 (bkz. Not 2)
HP-UX	932, 954, 5039 (bkz. Not 3)
Windows	932, 943 (bkz. Not 4 ve 5)
Linux, HP Integrity NonStop Server, Solaris	943, 5050

Not:

1. CCSID 1390, küçük harf karakterleri kabul etmez. 5026, 930 kod sayfasıyla ilgili bir CCSID ' dir. CCSID 5026, Japonca Katakana (DBCS) özelliği seçildiğinde IBM i üzerinde bildirilen CCSID ' dir.
2. 5050 ve 33722, AIX üzerinde 954 temel kod sayfasıyla ilgili CCSID'lerdir. İşletim sistemi tarafından bildirilen CCSID değeri 33722 'dir.
3. Code sets japan15 and SJIS on HP-UX are represented by CCSID 932. Bu karakterler, HP-UX sisteminde dönüştürme gerçekleştirilmezse, SJIS 932 'de farklı gösterimler içeren birkaç DBCS karakterinin yanlış dönüştürülebileceği bir durum ortaya çıkabilir. IBM MQ for HP-UX , HP SJIS için doğru CCSID ' yi (5039) destekler. 932 'den 5039 'a kullanılan CCSID ' yi değiştirmek için /var/mqm/conv/ccsid.tbl kütüğüne yapılan bir değişiklik yapılabilir.
4. 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 üzerinde, 932 kod sayfasını temsil etmek için CCSID 932 kullanılır, ancak 943 'te kullanılan CCSID ' yi değiştiren . ./conv/table/ccsid.tbl kütüğüne yapılan bir değişiklik yapılabilir.
5. IBM MQ , JIS X 0213 (JIS2004) standardına dayalı kod sayfalarını desteklemez.

Tüm platformlar, yerel CCSID 'leri ile diğer platformların yerel CCSID' leri arasında, aşağıdaki istisnalar dışında dönüştürmeyi destekler.

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

Kod sayfası:

5026

1390, 5039 kod sayfalarına d " nYtrme

HP-UX

Kod sayfası:

932

942, 943, 1390 kod sayfalarına dönüştürülmez

954

942, 943, 1390 kod sayfalarına dönüştürülmez

5039

942, 943, 1390 kod sayfalarına dönüştürülmez

HP Integrity NonStop Server

Kod sayfası:

943

1390 kod sayfasına dönüştürülmez

5050

1390 kod sayfasına dönüştürülmez

Korece

Korece için CCSID ' lerin ve CCSID dönüştürmenin ayrıntıları.

Aşağıdaki çizelge, desteklenen platformlarda Korece ilişkin yerel CCSID ' leri göstermektedir:

Altyapı	Yerel CCSID ' ler
z/OS, IBM i	933, 1364
AIX, HP-UX, Linux, HP Integrity NonStop Server, Solaris	970
Windows	949, 1363

Tüm platformlar, yerel CCSID 'leri ile diğer platformların yerel CCSID' leri arasında, aşağıdaki istisnalar dışında dönüştürmeyi destekler.

z/OS

Kod sayfası:

933

970 kod sayfasına dönüştürülmez

1364

970 kod sayfasına dönüştürülmez

HP-UX

Kod sayfası:

970

949, 1363, 1364 kod sayfalarına dönüştürülmez

Yalınlaştırılmış Çince

Basitleştirilmiş Çince için CCSID ' lerin ayrıntıları ve CCSID dönüştürmesi.

Aşağıdaki çizelge, desteklenen altyapılarda Yalınlaştırılmış Çince için yerel CCSID ' leri göstermektedir:

Altyapı	Yerel CCSID ' ler
z/OS	935, 1388
IBM i	935, 1388
AIX	1383, 1386
HP-UX	1381 (bkz. Not 1)
Windows	1381, 1386 (bkz. Not 2)
Linux, HP Integrity NonStop Server, Solaris	1383

Not:

1. Code sets prc15 and hp15CN on HP-UX are represented by CCSID 1381.
2. 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.

3. IBM MQ , Çince GB18030 standardından birinin birinci aşamasını destekler.

z/OS, Linux, Windows ve Solaris üzerinde dönüştürme desteği, Unicode (UTF-8 ve UCS-2) ve CCSID 1388 (GB18030 uzantıları içeren EBCDIC), Unicode (UTF-8 ve UCS-2) ve CCSID 5488 (GB18030 birinci aşaması) ve CCSID 1388 ile CCSID 5488 arasında sağlanır.

Not:

IBM işletim sisteminde, işletim sistemi tarafından Unicode (UTF-8 ve UCS-2) ve CCSID 1388 (GB18030 uzantıları ile EBCDIC) arasında dönüştürme için destek sağlanır.

On HP-UX there is currently no support available on the HP11 operating system for GB18030. HP11 üzerinde PHCO_26456 yaması, GB18030 (CCSID 5488) ve Unicode arasında dönüştürme desteği sağlar. GB18030 ve 1388 (EBCDIC) arasında dönüştürme için destek sağlanmaz.

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

HP-UX

Kod sayfası:

1381

1383, 1386, 1388 kod sayfalarına dönüştürülmez

Geleneksel Çince

Geleneksel Çince için CCSID ' lerin ayrıntıları ve CCSID dönüştürmesi.

Aşağıdaki çizelgede, desteklenen platformlarda Geleneksel Çince için yerel CCSID ' ler gösterilmektedir:

Altyapı	Yerel CCSID ' ler
z/OS, IBM i	937
HP-UX	938, 950, 964 (bkz. Not)
Windows	950
AIX, HP Integrity NonStop Server, Solaris, Linux	950, 964

Not: Code set roc15 on HP-UX is represented by CCSID 938.

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

HP-UX

Kod sayfası:

938

948 kod sayfasına dönüştürülmez

950

948 kod sayfasına dönüştürülmez

964

948 kod sayfasına dönüştürülmez

Linux, Solaris

Kod sayfası:

964

938 kod sayfasına dönüştürülmez

z/OS dönüştürme desteği

Desteklenen CCSID dönüştürmelerinin bir listesi.

Çizelge 121. 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 121. 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 121. 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

Çizelge 121. IBM MQ for z/OS CCSID dönüştürme desteği (devamı var)

CCSID	CCSID 'lere ve CCSID' ye dönüştürür
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 121. 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

Çizelge 121. IBM MQ for z/OS CCSID dönüştürme desteği (devamı var)

CCSID	CCSID 'lere ve CCSID' ye dönüştürür
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 121. 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

Çizelge 121. IBM MQ for z/OS CCSID dönüştürme desteği (devamı var)

CCSID	CCSID 'lere ve CCSID' ye dönüştürür
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

Çizelge 121. IBM MQ for z/OS CCSID dönüştürme desteği (devamı var)

CCSID	CCSID 'lere ve CCSID' ye dönüştürür
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

Çizelge 121. IBM MQ for z/OS CCSID dönüştürme desteği (devamı var)	
CCSID	CCSID 'lere ve CCSID' ye dönüştürür
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

Çizelge 121. IBM MQ for z/OS CCSID dönüştürme desteği (devamı var)

CCSID	CCSID 'lere ve CCSID' ye dönüştürür
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

Çizelge 121. IBM MQ for z/OS CCSID dönüştürme desteği (devamı var)

CCSID	CCSID 'lere ve CCSID' ye dönüştürür
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

Çizelge 121. IBM MQ for z/OS CCSID dönüştürme desteği (devamı var)

CCSID	CCSID 'lere ve CCSID' ye dönüştürür
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

Çizelge 121. IBM MQ for z/OS CCSID dönüştürme desteği (devamı var)	
CCSID	CCSID 'lere ve CCSID' ye dönüştürür
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

Çizelge 121. IBM MQ for z/OS CCSID dönüştürme desteği (devamı var)

CCSID	CCSID 'lere ve CCSID' ye dönüştürür
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, 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, 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

Çizelge 121. IBM MQ for z/OS CCSID dönüştürme desteği (devamı var)	
CCSID	CCSID 'lere ve CCSID' ye dönüştürür
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
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

Çizelge 121. IBM MQ for z/OS CCSID dönüştürme desteği (devamı var)

CCSID	CCSID 'lere ve CCSID' ye dönüştürür
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 121. 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 121. 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

Çizelge 121. IBM MQ for z/OS CCSID dönüştürme desteği (devamı var)

CCSID	CCSID 'lere ve CCSID' ye dönüştürür
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 121. 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

Çizelge 121. IBM MQ for z/OS CCSID dönüştürme desteği (devamı var)

CCSID	CCSID 'lere ve CCSID' ye dönüştürür
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 121. 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

Çizelge 121. IBM MQ for z/OS CCSID dönüştürme desteği (devamı var)

CCSID	CCSID 'lere ve CCSID' ye dönüştürür
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 121. 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 dönüştürme desteği

CCSID 'lerin tam listesi ve IBM itarafından desteklenen dönüştürmeler, uygun IBM i yayınında bulunabilir.

Desteklenen kod sayfaları, [Desteklenen CCSID eşlemeler](#) listesinde yer alıyor.

Unicode dönüştürme desteği

Bazı platformlar, kullanıcı verilerinin Unicode kodlamasına ya da Unicode kodlamasına dönüştürülmesini destekler. Desteklenen Unicode kodlamasının iki biçimi şunlardır: UCS-2 (CCSID 1200, 13488 ve 17584) ve UTF-8 (CCSID 1208).

UCS-2 terimi genellikle değişimli olarak kullanılır, ancak UTF-16 ile yanlış kullanılırdır. UCS-2, her karakterin 2 baytı kapladığı, sabit genişlikli bir kodlamadır. UTF-16 is variable-width encoding that is a

superset of UCS-2. 2 baytlık UCS-2 karakterlerine ek olarak, UTF-16 , 4 bayt uzunluğundaki taşıyıcı çiftler olarak bilinen karakterler içerir. IBM MQ , vekil çiftleri desteklemez. Bu nedenle, IBM MQ içindeki UTF-16 ve UTF-8 desteği, bu nedenle UCS-2 içinde kodlanabilen Unicode karakterlerle sınırlıdır.

Not: IBM MQ does not support UCS-2 queue manager CCSIDs so message header data cannot be encoded in UCS-2.

Unicode için IBM MQ AIX desteği

On IBM MQ for AIX conversion to and from Unicode CCSIDs is supported for the CCSIDs in the following table.

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	1200
1153	1156	1157	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

Unicode için IBM MQ HP-UX desteği

IBM MQ for HP-UX 'de aşağıdaki çizelgede listelenen CCSID' ler için Unicode CCSID 'lere ve Unicode CCSID' lere dönüştürme desteklenir.

437	737	813	819	850	852
855	857	861	864	865	866
869	874	912	915	916	920
932	938	950	954	964	970
1051	1089	1140	1141	1142	1143
1144	1145	1146	1147	1148	1149
1200	1208	1250	1251	1252	1253
1254	1255	1256	1257	1258	1381
5050	5488	13488	33722		

Unicode için IBM MQ for Windows, Solaris ve Linux desteği

On IBM MQ for Windows, IBM MQ for Solaris, and IBM MQ for Linux conversion to, and from, Unicode CCSIDs is supported for the CCSIDs in the following table.

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 (5)	915	916	918
920	921	922	923	924	927
928	930	931 (1)	932 (2)	933	935
937	938 (3)	939	941	942	943
947	948	949	950	951	954 (4)
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	1380	1381
1383	1386	1388	4899	5050	5346
5347	5348	5349	5350	5351	5352
5353	5354	5488 (5)	9044	9048	9449
12712	13488	17584	33722 (4)		

Notlar:

1. 931, dönüştürme için 939 kullanır.
2. 932, dönüştürme için 942 kullanır.
3. 938, dönüştürme için 948 'i kullanır.
4. 954 ve 33722, dönüştürme için 5050 kullanır.
5. Yalnızca Windows, Linux ve Solaris üzerinde.

Unicode için IBM i desteği

UNICODE desteği ile ilgili ayrıntılar için, işletim sisteminizle ilgili IBM i yayınına bakın.

Unicode için IBM MQ for z/OS desteği

On IBM MQ for z/OS conversion to and from the Unicode CCSIDs is supported for the following CCSIDs:

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	1162
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	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 değiştirmez ve hem 32 bit, hem de 64 bit IBM MQ altyapılarında kullanılabilir:

Ad	Uzunluk
MQLONG	4 Byte
MQULONG	4 Byte
MQINT32	4 Byte
MQUINT32	4 Byte
MQINT64	8 Byte
MQUINT64	8 Byte

Standart veri tipleri

32 bit UNIX, 64 bit UNIX ve 64 bit Windows uygulamalarındaki standart veri tipleri hakkında bilgi edinin.

32 bit UNIX uygulamaları

Bu bölüm karşılaştırmaya dahil edilir ve Solaris' a dayalıdır. Diğer UNIX platformları ile ilgili farklar belirtilmektedir:

Ad	Uzunluk
DAMGA	1 Byte
kısa	2 Byte
int	4 Byte
uzun	4 Byte
kayan nokta	4 Byte
çift	8 Byte
uzun çift	16 Byte

AIX ve Linux PPC üzerinde uzun bir çift 8 byte olduğunu unutmayın.

işaretçi	4 Byte
ptrdiff_t	4 Byte
size_t	4 Byte
tima_t	4 Byte
saat_t	4 Byte

Ad	Uzunluk
wchar_t	4 Byte
	AIX a wchar_t için 2 byte olduğunu unutmayın.

64 bit UNIX uygulamaları

Bu bölüm, Solaristemel alınarak ifade edilir. Diğer UNIX platformları ile ilgili farklar belirtilmektedir:

Ad	Uzunluk
DAMGA	1 Byte
kısa	2 Byte
int	4 Byte
uzun	8 Byte
kayan nokta	4 Byte
çift	8 Byte
uzun çift	16 Byte
	AIX ve Linux PPC üzerinde uzun bir çift 8 byte olduğunu unutmayın.
işaretçi	8 Byte
ptrdiff_t	8 Byte
size_t	8 Byte
tima_t	8 Byte
saat_t	8 Byte
	Diğer UNIX platformları a clock_t ise 4 byte olduğunu unutmayın.
wchar_t	4 Byte
	AIX a wchar_t için 2 byte olduğunu unutmayın.

Windows 64 bit uygulamaları

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
işaretçi	8 Byte
	Tüm işaretçilerin 8 bayt olduğunu unutmayın.
ptrdiff_t	8 Byte
size_t	8 Byte

Ad	Uzunluk
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

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);
```

%ld , 4 baytlık bir tip için 8 byte 'lık bir tip yerleştirmeyi dener; yalnızca gerçek bir long veri tipi ile ilgileniyorsanız %l 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 i için uygulama geliştirmenize yardımcı olması için aşağıdaki konuları kullanın

- “Veri tipi açıklamaları” sayfa 967
- “İşlev çağrılar” sayfa 1213
- “Nesnelerin öznitelikleri” sayfa 1331
- “Uygulamalar” sayfa 1375
- “IBM i için dönüş kodları (ILE RPG)” sayfa 1388
- “IBM i için MQI seçeneklerinin geçerliliğini denetleme kuralları (ILE RPG)” sayfa 1389
- “Machine encodings on IBM i” sayfa 1392
- “Rapor seçenekleri ve ileti işaretleri” sayfa 1395

IBM üzerindeki RPG uygulamaları için uyumluluk kipi

Bu işlem kipi, MQSeries Sürüm 5.1 öncesi yazılmış uygulamalar için gereklidir. Ürünün sonraki sürümleri, derlemek için gerekli olan telif kitapları IBM WebSphere MQ 6.0' ta kaldırılrsa da, bu uygulamalar için uyumlu bir yürütme ortamı ortamı sağlar. Although dynamic linkage (compatibility mode) is provided in library QMQM for versions of the product after MQSeries Version 5.1, applications that use this compatibility mode of operation, need to be recompiled to use the static bound MQ calls that are provided by the LIBMQM and LIBMQM_R service programs, instead.

İlgili bilgiler

[Uygulamaların geliştirilmesi](#)

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 yaratınız:

- 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 i nesneleri için 'FIELD REFERENCE FILE' olarak kullanılabilmesinin sağlandığı. Bir IBM MQ veri tipi tanımlaması değişirse, bu nesneleri 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.

Çizelge 122. Temel veri tipleri

Veri türü	Gösterim
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
MQPID	İşlem tanıtıcısı
MQPTR	İşaretçi
MQTID	İş parçacığı tanıtıcısı
MQUINT8	8 bitlik imzalanmamış tamsayı
MQUINT16	16 bit işaretsiz tamsayı

Çizelge 122. Temel veri tipleri (devamı var)

Veri türü	Gösterim
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	MQDMMPO tipinde bir veri yapısına işaret eden gösterge
PMQEPH	MQEPH tipinde bir veri yapısına işaretçi
PMQFLOAT32	MQFLOAT32tipi veriye ilişkin gösterge
PMQFLOAT64	MQFLOAT64tipi veriye ilişkin gösterge
PMQFUNC	İşleve ilişkin gösterge
PMQGMO	MQGMO tipinde bir veri yapısına ilişkin gösterge

Çizelge 122. Temel veri tipleri (devamı var)

Veri türü	Gösterim
PMQHCONFIG	MQHCONFIG tipli verilere g " sterger
PMQHCONN	MQHCONN tipinde veri işaretçisi
PMQHMSG	MQHMSG tipli verilere g " sterger
PMQHOBJS	MQHOBJS 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	İmlecin 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
PMQMHBO	MQMHBO 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 tanıtcısı 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
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

Çizelge 122. Temel veri tipleri (devamı var)	
Veri türü	Gösterim
PMQTID	Bir iş parçacığı tanıtıcısı MQTID ' ye işaretçi
PMQTM	MQTM tipinde bir veri yapısına işaret eden gösterge
PMQTC2	MQTMC2tipinde 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

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.

MQBYTE-Bayt

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 nitesiyle uyumlu olmalıdır.

MQBYTEN-n byte dizgisi

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 deşîřmezler kullanılabilir.

MQCHAR-karakteri

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.

MQCHARn-n karakter dizgisi

Her bir MQCHARn veri tipi, bir *n* karakteri dizesini 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.

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.

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.

MQFLOAT32

MQFLOAT32 veri tipi, standart IEEE kayar noktalı biçim kullanılarak gösterilen, 32 bitlik bir kayan noktalı sayıdır.

Bir MQFLOAT32 , 4 baytlık bir sınır üzerinde hizalanmalıdır.

MQFLOAT64

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-yapılandırma tanıtıcısı

MQHCONFIG veri tipi, belirli bir kurulabilir hizmet için yapılandırılmakta olan bir yapılandırma tanıtıcısını (yani, bileşen) gösterir. Bir yapılandırma tanıtıcısı, doğal sınırı üzerinde hizalanmalıdır.

Not: Uygulamalar, yalnızca eşitlik için bu tipteki değişkenleri test etmelidir.

MQHCONN-Bağlantı tanıtıcısı

MQHCONN veri tipi bir bağlantı tanıtıcısını, yani belirli bir kuyruk yöneticisiyle olan bağlantıyı gösterir.

Bir bağlantı tanıtıcısı, doğal sınırı üzerinde hizalanmalıdır.

Not: Uygulamalar, yalnızca eşitlik için bu tipteki değişkenleri test etmelidir.

MQHMSG-İleti tanıtıcısı

MQHMSG veri tipi, bir iletiye erişim veren bir ileti tanıtıcısını temsil eder.

Bir ileti tanıtıcısı, 8 byte 'lık bir sınırdan hizalanmalıdır.

Not: Uygulamalar, yalnızca eşitlik için bu tipteki değişkenleri test etmelidir.

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.

Bir MQUINT16 , 2 baytlık bir sınır üzerinde hizalanmalıdır.

MQUINT32 -32 bitlik imzalanmamış tamsayı

MQUINT32 veri tipi, 32 bitlik imzalanmamış bir tamsayıdır. Bu, MQULONG ' un eşdeğeridir.

MQUINT64 -64 bit işaretsiz 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 sınırlandırılır. Bir MQUINT64 , 8 baytlık bir sınır üzerinde hizalanmalıdır.

MQULONG-32 bit işaretsiz 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.

PMQGMO-tipi MQGMO tipinde bir veri yapısı göstergesi

MQGMO 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.

PMQHOBJS-İşaretçiyi MQHOBJS 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.

PMQINT32 - MQINT32tipinde veriler arasında yer alan veri tipi

PMQINT32 veri tipi, MQINT32tipindeki verilere ilişkin bir göstergedir. Bu, PMQlong ile eşdeğerdir.

PMQINT64 - MQINT64tipinde veriler arasında yer alan 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.

PMQPMC2 - MQPMC2tipinde bir veri yapısı göstergesi.

MQPMC2tipinde bir veri yapısına ilişkin gösterge.

PMQUINT8 - pointer to data of type MQUINT8

MQUINT8tipine ilişkin bir veri göstergesi.

PMQUINT16 - pointer to data of type MQUINT16

MQUINT16tipindeki verilere ilişkin gösterge.

PMQUINT32 - MQUINT32tipinde veriler arasında yer alan veri tipi

PMQUINT32 veri tipi, MQUINT32tipindeki verilere ilişkin bir göstergedir. Bu, PMQULONG ' un eşdeğeridir.

PMQUINT64 - MQUINT64tipinde veriler arasında yer alan veri tipi

PMQUINT64 veri tipi, MQUINT64tipindeki 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 980
- “Çağrılar” sayfa 982
- “Arama değiştirgeleri” sayfa 982
- “Yapılar” sayfa 982
- “Adlandırılmış Değişmezler” sayfa 982
- “MQI yordamları” sayfa 982
- “Göz önünde bulundurulması gerekenler” sayfa 983
- “Kesinleştirme denetimi” sayfa 983
- “Bağlı çağrılar kodlanması” sayfa 983
- “Notasyonla ilgili kurallar” sayfa 985

Kütüklerin KOPYASI

İleti kuyruk 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 982’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 982’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 123 sayfa 980’de özetlenir.

Çizelge 123. 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
CMQCFBFG	CMQCFBFH	-	PCF bit süzgeç değiştirgesi
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 üstbilgi
CMQCFIGG	CMQCFILH	-	PCF tamsayı listesi değiştirgesi yapısı
CMQCFING	CMQCFINH	-	PCF tamsayı değiştirgesi yapısı

Çizelge 123. 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
CMQCFSG	CMQCFSFH	-	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
CMQCNORG	CMQGNOH	-	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
CMQGMORG	CMQGMOH	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
CMQPMORG	CMQPMOH	CMQPMOR	İleti seçenekleri yapısını koy
CMQPSG	-	-	Yayınlama/abone olma değişmezleri
CMQRFHG	CMQRFHH	-	Kurallar ve biçimlendirme üstbilgisi yapısı
CMQRFH2G	CMQRFH2H	-	Kurallar ve biçimlendirme üstbilgisi 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

Çizelge 123. 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
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 koşut zamanlı 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 bildirimleri DS deyimlerini içermez. Bu, uygulamanın, DS deyimini kodlayarak ve daha sonra bildirimin geri kalan kısmında kopyalamak için /COPY deyimini kullanarak tek bir veri yapısı ya da çoklu geçiş veri yapısı bildirmesine olanak sağlar:

```
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 çağrılar kullanırken, programınızı yaratırken MQI yordamlarına bağlanmanız gerekir. Bu yordamlar, aşağıdaki hizmet programlarından uygun şekilde ihraç edilir:

QMQM/AMQZSTUB

This service program provides compatibility bindings for applications written before MQSeries V5.1 that do not require access to any of the new capabilities provided in version 5.1. The signature of this service program matches that contained in version 4.2.1.

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 çağrılar 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.

MQCMIT ve MQBACK çağrılar, uyumluluk kipinde çalışan ILE RPG programları için kullanılamaz. Bu programlar için COMMIT ve ROLBK işlem kodlarını kullanmanız gerekir.

Önemli:  Uyumluluk kipini kullanmaya devam etmeniz, uygulamanızı yeniden derlemeyi ve bunun yerine statik bağlı çağrılar kullanmayı düşünün. Bkz. [IBM üzerindeki RPG uygulamaları için uyumluluk kipi](#)

Bağlı çağrılar kodlanması

MQI ILE yordamları [Çizelge 124 sayfa 984](#) listesinde yer alıyor.

Çizelge 124. Her hizmet programı tarafından desteklenen ILE RPG bağlı çağrıları

Arama adı	LIBMQM ve LIBMQM_R	AMQZKALINTI	AMBVSTUB	LIBMQIC ve LIBMQIC_R
MQBACK	Y			Y
MQBEGIN	Y			Y
MQCMIT	Y			Y
MQCLOSE	Y	Y		Y
MQCONN	Y	Y		Y
MQCONNX	Y			Y
MQDISC	Y	Y		Y
MQGet	Y	Y		Y
MQINQ	Y	Y		Y
MQOPEN	Y	Y		Y
MQPUT	Y	Y		Y
MQPUT1	Y	Y		Y
MQSET	Y	Y		Y
MQXCNVC	Y		Y	Y

Bu yordamları kullanmak için aşağıdaki işlemleri yapmak gerekir:

1. 'D' belirtimlerinizde dış yordamları tanımlayın. Bunların tümü, adı belirtilen değişmezleri içeren COPY dosya üyesi CMQG içinde kullanılabilir.
2. Yordamlarıyla birlikte yordamı çağırmak 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 çağırmak için, çeşitli değıştirmeleri 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.

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: Data in MQAIR 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 985
- “Başlangıçtaki değerler” sayfa 987
- “RPG bildirimi” sayfa 987

Alanlar

MQAIR yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

AICN (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, kayıttaki 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 yoksaılır.

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 yoksaılır.

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 seçiliyse, *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 tanıtıcı.

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 125. 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 değeri
AILUP	Yok	Boş değeri gösterge ya da boş değeri byte
AILUO	Yok	0
AILUL	Yok	0
AIPW	Yok	Dizgi ya da boşluk boş değeri

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
```

MQBMHO-Arabelleğiyle ileti tanıtıcısı seçenekleri

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 değiştirir.

Karakter kümesi ve kodlama: MQBMHO içindeki veriler, uygulamanın (ENNAT) uygulamanın uygulama ve kodlamasının karakter kümesinde yer almalıdır.

- “Alanlar” sayfa 988
- “Başlangıçtaki değerler” sayfa 989
- “RPG bildirimi” sayfa 989

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 126. MQBMHO 'daki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
BMSID	MMSIDV	'BMHO'
BMVER	BMVER1	1
BMOPT	BYT	0

RPG bildirimi

```
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)
```

MQBO-Başlat seçenekleri

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 989
- “Başlangıçtaki değerler” sayfa 990
- “RPG bildirimi” sayfa 990

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 tanıtcısı.

Değer şu olmalıdır:

BOSDİS

Başlangıç seçenekleri yapısına ilişkin tanıtcı.

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 127. MQBO 'daki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
BOSID	BOSDİS	'B0↯↯'
BOVER	BOVER1	1
BOOPT	BONONE	0

Notlar:

1. ↯ simgesi tek bir boş karakteri temsil eder.

RPG bildirimi

```
D*..1.....2.....3.....4.....5.....6.....7..  
D* MQBO Structure  
D*  
D* Structure identifier  
D BOSID 1 4 INZ('B0↯↯')  
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)
```

MQCBC-Callback bağlamı

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 991
- “Başlangıçtaki değerler” sayfa 995
- “RPG bildirimi” sayfa 996

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 *CBCHOBJ* 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 tanımdır.

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 gmatmbelirtmiyor

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 bildirimin denetime ilişkin bilgileri içerir ve bir iletiyle ilgili ayrıntıları içermez. Bu çağrı tipleri, MQCBD yapısındaki CBDOPT kullanılarak istenir.

CBCLLEN 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 MQCB 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 MQCB 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 çağrılır:

- 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 çağırılmadan 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 geçilir.

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.

CBCLLEN (10 basamaklı işaretli tamsayı)

Bu, iletteki uygulama verilerinin bayt cinsinden uzunluğudur. Değer sıfırsa, iletinin uygulama verisi içermediği anlamına gelir.

CBLLEN 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 CBDMMML '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 CBDMMML '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 kullanılamaması 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 verilmediğini 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.

CBRCRD (10 basamaklı işaretli tamsayı)

CBRCRD , 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 **CBRCRD** field only if the value of the **Reason** field in the Callback Context is RC2545.

On entry to the event handler the value of **CBRCRD** is the number of milliseconds the queue manager is going to wait before making a reconnection attempt. [Çizelge 128 sayfa 994](#) , olay işleyicisinden dönüşte kuyruk yöneticisinin davranışını değiştirmek için ayarlayabileceğiniz değerleri listeler.

Çizelge 128. CBRCRD değerler	
Değer	Tanım
-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.

CBBSTATE (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.

Durum	Kuyruk yöneticisi işlemi	Değişmez değeri
<i>CSNONE</i> Bu neden kodu, ek neden bilgisi olmayan olağan çağrıyı 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 çağrılır. 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 bildirimin koşulu çözümlemek için harekete geçeceği 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>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 çağrılarına teslim edildi. Başka ileti tüketilemez.	4

CBCSID (10 basamaklı işaretli tamsayı)

Geri bildirme bağlamı yapısı- StrucId alanı.

Bu, yapı tanıtıcısıdır; değer şöyle olmalıdır:

CBCSI

Geri çağırma bağlamı yapısına ilişkin tanıtıcı.

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 129. MQCBC ' deki alanların ilk değerleri		
Alan adı	Değişmez adı	Değişmez değeri
<i>CBCSID</i>	CBCSI	'CBC-'

Çizelge 129. MQCBC 'deki alanların ilk değerleri (devamı var)

Alan adı	Değişmezin adı	Değişmezin değeri
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
CBCLEN	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  CBCLEN          61          64I 0 INZ(0)
D*
D* Buffer Length
D  CBCBUFFLEN      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*
```

MQCBD-Callback tanımlayıcısı

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 değiştirmedir.

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 997
- “Başlangıçtaki değerler” sayfa 1001
- “RPG bildirimi” sayfa 1001

Alanlar

MQCBD yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

CBDSCALLBA (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 alanından değiştirilmeden geçirilir.

Bu değer yalnızca, tanımlı bir geri çağırma 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.

CBDSCALLBF (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* 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, RC2486 neden 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* 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 RC2486 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 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.

UNIX sistemleri

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 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.

CBDSCALLBT (10 basamaklı işaretli tamsayı)

Bu, geri bildirme işlevinin tipidir. Değer aşağıdakilerden biri olmalıdır:

CCBTMC

Bu geri çağırıcı 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 çağırıcı 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

Çağırıcı bu değerlerden birini geçmezse, çağrı bir neden kodu RC2483 ile başarısız olur

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri CBTMC 'dir.

CBDMMML (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 [“CBCLEN \(10 basamaklı işaretli tamsayı\)”](#) sayfa 993 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 CDFM ' 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 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.

CBDKF

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, susturma sırasında ileti tüketicilerine bildirim yol açabilmek için GMFIQ değerini belirtin.

Denetim seçenekleri: Aşağıdaki seçenekler, tüketici değişikliklerinin durumu değiştiğinde, bir ileti olmadan geri çağırma işlevinin çağrılıp çağrılmayacağını denetler:

CCBDRC

Geri bildirme işlevi CBCTRC çağrısıyla çağrılır

CBDSC

Geri bildirme işlevi, CBCTSC çağrı tipiyle çağrılır.

CBDTC

Geri bildirme işlevi, CBCTTC çağrı tipiyle çağrılır.

CCBDDC

Geri bildirme işlevi CBCTDC (CBCTDC) çağrısıyla çağrılır.

Bu çağrı tipleriyle ilgili daha fazla ayrıntı için bkz. [“CBCCALLT \(10 basamaklı işaretli tamsayı\)” sayfa 991](#).

Varsayılan seçenek: Açıklanılan seçeneklerden herhangi birine gereksiniminiz yoksa, aşağıdaki seçeneği kullanın:

GCBDNA

Başka bir seçenek belirtilmemeyi belirtmek için bu değeri kullanın; tüm seçenekler varsayılan değerlerini kabul eder.

CBDNA, 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 CBDNA ' dur.

CBDSID (10 basamaklı işaretli tamsayı)

Geri çağırma tanımlayıcısı yapısı- StrucId alanı.

Bu, yapı tanıttıcısıdır; değer şöyle olmalıdır:

CBDSI

Geri çağırma tanımlayıcısı yapısına ilişkin tanıttıcı.

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; değer 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 130. MÖCBD içindeki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
StrucId	CBDSI	' CBD↵ '
Version	CBDV1	1
CallbackType	CCBTMC	1
Options	GCBDNO	0
CallbackArea	Yok	Boş bayt
CallbackFunction	Yok	Boş bayt
CallbackName	Yok	Boşluklar
MaxMsgLength	CDFM	-1

Not:

1. ↵ simgesi tek bir boş karakteri temsil eder.

RPG bildirimi

```
D* MÖCBD 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)
```

MÖCHARV-Değişken Uzunluklu Dizgi

Değişken uzunluklu bir dizilimi tanımlamak için MÖCHARV yapısını kullanın.

Genel Bakış

Karakter kümesi ve kodlama: MÖCHARV 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 biriye, kullanılan kodlama, MQCHARV içindeki diğer alanlarla aynı kodlamayla aynı kodda olur. VSCCSID ile tanıtılan 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 1002](#)
- [“Başlangıçtaki değerler” sayfa 1003](#)
- [“RPG bildirimi” sayfa 1003](#)
- [“CSAPL ' nin yeniden tanımlanması” sayfa 1003](#)

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 dizgisiyle 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, dizeyi 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 (UCS-2 CCSID 1200 ve 13488), bu, 2 baytlık Unicode kodlamasıdır; burada boş değer, tüm sıfırların 16 bitlik 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 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.

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 değildir. Bu, bu alanın ilk değeridir.

Başlangıçtaki değerler

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 bildirimi

```
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.

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 *CICSI* 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 132 sayfa 1013](#)' 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 131 sayfa 1004](#) , farklı *CIRET* değerleri için hangi alanların belirlendiğini gösterir.

Çizelge 131. MQCIH yapısındaki hata bilgileri alanlarının içeriği				
<i>CIRET</i>	<i>CIFNC</i>	<i>CICC</i>	<i>CIREA</i>	<i>CIAC</i>
CRC000	-	-	-	-
CRC003	-	-	GB*	-
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 1005
- “Başlangıçtaki değerler” sayfa 1013
- “RPG bildirimi” sayfa 1014

Alanlar

MQCIH 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 kullanılmakta olan *değilse* , 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 kullanılmakta olan *değilse* , 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 131 sayfa 1004](#).

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 LNFACT 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 öznelilikleri.

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 LNFACT 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 131 sayfa 1004](#). *CIFNC* bir IBM MQ arama adı içerdiğinde aşağıdaki değerler kullanılabilir:

CFCONN

MQCONN çağrısı.

CFGET

MQGET çağrısı.

CINGER

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 CUFRST 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* , *CIVER2* değerinden 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 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 yapısını *hariç* iletinin uzunluğudur.

İstek iletilinde 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 131 sayfa 1004](#).

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 131 sayfa 1004](#)). 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 çağırısı 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 kullanılamıyor.

CRC008

(8, X'008 ') Yürürlükteki iş birimi içinde ikinci ya da daha sonraki bir ileti belirlenen süre içinde alınmadı.

CRC009

(9, X'009 ') İşlem kullanılamıyor.

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 Sürüm 1.2 için, bu alan yalnızca bir istek alanıdır; yanıtta değeri tanımsız olur.

CICS Transaction Server Sürüm 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 geriletilecek. 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 gerçekleştirmesi 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*).

GERİ 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

Çizelge 132. MQCIH içindeki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
CISID	CISIDV	'CIH~'
CIVER	CIVER2	2
CILEN	CILEN2	180
CIENC	Yok	0
CICSI	Yok	0
CIFMT	FMNONE	Boşluklar
CIFLG	DOSYA Dışı	0
CIRET	CRC000	0
CICC	CCOK	0
CIREA	YOK	0
CIUOW	SADECE	273
CIGWI	WIDFLT	-2
CILT	TLTPROG	1
CIODL	OLINT	-1
CIFKT	Yok	0
CIADS	YOK	0
CICT	CTNO	0
CITES	TENOSY	0
CIFAC	FCNONE	Boş Değerler
CIFNC	CFYOK	Boşluklar

Çizelge 132. MQCIH içindeki alanların ilk değerleri (devamı var)

Alan adı	Değişmezin adı	Değişmezin değeri
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* MQCIH 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 MQCIH structure
D  CILEN          9     12I 0 INZ(180)
D* Reserved
D  CIENC         13     16I 0 INZ(0)
D* Reserved
D  CICS1         17     20I 0 INZ(0)
D* MQ format name of data that followsMQCIH
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 issuedby 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	CIADS	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(X'0000000000000000-00')
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*			

MQCMHO-İleti tanıtıcısı yaratma seçenekleri

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 1016
- “Başlangıçtaki değerler” sayfa 1017
- “RPG bildirimi” sayfa 1017

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 **PDSUPO** 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 hiçbirisi 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 133. MQCMHO 'daki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
CMSID	CMSIDV	' CMHO '
CMVER	CMVER1	1
CMOPT	CMDEFV	0

RPG bildirimi

```
D* MQCMHO 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*
D* Options that control the action of MQCRTMH
D  CMOPT          9     12I 0  INZ(0)
```

MQCNO-Bağlantı seçenekleri

MQCNO yapısı, uygulamanın yerel kuyruk yöneticiyle bağlantıyla ilgili seçenekleri belirlemesine olanak sağlar.

Genel Bakış

Amaç: Yapı, MQCONNX çağrısında bir giriş/çıkış değiştirgesi.

Sürüm: MQCNO 'nun yürürlükteki sürümü CNVER5' dir. Yalnızca yapının daha yeni sürümlerinde var olan alanlar, izleyen açıklamalarda olduğu gibi tanımlanır.

Sağlanan COPY dosyası, ortam tarafından desteklenen en son MQCNO sürümünü içerir; ancak, *CNVER* alanının ilk değeri CNVER1olarak ayarlıdır. version-1 yapısında mevcut olmayan alanları kullanmak için, uygulamanın *CNVER* alanını, gereken sürümün sürüm numarasına ayarlaması gerekir.

Karakter kümesi ve kodlama: MQCNO içindeki veriler, ENNAT tarafından verilen yerel kuyruk yöneticisinin *CodedCharSetId* kuyruk yöneticisi özniteliği ve kodlaması ile belirtilen karakter kümesinde olmalıdır.

- “Alanlar” sayfa 1018
- “Başlangıçtaki değerler” sayfa 1022
- “RPG bildirimi” sayfa 1023

Alanlar

MQCNO yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

CNCCO (10 basamaklı işaretli tamsayı)

Bu, MQCNO yapısının başlangıcındaki bir MQCD kanal tanımlama yapısının bayt cinsinden görelî konudur.

CNCCP (işaretçi)

Bu, bir MQCD kanal tanımlama yapısına ilişkin bir göstergedir.

CNCONID (24 baytlık karakter dizilimi)

Benzersiz bağlantı tanıtıcısı. Bu alan, kuyruk yöneticisinin bir uygulama işlemini, kuyruk yöneticisine ilk bağlandığında benzersiz bir tanıtıcı atayarak tanımlamasına olanak tanır.

Uygulamalar, PUT ve GET çağrıları yapılırken ilinti amaçları için bağlantı tanıtıcısını kullanır. Bağlantı kurulduğunda tüm bağlantılara kuyruk yöneticisi tarafından bir tanıtıcı atanır.

Uzun bir çalışma biriminin sonunu zorlamak için bağlantı tanıtıcısını kullanmak mümkündür. Bunu yapmak için, 'Bağlantıyı Durdur' PCF komutunu ya da MQSC komutunu STOP CONN ' i kullanarak bağlantı tanıtıcısını belirtin. Bu komutların kullanılmasıyla ilgili daha fazla bilgi için ilgili bağlantılara bakın.

Alanın ilk değeri 24 boş byte 'tır.

CNCT (128 baytlık bit dizesi)

Bu, kuyruk yöneticisinin, bu bağlantı sırasında uygulamadan etkilenen kaynaklarla ilişkilendirdiği bir etikettir.

Kuyruk yöneticisi bağlantı etiketi.

Kuyruk yöneticisinin etkilenen kaynaklara erişimi doğru olarak diziselleştirebilmesi için, her uygulama ya da uygulama eşgörünümünün etiket için farklı bir değer kullanması gerekir. Ayrıntılı bilgi için CN* CT* seçeneklerinin açıklamalarına bakın. Etiket, uygulama sona erdirildiğinde geçerli olacak ya da MQDISC çağrısını yayınlarken geçerli olacak.

Herhangi bir etiket gerekmiyorsa aşağıdaki özel değeri kullanın:

CTNONE

Bağlantı etiketi belirtilmedi.

Değer, alanın uzunluğuna ilişkin ikili sıfır değeridir.

Bu bir giriş alanıdır. Bu alanın uzunluğu LNCTAG tarafından verilir. Bu alanın ilk değeri CTNONE olur. *CNVER* , CNVER3değerinden küçükse bu alan yok sayılır.

Bir z/OS kuyruk yöneticisine bağlanılırken ConnTag alanını kullanın.

CNOPT (10 basamaklı işaretli tamsayı)

MQCONNXişlemini denetleyen seçenekler.

Bağlama seçenekleri

Bağlama seçenekleri, kullanılan IBM MQ bağ tanımı tipini denetler; bu seçeneklerden yalnızca birini belirtin:

CNSBND

Standart bağ.

Standart bağ tanımlama seçeneği, uygulamanın ve yerel kuyruk yöneticisi aracısının genellikle ayrı süreçlerde, ayrı yürütme birimlerinde çalıştırılmasına neden olur. Düzenleme kuyruk yöneticisinin bütünlüğünü korur; yani, kuyruk yöneticisini errant programlarından korumaktadır.

Uygulamanın tam olarak sınanmamış olabileceği ya da güvenilir olmayan ya da güvenilmez olduğu durumlarda CNSBND seçeneğini kullanın. Varsayılan değerCNSBND ' dir.

CNSBND , program belgelerine yardımcı olmak için tanımlıdır. Bu seçeneği, kullanılan bağ tanımı tipini denetleyen başka bir seçenekle kullanmayın; ancak değeri sıfır olduğu için, bu tür bir kullanım saptanamaz.

Bu seçenek tüm ortamlarda desteklenir.

CNFBND

Hızlı yol bağı.

Hızlı yol bağ tanımı 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 çalıştırıldığı standart bağlayıcıyla karşıttır.

Kuyruk yöneticisi bu bağ tanımlama tipini desteklemiyorsa,CNFBND yoksayılır; işlem, seçenek belirlenmemiş gibi devam eder.

CNFBND , birden çok işlemin uygulama tarafından kullanılan genel kaynaktan daha fazla kaynak tükettiği durumlarda avantaj olabilir.Hızlı yol bağ tanımını kullanan bir uygulama, *güvenilir uygulama*olarak bilinir.

Hızlı yol bağ tanımını kullanıp kullanmayacağına karar verirken aşağıdaki önemli noktaları göz önünde bulundurun:

- **CNFBND seçeneğinin kullanılması, bir uygulamanın kuyruk yöneticisine ait iletileri ve diğer veri alanlarını değiştirmesini ya da değiştirmesini engellememektedir. Bu seçeneği yalnızca, bu sorunları tam olarak değerlendirdiğiniz durumlarda kullanın.**
- Uygulama, zamanuyumsuz sinyalleri ya da süreölçer kesintilerini (sigkillgibi) CNFBNDile birlikte kullanmamalıdır. Paylaşılan bellek kesimlerinin kullanımına ilişkin kısıtlamalar da vardır.
- Uygulamanın, herhangi bir zamanda kuyruk yöneticisine birden fazla iş parçacığı bağlanmamış olması gerekir.
- Uygulama, kuyruk yöneticisinden bağlantı kesmek için MQDISC çağrısını kullanmalıdır.
- Kuyruk yöneticisini endmqm komutuyla sona erdirmeden önce uygulamanın bitmesi gerekir.

Belirtilen ortamlarda CNFBND kullanımı için aşağıdaki noktalar geçerlidir:

- On IBM i, the job must run under user profile QMQM that belongs to the QMQMADM group. Ayrıca, programın olağan dışı bir şekilde sonlanmaması gerekir; tersi durumda, önceden kestirilemeyen sonuçlar ortaya çıkabilir.

Güvenilen uygulamaların kullanılmasıyla ilgili daha fazla bilgi için [MQCONNX çağrısını kullanarak kuyruk yöneticisine bağlanma](#) ve [Güvenilir uygulamalara ilişkin kısıtlamalar](#)başlıklı konuya bakın.

CNSHBD

Paylaşılan Bağ Tanımları.

Paylaşılan bağ tanımları seçeneği, uygulamanın ve yerel kuyruk yöneticisi aracısının genellikle ayrı süreçlerde olmak üzere ayrı yürütme birimlerinde çalışmasına neden olur. Düzenleme kuyruk yöneticisinin bütünlüğünü korur; yani, kuyruk yöneticisini errant programlarından korumaktadır. Ancak, uygulama ile yerel kuyruk yöneticisi aracı arasında bazı kaynaklar paylaşılır. Kuyruk yöneticisi bu bağ tanımı tipini desteklemiyorsa, CNSHBD yoksayılr. Bu seçenek belirlenmemiş gibi işleme devam eder.

CNIBND

Yalıtılmış Bağ Tanımları.

Yalıtılmış bağ tanımları seçeneği, uygulamanın ve yerel kuyruk yöneticisi aracısının genellikle ayrı süreçlerde olmak üzere ayrı yürütme birimlerinde çalışmasına neden olur. Düzenleme kuyruk yöneticisinin bütünlüğünü korur; yani, kuyruk yöneticisini errant programlarından korumaktadır. Uygulama işlemi ve yerel kuyruk yöneticisi aracı, kaynakları paylaşmadıkları için birbirlerinden yalıtılır. Kuyruk yöneticisi bu bağ tanımı tipini desteklemiyorsa, CNIBND yoksayılr. Bu seçenek belirlenmemiş gibi işleme devam eder.

Tanıtıcı paylaşım seçenekleri

Aşağıdaki seçenekler, aynı işlem içinde farklı iş parçacıkları (koşut işleme birimleri) arasındaki tutamaçların paylaşımını denetler. Bu seçeneklerden yalnızca biri belirlenebilir.

CNHSN

İş parçacıkları arasında tanıtıcı paylaşımı yok.

İş parçacıkları arasında yapılan paylaşım yok seçeneği, bağlantı ve nesne tutamaçlarının yalnızca, ayrılmasına neden olan iş parçacığının ayrılmasına neden olduğunu; yani, MQCONN, MQCONNX ya da MQOPEN çağrısını yayınlayan iş parçacığının kullanılacağını belirtir. Tutamaçlar, aynı işleme ait diğer iş parçacıkları tarafından kullanılamaz.

CNHSB

Arama engellemesi ile iş parçacıkları arasında dizisel tanıtıcı paylaşımı.

İş parçacıkları arasındaki dizisel tanıtıcı paylaşımı, çağrı engellemesi ile, bir işlemin bir iş parçacığı tarafından ayrılan bağlantı ve nesne tutamaçlarının aynı işleme ait diğer iş parçacıkları tarafından kullanılabilmesi anlamına gelir. Ancak, bir defada yalnızca bir iş parçacığı belirli bir tutamacı kullanabilir; bu, yalnızca bir tutamaçtan yalnızca 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).

CNHSNB

Arama engellemesi olmadan, iş parçacıkları arasında dizisel tanıtıcı paylaşımı.

The serial handle sharing between threads, *bu olmadan* call blocking, option is the same as the " *bununla* blocking" option, except that, if the handle is in use by another thread, the call completes immediately with CCFAIL and RC2219 instead of blocking until the handle becomes available.

Bir iş parçacığında sıfır ya da bir paylaşılmayan tutamaçlar, artı sıfır ya da daha fazla paylaşılan tanıtıcı olabilir:

- CNHSN belirten her MQCONN ya da MQCONNX çağrısı, ilk çağrıda yeni paylaşılmayan bir tanıtıcı döndürür ve sonraki çağrılarda da aynı paylaşılmayan tanıtıcı döndürür (MQDISC çağrısına müdahale edilmezse). İkinci ve daha sonraki aramalar için neden kodu RC2002 ' dir.
- CNHSB ya da CNHSNB belirten her MQCONNX çağrısı, her çağrıda yeni bir paylaşılan tanıtıcı döndürür.

Nesne tutamaçları, nesne tanıtıcısını yaratan MQOPEN çağrısında belirlenen bağlantı tanıtıcısı ile aynı paylaşım özelliklerini devralır. 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 parçacığında güncellenebilir.

Bir tutamaç paylaşımı seçeneği belirtmezseniz, varsayılan değer ortam tarafından belirlenir:

- Microsoft Transaction Server (MTS) ortamında varsayılan değer, CNHSBile aynıdır.
- Diğer ortamlarda, varsayılan değer CNHSNile aynıdır.

Yeniden bağlanma seçenekleri

Yeniden bağlantı seçenekleri, bir bağlantının yeniden bağlanabilir olup olmadığını belirler. Yalnızca istemci bağlantıları yeniden bağlanabilir.

CNRCDF

Yeniden bağlanma seçeneği, varsayılan değerine çözümlenir. Varsayılan değer ayarlanmazsa, bu seçeneğin değeri DISABLEolarak çözülüyor. Seçeneğin değeri sunucuya geçirilir ve **PCF** ve **MQSC**tarafından sorgulanabilir.

CNRC

Uygulama, MQCONNX **QMNAME** parametresindeki değerle tutarlı herhangi bir kuyruk yöneticisine yeniden bağlanabilir. Yalnızca istemci uygulaması ile bir bağlantı kurmasıyla kuyruk yöneticisi arasında bir benzerlik yoksa CNRC seçeneğini kullanın. Seçeneğin değeri sunucuya geçirilir ve **PCF** ve **MQSC**tarafından sorgulanabilir.

CNRC

Uygulama yeniden bağlanamaz. Bu seçeneğin değeri, sunucuya *iletmiyor*.

CNRCQM

Uygulama yalnızca, başlangıçta bağlı olduğu kuyruk yöneticisiyle yeniden bağlantı kurulabilir. 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. Seçeneğin değeri sunucuya geçirilir ve **PCF** ve **MQSC**tarafından sorgulanabilir.

Yalnızca istemci bağlantıları için CNRC, CNRC ve CNRCQM seçeneklerini kullanın. Bir bağlama bağlantısı için seçenekler kullanılırsa, MQCONNX tamamlanma kodu, MQCC_FAILED ve neden kodu MQRC_OPTIONS_ERRORile başarısız olur.

Varsayılan seçenek: Tanımlanan seçeneklerden hiçbiri gerekli değilse, aşağıdaki seçenek kullanılabilir:

CNNONE

Seçenek belirlenmez.

CNNONE , program belgelerine yardımcı olmak için tanımlıdır. Bu seçeneğin diğer herhangi bir CN* seçeneğiyle birlikte kullanılması amaçlanmaz, ancak değeri sıfır olduğu için bu tür bir kullanım algılanamaz.

CNSCO (10 basamaklı işaretli tamsayı)

Bu, MQCNO yapısının başlangıcındaki bir MQSCO yapısının byte cinsinden görelî konudur.

CNVER , CNVER4değerinden küçükse bu alan yok sayılır.

CNSCP (işaretçi)

Bu, bir MQSCO yapısının adresidir.

CNVER , CNVER4değerinden küçükse bu alan yok sayılır.

CNSECP (10 basamaklı işaretli tamsayı)

Güvenlik değiştiricileri 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 'tır.

CNVER , CNVER5değerinden küçükse bu alan yok sayılır.

CNSECPP (işaretçi)

Güvenlik değıştirmeleri göstergesi. Bir kullanıcı kimliği ve parola belirtmek için kullanılan MQCSP yapısının adresi.

Bu alanın başlangıç değeri, boş değeri ya da boş değeri bayttır.

CNVER , CNVER5değerinden küçükse bu alan yok sayılır.

CNSID (4 baytlık karakter dizgisi)

MQCNO yapısına ilişkin yapı tanıtıcısı.

Değer şu olmalıdır:

CNSIDV

Bağlantı seçenekleri yapısına ilişkin tanıtıcı.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri CNSIDV ' dir.

CNVER (10 basamaklı işaretli tamsayı)

MQCNO yapısına ilişkin yapı sürüm numarası.

Değer şu olmalıdır:

CNVER5

Version-5 bağlantı seçenekleri yapısı.

Bu sürüm, tüm ortamlarda desteklenir.

Aşağıdaki değışmez, yürürlükteki sürümün sürüm numarasını belirtir:

CNVERC

Connect-options yapısının yürürlükteki sürümü.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri CNVER5' dir.

Başlangıçtaki değerler

Çizelge 134. MQCNO içindeki alanların ilk değerleri		
Alan adı	Değışmezin adı	Değışmezin değeri
CNSID	CNSIDV	' CNO→
CNVER	CNVER5	1
CNOPT	CNNONE	0
CNCCO	Yok	0
CNCCP	Yok	Boş değeri gösterge ya da boş değeri byte
CNCT	CTNONE	Boş Değerler
CNSCP	Yok	Boş değeri gösterge ya da boş değeri byte
CNSCO	Yok	0
CNCONID	Yok	Boş Değerler
CNSECPO	Yok	0
CNSECPP	Yok	Boş değeri gösterge ya da boş değeri byte

Notlar:

1. $\neg$ simgesi tek bir boş karakteri temsil eder.

RPG bildirimi

```
D*..1.....2.....3.....4.....5.....6.....7..
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 ofMQCONN
D CNOPT             9          12I 0 INZ(0)
D* Offset of MQCD structure for clientconnection
D CNCCO            13          16I 0 INZ(0)
D* Address of MQCD structure for clientconnection
D CNCCP            17          32*   INZ(*NULL)
D* Queue-manager connection tag
D CNCT              33        160    INZ(X'00000000000000-
                                00000000000000000000-
                                00000000000000000000-
                                00000000000000000000-
                                00000000000000000000-
                                00000000000000000000-
                                00000000000000000000-
                                00000000000000000000-
                                00000000000000000000-
                                00000000000000000000-
                                00000000000000000000-
                                00000000000000000000-
                                00000000000000000000-
                                00000000000000000000-
                                ')
D* Address of MQSCO structure forclient connection
D CNSCP            161        176*   INZ(*NULL)
D* Offset of MQSCO structure for clientconnection
D CNSCO            177        180I 0 INZ(0)
D* Unique Connection Identifier
D CNCONID           181        204    INZ(X'00000000000000-
                                00000000000000000000-
                                000000000000')
D* Offset of MQCSP structure
D CNSECP           205        208I 0 INZ(0)
D* Address of MQCSP structure
D CNSECPP           209        224*   INZ(*NULL)
```

MQCSP-Güvenlik değiştirgeleri

IBM MQ for IBM için MQCSP yapısına ilişkin özet.

Genel Bakış

Amaç: MQCSP yapısı, yetki hizmetinin bir kullanıcı kimliği ve parola doğrulamasına olanak sağlar. MQCONNX çağrısında MQCSP bağlantı güvenliği değiştiricileri yapısını belirtmenizi sağlar.

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 1023
- “Başlangıçtaki değerler” sayfa 1025
- “RPG bildirimi” sayfa 1025

Alanlar

MQCSP yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

CSAUTH (10 basamaklı işaretli tamsayı)

Bu, gerçekleştirilecek kimlik doğrulamasının tipidir.

Geçerli değerler şunlardır:

CSAN

Kullanıcı kimliği ve parola alanlarını kullanmayın.

CSAUIAP

Kullanıcı kimliği ve parola alanlarını doğrulayın.

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.

Görelilik 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.

Görelilik 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 tanıtcısı.

Değer şu olmalıdır:

CSSIDV

Güvenlik parametreleri yapısına ilişkin tanıtcı.

CSVER (10 basamaklı işaretli tamsayı)

Yapı sürüm numarası.

Değer şu olmalıdır:

CSVER1

Version-1 güvenlik parametreleri yapısı.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

CSVERC

Güvenlik değiştirgeleri yapısının yürürlükteki sürümü.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri CSVER1' dir.

Başlangıçtaki değerler

Çizelge 135. MQCNO içindeki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
CSSID	CSSIDV	' CSP¬ '
CSVER	CSVER1	1
CSAUTHT	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 bildirimi

```
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 CSVER 5 8I 0 INZ(1)
D* Type of authentication
D CSAUTHT 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)

MQCTLO-Geri çağırma seçenekleri yapısı

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\u00f0fcr\u00f0fcm

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 1026](#)
- [“Başlangıçtaki değerler” sayfa 1027](#)
- [“RPG bildirimi” sayfa 1027](#)

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 tanıttıcısıdır; değer şöyle olmalıdır:

CTLSI

Denetim Seçenekleri yapısına ilişkin tanıttıcı.

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

Çizelge 136. MQCTLO içindeki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
COSID	CTLSI	'CTLO'
COVER	CTLV1	1
COOPT	CCTLNO	Boş Değerler
CORSV	Ayrılmış alan	
COCONNAREA	Yok	Boş değerli gösterge ya da boş değerli byte

RPG bildirimi

```
D* MQCTLO 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 MQCTL
D COOPT          9     12I 0 INZ(0)
D*
```

```
D* Reserved
D CORSV                13      16I 0 INZ(-1)
D*
D* MQCTL Data area passed to the function
D COCONNAREA           17      32*  INZ(*NULL)
```

MQDH-Dağıtım üstbilgisi

MQDH yapısı, ileti bir iletim kuyruğunda saklanan bir dağıtım listesi iletilisi olduğunda, bir iletime var olan ek verileri açıklar.

Genel Bakış

Amaç: Dağıtım listesi iletilisi, 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 aracılığıyla) özel uygulamalar tarafından kullanılmak üzere kullanılır.

Bu yapı, yalnızca dağıtım listelerine ileti koymak isteyen normal uygulamalar tarafından kullanılmamalıdır *olmamalıdır*. 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 iletilisi 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 iletilisi verileri

Variş noktalarına bağlı olarak, kuyruk yöneticisi tarafından birden çok ileti oluşturulabilir ve farklı iletim kuyruklarına yerleştirilebilir. Bu durumda, bu iletilerdeki MQDH yapıları, uygulama tarafından açılan dağıtım listesi tarafından tanımlanan hedeflerin farklı alt kümelerini tanımlar.

Dağıtım listesi iletilisini 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 1331 içinde açıklanan *DistLists* kuyruğu özniteliğine bakın). Bir uygulama dağıtım listesi iletilisini dağıtım listelerini desteklemeyen bir kuyruğa doğrudan koyarsa, kuyruk yöneticisi dağıtım listesi iletilisini tek tek iletilere ayırır ve bunun yerine bunları kuyruğun üzerine yerleştirir.

- "[Alanlar](#)" sayfa 1028
- "[Başlangıçtaki değerler](#)" sayfa 1031
- "[RPG bildirimi](#)" sayfa 1032

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:

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.

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 iletiyi dağıtım listesine koyduğunda, kuyruk yöneticisi, aşağıdaki her ikisinin de doğru olduğu durumlarda, MQDH 'de DHFNEW' yi 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 değeri 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 değeri 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 içeriği kuyruk yöneticisi tarafından korunmaz. MQPMR kayıtları dizisi için, MQOR kayıtları diziden önce gelen dizi için geçerlidir.

Bu alanın ilk değeri 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 uzunluğa eklenir.

Bir dağıtım listesi her zaman en az bir hedef içermeli, bu nedenle *DHORO* her zaman sıfırdan büyük olmalıdır.

Bu alanın ilk değeri 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 belirtici alanı var.

Hiçbir MQPMR alanı yoksa, aşağıdaki değer 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 PFNONE '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 iletisi 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 137. 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

Çizelge 137. MQDH ' deki alanların ilk değerleri (devamı var)

Alan adı	Değişmezin adı	Değişmezin değeri
DHFLG	DHFNON	0
DHPRF	FPNONE	0
DHCNT	Yok	0
DHORO	Yok	0
DHPRO	Yok	0

Notlar:

1. → simgesi tek bir boş karakteri temsil eder.

RPG bildirimi

```

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 plusfollowing 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)

```

MQDLH -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 iletilisi 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 1033
- “Ölü-mektup kuyruğundan ileti alınması” sayfa 1034
- “Alanlar” sayfa 1034
- “Başlangıçtaki değerler” sayfa 1037
- “RPG bildirimi” sayfa 1037

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 PMDEFC seçeneğini kullanmaları gerekir. PMDEFC 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 değildir.
- 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 tanıtıcısını 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 tanıtıcısını devralır.

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. Provided no error occurs, the value CSINHT is not returned by the MQGET call.

MQMD içindeki MDPAT alanının değeri ATBRK Rise,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 LNMNtarafından 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-İleti tanımlayıcısı” sayfa 1076içindeki MDPAN alanının açıklamalarına bakın.

İletiyi ölü-mektup kuyruğuna yeniden yönlendiren kuyruk yöneticiyse, DLPAN kuyruk yöneticisi adının ilk 28 karakterini içerir. Ad, gerekirse boşluklarla doldurulur.

Bu alanın uzunluğu LNPANTarafından 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-İleti tanımlayıcısı” sayfa 1076 konusuna bakın).

İletiyi ölü-harfli kuyruğuna yeniden yönlendiren kuyruk yöneticiyse, DLPAT değeri ATQMolur.

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 LNPDATtarafından 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 sığrama 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 iletisi) 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-İleti tanımlayıcısı” sayfa 1076](#) 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 1198](#).

FBATYP

Uygulama tipi hatası.

Tetikleme iletisinin TMAI alanı geçerli olmadığından, bir tetikleme iletisini işleyen bir uygulama, uygulamayı başlatamadı; bkz. [“MQTM-Tetikleme iletisi” sayfa 1198](#).

FBBODC

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

<i>Çizelge 138. Initial values of fields in MQDLH.</i> MQDLH değişmezlerinin adlarını ve değerlerini listeler.		
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 bildirimi

D*..1.....2.....3.....4.....5.....6.....7..
D* MQDLH 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 that follows 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 message on dead-letter
D* (undelivered-message)queue
D  DLPAT         125        128I 0 INZ(0)
D* Name of application that put message on dead-letter
D* (undelivered-message)queue
D  DLPAN         129        156      INZ
D* Date when message was put on dead-letter (undelivered-message)queue
D  DLPD          157        164      INZ
D* Time when message was put on the dead-letter (undelivered-message)queue
D  DLPT          165        172      INZ

```

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 1038
- “Başlangıçtaki değerler” sayfa 1039
- “RPG bildirimi” sayfa 1039

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 139. 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)
```

MQDMMPO-İleti özelliği seçeneklerini sil

İleti silme özelliği seçeneklerini tanımlayan yapı.

Genel Bakış

Amaç: MQDMMPO yapısı, uygulamaların, iletilerin özelliklerini nasıl sileceğini denetleyen seçenekleri belirlemesine olanak sağlar. Yapı, MQDLTMP çağrısına ilişkin bir giriş değiştirir.

Karakter kümesi ve kodlama: MQDMMPO 'daki veriler, uygulamanın (ENNAT) uygulama ve kodlamasının karakter kümesinde yer almalıdır.

- “Alanlar” sayfa 1039
- “Başlangıçtaki değerler” sayfa 1040
- “RPG bildirimi” sayfa 1041

Alanlar

MQDMMPO 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 görel konularıyla ilgilidir.

DPDELF

Belirtilen adla eşleşen ilk özelliğin silinmesine neden olur.

DPDELC

Özellik imlecinin işaret ettiği özelliği 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 hiçbirisi 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 140. MQDPMO ' daki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
DPSID	DPSİDETİ	' DPMO '
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)
```

MQEPH-Embedded PCF üstbilgisi

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 oluşturmaz.

- [“Alanlar” sayfa 1041](#)
- [“Başlangıçtaki değerler” sayfa 1043](#)
- [“RPG bildirimi” sayfa 1043](#)

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 miktarıdır. 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 141. 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 142. MQEPH '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 141 sayfa 1042 içinde 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* MQEPH Structure
D*
D* Structure identifier
D EPSID 1 4
D* Structure version number
D EPVER 5 8I 0
D* Total length of MQEPH including MQCFHand 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

MQGMO-Get-message seçenekleri

MQGMO yapısı, uygulamanın iletilerin kuyruktan nasıl kaldırıldığını denetleyen seçenekleri belirlemesine olanak sağlar.

Genel Bakış

Amaç

Yapı, MQGET çağrısındaki bir giriş/çıkış parametresidir.

S\u00f0fcr\u00f0fcm

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 1044
- “Başlangıçtaki değerler” sayfa 1063
- “RPG bildirimi” sayfa 1064

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* , GMVER2değerinden küçükse bu alan yok sayılır.

GMMH (10 basamaklı işaretli tamsayı)

İleti Tanıtıcısı

GMPRAQ 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.

GMPRRF belirtilirse ya da GMPRAQ 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; 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 GMVER4 then you must supply a valid message descriptor on the MQGET call. 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 yoksayılır. 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 aynı.

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.

MOFFFS

İletiyi belirtilen görelî konum ile al.

Bu seçenek, alınacak iletinin, MQGET çağrısının *MSGDSC* parametresindeki *MDOFF* alanının değeriyle eşleşen bir görelî 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 görelî 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* , GMVER2değerinden 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 GMVER2ve *GMMO* ' dan MONONE ' a ayarlamaktan kaçınılabılır.

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 değişmezi 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 *GMWIT* 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 hiçbir, 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 beklemes. 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 getiriletilirse, 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 hiçbirisiyle geçerli değildir:

- GGMSYP
- GMPSYP

Göz at seçenekleri: Aşağıdaki seçenekler kuyruklardaki iletilere göz atmak için ilişkilendirilir:

GMBRWF

Kuyruğun başlangıcından göz atın.

OBRW 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. Kuyrukta 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 kuyrukta konumdaki konumda kalır.

GMMUC seçeneği, iletiyi kuyruktan kaldırmak için, gerekliyse, göz atmayan bir MQGET çağrısıyla birlikte kullanılabilir.

Göz atma imleci, aynı HOBJS 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 kilitlenmiş 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 kaldıran 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. theMQGET çağrısı başarılı olursa (tamamlanma kodu CCOK ya da CCWARN), göz

atma için grup ve bölüm 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 ilerletildiğinden 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 1331](#) 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 kilitlenmiş 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, gerekliyse, 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 kilitlenmiş bir ileti varsa, bu ileti, imlecin altındaki, bu nedenle kilidi açmadan ve yeniden kilitlenmeden döndürülmesi gerekir; ileti kilitli kalır.
- Kilitli bir ileti yoksa, göz at imlecinin altındaki ileti (varsa) kilitlenir ve uygulamaya geri döndürülür; Göz At imlecinin altında bir ileti yoksa, arama başarısız olur.

GMBRWC, GMLK olmadan belirtilirse:

- Önceden kilitlenmiş 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 alınamadığına 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 ayrılır.

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 iletin 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 kilitlenir, ancak bu bir mantıksal ileti ya da fiziksel bir ileti olabilir:

- GMCMPM belirtilirse, mantıksal iletiyi oluşturan tüm ileti bölümleri kuyruk tanıtıcılarına kilitlenir (kuyruğun tümü varsa ve alma için kullanılabilir).
- GMCMPM belirtilmemişse, kuyruk sapına yalnızca tek bir fiziksel ileti kilitlenir. Bu ileti bir mantıksal iletinin bir bölümü olursa, kilitli kesim mantıksal iletiyi almak ya da göz atmak için GMCMPM kullanan diğer uygulamaları öner.

Kilitlenen 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 çağrıları 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 OOBROW dışında, bu seçeneği belirtmek için özel bir açma seçeneği gerekli değildir.

Bu seçenek, aşağıdaki seçeneklerin hiçbirisiyle geçerli değildir:

- GGMSYP
- GMPSYP
- GGMUNLK

GMUNLK

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, GMUNLK belirtilirse denetlenmez ya da değiştirilmez. *BUFFER*' da ileti döndürülmez.

Bu seçeneği belirlemek için özel bir açma seçeneği gerekmez (kilit isteğini ilk sırada OOBROW komutu verebilmek için OOBROW 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 dönüştürülebiliyor.

- 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, tamamlanma 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-İleti tanımlayıcısı” sayfa 1076](#) 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 'deMDMID 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 '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 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 ayırt edilir; burada n, gruptaki mantıksal iletilerin sayısıdır. Mantıksal iletilerden biri ya da daha fazlası bölümlendiye, grupta n fiziksel ileti sayısı daha fazla olur.

GGMLOGO

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 MFMIĞ 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 MFLMIĞ (yani, gruptaki son mantıksal ileti kesimlere ayrılmış değildir) ya da
- MFLSEG ile MFLMIĞ (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 143 sayfa 1056 , 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 143. Mantıksal iletilerin gruplarındaki ve kısımlarındaki iletilere ilişkin MQGET seçenekleri

Belirttiğiniz seçenekler	Aramadan önce grup ve günlük-msg durumu		Kuyruk yöneticisinin bakışlarına ilişkin değerler				
	Cur Grp	Ccur günlük iletisi	MDMID	MDCID	MDGID	MDSEQ	MDOFF
Günlük ORD							
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 alınır, aynı kuyruk tanıtıcısı kullanılırsa, 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 *alınmamışsa*, aynı kuyruk tanıtıcısı kullanılırsa, gruptaki diğer mantıksal ileti ve kesimlerin hiçbirisi bir iş birimi içinde alınamaz.

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 “MQGMO-Get-message seçenekleri” sayfa 1044 içinde açıklanan GMMO alanına bakın).

Sistem başarısızlığı oluştuktan sonra, ortadaki bir ileti grubunu ya da mantıksal iletiyi yeniden başlatmak için kullanılabilecek 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 GMMO 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 144 sayfa 1057 , 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 144. 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.

GMCMM

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 karşılarlar). Bu nedenle, tek tek parça almak istemeyen uygulamaların her zaman GMCMM '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), GMCMPM 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 tanıttıcılarına 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 özneliğinin 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 belirtilirse, GMAMSA, geçerli grup ya da mantıksal ileti olmadığında *yalnızca* seçeneğini etkiler. If there : a current group or logical message, GMAMSA is ignored. Bu, 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 kilitlenmez).

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 iletile ait kesimlerin alınmasını engeller. Ancak bu kesimler yine de *CurrentQDepth* kuyruk özniteliğinin değerine katkıda bulunur; bu, *CurrentQDepth* sıfırdan büyük olsa bile, yeniden alınabilir mantıksal iletler 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 iletilerinde 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 hiçbirisi 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 dizgisi)

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.

GMSG1 (10 basamaklı işaretli tamsayı)

Sinyal.

Bu ayrılmış bir alandır; değeri anlamlı değildir. Bu alanın ilk değeri 0 'tır.

GMSG2 (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.

SSSEEG

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-İleti tanımlayıcısı” sayfa 1076 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. Belirtirirse, 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 145. 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                      000000000000000000')
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)
```

MQIIH- IMS bilgi üstbilgisi

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 iletili verilerini dönüştürmek için kullanıcı tarafından bir veri dönüştürme çıkışı sağlanmalıdır.

- [“IMS köprü uygulamaları için passtickets doğrulanıyor” sayfa 1065](#)
- [“Alanlar” sayfa 1065](#)
- [“Başlangıçtaki değerler” sayfa 1068](#)
- [“RPG bildirimi” sayfa 1068](#)

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 çalışamaz.

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 146. MQIIH 'daki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
IISID	IISIDV	' I IH ¬ '
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 bildirimi

```
D*..1.....2.....3.....4.....5.....6.....7..
D*
D* MQIIH Structure
D*
D* Structure identifier
D IISID 1 4 INZ(' I IH ¬ ')
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'00000000000000-0000000000000000')
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

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 1069
- “Başlangıçtaki değerler” sayfa 1075
- “RPG bildirimi” sayfa 1075

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 ilerletilir.

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 ilerletilir. 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 değerin 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 dönüştürmeleri desteklenmektedir:

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 sıfıra 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.

İmlecin 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 IPINQN 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.

IPREQCSI (10 basamaklı işaretli tamsayı)

Değer, bir karakter dizgisiyse, sorgulayan özellik değerinin dönüştürülmesini sağlar. Bu, IPCVAL ya da IPCTYP belirtildiğinde, *ReturnedName* ' in dönüştürüleceği karakter kümesidir.

Bu alanın ilk değeri CSAPL ' dir.

IPREQENC (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 değerin 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 değerin 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 tanıttıcısıdır. Değer şu olmalıdır:

IPSIDV

Sorgu iletisi özellik seçenekleri yapısının tanıttıcısı.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri IPSIDV 'dir.

IPITYP (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:

İPVERC

Sorgu iletisi özellik seçenekleri yapısının yürürlükteki sürümü.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri IPVER1' dir.

Başlangıçtaki değerler

Çizelge 147. 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	
IPRETNAMCHRP		
IPTYP		boşluk

RPG bildirimi

```
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 IPRETNAMCHRP   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
```

MQMD-İleti tanımlayıcısı

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 değiştirir.

Sürüm: MQMD 'nin yürürlükteki sürümü MDVER2' dir. Yalnızca yapının daha yeni sürümlerinde var olan alanlar, izleyen açıklamalarda olduğu gibi tanımlanır.

Sağlanan COPY dosyası, ortam tarafından desteklenen MQMD ' nin en son sürümünü içerir, ancak MDVER alanının ilk değeri 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 özniteliği 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 kuyruğu yöneticileri farklı karakter kümeleri ya da kodlamalar kullanırsa, MQMD ' deki veriler otomatik olarak dönüştürülür. Uygulamanın MQMD ' yi dönüştürmesi gerekli değildir.

- “MQMD ' nin farklı sürümlerinin kullanılması” sayfa 1076
- “İleti bağlantısı” sayfa 1077
- “İleti süre bitimi” sayfa 1077
- “Alanlar” sayfa 1077
- “Başlangıçtaki değerler” sayfa 1116
- “RPG bildirimi” sayfa 1117

MQMD ' nin farklı sürümlerinin kullanılması

version-2 MQMD, genellikle bir version-1 MQMD kullanılmasına ve ileti verilerinin MQMDE yapısıyla önceden düzeltilmesine eşdeğerdir. Ancak, MQMDE yapısındaki tüm alanların varsayılan değerleri varsa, MQMDE atlanabilir. Bir version-1 MQMD artı MQMDE, daha sonra bu bölümde anlatıldığı gibi kullanılır.

- 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.

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 *değil* , MQPUT ve MQPUT1 çağrılarının çıktısındaki MQMD ' deki eşdeğer alanlara herhangi bir değer döndürür; uygulama bu çıkış değerlerini gerektiriyorsa, bu değer bir version-2 MQMD kullanılmalıdır.

- MQGET çağrısında, uygulama bir version-1 MQMD sağlıyorsa, kuyruk yöneticisi örnekleri bir MQMDE ile döndürülür; ancak, MQMDE içindeki alanlardan biri ya da daha fazlası varsayılan olmayan bir değere 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 148 sayfa 1116](#) ' ta gösterilen bu alanların ilk değerleriyle aynıdır.

Bir ileti iletim kuyruğunda olduğunda, MQMD ' deki bazı alanlar belirli değerlere ayarlanır; ayrıntılar için “MQXQH-İletim kuyruğu üstbilgisi” sayfa 1208 konusuna bakın.

İleti bağlamı

MQMD ' deki bazı alanlar ileti bağlamını içerir. Tipik olarak:

- **Kimlik** bağlamı, iletiyi *ilk olarak* koyan uygulamayla ilgilidir.
- **Köken** bağlamı, iletiyi *en son* koyan uygulamaya ilişkindir
- **Kullanıcı** bağlamı, iletiyi *ilk olarak* koyan uygulamayla ilişkilidir.

Bu iki uygulama aynı uygulama olabilir, ancak bunlar aynı zamanda farklı uygulamalar da olabilir (örneğin, bir ileti bir uygulamadan başka bir uygulamaya iletildiğinde).

Tanıtıcı ve kaynak bağlamı genellikle daha önce açıklanan anlamlara sahip olsa da, MQMD ' deki bağlam alanlarının her iki tipinin içeriği aslında, ileti konduğunda 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).

- IMSiçin, 16 karakterlik IMS kurtarma simgesiyle bitıştırılan 8 karakterlik PSB adı.
- IBM i' ta, muhasebe bilgileri işe ilişkin muhasebe koduna ayarlanır.
- HP Integrity NonStop Serverve UNIX sistemleriüzerinde, hesap bilgileri ASCII karakterlerinde sayısal kullanıcı tanımlayıcıyla ayarlanır.
- Windowsüzerinde, muhasebe bilgileri sıkıştırılmış biçimde bir Windows NT güvenlik tanıtıcısına (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

UNIX sistemleri sayısal tanıtıcısı.

AYIKTIR

Kullanıcı tanımlı hesap belirteci.

KATIL

Bilinmeyen muhasebe belirteci tipi.

The accounting-token type is set to an explicit value only in the following environments: AIX, HP-UX, IBM i, Solaris, Windows, plus IBM MQ MQI clients connected to these systems. 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 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 *MDACC* ögesini içerir. Bu, alıkonursa iletiyle birlikte tutulan *MDACC* değeri olacaktır (alıkonan yayınlara ilişkin ek ayrıntılar için [“MQPMO-put-message seçenekleri” sayfa 1138](#) 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 dizgisi olarak değil, bitler dizgisi 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 LNACCT 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 kullanılabilecek uygulama takımı tarafından tanımlanan bilgilerdir. 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 1331](#).

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 belirlen abonelere bir yayın olarak gönderilirken ilinti tanıtıcısı olarak kullanılır.

Alıkonan yayınlarla ilgili daha fazla ayrıntı için bkz. [“MQPMO-put-message seçenekleri” sayfa 1138](#)

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.

Specifying CINONE as the correlation identifier has the same effect as *değil* specifying MOCORI, that is, *herhangi biri* correlation identifier will match.

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 değer, CICS bridge tarafından yeni bir oturumun başlangıcının, yani yeni bir ileti dizisinin başlangıcının belirtildiği gibi tanınır.

MQGET çağrısına ilişkin bu bir giriş/çıkış alanıdır. MQPUT ve MQPUT1 çağrıları için, PMNCID *değil* belirtildiyse, bu bir giriş alanıdır ve PMNCID : belirtilmişse bir çıkış alanı olur. Bu alanın uzunluğuna LNCID değeri verilir. Bu alanın ilk değeri CINONE 'tır.

MDCSI (10 basamaklı işaretli tamsayı)

Bu, iletteki karakter verilerinin karakter kümesi tanıtıcısını belirtir.

Not: MQMD ' deki karakter verileri ve çağrılar üzerinde değiştirge olan diğer IBM MQ veri yapıları, kuyruk yöneticisinin karakter kümesinde bulunmalıdır. Bu, kuyruk yöneticisinin *CodedCharSetId* özneliği tarafından tanımlanır; bu özneliğe ilişkin ayrıntılar için [“Kuyruk yöneticisine ilişkin öznitelikler” sayfa 1362](#) ' e bakın.

Aşağıdaki özel değerler kullanılabilir:

CSQM

Kuyruk yöneticisinin karakter kümesi tanıtıcısı.

İletideki karakter verileri, kuyruk yöneticisinin karakter takımında yer alıyor.

MQPUT ve MQPUT1 çağrılarında, kuyruk yöneticisi bu değeri, ileti ile gönderilen MQMD içinde, kuyruk yöneticisinin gerçek karakter kümesi tanıtıcısına (true character-set) değiştirir. Sonuç olarak, MQGET çağrısıyla CSQM değeri hiçbir zaman döndürülmez.

CINT

Bu yapının karakter kümesi tanıtıcısını devralır.

İletideki karakter verileri, bu yapıyla aynı karakter takımında yer alıyor; bu, kuyruk yöneticisinin karakter takımıdır. (Yalnızca MQMD için, CSINHT, CSQM ile aynı anlamlara sahiptir).

Kuyruk yöneticisi, MQMD 'nin gerçek karakter kümesi tanıtıcısına iletiyle gönderilen MQMD' de bu değeri değiştirir. Hata oluşmaması durumunda, MQGET çağrısıyla CSINHT değeri döndürülmez.

MQMD ' deki MDPAT alanının değeri ATBRKR ise, CSINHT kullanılamaz.

CSEMBO

Gömülü karakter takımı tanıtıcısı.

İletideki karakter verileri, ileti verilerinin kendisinde yer alan tanıtıcıyı içeren bir karakter kümesinde bulunur. İletinin farklı bölümlerine uygulanırken, ileti verileri içine yerleştirilmiş herhangi bir sayıda karakter kümesi tanıtıcısı olabilir. Bu değer, karakter kümelerinin karışımında veri içeren PCF iletileri için kullanılmalıdır. PCF iletileri, FMPCF biçim adına sahiptir.

Bu değeri yalnızca MQPUT ve MQPUT1 çağrılarında belirtin. MQGET çağrısında belirtilirse, iletinin dönüştürülmesini önler.

MQPUT ve MQPUT1 çağrılarında, kuyruk yöneticisi daha önce açıklandığı gibi iletiyle birlikte gönderilen MQMD 'deki CSQM ve CSINHT değerlerini değiştirir, ancak MQPUT ya da MQPUT1 çağrısında belirtilen MQMD' yi değiştirmez. Belirtilen değerde başka bir denetim gerçekleştirilmez.

İletileri alan uygulamaların bu alanı uygulamanın beklediği değerle karşılaştırmaları gerekir; değerler değişirse, uygulamanın karakter verilerini iletide dönüştürmesi gerekebilir.

MQGET çağrısında GMCONV seçeneği belirtilirse, bu alan bir giriş/çıkış alanıdır. Uygulama tarafından belirlenen değer, gerekli olduğunda ileti verilerinin dönüştürülmesi gereken kodlanmış karakter takımı tanıtıcısıdır. Dönüştürme işlemi başarılı ya da gereksizse, değer değişmeden kalır (CSQM ya da CSINHT değeri gerçek değere dönüştürülürse). Dönüştürme başarısız olursa, MQGET çağrısından sonraki değer, uygulamaya döndürülen dönüştürülmemiş iletinin kodlanmış karakter kümesi tanıtıcısını temsil eder.

Tersi durumda, bu, MQGET çağrısına ilişkin bir çıkış alanıdır ve MQPUT ve MQPUT1 çağrılarında ilişkin bir giriş alanı olur. Bu alanın ilk değeri CSQM ' dir.

MDENC (10 basamaklı işaretli tamsayı)

İleti verilerinin sayısal kodlaması.

Bu, iletideki 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 değışmezin değeri, programlama diline ve ortama bağıdır. Bu nedenle, uygulamaların, uygulamanın çalışacağı ortama uygun olan üstbilgi, makro, COPY ya da INCLUDE dosyaları kullanılarak derlenmeleri gerekir.

İletilerin yerleştirdiği uygulamaların olağan durumda ENNAT belirtmesi gerekir. İletileri alan uygulamaların bu alanı ENNAT değeriyle karşılaştırmaları gerekir; değerler değışirse, 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 değışmeden 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 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 harcadığı süreyi ve uzak bir kuyruğa konursa ara iletim kuyruklarında da yer alan süreyi yansıtacak şekilde belirlenir. İleti kanalı araçlarının iletim sürelerini yansıtmaması 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 hiçbirini 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 iletisi 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 tetiklemesi de içinde olmak üzere, kuyruğun tüm amaçlar için kuyruklardaki ileti sayısına kadar sayılabilir.

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 ileti gönderdiğinde aşağıdaki işlemlerden birini alabilir:
 - Kalan süre bitimini istek ileti üzerinden yanıt iletiye kopyalayın.
 - Yanıt ileti süresi bitimi süresini sıfırdan büyük bir açık değere ayarlayın.
 - Yanıt ileti süresi 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 ileti verilerinin 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 *her ikisi* 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 tanıdır:

EIULIM

Sınırsız ömür.

İletin süre bitimi sınırsız.

Bu, MQGET çağrısına ilişkin bir çıkış alanıdır ve MQPUT ve MQPUT1 çağrılarına 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 konamadığında 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, *MDMT* değeri 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 bildirimi (ROPAN ' a bakın).

FBNAN

Negatif işlem bildirimi (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 iletili, 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 iletilisinin *MDFB* alanına yerleştirir. Bu, bir IMS-OTMA hatası olduğunda FBILST (399) aralığında FBIFST (301) aralığında bir değere sahip olan geri bildirim kodunda sonuçlanır.

IMS köprüsü tarafından aşağıdaki geri bildirim kodları oluşturulabilir:

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.

The *MDFMT* field in MQMD specifies FMIMS, but the message does not begin with a valid MQIIH structure.

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.

FBCDLQ

Ölü-mektup kuyruğu kullanılamıyor.

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 geriletti.

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 addır. 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.

FMCMD2

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 “MQDLH -Dead-letter üstbilgisi” sayfa 1032 ' 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 “MQDH-Dağıtım üstbilgisi” sayfa 1028 ' 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 yapılar](#) başlıklı konuya bakın. Olaylarla ilgili bilgi için [Olay izleme](#) başlıklı 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 iletisi verileri) izler. MQMDE 'yi izleyen verilerin biçim adı, karakter kümesi ve kodlaması, MQMDE' deki *MEFMT*, *MECSI* ve *MEENC* alanları tarafından verilir. Bu yapıyla ilgili ayrıntılar için bkz. “MQMDE-İleti tanımlayıcı uzantısı” sayfa 1118 . MQGET çağrısında GMCONV seçeneği belirtilirse, bu biçimin iletileri dönüştürülebilmektedir.

FMPCF

Programlanabilir komut biçiminde (PCF) kullanıcı tanımlı ileti.

İleti, programlanabilir komut biçimi (PCF) iletilerinin yapısına uygun, kullanıcı tanımlı bir iletidir. MQGET çağrısında GMCONV seçeneği belirtilirse, bu biçimin iletileri dönüştürülebilmektedir. Programlanabilir komut biçimi iletileri kullanmaya ilişkin ek bilgi için [Programlanır Komut Biçimlerinin Kullanılması](#) konusuna bakın.

FMRMH

Başvuru iletisi üstbilgisi.

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 *RMFMT*, *RMCS* ve *RMENC* alanları tarafından verilir. Bu yapıyla ilgili ayrıntılar için bkz. "[MQRMH-Başvuru iletisi üstbilgisi](#)" sayfa 1163 . 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*, *RFCS* ve *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*, *RF2CS* ve *RF2ENC* alanları tarafından verilir. MQGET çağrısında GMCONV seçeneği belirtilirse, bu biçimin iletileri dönüştürülebilmektedir.

FMSTR

Tamamen karakterlerden oluşan ileti.

Uygulama iletisi 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 iletisi](#)" sayfa 1198 konusuna bakın. MQGET çağrısında GMCONV seçeneği belirtilirse, bu biçimin iletileri dönüştürülebilmektedir.

FNWIIH

İş bilgileri üstbilgisi.

İleti verileri, uygulama verileri tarafından izlenen iş bilgisi üstbilgisi MQWIIH ile başlar. Uygulama verilerine ilişkin biçim adı, MQWIIH 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. "[MQXQH-İletim kuyruğu üstbilgisi](#)" sayfa 1208 .

Bir *MDFMT* FMXQH değeri olan iletiler için COA ve COD raporları oluşturulmaz.

Bu, MQGET çağrısına ilişkin bir çıkış alanıdır ve MQPUT ve MQPUT1 çağrılarına 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:

- MFMIG
- MFLMIG
- 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 *değil* belirtildi.

Bu çağrılar, rapor ileti olmayan iletiler için kullanmayı düşünün. Ancak, uygulama daha fazla denetim gerektiriyorsa ya da çağrı MQPUT1 ise, uygulama *MDGID* ' 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:

- 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 *değil* 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 verilen *ilk* MQPUT ya da MQPUT1 çağrısı tarafından 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* , *MDVER2* değerinden 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:

- Bölümlendirme işareti
- Durum işaretleri

Bunlar sırayla anlatılır.

Bölümlendirme işareti: Bir ileti kuyruk için çok büyük olduğunda, kuyruğa ileti koyma girişimi genellikle başarısız olur. Bölümlendirme, 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. “MQGMO-Get-message seçenekleri” sayfa 1044 .) 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

Bölümlendirmeye 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.

the application must remember to reset the *MDGID* field to GINONE before *Her biri* MQPUT or MQPUT1 call, in order to cause a unique group identifier to be generated by the queue manager for each message. 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 bölünüdür. 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 iletisi verilerinin başlangıcı
- Uygulama iletisi 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*, *MDCSI* ve *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 *MDOFF* 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 baytından 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

Kuyruk yöneticisi, *kalıcı* iletiler için, bölümleme 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 yapıyorsa, kuyruk yöneticisi bölümleme işlemini gerçekleştiremez. İleti bölümlenmeye gerek duymuyorsa, arama hala başarılı olabilir. Ancak *yapar* iletisi kesimlere ayırma gerektiriyorsa, çağrı neden kodu RC225 ile başarısız olur.

kalıcı olmayan iletileri için, kuyruk yöneticisi bölümlemenin 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 dönüştürülemeyecek şekilde sonuçlanabileceği gibi, ya da FMSTR ve karakter kümesi DBCS ya da SBCS/DBCS ise, gönderme uygulaması kendi kendine yaratılamayı ve segmentasyonu engellemek için MFSEGI belirterek, kesimlere yol açabilir ve kesimler yerleştirebilir. 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 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 yansıtabiliyor.

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:

MFMI

İleti, bir grubun üyesidir.

MFLMI

İleti, bir gruptaki son mantıksal iletidir.

Bu işaret ayarlanırsa, kuyruk yöneticisi iletiyle birlikte gönderilen MQMD 'nin kopyasında MFMI '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, MFLMI ayarlanır, ancak *MDSEQ* alanında değer bir değer olur.

MFSEG

İleti, mantıksal bir iletinin bir bölümüdür.

MFSEG, MFLSEG olmadan belirtildiğinde, kesimde uygulama ileti verilerinin uzunluğu (var olabilecek MQ üstbilgi yapılarının uzunlukları *hariç tutma*) en az bir olmalıdır. Uzunluk sıfırsa, 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, kesimde uygulama ileti verilerinin (var olabilecek herhangi bir üstbilgi yapısının uzunlukları *hariç tutma*) 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* MQPUT çağrıları için geçerlidir:

- Yürürlükte bir grup ya da mantıksal ileti yoksa, bu işaretlerin tümü (ve bunların birleşimleri) geçerli olur.
- MFMI belirtildikten sonra, MFLMI belirtinceye kadar bu durumda kalmalıdır. Bu koşul karşılanmazsa, arama neden kodu RC2241 ile başarısız olur.
- MFSEG belirtildikten sonra, MFLSEG belirtinceye kadar bu durumda kalmalıdır. Bu koşul karşılanmazsa, arama neden kodu RC2242 ile başarısız olur.
- MFMI olmadan MFSEG belirlendikten sonra MFMI, MFLSEG belirtildikten sonra *off* (kapalı) durumda 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 çağrılarına 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 [“Rapor seçenekleri ve ileti işaretleri” sayfa 1395](#)'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 bir kuyruk yöneticisinden diğerine ileti akışı olduğunda ileti tanıtıcısı karakter kümeleri arasında *dönüştürülmez* .

MQPUT ve MQPUT1 çağrıları için, uygulama tarafından MINONE ya da PMNMID belirtilirse, kuyruk yöneticisi benzersiz bir ileti tanıtıcısı oluşturur.⁴İleti 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 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 dönmede 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 yaymak için kullanabilir.

Kuyruk yöneticisi şu hariç olmak üzere, bu alanın herhangi bir kullanımını gerçekleştirmiyor:

⁴ 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 oluşturulamaması gerekir. Ancak, uygulamanın bu aralıklarda başlangıç karakterleri olan tanıtıcılar oluşturmasını önlemez.

- 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 iletilisinin *MDCID* alanına kopyalayın (*MDREP* seçeneklerine bağlı olarak).

Kuyruk yöneticisi ya da bir ileti kanalı aracısı bir rapor iletilisi oluşturduğunda, *MDMID* alanını, özgün iletinin *MDREP* alanı, *RONMI* ya da *ROPMI* iletilisinin belirlediği şekilde ayarlar. Rapor iletileri oluşturan uygulamalar da bunu yapmalıdır.

MQGET çağrısına ilişkin *MDMID* , kuyruktan alınacak belirli bir iletiyi seçmek için kullanılabilecek beş alandan biridir. Olağan durumda *MQGET* çağrısı kuyruksa bir sonraki iletiyi döndürür, ancak belirli bir ileti gerekiyorsa, 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 uyan *ilk* iletidir. Ancak, *GMBRWN* belirtilirse, döndürülen ileti seçim ölçütlerine uyan *sonraki* iletidir; bu iletinin taranması, imlecin 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.

Seçim ölçütlerinin çeşitli durumlarda nasıl kullanılırla ilgili daha fazla bilgi için [Tablo 1](#) 'e bakın.

MINONE 'ın ileti tanıtıcısı olarak belirtilirse, *MOMSGI* belirtilerek *değil* ile aynı etkiye sahip olur; bu, *herhangi biri* ileti tanıtıcısının eşleşmesini sağlar.

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 *LNMIID* 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 çağrılarını 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 çağrılarını 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 çağrılarını için bir giriş alanı. Bu alanın ilk değeri MTDGRM ' dir.

MDOFF (10 basamaklı işaretli tamsayı)

Mantıksal iletinin başlangıcındaki fiziksel iletilerde verilerin görelî 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 aralığıdır. 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 *değil* belirtildi.

Bunlar, rapor iletisi olmayan iletiler için bu çağrılarını kullanmanın önerilen yollarıdır. Ancak, uygulama bu koşullara uymuyorsa ya da çağrı MQPUT1 ise, uygulama *MDOFF* ' 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 ileti 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* , *MDVER2* değerinden küçükse bu alan yok sayılır.

MDOLN (10 basamaklı işaretli tamsayı)

Özgün iletinin uzunluğu.

Bu alan yalnızca bölüm olan rapor iletileri için ilgi alanıdır. Rapor iletinin ilişkilendirdiği ileti kesiminin uzunluğunu belirtir; bu, bölüm formlarının parçası olan mantıksal iletinin uzunluğunu ya da rapor iletilerinde 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*, *MDOFF*, and *MDMFL*, fields from the original message. Sonuç olarak, rapor ileti de bir bölümdür. Rapor iletileri oluşturan uygulamaların aynı yapılması ve *MDOLN* alanının doğru olarak ayarlandığından emin olmak için önerilir.

Aşağıdaki özel değer tanımlıdır:

OUNDF

İletin 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 iletiyse, 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 RC2252 ile başarısız olur.

- Bir rapor ileti gönderiliyorsa, ancak bir rapor ileti değilse, kuyruk yöneticisi alanı yoksayar ve bunun yerine uygulama ileti 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* , *MDVER2* değerinden 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:

- 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 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.

- 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 üzerinde, kuyruk yöneticisi tam olarak nitelenmiş iş adını kullanır.
- HP Integrity NonStop Server üzerinde, kuyruk yöneticisi kullanılır: yürütülebilir dosyanın tam olarak nitelenmiş 28 karakteri, eğer bu kuyruk yöneticisi kullanılabiliriyorsa ve tersi durumda boşluk olmalıdır.
- On UNIX sistemleri, 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 yoksayılr 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 , aşağıdaki standart tiplerden birine sahip olabilir. Kullanıcı tanımlı tipler de kullanılabilir, ancak ATULST ile ATUFST aralığındaki değerlerle sınırlandırılmalıdır.

SAATAIX

AIX uygulaması (UNIX ile aynı değer).

ATBRKR

Aracı.

SAATCICS

CICS işlemi.

ATCIB

CICS bridge.

ATVSE

CICS/VSE işlemi.

ATDOS

PC DOS üzerinde IBM MQ MQI client uygulaması.

ATDQM

Dağıtılmış kuyruk yöneticisi aracı.

ATGUAR

Tandem Guardian uygulaması (ATNSK ile aynı değer).

SAATIMS

IMS uygulaması.

ATIMSB

IMS köprüsü.

ATJAVA

Java.

ATMVS

MVS ya da TSO uygulaması (ATZOS ile aynı değer).

ATNOT

Lotus Notes Agent uygulaması.

ATNKK

Tandem NonStop Kernel uygulaması.

AT390

OS/390 uygulaması (ATZOS ile aynı değer).

AT400

IBM i uygulaması.

ATQM

Kuyruk yöneticisi.

SAATUNIX

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 değişimin değeri ortamdan özgüdür.

ATUNK

Bilinmeyen uygulama tipi.

Bu değer, diğer bağlam bilgileri mevcut olsa da, uygulama tipinin bilinmediğini göstermek için kullanılabilir.

ATUFST

Kullanıcı tanımlı uygulama tipi için en düşük değer.

ATULST

Kullanıcı tanımlı uygulama tipi için en yüksek değer.

Aşağıdaki özel değer de oluşabilir:

ATNCON

İletide bağlam bilgisi yok.

Bu değer, 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 değer için test edilebilir (örneğin, diğer 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 değere ayarlanır. Note that on IBM i, it is set to AT400; the queue manager never uses ATCICS on IBM i.

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 yoksayılr 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* değerinin (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 değer 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:

• 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 *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 kaldırılincaya 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 kurtarılabildiği 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 bulunsun bile geçerlidir.

Paylaşılan kuyrukların özel durumunda, kalıcı olmayan iletiler *do* kuyruk-paylaşım grubundaki kuyruk yöneticilerinin yeniden başlatılmasını sağlar, ancak paylaşılan kuyruklarda iletileri saklamak için kullanılan bağlaşım tesisinin başarısızlıklarını atlamaz.

PPEQDEF

İleti varsayılan kalıcılığı içeriyor.

- Kuyruk bir küme kuyruğıysa, iletinin kalıcılığı, iletinin yerleştirdiği kuyruğun belirli bir örneğinin sahibi olan *hedef* kuyruk yöneticisinde tanımlanan *DefPersistence* özneliğinden alınır. Genellikle, bir küme kuyruğunun tüm eşgörünümlerinin *DefPersistence* özneliği 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.

- If the queue is not a cluster queue, the persistence of the message is taken from the *DefPersistence* attribute defined at the *yerele* queue manager, even if the destination queue manager is remote.

Kuyruk adı çözümleme yolunda birden çok tanımlama varsa, yoldaki *ilk* tanımındaki bu özniteliğin değerinden 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 bulunduğu, *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 letisi için uygulamaların, istek letisinin kalıcılık 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, değer 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 eşgörünümünün *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, ileti için öncelik, hedef kuyruk yöneticisi uzak olsa bile, *yerel* kuyruk yöneticisinde tanımlandığı şekilde *DefPriority* özniteliğinden alınır.

Kuyruk adı çözümleme yolunda birden çok tanımlama varsa, yoldaki *ilk* tanımlamasındaki bu özniteliğin değerinden varsayılan öncelik 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 bulunduğu, *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 özniteliği 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 çağrılarına ilişkin bir giriş alanı. Bu alanın ilk değeri PRQDEF olur.

MDPT (8 baytlık karakter dizilimi)

İletin konulduğu saat.

Bu, iletin **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 bulunduğu 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 konulursa, iletiyle birlikte gönderilen *MDPT* ögesini 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. İletin 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 bildirimi (PAN)
- Negatif işlem bildirimi (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 raporlarına geçiş yapamazsa, kuyruk yöneticisi yöneticileri tarafından sağlanan bir API geçiş çıkışı kullanılarak bu rapor açılabilir.

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 aracılığıyla oluşturulabilir. Örneğin, hedef kuyruk ya da ara iletim kuyruğu dolu olabilir ya da ileti kuyruk için çok büyük olabilir.

Kural dışı durum raporu iletilerinin oluşturulması, özgün iletinin kalıcılarına ve özgün iletinin seyahat ettiği ileti kanalının (olağan ya da hızlı) hıza bağlıdır.

- Tüm kalıcı iletiler ve olağan ileti kanalları üzerinden seyahat eden kalıcı olmayan iletiler için, hata koşuluna ilişkin gönderme uygulaması tarafından belirtilen işlem başarıyla tamamlanabiliyorsa,

kural dışı durum raporu *yalnızca* oluşturulur. 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 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, RODLQ belirtilirse, ancak ilk ileti kuyruğun dolu olduğu için (say), kural dışı durum raporu iletisi 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 iletisine dahil edilmiyor.

ROEXC, ROEXCD ve ROEXCF ' den birden fazlasını belirtmeyin.

ROEXCD

Veri girilmesi zorunlu olan kural dışı durum raporları.

Bu, özgün iletiden gelen uygulama iletisi verilerinin ilk 100 baytı rapor iletisine dahil etmek dışında, ROEXC ile aynıdır. Özgün ileti bir ya da daha çok MQ üstbilgisi yapısı içeriyorsa, bunlar rapor iletisine 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 iletisi verilerinin rapor iletisine dahil olması dışında, ROEXC ile aynıdır.

ROEXC, ROEXCD ve ROEXCF ' den birden fazlasını belirtmeyin.

Süre bitimi seçenekleri: Bir süre bitimi rapor iletisi 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 üstbilgisi yapısı 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-gelism 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 raporlarda 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 üstbilgisi yapısı 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 atmayacağınızı (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 raporsa, 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 kalmaz.
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 alınırsa, 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.

ROCODD

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 ROCODD 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 raporu iletisi göndermesini istemek için aşağıdaki seçeneklerden birini ya da her ikisini belirleyebilirsiniz.

ROPAN

Olumlu eylem bildirimi 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 bildirimi raporları gerekli.

Bu rapor tipi, iletiyi alan ve üzerinde işlem yapan uygulama tarafından oluşturulur. İletide istenen işlemin *değil* işlemi başarıyla gerçekleştirildiğini 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 sorumluluğundur.

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 varsayılır.

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 *CorrelId* 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* , *SOSCID* 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, ROCMTC 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 istendiyse, 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 işlemdir 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 istendiyse, bir kural dışı durum raporu iletisi üretilir.
- Özgün iletiyi koyan uygulama bir konuya yerleştirildi

Gönderen tarafından istendiyse, bir kural dışı durum raporu 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 istendiyse, bir kural dışı durum raporu iletisi üretilir.
- Özgün iletiyi koyan uygulama bir konuya yerleştirildi

Gönderen tarafından istendiyse, bir kural dışı durum raporu iletisi oluşturulacak.

Özgün iletiyi gönderene geri vermek isteniyorsa, özgün ileti çıkış kuyruğu 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 aracı (MCA) tarafından hiçbir rapor iletisi oluşturulmadı.

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 1395](#).

Bir rapor iletisi istenirse, raporun gönderilmesi gereken kuyruğun adı, *MDRQ* alanında belirtilmelidir. Bir rapor iletisi alındığında, raporun niteliği, ileti tanımlayıcısındaki *MDFB* alanı incelenerek saptanabilir.

2. Bir rapor iletisi oluşturan kuyruk yöneticisi ya da MCA, rapor iletisini yanıt kuyruğuna (örneğin, yanıt kuyruğu ya da iletim kuyruğu dolu olduğu için) koyamıyorsa, rapor iletisi ölü harf kuyruğunda yerine yerleştirilir. Aynı zamanda da başarısız olursa ya da herhangi bir ölü harf kuyruğu yoksa, işlem, rapor iletisinin tipine bağlıdır:
 - Rapor iletisi bir kural dışı durum bildirimiye, kural dışı durum raporunun oluşturulmasına neden olan ileti, iletim kuyruğunda bırakılır; bu, iletinin kaybolmamasını sağlar.
 - Diğer tüm rapor tipleri için, rapor iletisi 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 iletisi 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 iletinin *MQMD* ' de *MDUID* yetki yetkisini kullanarak rapor iletilisi açılır:

- Alıcı bir MCA tarafından oluşturulan kural dışı durum raporları, rapora neden olan iletiyi yerleştirmeye çalıştığında, 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 iletilide 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 iletinin 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 iletinin *MDFMT* değeri *FMXQH* ise, rapordaki veriler *MQXQH* ' yi içermez. Rapor verileri, özgün iletilide *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 iletinin teslim edildiği, teslim edildiği ya da gerektiği gibi süresi dolduğu garanti edilir. Ancak, bu rapor iletililerinden biri ya da daha fazlası istenirse ve *değil* alındıysa, aşağıdakilerden biri gerçekleşmiş olabileceği için ters varsayılmaz:

- 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 alınmadır.

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 iletili 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, NPMGDLV ya da PMSGDLV (iletinin kalıcısına bağlı olarak) ayarına bağlıdır. Ayar, hedef kuyruğa ileti tesliminin başarılı olması gerekiyorsa (örneğin, kalıcı bir aboneye kalıcı bir ilettir ve PMSGDLV ALL ya da ALLDIR olarak ayarlanır), başarı aşağıdaki ölçütlerden biri olarak tanımlıdır:
 - Abone kuyruğuna başarılı bir şekilde yerleştirilecek
 - Abone kuyruğu iletili alamazsa, RODLQ ve Dead-letter kuyruğuna başarılı bir şekilde konun.
 - Abone kuyruğu iletili alamazsa RODISC ' yi kullanın.

İleti bölümleri için rapor iletileri:

1. Kesimlemeye izin verilen iletiler için rapor iletileri istenebilir (MFSEGA işaretinin açıklamasına bakın). Kuyruk yöneticisi iletili bölümlemek için gerekli bulursa, ilgili koşulla daha sonra karşılaştıran her bir kesim için bir rapor iletili oluşturulabilir. Bu nedenle, istekte bulunulan her bir rapor iletili tipi için birden çok rapor iletili almak üzere uygulamalar hazırlanmalıdır. Rapor iletilisinde yer alan *MDGID* alanı, özgün iletinin 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 GMLOGO kullanılıyorsa, arka arkaya gelen MQGET çağrılarını tarafından *farklı tiplerde* 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 dönebilmesini sağlar. Bu, GMCMM seçeneği kullanılarak (isteğe bağlı olarak GMATM ' de) önlenebilir. 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 iletili 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, yalnızca *ilk* bölümündeki rapor seçenekleri kuyruk yöneticisi tarafından 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 iletili oluşturulursa, kuyruk yöneticisi ya da MCA varsayılan olarak rapor iletilisinde 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 iletilisiyle veri istenerek önlenebilir. Ancak, RO* D belirtilirse, rapor iletilisi bir kuyruk yöneticisi tarafından oluşturulursa ya da segmentasyonu desteklemeyen bir MCA tarafından oluşturulduysa, rapor iletilisini alan uygulamaya *küçüktür* 100 bayt uygulama iletili verisi döndürülebileceğini unutmayın.

Rapor iletili için ileti tanımlayıcısının içeriği: Kuyruk yöneticisi ya da ileti kanalı aracısı (MCA) bir rapor iletili oluşturduğunda, ileti tanımlayıcısındaki alanları aşağıdaki değerlere ayarlar ve sonra iletili olağan şekilde yerleştirir.

MQMD ' de alan	Kullanılan değer
<i>MDSID</i>	MDSIDV
<i>MDVER</i>	MDVER2

MQMD ' de alan	Kullanılan değ�r
<i>MDREP</i>	ROYOK
<i>MDMT</i>	MTRPRT
<i>MDEXP</i>	EIULIM
<i>MDFB</i>	Raporun niteine uygun olarak (FBCOA, FBCOD, FBEXP ya da RC* değ�ri)
<i>MDENC</i>	�zg�n ileti tanımlayıcısından kopyalandı
<i>MDCSI</i>	�zg�n ileti tanımlayıcısından kopyalandı
<i>MDFMT</i>	�zg�n ileti tanımlayıcısından kopyalandı
<i>MDPRI</i>	�zg�n ileti tanımlayıcısından kopyalandı
<i>MDPER</i>	�zg�n ileti tanımlayıcısından kopyalandı
<i>MDMID</i>	�zg�n ileti tanımlayıcısındaki rapor se�enekleri tarafından belirlendiđi gibi
<i>MDCID</i>	�zg�n ileti tanımlayıcısındaki rapor se�enekleri tarafından belirlendiđi gibi
<i>MDBOC</i>	0
<i>MDRQ</i>	Boşluklar
<i>MDRM</i>	Kuyruk yöneticisinin adı
<i>MDUID</i>	PMPASI se�eneđi tarafından belirlenen se�enek
<i>MDACC</i>	PMPASI se�eneđi tarafından belirlenen se�enek
<i>MDAID</i>	PMPASI se�eneđi tarafından belirlenen se�enek
<i>MDPAT</i>	ATQM ya da ileti kanalı aracısına uygun olan
<i>MDPAN</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>MDPD</i>	Rapor iletisinin g�nderildiđi tarih
<i>MDPT</i>	Rapor iletisinin g�nderildiđi saat
<i>MDAOD</i>	Boşluklar
<i>MDGID</i>	�zg�n ileti tanımlayıcısından kopyalandı
<i>MDSEQ</i>	�zg�n ileti tanımlayıcısından kopyalandı
<i>MDOFF</i>	�zg�n ileti tanımlayıcısından kopyalandı
<i>MDMFL</i>	�zg�n ileti tanımlayıcısından kopyalandı
<i>MDOLN</i>	OLUNDF deđilse, �zg�n ileti tanımlayıcısından kopyalanır ve �zg�n ileti verilerinin uzunluđuna g�re ayarlanır

Aşađıdakilerin dıřında, benzer değ rleri ayarlamak i in rapor oluřturan bir uygulama  nerilir:

- *MDRM* alanı boşluklara ayarlanabilir (kuyruk yöneticisi, ileti konuđunda bu değ ri yerel kuyruk yöneticisinin adıyla deđiřtirecektir).
- Bađlam alanları, genellikle PMPASI gibi bir yanıt i in kullanılmıř olan se enek kullanılarak ayarlanmalıdır.

Rapor alanının çözümlenmesi: *MDREP* alanı alt alanlar içerir; bunun nedeniyle, ileti gönderenin belirli bir rapor istemesinin "Rapor alanının çözümleniyor" sayfa 1396' de açıklanan tekniklerden birini kullanması gerekip gerekmediğini denetlemesi gereken uygulamalar.

Bu, MQGET çağrısına ilişkin bir çıkış alanıdır ve MQPUT ve MQPUT1 çağrılarına ilişkin bir giriş alanı. Bu alanın ilk değeri RONONE olur.

MDRM (48 baytlık karakter dizgisi)

Yanıt kuyruğu yöneticisinin adı.

Bu ad, yanıt iletilisinin ya da rapor iletilisinin 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* özniteliğinin 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 çağrılarına ilişkin bir giriş alanı. Bu alanın uzunluğu LNQMNM 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 tanımlı kuyruk yöneticisinde tanımlı olan bir kuyruğun yerel adıdır. Kuyruk yöneticisi, ileti konusunda 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 iletilisi 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 *MDRQ* adını kendi kuyruk tanımlamalarında 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 iletilisi gerektiren bir ileti teslim edilemezse ve rapor iletilisi de belirtilen kuyruğa teslim edilemezse, hem özgün ileti, hem de rapor iletilisi ölü-mektup (teslim edilemeyen ileti) kuyruğuna gider

(“Kuyruk yöneticisine ilişkin öznitelikler” sayfa 1362 ' ta açıklanan *DeadLetterQName* özniteliğini göreceldir).

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 *değil* belirtildi.

Bunlar, rapor iletisi olmayan iletler için bu çağrılar 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 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.

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 ortamdan saptanan bir kullanıcı kimliğini kullanır.

Kullanıcı kimliği ortamdan saptanırken:

- 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 *yayınlanmamış* 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:

- *sahip* ' in başarılı bir GU çağrısı yaptığı ileti BMP ve ileti IFP bölgeleri
- MPP bölgeleri

Kuyruk yöneticisi aşağıdakilerden birini kullanır:

- İletiyile ilişkilendirilen oturum açmış kullanıcı kimliği
- Mantıksal uçbirim (LTERM) adı
- Bölge JES JOB kartından kullanıcı kimliği
- TSO kullanıcı kimliği
- PSB adı

- On IBM i, the queue manager uses the name of the user profile associated with the application job.
- On HP Integrity NonStop Server, the queue manager uses the MQSeries principal that is defined for the Tandem user identifier in the MQSeries principal database.
- On UNIX sistemleri, the queue manager uses:
 - Uygulamanın oturum açma adı
 - Oturum açma kullanılmıyorsa, işlemin etkin kullanıcı kimliği
 - Uygulama bir CICS hareketiye, işlemle ilişkili kullanıcı kimliği
- VSE/ESA' da bu ayrılmış bir alandır.
- Windowsüzerinde, 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.

Çizelge 148. MQMD 'deki alanların ilk değerleri (devamı var)

Alan adı	Değişmezin adı	Değişmezin değeri
MDOFF	Yok	0
MDMFL	MFYOK	0
MDOLN	OUNDF	-1
Notlar: 1. ↗ simgesi tek bir boş karakteri temsil eder.		

RPG bildirimi

```

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                          00000000000000000000-
D                          000000000000')
D* Correlation identifier
D MDCID         73     96    INZ(X'00000000000000-
D                          00000000000000000000-
D                          000000000000')
D* Backout counter
D MDBOC         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(X'00000000000000-
D                          00000000000000000000-
D                          00000000000000000000-
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(X'0000000000000000-
D				00000000000000000000-
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)

MQMDE-İleti tanımlayıcı uzantısı

Genel Bakış

Amaç: MQMDE yapısı, bazen uygulama iletisi 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 özniteliği 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 ileti verilerinin 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 ileti verileri 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 1118](#)
- [“MQGET çağrısıyla döndürülen MQMDE” sayfa 1119](#)
- [“İletim kuyruklarına ilişkin iletilerde MQMDE” sayfa 1119](#)
- [“Alanlar” sayfa 1120](#)
- [“Başlangıçtaki değerler” sayfa 1121](#)
- [“RPG bildirimi” sayfa 1122](#)

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 150 sayfa 1121](#).

Uygulama bir version-2 MQMD ve örnekleri sağlıyorsa, uygulama ileti verilerini bir MQMDE ile ekler; yapılar, [Çizelge 149 sayfa 1119](#) içinde gösterildiği gibi işlenir.

Çizelge 149. 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 *değil*, MQPUT ve MQPUT1 çağrılarının çıktısındaki MQMD 'deki eşdeğer alanlara herhangi bir değer döndürür; 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 yoksayılr. 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 149 sayfa 1119](#) 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 nesnelerinin güvenliği](#) konusuna bakın).

İletim kuyruklarına ilişkin iletilerde MQMDE

İletim kuyruklarındaki iletiler, bir version-1 MQMD içinde bulunan MQXQH yapısıyla önlenir. MQXQH yapısı ve uygulama ileti verileri arasında konumlandırılmış bir MQMDE da var olabilir; ancak, yalnızca MQMDE içindeki bir ya da daha çok alanın varsayılan değeri olmayan bir değer varsa geçerli olur.

MQXQH yapısı ile uygulama ileti verileri 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 iletisi verileri

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, 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.

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-İleti tanımlayıcısı” sayfa 1076](#) 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 değeri MEFON değeridir.

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 değere ayarlamalıdır. Kuyruk yöneticisi bu alanın geçerli olup olmadığını denetmiyor. Biçim adlarıyla ilgili daha fazla bilgi için [“MQMD-İleti tanımlayıcısı” sayfa 1076](#) içinde açıklanan MDFMT alanına bakın.

Bu alanın ilk değeri FMNONE ' dir.

MEGID (24 baytlık bit dizesi)

Grup tanıtıcısı.

[“MQMD-İleti tanımlayıcısı” sayfa 1076](#) içinde açıklanan MDGID alanına bakın. Bu alanın ilk değeri GINONE 'tır.

MELEN (10 basamaklı işaretli tamsayı)

MQMDE yapısının uzunluğu.

Aşağıdaki değer tanımlıdır:

MELEN2

version-2 ileti tanımlayıcı uzantısı yapısının uzunluğu.

Bu alanın ilk değeri MELEN2' dir.

MEFL (10 basamaklı işaretli tamsayı)

Mesaj bayrakları.

"MQMD-İleti tanımlayıcısı" sayfa 1076 içinde açıklanan *MDMFL* alanına bakın. Bu alanın ilk değeri MFNONE 'dir.

MEOFF (10 basamaklı işaretli tamsayı)

Mantıksal iletinin başlangıcındaki fiziksel iletilerde verilerin görelî konumu.

"MQMD-İleti tanımlayıcısı" sayfa 1076 içinde açıklanan *MDOFF* alanına bakın. Bu alanın ilk değeri 0 'tır.

MEOLN (10 basamaklı işaretli tamsayı)

Özgün iletinin uzunluğu.

"MQMD-İleti tanımlayıcısı" sayfa 1076 içinde açıklanan *MDOLN* alanına bakın. Bu alanın ilk değeri OLUNDF ' dir.

MESEQ (10 basamaklı işaretli tamsayı)

Grup içindeki mantıksal iletinin sıra numarası.

"MQMD-İleti tanımlayıcısı" sayfa 1076 içinde açıklanan *MDSEQ* alanına bakın. Bu alanın ilk değeri 1 'dir.

MESID (4 baytlık karakter dizilimi)

Yapı tanıtıcısı.

Değer şu olmalıdır:

MESIDV

İleti tanımlayıcı uzantısı yapısına ilişkin tanıtıcı.

Bu alanın ilk değeri 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 150. MQMDE içindeki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
MESID	MESIDV	'MDE↵'
MEVER	MEVER2	2

Çizelge 150. MQMDE içindeki alanların ilk değerleri (devamı var)

Alan adı	Değişmezin adı	Değişmezin değeri
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:

1. - 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                      00000000000000000000-
D                      000000000000')
D* Sequence number of logical messagewithin group
D MESEQ         57     60I 0 INZ(1)
D* Offset of data in physical messagefrom 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)

```

MQMHBO-İleti tanıtıcısı-arabellek seçenekleri

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 değiştirmedir.

Karakter kümesi ve kodlama: MQMHBO içindeki veriler, uygulamanın uygulama ve kodlamasının (ENNAT) karakter kümesinde yer almalıdır.

- “Alanlar” sayfa 1123
- “Başlangıçtaki değerler” sayfa 1123
- “RPG bildirimi” sayfa 1124

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:

MPPRRF

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).

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 151. 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)
```

MQOD-Nesne tanımlayıcısı

MQOD yapısı, ada göre bir nesneyi belirtmek için kullanılır.

Genel Bakış

Amaç: Aşağıdaki nesne tipleri geçerlidir:

- Kuyruk ya da dağıtım listesi
- Ad Listesi
- Süreç tanımlaması
- Kuyruk yöneticisi
- Konu

Yapı, MQOPER ve MQPUT1 çağrılarında bir giriş/çıkış değiştiricidir.

Sürüm: MQOD 'nin yürürlükteki sürümü ODVER4' tir. Yalnızca yapının daha yeni sürümlerinde var olan alanlar, izleyen açıklamalarda olduğu gibi tanımlanır.

Sağlanan COPY dosyası, ortam tarafından desteklenen en son MQOD sürümünü içerir, ancak ODVER alanının ilk değeri ODVER1olarak ayarlıdır. version-1 yapısında mevcut olmayan alanları kullanmak için, uygulamanın ODVER alanını, gereken sürümün sürüm numarasına ayarlaması gerekir.

Bir dağıtım listesi açmak için ODVER , ODVER2 ya da sonraki bir değer olmalıdır.

Karakter kümesi ve kodlama: Data in MQOD 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 1124](#)
- [“Başlangıçtaki değerler” sayfa 1131](#)
- [“RPG bildirimi” sayfa 1132](#)

Alanlar

MQOD yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

ODASI (40 baytlık bit dizesi)

Diğer güvenlik tanıtıcısı.

Bu, uygun yetki denetimlerinin gerçekleştirilmesine olanak sağlamak için yetki hizmetine ODAU ile geçirilen bir güvenlik tanıtıcısıdır. ODASI yalnızca aşağıdaki durumlarda kullanılır:

- OOALTU, MQOPEN çağrısında belirtilir ya da
- PMALTU, MQPUT1 çağrısında belirtilir,

ve the *ODAU* field is not entirely blank up to the first null character or the end of the field.

ODASI 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:

ITWNT

Windows güvenlik tanıtıcısı.

DİZMEYEN

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 değer kullanılabilir:

SATIR

Güvenlik tanıtıcısı belirtilmedi.

Değer, alanın uzunluğuna ilişkin ikili sıfır değeridir.

Bu bir giriş alanıdır. Bu alanın uzunluğuna *LNSCID* değeri verilir. Bu alanın ilk değeri *SINONE* 'tır. *ODVER* , *ODVER3* değerinden küçükse bu alan yok sayılır.

ODAU (12 baytlık karakter dizilimi)

Diğer kullanıcı kimliği.

If *OOALTU* is specified for the *MQOPEN* call, or *PMALTU* for the *MQPUT1* call, this field contains an alternate user identifier that is to be used to check the authorization for the open, in place of the user identifier that the application is currently running under. Ancak, bazı denetimler yürürlükteki kullanıcı kimliğiyle (bağlam denetimleri gibi) hala gerçekleştirilmeye devam eder.

OOALTU ve *PMALTU* belirtilmemişse ve bu alan, ilk boş karakteri ya da alanın sonuna kadar tamamen boşsa, bu nesneyi belirtilen seçeneklerle açmak için kullanıcı yetkisi gerekmiyorsa, açma işlemi başarılı olabilir.

AOALTU ya da *PMALTU* belirtilmiyorsa, bu alan yoksayılr.

Bu bir giriş alanıdır. Bu alanın uzunluğu *LNUID* tarafından verilir. Bu alanın ilk değeri 12 boş karakterdir.

ODDN (48 baytlık karakter dizilimi)

Dinamik kuyruk adı.

Bu, *MQOPEN* çağrısıyla yaratılacak dinamik bir kuyruğun adıdır. This is of relevance only when *ODON* specifies the name of a model queue; in all other cases *ODDN* is ignored.

Adda geçerli olan karakterler, bir yıldız işaretinin de geçerli olması dışında, *ODON* için geçerli olan karakterlerle aynıdır. A name that is blank (or one in which only blanks are shown before the first null character) is not valid if *ODON* is the name of a model queue.

Addaki son boş olmayan karakter bir yıldız işaretiyse (*), kuyruk yöneticisi, yerel kuyruk yöneticisinde kuyruk için oluşturulan adın benzersiz olduğunu garanti eden bir karakter dizisiyle 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.

Bu bir giriş alanıdır. Bu alanın uzunluğu *LNQN* tarafından verilir. Bu alanın ilk değeri ' *AMQ. ** ' olur ve boşluklarla doldurulur.

ODIDC (10 basamaklı işaretli tamsayı)

Açılmayan kuyrukların sayısı.

Dağıtım listesindeki kuyrukların sayısı, başarıyla açılmamaktadır. Bu alan varsa, dağıtım listesinde olmayan tek bir kuyruk açılırken da bu alan ayarlanır.

Not: If present, this field is set *yalnızca* if the *CMPCOD* parameter on the MQOPEN or MQPUT1 call is CCOK or CCWARN; it is *değil* set if the *CMPCOD* parameter is CCFAIL.

Bu bir çıkış alanıdır. Bu alanın ilk değeri 0 'tır. *ODVER* , *ODVER2*değerinden küçükse bu alan yok sayılır.

ODKDC (10 basamaklı işaretli tamsayı)

Yerel kuyrukların sayısı başarıyla açıldı.

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. *ODVER* , *ODVER2*değerinden küçükse bu alan yok sayılır.

ODMN (48 baytlık karakter dizilimi)

Nesne kuyruğu yöneticisi adı.

Bu ad, *ODON* nesnesinin tanımlandığı kuyruk yöneticisinin adıdır. Adda geçerli olan karakterler, *ODON* ile aynıdır (daha önce bakın). 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:

- *ODOT* *OTTOP*, *OTNLST*, *OTPRO* ya da *OTQM* ise, *ODMN* boş bırakılmalı ya da yerel kuyruk yöneticisinin adı olmalıdır.
- If *ODON* 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 *ODMN* 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.
- *ODON* bir küme kuyruğunun adıysa ve *ODMN* boşsa, kuyruk yöneticisi (ya da kuruluysa, küme iş yükü çıkışı) tarafından döndürülen kuyruk tanıtıcısı kullanılarak gönderilen kuyruk tanıtıcısı kullanılarak gönderilen iletilerin gerçek hedefi aşağıdaki gibi seçilir:
 - *OOBND0* belirtilirse, kuyruk yöneticisi, *MQOP* çağrısının işlenmesi sırasında küme kuyruğunun bir eşgörünümünü seçer ve bu kuyruk tanıtıcısı kullanılarak yapılan tüm iletiler o yönetim ortamına gönderilir.
 - *OOBNDN* değeri belirlenirse, 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, uygulama, *ODMN* alanında o kuyruk yöneticisinin adını belirtmelidir. Bu, yerel kuyruk yöneticisini, iletiyi belirtilen hedef kuyruk yöneticisine göndermesini sağlar.

- Açılmakta olan nesne bir dağıtım listese (yani, *ODREC* sıfırdan büyükse), *ODMN* boş olmalı ya da boş dizgi olmalıdır. Bu koşul karşılanmazsa, çağrı neden kodu RC2153 ile başarısız olur.

Bu, *ODON* , bir model kuyruğunun adı ve diğer tüm durumlarda yalnızca giriş alanı olduğunda, *MQOPER* çağrısına ilişkin bir giriş/çıkış alanıdır. Bu alanın uzunluğu *LNQMN* tarafından verilir. Bu alanın ilk değeri 48 boş karakterdir.

ODON (48 baytlık karakter dizilimi)

Nesne adı.

Bu, *ODMN* ile tanıtilan kuyruk yöneticinde 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 (/), 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:

- EBCDIC Katakana kullanan sistemlerde küçük harfli karakterler kullanılamaz.
- IBM üzerinde, komutlar üzerinde belirtildiğinde, küçük harf, eğik çizgi ya da yüzde içeren adların tırnak işareti içine alınması gerekir. Bu tırnak işaretleri, yapılarda alan olarak ya da çağrılarda parametre olarak ortaya çıkan adlar için belirtilmemelidir.

Aşağıdaki noktalar, gösterilen nesne tipleri için geçerlidir:

- *ODON* bir model kuyruğunun adıysa, kuyruk yöneticisi, model kuyruğunun öznitelikleriyle dinamik bir kuyruk yaratır ve *ODON* alanında, yaratılan kuyruğun adını döndürür. Model kuyruğu yalnızca *MQOPEN* çağrısında belirtilebilir; *MQPUT1* çağrısında bir model kuyruğu geçerli değildir.
- Açılmakta olan nesne bir dağıtım listesi (yani, *ODREC* mevcut ve sıfırdan büyük), *ODON* boş olmalı ya da boş dizgi olmalıdır. Bu koşul karşılanmazsa, çağrı neden kodu RC2152 ile başarısız olur.
- *ODOT* *OTQM* ise, özel kurallar geçerli olur; bu durumda adın ilk boş karakteri ya da alanın sonuna kadar tamamen boş olması gerekir.
- *ODON*, *TARGETYPE* (*KONU*) içeren bir diğer ad kuyruğunun adıysa, diğer ad kuyruklarının kullanımı için olağan olduğu gibi, adı belirtilen diğer ad kuyruğunda bir güvenlik denetimi yapılır. Bu güvenlik denetimi başarılı olursa, bu *MQOPEN* çağrısı devam eder ve denetim konusu nesnesine ilişkin bir güvenlik denetimi yapmak da dahil olmak üzere, bir *OTTOP* 'un *MQOI* 'ı gibi hareket eder.

Bu, *ODON* , bir model kuyruğunun adı ve diğer tüm durumlarda yalnızca giriş alanı olduğunda, *MQOPER* 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 oluşturulabiliyor: *ODON* ve *ODOS*. Bu iki alanın nasıl kullanılma hakkında ayrıntılı bilgi için bkz. "[Konu dizelerinin kullanılması](#)" sayfa 560.

ODORO (10 basamaklı işaretli tamsayı)

MQOD başlangıcından ilk nesne kaydının kayması.

Bu, *MQOD* yapısının başlangıcındaki ilk *MQOR* nesne kaydının bayt cinsinden görelî konudur. Görelî konum pozitif ya da negatif olabilir. *ODORO* , yalnızca bir dağıtım listesi açılırken kullanılır. *ODREC* sıfırsa, alan yoksayılır.

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:

- By using the offset field *ODORO*

Bu durumda, uygulama bir *MQOD* içeren kendi yapısını, izleyen *MQOR* kayıtları dizisini (gereken sayıda dizi ögesi ile) bildirmelidir ve *ODORO* değerini, *MQOD* 'nin başlangıcındaki dizideki ilk ögenin görelî konumuna ayarlamalıdır. Bu görelî konumun doğru olduğundan emin olmak için dikkatli olmanız gerekir.

- *ODORP* işaretçi alanını kullanarak

Bu durumda uygulama, *MQOR* yapılarının dizisini, *MQOD* yapısından ayrı olarak bildirebilir ve *ODORP* 'yi dizinin adresine ayarlayabilir.

Whichever technique is chosen, one of *ODORO* and *ODORP* must be used; the call fails with reason code RC2155 if both are zero, or both are nonzero.

Bu bir giriş alanıdır. Bu alanın ilk değeri 0 'tır. *ODVER* , *ODVER2* değerinden küçükse bu alan yok sayılır.

ODORP (işaretçi)

İlk nesne kaydının adresi.

Bu, ilk MQOR nesne kaydının adresidir. *ODORP* , yalnızca bir dağıtım listesi açılırken kullanılır. *ODREC* sıfırsa, alan yoksayılır.

Bu bir giriş alanıdır. Bu alanın başlangıç değeri, boş değerli göstergedir. Nesne kayıtlarını belirtmek için *ODORP* ya da *ODORO* kullanılabilir, ancak her ikisi de değil; ayrıntılar için daha önce *ODORO* alanının açıklamasına bakın. *ODORP* kullanılmazsa, boş (null) işaretçi ya da boş değerli (null) byte olarak ayarlanmalıdır. *ODVER* , *ODVER2* değerinden küçükse bu alan yok sayılır.

ODOS (MQCHARV)

ODOS, kullanılacak uzun nesne adını belirtir.

Bu alana yalnızca belirli *ODOT* değerleri için başvuruluyor. Hangi değerlerin bu alanın kullanıldığını gösterdiğine ilişkin ayrıntılar için *ODOT* ' in açıklamasına bakın.

If *ODOS* is specified incorrectly, according to the description of how to use the *MQCHARV* structure, or if it exceeds the maximum length, the call fails with reason code RC2441.

Bu bir giriş alanıdır. Bu yapıdaki alanların ilk değerleri, *MQCHARV* yapısındaki alanlarla aynıdır.

Tam konu adı iki farklı alandan oluşturulabiliyor: *ODON* ve *ODOS*. Bu iki alanın nasıl kullanılma hakkında ayrıntılı bilgi için bkz. “Konu dizelerinin kullanılması” sayfa 560. *ODVER* , *ODVER4* değerinden küçükse bu alan yok sayılır.

ODOT (10 basamaklı işaretli tamsayı)

Nesne tipi.

ODON içinde adlandırılmakta olan nesnenin tipi. Olası değerler şunlardır:

OTQ

Kuyruk. Nesnenin adı *ODON* içinde bulunur.

OTNLST

İsim listesi. Nesnenin adı *ODON* içinde bulunur.

OTPRO

Süreç tanımlaması. Nesnenin adı *ODON* içinde bulunur.

OTQM

Kuyruk yöneticisi. Nesnenin adı *ODON* içinde bulunur.

OTTOP

Konu. Tam konu adı iki farklı alandan oluşturulabiliyor: *ODON* ve *ODOS*.

Bu iki alanın nasıl kullanılma hakkında ayrıntılı bilgi için bkz. “Konu dizelerinin kullanılması” sayfa 560.

ODON alanı tarafından tanımlanan nesne bulunamazsa, *ODOS* içinde belirtilmiş bir dizgi olsa bile, çağrı neden kodu RC2425 ile 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. If this number is greater than zero, it indicates that a distribution list is being opened, with *ODREC* being the number of destination queues in the list. Bir dağıtım listesi için geçerli, yalnızca bir hedef içerilir.

The value of *ODREC* must not be less than zero, and if it is greater than zero *ODOT* must be OTQ; the call fails with reason code RC2154 if these conditions are not satisfied.

Bu bir giriş alanıdır. Bu alanın ilk değeri 0 'tır. *ODVER* , *ODVER2* değerinden küçükse bu alan yok sayılır.

ODRMN (48 baytlık karakter dizilimi)

Çözölmüş kuyruk yöneticisi adı.

Bu ad, yerel kuyruk yöneticisi tarafından ad çözme işlemi gerçekleştirildikten sonra hedef kuyruk yöneticisinin adıdır. Döndürölen ad, *ODRQN* ile tanıtılan kuyruğa sahip olan kuyruk yöneticisinin adıdır. *ODRMN* , yerel kuyruk yöneticisinin adı olabilir.

ODRQN , yerel kuyruk yöneticisinin ait olduđu kuyruk paylaşım grubunun sahip olduđu bir paylaşılan kuyruksa, *ODRMN* kuyruk paylaşım grubunun adıdır. If the queue is owned by some other queue-sharing group, *ODRQN* 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 değeri yalnızca, nesne göz atma, giriş ya da çıkış (ya da herhangi bir birleşim) için açılmış tek bir kuyruksa döndürölür. Açılan nesne aşğıdakilerden herhangi biriye, *ODRMN* boşluklara ayarlanır:

- Kuyruk değeri
- Bir kuyruk, ancak göz atma, giriş ya da çıkış için açılmadı
- Belirtilen OOBNDN değeriye sahip bir küme kuyruđu (ya da *DefBind* kuyruk özniteliđi BNDNOT değeriye sahip olduđuunda OOBNDQ geçerli olur.)
- Bir dağıtım listesi

Bu bir çıkış alanıdır. Bu alanın uzunluđu LNQN tarafından verilir. Bu alanın başlangıç değeri, C ' de boş değeri dizgisidir ve diđer programlama dillerindeki 48 boş karakter dizilimidir. *ODVER* , *ODVER3* değeriinden küçöксе bu alan yok sayılır.

ODRO (MQCHARV)

ODRO, kuyruk yöneticisi *ODON* ' ta sağlanan adı çözdükten sonra uzun nesne adıdır.

Bu alan yalnızca, bir konu nesnesine gönderme yapan belirli nesne, konu ve kuyruk diđer adları için döndürölür.

ODOS ' ta uzun nesne adı sağlanırsa ve *ODON* ' ta hiçbir şey sağlanmıyorsa, bu alanda döndürölen değeri, *ODOS* ' de sağlanan değeriyle aynıdır.

Bu alan atlanırsa (*ODRO.VSBufSize* sıfırdır), *ODRO* döndürölmez, ancak uzunluk *ODRO.VSLength* ' ta döndürölür. If the length is shorter than the full *ODRO* then it is truncated and returns as many of the rightmost characters as can fit in the provided length.

ODRO yanlış belirtilirse, *MQCHARV* yapısının nasıl kullanılacağı ya da uzunluk üst sınırını aştıysa, çağrı neden kodu RC2520 ile başarısız olur. *ODVER* , *ODVER4* değeriinden küçöксе bu alan yok sayılır.

ODRQN (48 baytlık karakter dizilimi)

Çözölmüş kuyruk 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 *ODRMN*.

Boş olmayan bir değeri yalnızca, nesne göz atma, giriş ya da çıkış (ya da herhangi bir birleşim) için açılmış tek bir kuyruksa döndürölür. Açılan nesne aşğıdakilerden herhangi biriye, *ODRQN* boşluklara ayarlanır:

- Kuyruk değeri
- Bir kuyruk, ancak göz atma, giriş ya da çıkış için açılmadı
- Bir dağıtım listesi
- Bir konu nesnesine gönderme yapan bir diđer ad kuyruđu (bunun yerine “*ODRO (MQCHARV)*” sayfa 1129 belgesine bakın)

Bu bir çıkış alanıdır. Bu alanın uzunluđu LNQN tarafından verilir. Bu alanın başlangıç değeri, C ' de boş değeri dizgisidir ve diđer programlama dillerindeki 48 boş karakter dizilimidir. *ODVER* , *ODVER3* değeriinden küçöксе bu alan yok sayılır.

ODRRO (10 basamaklı işaretli tamsayı)

MQOD başlangıcından ilk yanıt kaydının kayması.

Bu, MQOD yapısının başlangıcındaki ilk MQRR yanıt kaydının bayt cinsinden görelî konutdur. Görelî konum pozitif ya da negatif olabilir. *ODRRO* , yalnızca bir dağıtım listesi açılırken kullanılır. *ODREC* sıfırda, alan yoksayılır.

Bir dağıtım listesi açılırken, açılmadığını (MQRR 'deRRCC alanı) ve her başarısızlığın nedenini (MQRR' deRRREA alanı) tanımlamak için bir ya da daha çok MQRR yanıt kaydı dizisi sağlanabilir. 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 nedenler için) ayarlar; bu durumda, çağrıdan RC2136 neden kodu belirtilir. 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 *ODREC* olması gerekir.

Yanıt kayıtları, nesne kayıtlarıyla aynı şekilde, *ODRRO* içinde bir görelî konum belirtilerek ya da *ODRRP* ' ta bir adres belirterek sağlanabilir. Bu işlemi nasıl yapacağına ilişkin ayrıntılar için *ODORO* ' in açıklamasını görürsünüz. Ancak, *ODRRO* ve *ODRRP* sayısı birden fazla kullanılamaz; her ikisi de sıfır değilse, çağrı neden kodu RC2156 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 ikincinin tamamlanma kodu CCOK ya da CCWARN ise, bu kuyruğa ilişkin açma işleminden sonra bu kuyruk için geçerli olan işlemlerin yerine geçer.

Bu bir giriş alanıdır. Bu alanın ilk değeri 0 'tır. *ODVER* , *ODVER2* değerinden küçükse bu alan yok sayılır.

ODRRP (işaretçi)

İlk yanıt kaydının adresi.

Bu, ilk MQRR yanıt kaydının adresidir. *ODRRP* , yalnızca bir dağıtım listesi açılırken kullanılır. *ODREC* sıfırda, alan yoksayılır.

Yanıt kayıtlarını belirtmek için *ODRRP* ya da *ODRRO* kullanılabilir; ayrıntılar için *ODRRO* alanının önceki açıklamalarına bakın. *ODRRP* kullanılmazsa, boş (null) işaretçi ya da boş değerli (null) byte olarak ayarlanmalıdır.

Bu bir giriş alanıdır. Bu alanın başlangıç değeri, boş değerli göstergedir. *ODVER* , *ODVER2* değerinden küçükse bu alan yok sayılır.

ODSID (4 baytlık karakter dizilimi)

Yapı tanıtıcısı.

Değer şu olmalıdır:

ODSIDV

Nesne tanımlayıcı yapısına ilişkin tanıtıcı.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri *ODSIDV* ' dir.

ODSS (MQCHARV)

ODSS, bir kuyruğun dışına ileti alınırken kullanılan seçim ölçütlerinin sağlanması için kullanılan dizgiyi içerir.

ODSS aşağıdaki durumlarda sağlanmamalıdır:

- *ODOT* OTQ değilse
- Açılmakta olan kuyruk, giriş seçeneklerinden biri kullanılarak açılmamışsa, *OOINP**

Bu durumlarda *ODSS* sağlandıysa, çağrı neden kodu RC2516 ile başarısız olur.

ODSS yanlış belirtilmişse, MÖCHARV yapısının nasıl kullanılacağı ya da uzunluk üst sınırını aşarsa, çağrı neden kodu RC2519 ile başarısız olur. ODVER , ODVER4 değerinden küçükse bu alan yok sayılır.

ODUDC (10 basamaklı işaretli tamsayı)

Başarıyla açılan uzak kuyrukların sayısı

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. ODVER , ODVER2 değerinden küçükse bu alan yok sayılır.

ODVER (10 basamaklı işaretli tamsayı)

Yapı sürüm 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 var olan alanlar, alanların açıklamalarında olduğu gibi 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 152. MÖOD 'daki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
ODSID	ODSIDV	'ODSID'
ODVER	ODVER1	1
ODOT	OTQ	1
ODON	Yok	Boşluklar
ODMN	Yok	Boşluklar
ODDN	Yok	'AMQ.*'
ODAU	Yok	Boşluklar
ODREC	Yok	0
ODKDC	Yok	0
ODUDC	Yok	0
ODIDC	Yok	0
ODORO	Yok	0
ODRRO	Yok	0

Çizelge 152. MQOD 'daki alanların ilk değerleri (devamı var)

Alan adı	Değişmezin adı	Değişmezin değeri
ODORP	Yok	Boş değerli gösterge ya da boş değerli byte
ODRRP	Yok	Boş değerli gösterge ya da boş değerli byte
ODASI	SATIR	Boş Değerler
ODRQN	Yok	Boşluklar
ODRMN	Yok	Boşluklar
ODOS	MQCHARV için tanımlandığı gibi	MQCHARV için tanımlandığı gibi
ODRO	ODOSiçinde sağlandığı gibi	ODOSiçinde sağlandığı gibi
ODSS	Yok	Boşluklar

Notlar:

1. → simgesi tek bir boş karakteri temsil eder.

RPG bildirimi

```

D*.1.....2.....3.....4.....5.....6.....7..
D*
D* MQOD 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)

```

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 özniteliği 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 1134
- “Başlangıçtaki değerler” sayfa 1134
- “RPG bildirimi” sayfa 1134

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 153. MQYA ' da alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
ORON	Yok	Boşluklar
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 1135
- “Başlangıçtaki değerler” sayfa 1137
- “RPG bildirimi” sayfa 1137

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 iletilmesine 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 hiçbirisi 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 tanıttıcısıdır; değer şöyle olmalıdır:

PSIDV

Özellik tanımlayıcı yapısına ilişkin tanıttıcı.

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 tanıttıcısı 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 alırsa, özellik şu şekilde ele alınır:

- 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 MQtanımlı olmayan özelliklere de atanır.

PDSSPR

Özelliğe ilişkin destek gereklidir. İleti, IBM MQtanımlı özelliği desteklemeyen bir kuyruk yöneticisi tarafından reddedilir. MQPUT ya da MQPUT1 çağrısı, tamamlanma kodu CCFAIL ve neden kodu RC2490ile başarısız olur.

PDSSUPL

İleti, yerel bir kuyruk için yazılmışsa, IBM MQtanımlı özelliği desteklemeyen bir kuyruk yöneticisi tarafından reddedilir. MQPUT ya da MQPUT1 çağrısı, tamamlanma kodu CCFAIL ve neden kodu RC2490ile 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 154. 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)
```

MQPMO-put-message seçenekleri

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\u00f0fcr\u00f0fcm

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 1138
- “Başlangıçtaki değerler” sayfa 1151
- “RPG bildirimi” sayfa 1152

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, açılmadı 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* değiştirgesi CCOK ya da CCWARN olduğunda ayarlanır; *CMPCOD* parametresi CCFAIL ise, bu değer 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 MQPUT1 işleminin denetlenmesini denetleyen seçenekler.

Aşağıdakilerin hiçbiri ya da hiçbiri belirlenebilir. Birden fazla değer gerekliyse, değerler eklenebilir (aynı değişmezi 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, yayınlandığı 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 değiştirilir. Bu nedenle, aynı konu üzerinde birden fazla yayınlı tutma iletileri 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 775](#) 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 iletiyle 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 iletiyle 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 ' de *MDGID* alanı) ve aynı ileti dizisi numarasına (MQMD ' de *MDSEQ* alanı) sahip iki ya da daha çok fiziksel ileten oluşur. Kesimler, mantıksal iletilerde verilerin başlangıcındaki fiziksel iletteki verilerin göreceli konumunu veren bölüm göreceli 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 ayırt edilir; burada n, gruptaki mantıksal iletilerin sayısıdır. Mantıksal iletilerden biri ya da daha fazlası bölümlendiye, grupta n fiziksel ileti sayısı daha fazla olur.

PLOGO

Mantıksal iletilerin gruplarındaki ve kısımlarındaki iletiler mantıksal sırada yer alıyor.

Bu seçenek, kuyruk yöneticisine uygulamanın, iletileri gruplar ve mantıksal iletiler 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 görelî 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 iletileri yerleştirin.
- Sonraki ileti grubuna mantıksal iletiler konmadan önce, tüm mantıksal iletileri bir ileti grubuna koyun.

Bu sipariş "mantıksal sipariş" olarak adlandırılır.

Uygulama, kuyruk yöneticisine, iletileri gruplar ve mantıksal iletiler 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, iletilerin ne zaman gruplara ait olduğunu ya da mantıksal iletilerin 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 iletileri verisi olmadan) göndererek bu bilgileri sonlandırabilir.

Çizelge 155 sayfa 1142 , 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:

GÜNLÜK ORD

Çağrıda PMLOGO seçeneğinin belirtilip belirtilmediğini belirtir.

MIG

Çağrıda MFMIG ya da MFLMIG seçeneğinin belirlenip belirlenmediğini belirtir.

SEG

Çağrıda MFSEG ya da MFLSEG seçeneğinin belirlenip belirlenmediğini belirtir.

SEÇ TAMAM

Çağrıda MFSEGA seçeneğinin belirlenip belirtilmediğini belirtir.

Cur Grp

Çağrıdan önce geçerli bir ileti grubunun var olup olmadığını belirtir.

Ccur günlük iletileri

Çağrıdan önce geçerli bir mantıksal iletinin var olup olmadığını belirtir.

Diğer kolonlar

Kuyruk yöneticisinin kullandığı değerleri gösterir. "Önceki" değeri, kuyruk tanıtıcısı için önceki ilerde 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 155. 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 iletişi	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
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

Çizelge 155. Mantıksal iletilerin gruplarındaki ve kısımlarındaki iletilere ilişkin MQPUT seçenekleri (devamı var)

Belirttiğiniz seçenekler	Aramadan önce grup ve günlük-msg durumu	Kuyruk yöneticisinin kullandığı değerler
Not: • PMLOGO, MQPUT1 çağrısında geçerli değil. • <i>MDMID</i> alanı için, kuyruk yöneticisi, PMNMID ya da MINONE belirtilirse yeni bir ileti tanıtıcısı oluşturur ve diğer bir değer olarak alandaki değeri kullanır. • <i>MDCID</i> 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 kullanılabilecek 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 karıştıramaz; 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 156 sayfa 1144 ,

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 156. 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 grupla 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, MDOFFve 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 alınırsa, 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, iletmeye 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 iletiyi 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:

MQMD ' de alan	Kullanılan değer
<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](#).

MQPUT çağrısı için, kuyruğun *OOPASI* seçeneğiyle (ya da bunu belirten bir seçenek) açılmış olması gerekir. MQPUT1 çağrısı için, *OOPASI* seçeneği ile MQOP 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 OOSETA seçeneğiyle açılmış olması gerekir. MQPUT1 çağrısı için, OOSETA seçeneği ile MQOP çağrısına aynı yetki denetimi gerçekleştirilir.

PM* bağlam seçeneklerinden yalnızca biri belirtilebilir. Bu seçeneklerden hiçbirini 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 çağrışı 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 geriletilir. Uygulama, önceki bir MQPUT ya da MQPUT1 hatasının nedenini saptamak için MQSTAT çağrısını yapabilir.

PMSRES

MQPMO yapısındaki bir put seçeneği için bu değer 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 Sürüm 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.

PFNONE, 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 *PFNONE* değerine sahipse, çağrı neden kodu RC2158 ile başarısız olur.

Bu bir giriş alanıdır. Bu alanın ilk değeri *PFNONE* '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 görelî konutdur. Görelî konum pozitif ya da negatif olabilir. *PMPRO* yalnızca ileti bir dağıtım listesine konduğunda kullanılır. *PMREC* sıfırsa, alan yoksayılr.

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, MQOD' daki nesne kayıtlarına benzer şekilde sağlanabilir. Bunu nasıl yapacağını ilişkin ayrıntılar için, “MQOD-Nesne tanımlayıcısı” sayfa 1124 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 MQPMR ' nin adresidir. *PMPRP* yalnızca ileti bir dağıtım listesine konduğunda kullanılır. *PMREC* sıfırsa, alan yoksayılr.

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 *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 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 tanımlanan kuyruğa sahip olan kuyruk yöneticisinin adıdır ve yerel kuyruk yöneticisinin adı olabilir.

PMRQN , yerel kuyruk yöneticisinin ait olduğu kuyruk paylaşım grubunun sahip olduğu bir paylaşılan 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 *LNQMN* 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 yoksayılır.

İleti bir dağıtım listesine konduğunda, iletinin başarıyla gönderilmediği kuyrukları (*MQRR* 'de *RRCC* alanı) ve her başarısızlığın nedenini (*MQRR* 'de *RRREA* 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 belirtilirse *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 yapacağını ilişkin ayrıntılar için, "MQOD- Nesne tanımlayıcısı" sayfa 1124 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 (istendiyse), 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 yok sayılır.

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 (istendiyse), 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 dizisi)

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 , PMVER2değerinden 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 157. MQPMO 'daki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
PMSID	PMSIDV	' PMO→ '
PMVER	PMVER1	1
PMOPT	PMYOK	0
PMT0	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

Çizelge 157. MQPMO 'daki alanların ilk değerleri (devamı var)

Alan adı	Değişmezin adı	Değişmezin değeri
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

```

MQPMR-Put-message kaydı

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 1153
- “Başlangıçtaki değerler” sayfa 1154
- “RPG bildirimi” sayfa 1154

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-İleti tanımlayıcısı” sayfa 1076 içindeki *MDACC* açıklamalarına bakın.

Bu alan mevcut değilse, MQMD 'deki değer kullanılır.

Bu bir giriş alanıdır.

PRCID (24 baytlık bit dizisi)

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 155 sayfa 1142](#) içinde belgelendiği şekilde işlenir, ancak aşağıdaki farklar söz alır:

- Yeni bir grup tanıtıcısının kullanılabileceği durumlarda, 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 şekline 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 bildirimde, alanların tümü gerekliyse, 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

MQRFH-Kurallar ve biçimleme 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 iletisi 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 1155
- “Başlangıçtaki değerler” sayfa 1157
- “RPG bildirimi” sayfa 1157

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 ' dır.

RFLN (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 içermemektedir .

Bazı ortamlarda kullanıcı verilerinin veri dönüştürmesiyle ilgili sorunları önlemek için, *RFLN* 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 değer, 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 değer bir ya da daha çok tırnak işareti içeriyorsa, ad ya da değer tırnak içine alınmalıdır ve dizedeki her 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.

RFNVS bayt cinsinden uzunluk, *RFLN* eksi *RFLNV* uzunluğuna eşittir. Bazı ortamlarda kullanıcı verilerinin veri dönüştürmesiyle ilgili sorunları önlemek için, bu uzunluğun dörtlerden biri olması önerilir. *RFNVS* , bu uzunluktaki boşluklarla doldurulmalı ya da dizedeki son anlamlı karakterin ardından boş bir karakter yerleştirilerek daha önce sonlandırılmalıdır. Boş değer karakteri ve bu karakteri izleyen byte 'lar, belirtilen *RFNVS* uzunluğuna kadar yoksayılır.

Not: Bu alanın uzunluğu değişmez olduğu için, alan, desteklenen programlama dilleri için sağlanan yapının bildirimlerinden çıkarılır.

RFSID (4 baytlık karakter dizilimi)

Yapı tanıtıcısı.

Değer şu olmalıdır:

RFSİDT

Kurallar ve biçimlendirme üstbilgisi yapısına ilişkin tanıtıcı.

Bu alanın ilk değeri RFSIDV ' dir.

RFVER (10 basamaklı işaretli tamsayı)

Yapı sürüm numarası.

Değer şu olmalıdır:

RFVER1

Version-1 kuralları ve biçimlendirme üstbilgisi yapısı.

Bu alanın ilk değeri RFVER1' dir.

Başlangıçtaki değerler

Çizelge 158. MQRFH içindeki alanların ilk değerleri		
Alan adı	Değişmez adı	Değişmez değeri
<i>RFSID</i>	RFSİDT	'RFH↵'
<i>RFVER</i>	RFVER1	1
<i>RFLN</i>	RFLNV	32
<i>RFENC</i>	ENNAT	Ortama göre değişir
<i>RFCSI</i>	CSUNDF	0
<i>RFFMT</i>	FMNONE	Boşluklar
<i>RFFLG</i>	YOK	0
Notlar:		
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 RFLN 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	RFC	SI	17	20I	0	INZ(0)
D*	Format name of data that followsNameValueString					
D	RFFMT	21	28	INZ('		')
D*	Flags					
D	RF	FLG	29	32I	0	INZ(0)

MQRFH2 -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, dizideki son MQRFH2 yapısını izleyen kullanıcı verileriyle birlikte iki ya da daha fazla MQRFH2 yapılarını 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 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 1158
- “Başlangıçtaki değerler” sayfa 1163
- “RPG bildirimi” sayfa 1163

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:

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 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 ' dır.

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 verilerini *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 değerlerden birine sahip olmalıdır:

CCSID	Anlamı
1200	UCS-2 açık uçlu

CCSID	Anlamı
13488	UCS-2 2.0 altkümesi
17584	UCS-2 2.1 altkümesi (Euro simgesini içerir)
1208	UTF-8

UCS-2 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 *değil* 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 is* , *MQGET* çağrısında belirtilen 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 çözümleyen 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:

Veri türü	Kullanılma Amacı
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 istiyor to imply a particular precision.

Veri türü	Kullanılma Amacı
14	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.
18	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, aşağıdaki şekilde ayrıntılı olarak açıklandığı gibi, herhangi bir karakterden oluşabilir. "Zorunlu" olarak işaretlenen bir karakterin değerindeki her bir oluşum, ilgili çıkış sırasına göre değiştirilmelidir. "İsteğe bağlı" olarak işaretlenen bir karakterin değerindeki her bir oluşum, ilgili çıkış sırasına göre değiştirilebilir, ancak bu zorunlu değildir.

Karakter	Çıkış sırası	Kullanım
&	&amp ;	Zorunlu
<	<	Zorunlu
>	&gt ;	İsteğe Bağlı
"	&quot ;	İsteğe Bağlı
'	&apos ;	İsteğe Bağlı

Not: Bir çıkış sırasının başlangıcındaki & karakteri, &amp ; ile değiştirilmemelidir *olmamalıdır* .

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. Verilerin (varsa) veri dönüştürmesiyle ilgili sorunları önlemek için *RF2NVL* (*RF2NVD*) alanını *şu şekilde*: (.), dört (4) birden çok 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:

RFSIDT

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 159. MQRFH2içindeki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
RF2SID	RFSİDT	' RFH↵ '
RF2VER	RFVER2	2
RF2LEN	RFLN2	36
RF2ENC	ENNAT	Ortama göre değişir
RF2CSI	CİNT	-2
RF2FMT	FMNONE	Boşluklar
RF2FLG	YOK	0
RF2NVC	Yok	1208
Notlar:		
1. ↵ simgesi tek bir boş karakteri temsil eder.		

RPG bildirimi

```
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)
```

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 görelî konum alanları tarafından adreslenen dizgiler, yerel kuyruk yöneticisinin karakter kümesinde olmalıdır; bu, *CodedCharSetId* kuyruk yöneticisi özneliği 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 ileti üstbilgisini işlemek için kullanıcı tarafından sağlanan bir ileti çıkışı çağırılır. Çıkış, MQRMH yapısı tarafından tanımlanan toplu verileri, MCA iletiyi kanaldan bir sonraki kuyruk yöneticisine göndermeden önce, başvuru ileti sonuna eklenebilir.

Alıcı uçta, başvuru iletileri için bekleyen bir ileti çıkışı var olmalıdır. Bir başvuru ileti alındığında, çıkış, iletteki MQRMH ' yi izleyen toplu verilerden nesneyi yaratmalı ve daha sonra, toplu veri olmadan başvuru ileti sonuna geçmelidir. Başvuru ileti daha sonra, bir kuyruktan başvuru ileti (toplu veriler olmadan) okunarak bir uygulama tarafından alınabilir.

Olağan durumda, MQRMH yapısı iletilerde 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 ileti 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 ileti gönderilmemelidir.

- “Veri dönüştürme” sayfa 1164
- “Alanlar” sayfa 1164
- “Başlangıçtaki değerler” sayfa 1168
- “RPG bildirimi” sayfa 1170

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, aşağıdakilerin tümünün doğru olduğu belirtilecek şekilde dönüştürülecektir:

- Toplu veriler, veri dönüştürme işlemi gerçekleştirildiğinde iletilerde 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 verilerin *olmadığını* ve sonuç olarak yığın verilerin GMCONV seçeneği tarafından dönüştürülmemesi için bu verilerinawareolmadığını unutmayın.

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:

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.

RMDEL (10 basamaklı işaretli tamsayı)

Hedef ortam verisinin uzunluğu.

Bu alan sıfırsa, hedef ortam verileri yoktur ve RMDEO yoksayılr.

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 RMLEN 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 RMLEN byte 'tan itibaren bir görelî konum olarak başlar. Tüm ileti eksi RMLEN 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* yoksayılır. Varsa, hedef nesne adının, yapının başlangıcından itibaren *RMLEN* bayt içinde tamamen bulunması gerekir.

Uygulamalar, hedef nesne adının *RMSEO*, *RMSNO* ve *RMDEO* alanları tarafından adreslenen herhangi bir veriyle bitişik olduğunu varsaymamalıdır.

Bu alanın ilk değeri 0 'tır.

RMD0 (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, toplu verilerin *MQRMH* yapısının başlangıcından fiziksel olarak görelî olarak *değil*, 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:

- *RMD0*, mantıksal görelî konum 1 000 000 000 olarak bölündüğünde elde edilen kalanı temsil eder. Böylece, 0 ile 999 999 999 aralığında bir değer vardır.
- *RMD02*, 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.

RMD02 (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. *RMD0*.

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 ileti başvuruda bulunulan 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 ' dır.

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 ileti tiplerini tanımak için kullanılabilecek bir addır. 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 iletinin 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*, *RMDEOve* *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 tanıttıcı.

Bu alanın ilk değeri RMSIDV ' dir.

RMSNL (10 basamaklı işaretli tamsayı)

Kaynak nesne adı uzunluğu.

Bu alan sıfırsa, kaynak nesne adı yoktur ve *RMSNO* yoksayılır.

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 *RMLen* 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 160. MQRMH ' deki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
<i>RMSID</i>	RMSIDR	' RMH- '
<i>RMVER</i>	RMVER1	1
<i>RMLen</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

Çizelge 160. MQRMH ' deki alanların ilk değerleri (devamı var)

Alan adı	Değişmezin adı	Değişmezin değeri
RMOII	OIINON	Boş Değerler
RMSEL	Yok	0
RMSEO	Yok	0
RMSNL	Yok	0
RMSNO	Yok	0
RMDEL	Yok	0
RMDEO	Yok	0
RMDNL	Yok	0
RMDNO	Yok	0
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 RMDEL         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	RMD0	101	104I 0 INZ(0)
D*	High offset of bulk data		
D	RMD02	105	108I 0 INZ(0)

RPG bildirimi

MQRR-Yanıt kaydı

MQRR 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 çağrıları için bir çıkış yapısıdır.

Karakter kümesi ve kodlama: MQRR ' deki veriler, ENNAT tarafından verilen yerel kuyruk yöneticisinin *CodedCharSetId* kuyruk yöneticisi özniteliği 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 1170
- “Başlangıçtaki değerler” sayfa 1170
- “RPG bildirimi” sayfa 1171

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 161. MQRR ' deki alanların ilk değerleri		
Alan adı	Değişmez adı	Değişmez değeri
RRCC	CCOK	0
RRREA	YOK	0

RPG bildirimi

```
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-SSL yapılandırma seçenekleri

The MQSCO structure (with the SSL fields in the MQCD structure) allows an application running as an IBM MQ MQI client to specify configuration options that control the use of SSL for the client connection when the channel protocol is TCP/IP.

Genel Bakış

Amaç: Yapı, MQCONNX çağırısına ilişkin bir giriş parametresidir.

İstemci kanalına ilişkin kanal protokolü TCP/IP değilse, MQSCO yapısı yoksayılr.

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 1171
- “Başlangıçtaki değerler” sayfa 1175
- “RPG bildirimi” sayfa 1175

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. “MQAIR-Kimlik doğrulama bilgileri kaydı” sayfa 985. 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 görelî konutdur. Görelî konum pozitif ya da negatif olabilir. SCAIC sıfırsa, alan yoksayılr.

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 sıfırsa, alan yoksayılr.

MQAIR kayıtları dizisini aşağıdaki iki yoldan biriyle sağlayabilirsiniz:

- SCAIPi 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* ' i 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 dizisi 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.

ibmwebspheremq<user_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 ibmwebspheremq<user_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
```

PKCS token belirtici 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 bir SSL şifrelemesi 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 SSL kanalı bağlantıları yalnızca aşağıdaki şifre belirtimlerinden 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 UNIX sistemleri ve 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.kdb` biçiminde olmalıdır; burada `zzz` kullanıcı tarafından seçilebilir. *SCKR* alanı, dosya adı gövdesiyle birlikte bu dosyanın yolunu içerir (dosya adı, son `.kdb` dahil 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 `.sth` sonekiyle sona ermelidir.

Örneğin, *SCKR* alanının değeri `/xxx/yyy/key` ise, anahtar veritabanı dosyası `/xxx/yyy/key.kdb` olmalı ve parola şifreleme dosyası `/xxx/yyy/key.sth` olmalıdır; 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 RC2381 ile başarısız olur.

Bir IBM MQ MQI client' den SSL 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 tanıttıcısıdır; değer şöyle olmalıdır:

SCSIDV

SSL yapılandırma seçenekleri yapısına ilişkin tanıttıcı.

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 SSL yapılandırma seçenekleri yapısı.

SCVER2

Version-2 SSL yapılandırma seçenekleri yapısı.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

SCVERC

SSL yapılandırma seçenekleri yapısının yürürlükteki sürümü.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri SCVER2' dir.

Başlangıçtaki değerler

Çizelge 162. 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
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 bildirimi” sayfa 1175.		

RPG bildirimi

```
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 SSL 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
D SCCERTVPOL           569      572I 0 INZ(0)
D* Ver:4 **

```

MQSD-Abonelik tanımlayıcısı

MQSD yapısı, yapılmakta olan aboneliğe ilişkin ayrıntıları belirtmek için kullanılır.

Genel Bakış

Amaç

Yapı, MQSUB çağrısında bir giriş/çıkış değiştirgesi.

Yönetilen abonelikler

Bir uygulamanın, aboneliğiyle eşleşen yayınların hedefi olarak belirli bir kuyruğu kullanmak için özel bir gereksinimi yoksa, yönetilen abonelik özelliğini kullanabilir. Bir uygulama yönetilen aboneliği kullanmayı seçerse, kuyruk yöneticisi, MQSUB çağrısından çıkış olarak bir nesne tanıtıcısı sağlayarak, yayınlanan iletilerin gönderildiği hedef ile ilgili aboneyi bilgilendirir. Ek bilgi için [HOBj \(10 basamaklı işaretli tamsayı\)-giriş/çıkış](#) başlıklı konuya bakın.

Abonelik kaldırıldığında, kuyruk yöneticisi aşağıdaki durumlarda, yönetilen hedeften alınmamış olan iletileri temizlemek için de gerekir:

- Abonelik kaldırıldığında, CORMSB ile MQCLOSE kullanımı-ve yönetilen Hobj kapatılır.
- Örtük olarak, kalıcı olmayan bir abonelik (SONDUR) kullanılarak bir uygulamaya bağlantının kaybolması anlamına gelir.
- Süresi dolduğu için abonelik kaldırıldığında ve yönetilen Hobj kapatıldığında sona erme tarihine kadar.

Temizleme işlemi gerçekleşebilmesi için, kalıcı olmayan aboneliklerle yönetilen abonelikleri kullanmanız gerekir; böylece, kapatılmış olmayan aboneliklere ilişkin iletiler kuyruk yöneticinizde yer kaplamaz. Dayanıklı abonelikler de yönetilen hedefleri kullanabilir.

Karakter kümesi ve kodlama

Data in MQSD 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 1176](#)
- [“Başlangıçtaki değerler” sayfa 1188](#)
- [“RPG bildirimi” sayfa 1189](#)

Alanlar

MQSD yapısı aşağıdaki alanları içerir; alanlar alfabetik sırayla anlatılır:

SDAID (32 baytlık karakter dizilimi)

Bu değer, bu abonelikte eşleşen tüm yayın iletilerinin (MQMD) İleti Tanımlayıcısının (MQMD) *MDAID* alanında yer alıyor. *SDAID*, iletinin tanıtıcı bağlamının bir parçasıdır. İleti bağlamına ilişkin ek bilgi için [İleti bağlamı](#) başlıklı konuya bakın.

MDAID ile ilgili daha fazla bilgi için bkz. [MDAID](#).

S0SETI seçeneği belirlenmezse, bu abonelik için yayınlanan her iletide belirtilen *MDAID* değeri, varsayılan bağlam bilgileri olarak boşluklarla birlikte ayarlanır.

SOSETI seçeneği belirtilirse, *SDAID* kullanıcı tarafından oluşturulmakta ve bu alan, bu abonelik için her yayında ayarlanacak *MDAID* ' u içeren bir giriş alanıdır.

Bu alanın uzunluğu LNAIDD tarafından verilir. Bu alanın ilk değeri 32 boş karakterdir.

SOALT seçeneğini kullanarak var olan bir aboneliğin değiştirilmesi durumunda, ilerideki yayın iletilerinin *SDAID* dosyası değiştirilebilir.

SORES kullanan bir MQSUB çağrısından dönüşte bu alan, abonelik için kullanılmakta olan geçerli *MDAID* değerine ayarlanır.

SDACC (32 baytlık karakter dizilimi)

Bu değer, bu abonelikte eşleşen tüm yayın iletilerinin (MQMD) İleti Tanımlayıcısının (MQMD) *MDACC* alanında yer alıyor. *MDACC* , iletinin tanıtıcı bağlamının bir parçasıdır. İleti bağlamına ilişkin ek bilgi için İleti bağlamı başlıklı konuya bakın.

MDACC ile ilgili daha fazla bilgi için bkz. MDACC.

SDACC alanı için aşağıdaki özel değeri kullanabilirsiniz:

YOK

Hesap belirteci belirtilmedi.

Değer, alanın uzunluğuna ilişkin ikili sıfır değeridir.

SOSETI seçeneği belirlenmezse, hesap simgesi kuyruk yöneticisi tarafından varsayılan bağlam bilgisi olarak oluşturulur ve bu alan, bu abonelik için yayınlanan her iletide ayarlanmış olan *MDACC* ' u içeren bir çıkış alanıdır.

SOSETI seçeneği belirtilirse, hesap simgesi kullanıcı tarafından oluşturulur ve bu alan, bu abonelik için her yayında ayarlanacak *MDACC* ' u içeren bir giriş alanıdır.

Bu alanın uzunluğu LNAICT tarafından verilir. Bu alanın ilk değeri ACNONE' dir.

If altering an existing subscription using the SOALT option, the value of *MDACC* in any future publication messages can be changed.

SORES kullanılarak bir MQSUB çağrısından dönüşte bu alan, abonelik için kullanılmakta olan *MDACC* değerine ayarlanır.

SDASI (40 baytlık bit dizesi)

Bu, uygun yetki denetimlerinin gerçekleştirilmesine olanak sağlamak için yetki hizmetine *SDAU* ile geçirilen bir güvenlik tanıtıcısıdır.

SDASI yalnızca *SOALTU* belirtilirse ve *SDAU* alanı, ilk boş karaktere ya da alanın sonuna kadar tamamen boş bırakılmamışsa kullanılır.

SORES kullanılarak bir MQSUB çağrısından dönüşte bu alan değişmez.

Daha fazla bilgi için, MQOD veri tipinde ODASI ' nin açıklamasına bakın.

SDAU (12 baytlık karakter dizilimi)

SOALTU değerini belirlerseniz, bu alan, uygulamanın çalışmakta olduğu kullanıcı kimliği yerine, abonelik yetkisini ve hedef kuyruğa (MQSUB çağrısının *Hobj* parametresinde belirlenen) çıkış için kullanılan bir yedek 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.

SOALTU belirtilirse ve bu alan ilk boş karaktere 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.

SOALTU belirtilmemişse, bu alan yoksayılr.

SORES kullanılarak bir MQSUB çağrısından dönüşte bu alan değişmez.

Bu bir giriş alanıdır. Bu alanın uzunluğu LNUID tarafından verilir. Bu alanın ilk değeri 12 boş karakterdir.

SDCID (24 baytlık bit dizesi)

Bu abonelik eşleşmesi için gönderilen tüm yayınlar, ileti tanımlayıcısında bu ilinti tanıtıcısını içerir. Birden çok abonelik, yayınlarını almak için aynı kuyruğu kullanıyorsa, MQGET by correlation ID ' nin kullanılması yalnızca belirli bir abonelik için yayınların elde edilebilmesini sağlar. Bu ilinti tanıtıcısı kuyruk yöneticisi ya da kullanıcı tarafından yaratılabilir.

SOSCID seçeneği belirlenmezse, kuyruk yöneticisi tarafından ilinti tanıtıcısı oluşturulur ve bu alan, bu abonelik için yayınlanan her iletide belirlenen ilinti tanıtıcısını içeren bir çıkış alanıdır.

SOSCID seçeneği belirtilirse, kullanıcı tarafından ilinti tanıtıcısı oluşturulmakta 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 belirlenen ilinti tanıtıcısı, iletinin özgün tanıtıcısıyla yaratılmış ilinti tanıtıcısıdır.

SOGRP seçeneği belirtilirse ve belirtilen ilinti tanıtıcısı, aynı kuyruğu ve çakışan bir konu dizesini kullanan var olan gruplanmış bir abonelik aynıysa, yalnızca grubun en önemli aboneliği yayının bir kopyasıyla sağlanır.

Bu alanın uzunluğuna LNCID değeri verilir. Bu alanın ilk değeri CINONE' dır.

Varolan bir aboneliği SOALT seçeneğini kullanarak değiştiriyorsa ve bu alan bir giriş alanıysa, abonelik, SOGRP seçeneği kullanılarak yaratılmadığı sürece, abonelik ilinti tanıtıcısı değiştirilebilir.

SORES kullanılarak bir MQSUB çağrısından dönüşte bu alan, abonelik için geçerli ilinti tanıtıcısı olarak ayarlanır.

SDEXP (10 basamaklı işaretli tamsayı)

Bu süre, aboneliğin sona ermesinden sonraki bir saniyenin onda biri olarak ifade edilir. Bu aralık iletildikten sonra, bu abonelik eşleşen yayın yok. Bu, bu aboneye gönderilen yayınların MQMD ' deki MDEXP alanındaki değer olarak da kullanılır.

Şu özel değer tanıdı:

EIULIM

Aboneliğin süre bitimi sınırsız.

If altering an existing subscription using the SOALT option, the expiry of the subscription can be changed.

Bu alan SORES seçeneğini kullanarak bir MQSUB çağrısından dönüşte, bu alan aboneliğin özgün süre bitimi olarak ayarlanır ve kalan süre bitimi olmaz.

SDON (48 baytlık karakter dizilimi)

Bu ad, 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 (/), alt çizgi (_), yüzde (%)

Ad, baştaki ya da gömülü boşluklar içermemeli, ancak sondaki boşlukları içermeyebilir. Addaki önemli verilerin sonunu belirtmek için boş değerli bir karakter kullanın; boş değer ve izleyen karakterler boşluk olarak işlenir. Aşağıdaki kısıtlamalar söz konusudur:

- EBCDIC Katakana kullanan sistemlerde küçük harfli karakterler kullanılamaz.
- Küçük harf karakterleri, eğik çizgi ya da yüzde içeren adlar, komutlarda belirtildiğinde tırnak işareti içine alınmalıdır. Bu tırnak işaretleri, yapılarda alan olarak ya da çağrılarda parametre olarak ortaya çıkan adlar için belirtilmemelidir.

SDON , tam konu adını oluşturmak için kullanılır.

Tam konu adı iki farklı alandan oluşturulabiliyor: *SDON* ve *SDOS*. Bu iki alanın nasıl kullanılma hakkında ayrıntılı bilgi için bkz. “Konu dizelerinin kullanılması” sayfa 560.

SORES seçeneği kullanılarak bir *MQSUB* çağrısından bu alan değiştirilmeden döndür.

Bu alanın uzunluğu *LNTOPN* tarafından verilir. Bu alanın ilk değeri 48 boş karakterdir.

Var olan bir aboneliği *SDALT* seçeneğini kullanarak değiştirebiliyorsa, abone olunan konu nesnesinin adı değiştirilemez. Bu alan ve *SDOS* atlanabilir. Sağlanırsa, aynı tam konu adına ya da çağrı *RC2510* ile başarısız olan çağrıya çözümleri gerekir.

SDOPT (10 basamaklı işaretli tamsayı)

Aşağıdaki seçeneklerden en az birini belirlemelisiniz:

- *SOALT*
- *SORES*
- *SOCRT*

Değerler eklenebilir. Aynı değışımezi bir kereden fazla eklemeyin. Tabloda bu seçeneklerin nasıl birleştirilebileceğinin gösterilmektedir: Geçerli olmayan birleşimler dikkat edilir; diğer birleşimler geçerli olur.

Erişim ya da yaratma seçenekleri

Erişim ve yaratma seçenekleri, bir aboneliğin oluşturulup oluşturulmadığını ya da var olan bir aboneliğin iade edilip değiştirilmeyeceğini denetler. Bu seçeneklerden en az birini belirtmeniz gerekir. Çizelgede, geçerli erişim ya da yaratma seçenekleri birleşimleri görüntülenir.

Seçenek bileşimi	Notlar
<i>SOCRT</i>	Eğer varsa abonelik yaratır; abonelik varsa başarısız olur.
<i>SORES</i>	Var olan bir aboneliğe devam eder, abonelik yoksa başarısız olur.
<i>SOKRAT + SOLAR</i>	Bir abonelik yaratır (varsa) ve varsa eşleşen bir aboneliği sürdürür. Bir uygulamada kaç kez çalıştırılabilen bir uygulamada kullanıldıysa, yararlı bir birleşim.
<i>SORES + SOALT</i> (nota bakın)	Var olan bir aboneliğe devam eder, herhangi bir alanı <i>MQSD</i> ' de belirtilenlerle eşleşecek şekilde değiştirir, abonelik yoksa başarısız olur.
<i>SORT + SOALT</i> (nota bakın)	Bir abonelik yaratır ve varsa, eşleşen bir abonelik yaratır (varsa), herhangi bir alanı <i>MQSD</i> ' de belirtilen alanlarla eşleşecek şekilde değiştirin. Devam etmeden önce aboneliğinin belirli bir durumda olmasını sağlamak isteyen bir uygulamada kullanıldıysa, yararlı bir birleşim.

Not:

SOALT belirtme seçenekleri de *SORES*belirtebilir, ancak bu birleşimin tek başına *SOALT* belirtmesi için herhangi bir ek etkisi yoktur. *SOALT* , bir aboneliği değiştirmek için *MQSUB* 'yi çağırdığı için, aboneliklerin de sürdürüldüğü anlamına gelen *SORES*' i belirtir. Bunun tersi doğru değil, ancak bir aboneliğin sürdürülmesi, bunun değiştirileceği anlamına da gelmez.

SOCRT

Belirtilen konu için bir abonelik yaratın. Aynı *SDSN* ' i kullanan bir abonelik varsa, çağrı RC2432 ile başarısız olur. This failure can be avoided by combining the SOCRT option with SORES. *SDSN* her zaman gerekli değildir. Daha fazla ayrıntı için, bu alanın açıklamasına bakın.

SOCRT olanağının SORES ile birleştirilmesi, önce belirtilen *SDSN* aboneliği olup olmadığını denetler ve var olan abonelik için bir tanıtıcı döndürürse, ancak var olan abonelik yoksa, MQSD ' de sağlanan tüm alanlar kullanılarak yeni bir abonelik yaratılır.

SOCRT , aynı etkiyi SOALT ile de birleştirebilir (bu konudaki SOALT ile ilgili ayrıntılara daha sonra bakın).

SORES

Return a handle to a preexisting subscription which matches those specified by *SDSN*. Eşleşen abonelik özniteliklerinde herhangi bir değişiklik yapılmadı ve bu öznitelikler MQSD yapısındaki çıkışa döndürülebilir. MQSD ' nin içeriğinin çoğu kullanılmaz: Kullanılan alanlar şunlardır: *SDSID*, *SDVER*, *SDOPT*, *SDAID* ve *SDASI*, ve *SDSN*.

The call fails with reason code RC2428 if a subscription does not exist matching the full subscription name. This failure can be avoided by combining the SOCRT option with SORES. SOCRT ile ilgili ayrıntılı bilgi için [SOCRT](#) başlıklı konuya bakın.

Aboneliğin kullanıcı kimliği, aboneliği yaratan kullanıcı kimliğidir ya da daha sonra farklı bir kullanıcı kimliği tarafından değiştirilmişse, en son, başarılı değişikliklerin kullanıcı kimliğidir. Bir *SDAID* kullanılırsa ve bu kullanıcı için diğer kullanıcı kimliklerinin kullanılmasına izin verilirse, *SDAID* , aboneliğin yapıldığı kullanıcı kimliği yerine aboneliği oluşturan kullanıcı kimliği olarak kaydedilir.

Aboneliği yaratan kullanıcı kimliği, bu alan kullanılıyorsa, *SDAU* olarak kaydedilir ve bu kullanıcı için diğer kullanıcı kimliklerine izin verilir.

SOAUID seçeneği olmayan bir eşleşen abonelik varsa ve aboneliğin kullanıcı kimliği, abonelik için bir tanıtıcı isteyen uygulamanın kullanıcılarından farklıysa, çağrı neden kodu RC2434 ile başarısız olur.

Eşleşen bir abonelik varsa ve şu anda başka bir uygulama tarafından kullanılmışsa, arama başarısız olur ve neden kodu RC2429 ile başarısız olur. Şu anda aynı bağlantı tarafından kullanılırsa, çağrı başarısız olmaz ve abonelik için bir tanıtıcı döndürülür.

SubName (Alt Ad) adlı abonelik, devam etmek ya da bir uygulamadan değiştirmek için geçerli bir abonelik değilse, çağrı RC2523 ile başarısız olur.

SORES , SOALT tarafından örtük olarak belirtilir ve bu nedenle bu seçenekle birleştirilmesi gerekmez; ancak, bu iki seçeneğin birleştirilmesi bir hata değildir.

SOALT

Return a handle to a preexisting subscription with the full subscription name matching those specified in *SDSN*. Bu öznitelik için değiştirilmeye izin verilmediği sürece, MQSD ' de belirtilenlerden farklı olan aboneliğin herhangi bir özniteliği, abonelikte değiştirilir. Ayrıntılar, her özniteliğin açıklamasında belirtilir ve aşağıdaki tabloda özetlenir. Değiştirilemeyecek bir özniteliği değiştirmeye çalışırsanız, arama aşağıdaki çizelgede gösterilen neden kodlarıyla başarısız olur.

The call fails with reason code RC2428 if a subscription does not exist matching the full subscription name. This failure can be avoided by combining the SOCRT option with SOALT.

Combining SOCRT with SOALT first checks whether there is an existing subscription for the specified full subscription name, and if there is returns a handle to that preexisting subscription with alterations made as previously detailed; but if there is no existing subscription, a new one is created using all the fields provided in the MQSD.

Aboneliğin kullanıcı kimliği, aboneliği yaratan kullanıcı kimliğidir ya da daha sonra farklı bir kullanıcı kimliği tarafından değiştirilmişse, en son başarılı değişikliklerin kullanıcı kimliğidir. *SDAU* kullanılırsa (ve o kullanıcı için diğer kullanıcı kimliklerinin kullanılmasına izin verilir), diğer kullanıcı

kimliği, aboneliğin yapıldığı kullanıcı kimliği yerine, aboneliği yaratan kullanıcı kimliği olarak kaydedilir.

SOAULID seçeneği olmadan oluşturulmuş bir abonelik aboneliği varsa ve aboneliğin kullanıcı kimliği, abonelik için bir tanıtıcı isteyen uygulamanın kullanıcı kimliği farklıysa, çağrı neden kodu RC2434 ile başarısız olur.

Eşleşen bir abonelik varsa ve şu anda başka bir uygulama tarafından kullanılmışsa, çağrı RC2429 ile başarısız olur. Aynı bağlantı tarafından kullanılmakta olan bağlantı başarısız olmaz ve aboneliğin bir tanıtıcısı döndürülür.

SubName (Alt Ad) adlı abonelik, devam etmek ya da bir uygulamadan değiştirmek için geçerli bir abonelik değilse, çağrı RC2523 ile başarısız olur.

Aşağıdaki tablolarda, SOALT tarafından değiştirilebilen abonelik öznitelikleri gösterilmektedir.

Veri tipi tanımlayıcısı ya da işlev çağırısı	Alan adı	Bu öznitelik SOALT kullanılarak değiştirilebiliyor mu?	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	Yayın seçenekleri	Evet (bkz. not 2)	Yok
MQSD	Joker 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)	Gruplanmış bir abonelikteRC2515
MQSD	SSDPRI	Evet	Yok
MQSD	SDACC	Evet	Yok
MQSD	SDAID	Evet	Yok
MQSD	SDSL	Hayır	RC2512
MQSUB	Hobj	Evet (bkz. not 6)	Gruplanmış bir abonelikteRC2515

Notlar:

1. SOGRP değiştirilemez.
2. SONEWP , 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 parçanın (SOGRP) parçası dışında değiştirilebilir.

Dayanıklılık seçenekleri: Aşağıdaki seçenekler aboneliğin ne kadar dayanıklı olduğunu denetler. Bu seçeneklerden yalnızca birini belirleyebilirsiniz. Var olan bir aboneliğin SOALT seçeneğini kullanarak değiştiriliyorsa, aboneliğin dayanıklılığını değiştiremezsiniz. SORESkullanılarak bir MQSUB çağrısından geri dönüş sırasında, uygun dayanıklılık seçeneği ayarlanır.

SODUR

Bu konuya ilişkin aboneliğin, CORMSB seçeneğiyle MQCLOSE kullanılarak belirtik olarak kaldırılincaya kadar devam etmesi için istekte bulunmayı isteyin. Bu abonelik belirtik olarak kaldırılmamışsa, bu uygulama kuyruk yöneticisine bağlandıktan sonra da devam eder.

Kalıcı aboneliklere izin vermemek olarak tanımlanmış bir konuya sürekli abonelik isteniyorsa, çağrı RC2436 ile başarısız olur.

SONDUR

Henüz açık bir şekilde kaldırılmamışsa, kuyruk yöneticisine uygulama bağlantısı kapatıldığında, bu konuya ilişkin aboneliğin kaldırıldığını isteyin. SONDUR , SODUR seçeneğinin tersi bir seçenektir ve program belgelerine yardımcı olmak için tanımlanır. İkisi de belirtilmediyse, bu varsayılan değerdir.

Hedef seçenekleri: Aşağıdaki seçenekler, abone olunan bir konuya ilişkin yayınların gönderileceği hedefi denetler. SOALT seçeneğini kullanarak var olan bir aboneliğin değiştirilmesi durumunda, abonelik için kullanılan yayınlar değiştirilebilir. SORESkullanılarak bir MQSUB çağrısından geri dönüş için, bu seçenek uygunsa ayarlanır.

SOMAN

Yayınların gönderileceği hedefin kuyruk yöneticisi tarafından yönetildiğinden emin olun.

HOBj içinde döndürülen nesne tanıtıcısı, bir kuyruk yöneticisi tarafından yönetilen kuyruğu temsil eder ve sonraki MQGET, MQCB, MQINQ ya da MQCLOSE çağrılarıyla kullanılmak içindir.

SOMAN belirtilmediğinde, *Hobj* parametresinde önceki bir MQSUB çağrısından döndürülen bir nesne tanıtıcısı sağlanamaz.

Kayıt seçenekleri: Aşağıdaki seçenekler, bu abonelik için kuyruk yöneticisine yapılan kayda ilişkin ayrıntıları denetler. If altering an existing subscription using the SOALT option, these registration options can be changed. SORES kullanılarak bir MQSUB çağrısından geri dönüş için uygun kayıt seçenekleri ayarlanır.

SOGRP

This subscription is grouped with other subscriptions of the same *SDSL* using the same queue and specifying the same correlation ID so that any publications to topics that would cause more than one publication message to be provided to the group of subscriptions, due to an overlapping set of topic strings being used, only causes one message to be delivered to the queue. Bu seçenek kullanılmıyorsa, her bir benzersiz abonelik (*SDSM* ile tanımlanır), bu yayının bir kopyasıyla eşleşen her bir abonelik sağlanır. Bu durumda, yayının birden çok kopyası, 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.

When creating a new grouped subscription it must still have a unique *SDSM*, but if it matches the full topic name of an existing subscription in the group, the call fails with RC2514 .

Grupdaki en önemli abonelik SONOLC ' yi de belirtiyorsa ve bu, aynı uygulamadan bir yayınıysa, kuyruğa yayın teslim edilmez.

When altering a subscription made with this option, the fields which imply the grouping, *Hobj* on the MQSUB call (representing the queue and queue manager name), and the *SDCID* cannot be changed. Bunların değiştirilmeye çalışılması, çağrının RC2515 ile başarısız olmasına neden olur.

This option must be combined with SOSCID with a *SDCID* that is not set to CINONE, and cannot be combined with SOMAN.

SOAULID

SOAULID belirtildiğinde, abonenin kimliği tek bir kullanıcı kimliğiyle sınırlı değildir. Bu, herhangi bir kullanıcının uygun yetkiye sahip olduğunda aboneliği değiştirmesine ya da sürdürmesine olanak sağlar. Herhangi bir zamanda yalnızca tek bir kullanıcı aboneliğe sahip olabilir. Şu anda başka bir uygulama tarafından kullanılmakta olan bir aboneliğin kullanımına devam etme girişimi, çağrının RC2429 ile başarısız olmasına neden olur.

To add this option to an existing subscription, the MQSUB call, using SOALT, must come from the same user ID as the original subscription itself.

Bir MQSUB çağrısı SOAULID ayarına sahip var olan bir aboneliğe gönderme yapıyorsa ve kullanıcı kimliği özgün abonelikten farklıysa, çağrı yalnızca yeni kullanıcı kimliğinin konuya abone olma yetkisi varsa başarılı olur. İşlem hatasız olarak tamamlandığında, bu aboneye gelecek yayınlar, yayınlama iletilerinde yeni kullanıcı kimliği kümesi ile abonenin kuyruğuna konabilir.

Hem SOAULID hem de SOFUID belirtmeyin. İkisi de belirlenmezse, varsayılan değer SOFUID' dir.

SOFUID

SOFUID değeri belirlendiğinde, abonelik değiştirilebilmesi için yalnızca son kullanıcı kimliği tarafından abonelik değiştirilebilir ya da sürdürülür. Abonelik değiştirilmediyse, aboneliği yaratan kullanıcı kimliğidir.

If an MQSUB verb references an existing subscription with SOAULID set and alters the subscription using SOALT to use the SOFUID, the user ID of the subscription is now fixed at this new user ID. Bu çağrı, yalnızca yeni kullanıcı kimliğinin konuya abone olma yetkisi varsa başarılı olur.

Abonelik sahibi olarak kaydedilenden başka bir kullanıcı kimliği SOFUID aboneliğini sürdürmeyi ya da değiştirmeyi denerse, çağrı RC2434 ile başarısız olur. Bir aboneliğin sahibi olan kullanıcı kimliği, **DISPLAY SBSTATUS** komutu kullanılarak görüntülenebilir.

Hem SOAULID hem de SOFUID belirtmeyin. İkisi de belirlenmezse, varsayılan değer SOFUID' dir.

Yayın seçenekleri: Aşağıdaki seçenekler, yayınların bu aboneye gönderilmesine yol açma şeklini denetler. If altering an existing subscription using the SOALT option, these publication options can be changed.

SONOLC

Aracıya uygulamanın kendi yayınlarından herhangi birini görmek istemediğini belirtir. Bağlantı tanıtıcıları aynıysa, yayınların aynı uygulamadan kaynaklandığı düşünülmektedir. SORES komutunu kullanarak bir MQSUB çağrısından geri dönerse, bu seçenek uygun olduğunda ayarlanır.

SONEWP

Bu abonelik oluşturulduğunda, yalnızca yeni yayınlar gönderildiğinde, yürürlükte tutulan yayınların gönderilmemesine neden olmaz. Bu seçenek yalnızca SOCRE belirtildiğinde geçerlidir. Abonelikte yapılacak her türlü değişiklik, yayınların akışını değiştirmez ve bu nedenle bir konuda alıkonan yayınlar, aboneye yeni yayınlar olarak gönderilmiştir.

Bu seçenek SOCRE olmadan belirtilirse, çağrının RC2046 ile başarısız olmasına neden olur. Bir MQSUB çağrısından SORES komutunu kullanarak dönüş için, bu seçenek kullanılarak abonelik yaratılsa bile bu seçenek belirlenmez.

Bu seçenek kullanılmıyorsa, önceden saklanan iletiler, sağlanan hedef kuyruğa gönderilir. Bu işlem bir hata nedeniyle başarısız olursa, RC2525 ya da RC2526 , aboneliğin yaratılması başarısız olur.

Bu seçenek, SOPYBR ile birlikte geçerli değildir.

SOPUBR

Bu seçeneğin ayarlanması, abonenin gerektiğinde özel olarak bilgi talep ettiğini belirtir. Kuyruk yöneticisi aboneye istenmeyen iletiler göndermez. Önceki bir MQSUB çağrısından Hsub tanıtıcısı kullanılarak bir MQSUBRQ çağrısı yapıldığında, alıkonan yayın (ya da konu içinde bir genel arama karakteri belirtilirse, büyük olasılıkla birden çok yayın) aboneye gönderilir. Bu seçeneği kullanarak MQSUB çağrısının sonucu olarak hiçbir yayın gönderilmez. SORES komutunu kullanarak bir MQSUB çağrısından geri dönerse, bu seçenek uygun olduğunda ayarlanır.

Bu seçenek, SONEWPile birlikte geçerli değildir.

Genel arama karakteri seçenekleri: Aşağıdaki seçenekler, genel arama karakterlerinin MQSD ' nin SDOS alanında sağlanan dizgide nasıl yorumlanana ilişkin denetim sağlar. Bu seçeneklerden yalnızca birini belirleyebilirsiniz. SOALT seçeneğini kullanarak var olan bir aboneliğin değiştirilmesi durumunda, bu genel arama karakteri seçenekleri değiştirilemez. SORES kullanılarak bir MQSUB çağrısından geri dönüş için, uygun genel arama karakteri seçeneği ayarlanır.

SOWCHR

Genel arama karakterleri yalnızca konu dizgisindeki karakterler üzerinde çalışır. SOWCHR (SOWCHR) alanı, özel önem taşıyan başka bir karakter olarak eğik çizgi (/) olarak işler.

SOWCHR ile tanımlanan işleyiş aşağıdaki çizelgede gösterilmiştir:

Özel Karakter	Davranış
*	Joker karakter, sıfır ya da daha fazla karakter
?	Genel arama karakteri, bir karakter
%	Bir dizgide '*', '?' ya da '%' karakterlerinin kullanılmasına izin vermek için çıkış karakteri; örneğin, '% *', '%?' ya da '%% ' gibi, özel bir karakter olarak yorumlanmaz.

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?el2/level3/level4
```

Not: Bu genel arama karakterleri kullanımı, Yayınlama/Abone Olma için MQRFH1 tarafından biçimlendirilmiş iletiler kullanılırken tam olarak IBM MQ V6 ve WebSphere MB V6 ' da sağlanan anlamlara erir. Bu, yeni yazılan uygulamalar için kullanılmaması ve yalnızca bu sürüme karşı daha önce çalışan uygulamalar için kullanılması önerilir ve SOWTOP içinde açıklandığı gibi varsayılan genel arama karakteri davranışını kullanmak üzere değiştirilmemiş olmalıdır.

SOWTOP

Genel arama karakterleri yalnızca konu dizgisindeki konu öğelerinde işlem görmektedir. Bu, hiçbirini seçilmezse, varsayılan davranıştır.

SOWTOP ile gerekli olan davranış aşağıdaki çizelgede gösterilmiştir:

Özel Karakter	Davranış
/	Konu düzeyi ayırıcı
#	Joker karakter: birden çok konu düzeyi
+	Joker karakter: tek konu düzeyi

Not:

'+' 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
```

Not: Bu joker karakterler kullanımı, Yayınlama/Abone Olma için MQRFH2 biçimlendirilmiş iletileri kullanırken WebSphere Message Broker Sürüm 6 ' ta sağlanan anlamı sağlar.

Diğer seçenekler: Aşağıdaki seçenekler, abonelik yerine API çağrısının yayınını kontrol eder. SORES kullanılarak bir MQSUB çağrısından geri dönüş sırasında bu seçenekler değiştirilmez.

SOALTU

SDAU alanı, bu MQSUB çağrısının geçerliliğini denetlemek için kullanılacak bir kullanıcı kimliği içerir. Bu çağrı, yalnızca bu SDAU ' nun, uygulamanın çalışmakta olduğu kullanıcı kimliğinin bu işlemi gerçekleştirme yetkisine sahip olup olmamasına bakılmaksızın, belirtilen erişim seçenekleriyle açma yetkisine sahip olması durumunda başarılı olabilir.

SOSCID

Abonelik, *SDCID* alanında sağlanan ilinti tanıtıcısını kullanmandır. Bu seçenek belirlenmezse, abonelik sırasında kuyruk yöneticisi tarafından otomatik olarak bir ilinti tanıtıcısı yaratılır ve *SDCID* alanında uygulamaya döndürülür. Ek bilgi için [SDCID \(24 baytlık bit dizgisi\) SDCID](#) konusuna bakın.

SOSETI

Abonelik, *SDACC* ve *SDAID* alanlarında sağlanan muhasebe belirteci ve uygulama kimliği verilerini kullanmandır.

Bu seçenek belirlenirse, aynı yetki denetimi, *SOMAN* seçeneğinin de kullanıldığı durumda, hedef kuyrukta kullanılan bir yetki denetimi olmadığında, *00SETI* ile bir *MQOPEN* çağrısı kullanılarak erişildiği gibi, aynı yetki denetimi gerçekleştirilir.

Bu seçenek belirlenmezse, bu aboneye gönderilen yayınların varsayılan bağlam bilgilerini aşağıda anlatıldığı gibi ilişkilendirir:

MQMD ' de alan	Kullanılan değer
<i>MDUID</i>	Aboneliğin yapıldığı sırada, abonelik ilişkili kullanıcı kimliği.
<i>MDACC</i>	Olanaklıysa, ortamdan saptanır; değilse, <i>ACNONE</i> olarak ayarlayın.
<i>MDAID</i>	Boşluklara ayarla

Bu seçenek yalnızca *SOCRE* ve *SOALT* ile geçerlidir. *SORES* ile kullanılırsa, *SDACC* ve *SDAID* alanları yoksayılır, bu nedenle bu seçeneğin herhangi bir etkisi yoktur.

Bir abonelik, önceden aboneliğin sağladığı kimlik bağlamı bilgilerini içeren bu seçenek kullanılmadan değiştirilirse, değiştirilen abonelik için varsayılan bağlam bilgileri oluşturulur.

If a subscription allowing different user IDs to use it with option SOAULID, is resumed by a different user ID, default identity context is generated for the new user ID now owning the subscription and any subsequent publications are delivered containing the new identity context.

SOFIQ

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.

SDAU (12 baytlık karakter dizilimi)

SOALTU değerini belirlerseniz, bu alan, uygulamanın çalışmakta olduğu kullanıcı kimliği yerine, abonelik yetkisini ve hedef kuyruğa (MQSUB çağrısının *Hobj* parametresinde belirlenen) çıkış için kullanılan bir yedek 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.

SOALTU değeri belirlenirse ve bu alan, ilk boş karaktere ya da alanın sonuna kadar tamamen boşsa, abonelik yalnızca, belirtilen seçeneklerle ya da çıkış için hedef kuyruğuyla bu konuya abone olmadıysa, bu alana abone olunabilir.

SOALTU belirtilmemişse, bu alan yoksayılr.

SORESkullanılarak bir MQSUB çağrısından dönüşte 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 İleti Tanımlayıcısının (MQMD) *MQPRI* alanında bulunan değerdir. MQMD ' deki *MQPRI* alanıyla ilgili ek bilgi için *MDPRI* başlıklı konuya bakın.

Değer sıfırdan büyük ya da sıfıra eşit olmalıdır; sıfır, en düşük önceliğe sahip olmalıdır. Aşağıdaki özel değerler de kullanılabilir:

PRQDEF

Bir abonelik kuyruğu, MQSUB çağrısındaki *Hobj* alanında sağlandığında ve yönetilen bir tanıtıcı değilse, ileti için öncelik bu kuyruğun *DefPriority* özniteliğinden alınır. If the queue so identified 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 *MDPRI*.

MQSUB çağrısı yönetilen bir tanıtıcı kullanıyorsa, ileti için öncelik, abone olunan konu ile ilişkilendirilmiş model kuyruğunun *DefPriority* özniteliğinden alınır.

PPUB

İletiye ilişkin öncelik, özgün yayının önceliğidir. Bu, alanın ilk değeridir.

If altering an existing subscription using the SOALT option, the *MQPRI* of any future publication messages can be changed.

SORESkullanılarak bir MQSUB çağrısından dönüşte bu alan, abonelik için kullanılmakta olan geçerli önceliğe ayarlıdır.

SDRO (MQCHARV)

SDRO, kuyruk yöneticisi *SDON* ' ta sağlanan adı çözdükten sonra uzun nesne adıdır.

SDOS ' ta uzun nesne adı sağlanırsa ve *SDON* ' ta hiçbir şey sağlanmıyorsa, bu alanda döndürülen değer, *SDOS* ' de sağlanan değerle aynıdır.

Bu alan atlanırsa (*SDRO.VSBufSize* sıfırdır), *SDRO* döndürülmez, ancak uzunluk *SDRO.VSLength* ' ta döndürülür. Uzunluk, tam *SDRO* değerinden daha kısaysa, kısaltılır ve sağlanan uzunluğa sığabileceği en sağdaki en sağdaki karakter döndürür.

SDRO yanlış belirtildiyse, *MQCHARV* yapısının nasıl kullanılacağını ya da uzunluk üst sınırını aştıysa, çağrı neden kodu RC2520 ile başarısız olur.

SDSID (4 baytlık karakter dizilimi)

Bu, yapı tanıttıcısıdır; değer şöyle olmalıdır:

SDSIDV

Abonelik Tanımlayıcısı yapısına ilişkin tanıttıcı.

Bu her zaman bir giriş alanıdır. Bu alanın ilk değeri *SDSIDV*' dir.

SDSL (10 basamaklı işaretli tamsayı)

Bu, abonelik ile ilişkili düzeydir. Bu abonelik yalnızca, yayınlama sırasında kullanılan *PubLevel* değerine eşit ya da bu değerden küçük olan en yüksek *SDSL* değeri olan abonelikler kümesinde yer alıyorsa, bu abonelik için teslim edilir.

Değer, sıfır ile 9 aralığında olmalıdır. Sıfır en düşük düzeydir.

Bu alanın ilk değeri 1 'dir.

If altering an existing subscription using the *SOALT* option, then *SDSL* cannot be changed.

SDSN (MQCHARV)

SDSN , abonelik adını belirtir.

This field is required only if *SDOPT* specifies the *SODUR* option, but if it is provided it is used by the queue manager for *SONDUR* as well. If specified, *SDSN* must be unique within the queue manager, because it is the field used to identify subscriptions.

SDSN uzunluğu üst sınırı 10240 'tır.

Bu alan iki amaca hizmet eder. *SODUR* aboneliği için, yarattığı yaratıldıktan sonra, aboneliğin tanıttıcısını (*COKPSB* seçeneğini kullanarak) kapatmış olması ya da kuyruk yöneticisinden bağlantınız varsa, bu aboneliği sürdürmek için bir abonelik tanımlamanız anlamına gelir. Bir aboneliği yaratıldıktan sonra kaldırmak için, *SORES* seçeneğiyle *MQSUB* çağrısını kullanarak bir abonelik tanımlanıyor. *SDSN* alanı ayrıca, *DISPLAY SBSTATUS* içindeki *SDSN* alanındaki aboneliklerin yönetim görünümünde de görüntülenir.

SDSN yanlış belirtilirse, *MQCHARV* yapısının nasıl kullanılacağı ya da uzunluk üst sınırını aşması durumunda ya da gerektiğinde atlanırsa (*SDSN* olur. *VCHRL* sıfır) ya da uzunluk üst sınırını aşarsa, çağrı neden kodu RC2440 ile başarısız olur.

Bu bir giriş alanıdır. Bu yapıdaki alanların ilk değerleri, *MQCHARV* yapısındaki alanlarla aynıdır.

SOALT seçeneğini kullanarak var olan bir aboneliğin değiştirilmesi durumunda, abonelik tanımlamak için kullanılan alan olduğundan, abonelik adı değiştirilemez. It is not changed on output from an *MQSUB* call with the *SORES* option.

SDSS (MQCHARV)

SDSS , bir konudan gelen iletilere abone olurken kullanılan seçim ölçütlerini sağlayan dizedir.

Bu değişken uzunluk alanı, *SORES* seçeneği kullanılarak bir *MQSUB* çağrısından çıkışta döndürülür; bir arabellek sağlandıysa ve *VSBufSize* içinde artı bir arabellek uzunluğu da varsa, bu değişken uzunluk alanı döndürülür. Çağrıda arabellek sağlanmazsa, *MQCHARV* ' ın *VSLcgrth* alanında yalnızca seçim dizgisinin uzunluğu döndürülür. Sağlanan arabellek, alanı döndürmek için gereken alandan küçükse, sağlanan arabellekte yalnızca *VSBufSize* byte değeri döndürülür.

SDSS yanlış belirtildiyse, *MQCHARV* yapısının nasıl kullanılacağını ya da uzunluk üst sınırını aştıysa, çağrı neden kodu RC2519 ile başarısız olur.

SDSUD (MQCHARV)

Bu alandaki aboneliğe ilişkin sağlanan veriler, bu aboneliğe gönderilen her yayının *mq.SubUserData* ileti özelliği olarak eklenir.

SDSUD uzunluğu üst sınırı 10240 'tır.

SDSUD yanlış belirtilmişse, *MQCHARV* yapısının nasıl kullanılacağı ya da uzunluk üst sınırını aşıyorsa, çağrı neden kodu RC2431 ile başarısız olur.

Bu bir giriş alanıdır. Bu yapıdaki alanların ilk değerleri, *MQCHARV* yapısındaki alanlarla aynıdır.

If altering an existing subscription using the *SOALT* option, the subscription user data can be changed.

Bu değişken uzunluk alanı, bir arabellek sağlanırsa ve *VSBufLen* içinde pozitif bir arabellek uzunluğu varsa, *SORES* seçeneğini kullanarak bir *MQSUB* çağrısından çıkışta döndürülür. Çağrıda arabellek sağlanmazsa, *MQCHARV* ' in *VCHRL* alanında yalnızca abonelik kullanıcı verileri uzunluğu döndürülür. Sağlanan arabellek, alanı döndürmek için gereken alandan küçükse, sağlanan arabellekte yalnızca *VSBufLen* byte 'ları döndürülür.

SDVER (10 basamaklı işaretli tamsayı)

Bu, yapı sürüm numarasıdır; değer şu olmalıdır:

SDVER1

Version-1 Abonelik Tanımlayıcısı yapısı.

Aşağıdaki değişmez, yürürlükteki sürümün sürüm numarasını belirtir:

SDVERC

Abonelik Tanımlayıcısı yapısının yürürlükteki sürümü.

Bu her zaman bir giriş alanıdır. Alanın ilk değeri *SDVER1* ' dir.

Başlangıçtaki değerler

Alan adı	Değişmezin adı	Değişmezin değeri
<i>SDSID</i>	<i>SDSIDV</i>	'SD--'
<i>SDVER</i>	<i>SDVER1</i>	1
<i>SDOPT</i>	<i>SONDUR</i>	0
<i>SDON</i>	Yok	Boşluklar
<i>SDAU</i>	Yok	Boşluklar
<i>SDASI</i>	<i>SATIR</i>	Boş Değerler
<i>SDEXP</i>	<i>EIULIM</i>	-1
<i>SDOS</i>	<i>MQCHARV</i> için tanımlandığı şekilde adlar ve değerler	
<i>SDSN</i>	<i>MQCHARV</i> için tanımlandığı şekilde adlar ve değerler	
<i>SDSUD</i>	<i>MQCHARV</i> için tanımlandığı şekilde adlar ve değerler	
<i>SDCID</i>	<i>CINONE</i>	Boş Değerler
<i>SDPRI</i>	<i>PRQDEF</i>	-3
<i>SDACC</i>	<i>YOK</i>	Boş Değerler
<i>SDAID</i>	Yok	Boşluklar
<i>SDSL</i>	Yok	1
<i>SDRO</i>	<i>MQCHARV</i> ' da tanımlandığı gibi adlar ve değerler	

Alan adı	Değişmezin adı	Değişmezin değeri
Not: 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 SDRDP 341 356*
D SDRDO 357 360I 0
D SDRDS 361 364I 0
D SDRDL 365 368I 0
D SDRDC 369 372I 0

```

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 1190
- “Başlangıçtaki değerler” sayfa 1191
- “RPG bildirimi” sayfa 1191

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. İmlecin 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 imlecinin 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. İmlecin 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 imlecinin 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.

SPVALENC (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 163. 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  SP SID      1      4  INZ('SMPO')
D*
D* Structure version number
D  SP VER      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)
```

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 1192
- “Başlangıçtaki değerler” sayfa 1192
- “RPG bildirimi” sayfa 1193

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

Alan adı	Değişmezin adı	Değişmezin değeri
SRSID	SİRSİS	' SRO→ '

Alan adı	Değişmezin adı	Değişmezin değeri
SERVER	SERVER1	1
SROPT	YOK	0
SRNMP	Yok	0
Notlar: ↪ simgesi tek bir boş karakteri temsil eder. - 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* MQSRO Structure
D*
D* Structure identifier
D SRSID 1 4
D* Structure version number
D SERVER 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

```

MQSTS-Durum raporlama yapısı

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 değer *CodedCharSetId* kuyruk yöneticisi özniteliği 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. [“MQSTAT-Durum bilgilerini al” sayfa 1321.](#)

- [“Alanlar” sayfa 1193](#)
- [“Başlangıçtaki değerler” sayfa 1196](#)
- [“RPG bildirimi” sayfa 1197](#)

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.

STSFC (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.

STS00 (10 basamaklı işaretli tamsayı)

The STS00 used to open the object being reported upon. Yalnızca MQSTS ya da daha sonraki sürümlerde bulunan Sürüm 2 'de bulunur.

STS00 değeri, MQSTAT STYPE parametresinin değerine bağlıdır.

STATAPT

Sıfır.

DURDURC

Sıfır.

DURDURUR

Hata ortaya çıktığında STS00 kullanılır. Başarısızlığın nedeni, MQSTS yapısındaki *STSCC* ve *STSRC* alanlarında raporlanır.

STS00 , bir çıkış alanıdır. Başlangıç değeri sıfır.

STSOS (MQCHARV)

Raporlanmakta olan başarısız nesnenin uzun nesne adı. Yalnızca MQSTS ya da daha sonraki sürümlerde bulunan Sürüm 2 'de bulunur.

STSOS , uzunluk üst sınırı 10240 olan bir MQCHARV alanıdır. MQCHARV yapısının nasıl kullanılacağını görmek için [MQCHARV](#) başlıklı konuya bakın.

STSOS ' ın 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

Sıfır 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 *STSRBJN* 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 sürümlerde bulunan Sürüm 2 'de bulunur.

STSSO ' in yorumu, MQSTAT STYPE parametresinin değerine bağlıdır.

STATAPT

Sıfır.

DURDURC

Sıfır.

DURDURUR

Hata ortaya çıktığında STSS0 kullanılır. Başarısızlığın nedeni, MQSTS 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 (MQCHARV)

Başarısız olan aboneliğin adı. Yalnızca MQSTS ya da daha sonraki sürümlerde bulunan Sürüm 2 'de bulunur.

STSSUN , maximum uzunluğuna 10240 uzunluğunda bir MQCHARV alanıdır. MQCHARV yapısının nasıl kullanılacağını görmek için [MQCHARV](#) başlıklı konuya bakın.

STSSUN ' in yorumu, MQSTAT STYPE parametresinin değerine bağlıdır.

STATAPT

Sıfır uzunluklu dizgi.

DURDURC

Sıfır uzunluklu dizgi.

DURDURUR

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.

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 zamanuysuz koyma çağrılarının sayısıdır.

Bu bir çıkış alanıdır. Bu alanın ilk değeri 0 'tır.

Başlangıçtaki değerler

Çizelge 164. MQSTS ' deki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
STSSID	STSID	
STSVR	STSVRC	STSVR1
STSCC	CCOK	0

Çizelge 164. MQSTS ' deki alanların ilk değerleri (devamı var)

Alan adı	Değişmezin adı	Değişmezin değeri
STSRC	YOK	0
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 bildirimi

```

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 STSSUNVSBS      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*, *TMTDve* *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. [“MQTMC2 -Tetikleyici iletisi 2 \(karakter biçimi\)” sayfa 1202](#).

- 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 1198](#)
- [“Alanlar” sayfa 1199](#)
- [“Başlangıçtaki değerler” sayfa 1201](#)
- [“RPG bildirimi” sayfa 1202](#)

Tetikleme iletisi için MQMD

Tetikleme iletisi için MQMD: Kuyruk yöneticisi tarafından oluşturulan bir tetikleme iletisinin MQMD ' deki alanları aşağıdaki gibi ayarlanır:

MQMD ' de alan	Kullanılan değer
MDSID	MDSIDV
MDVER	MDVER1

MQMD ' de alan	Kullanılan değer
<i>MDREP</i>	ROYOK
<i>MDMT</i>	MTDGRM
<i>MDEXP</i>	EIULIM
<i>MDFB</i>	YOK
<i>MDENC</i>	ENNAT
<i>MDCSI</i>	Kuyruk yöneticisinin <i>CodedCharSetId</i> özniteliği
<i>MDFMT</i>	FMTM
<i>MDPRI</i>	Başlatma kuyruğunun <i>DefPriority</i> özniteliği
<i>MDPER</i>	PENPER
<i>MDMID</i>	Benzersiz bir değer
<i>MDCID</i>	CINONE
<i>MDBOC</i>	0
<i>MDRQ</i>	Boşluklar
<i>MDRM</i>	Kuyruk yöneticisinin adı
<i>MDUID</i>	Boşluklar
<i>MDACC</i>	YOK
<i>MDAID</i>	Boşluklar
<i>MDPAT</i>	ATQM ya da ileti kanalı aracısına uygun olan
<i>MDPAN</i>	Kuyruk yöneticisi adının ilk 28 byte 'ı
<i>MDPD</i>	Tetikleme iletisinin gönderildiği tarih
<i>MDPT</i>	Tetikleme iletisinin gönderildiği saat
<i>MDAOD</i>	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 *AppId* özniteliğinin değeriyle birlikte başlatır; bu özniteliğe ilişkin ayrıntılar için “Süreç tanımlamalarına ilişkin öznitelikler” sayfa 1360 ' 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 *AppType* özniteliğinin değeriyle birlikte başlatır; bu özniteliğe ilişkin ayrıntılar için [“Süreç tanımlamalarına ilişkin öznitelikler” sayfa 1360](#) ' 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 [“Süreç tanımlamalarına ilişkin öznitelikler” sayfa 1360](#) ' 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 1331](#) ' 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 dizgisi)

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 1331 ' 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 1331 ' 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 “[Süreç tanımlamalarına ilişkin öznitelikler](#)” sayfa 1360 ' 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 165. MQTM ' deki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
<i>TMSID</i>	TMSIDV	' TM-- '
<i>TMVER</i>	TMVER1	1
<i>TMQN</i>	Yok	Boşluklar

Çizelge 165. 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 bildirimi

```

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

```

MQTMC2 -Tetikleyici iletisi 2 (karakter biçimi)

Bir tetikleme izleme programı bir başlatma kuyruğundan bir tetikleme iletisi (MQTM) aldığı anda, 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*, *TC2TD* ve *TC2UD* içerir. 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 1203
- “Başlangıçtaki değerler” sayfa 1204
- “RPG bildirimi” sayfa 1204

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 iletisinin 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.

TC2QM (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:

TC2SIDV

Tetikleyici iletisi (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 166. 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:		
1. ¬ simgesi tek bir boş karakteri temsil eder.		

RPG bildirimi

```
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
```

MQWIIH-İş bilgisi üstbilgisi

MQWIIH yapısı, z/OS iş yükü yöneticisi tarafından işlenecek bir iletinin başlangıcında var olması gereken bilgileri açıklar.

Genel Bakış

Biçim adı: FMWIIH.

Karakter kümesi ve kodlama: MQWIIH yapısındaki alanlar, MQWIIH ' den önce gelen üstbilgi yapısındaki *MDCSI* ve *MDENC* alanları tarafından verilen karakter kümesi ve kodlamadır ve MQWIIH uygulama iletisi verilerinin başlangıcındaki bu alanlar tarafından MQWIIH 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 MQWIIH structure.

- “Alanlar” sayfa 1205
- “Başlangıçtaki değerler” sayfa 1207
- “RPG bildirimi” sayfa 1207

Alanlar

MQWIIH yapısı aşağıdaki alanları içerir; alanlar **alfabetik sırayla** anlatılır:

WICSI (10 basamaklı işaretli tamsayı)

MQWIIH ' ı izleyen verilerin karakter kümesi tanıtıcısı.

Bu, MQWIIH yapısını izleyen verilerin karakter kümesi tanıtıcısını belirtir; bu, MQWIIH yapısındaki 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 CSUNDF 'dir.

WIENC (10 basamaklı işaretli tamsayı)

MQWIIH ' ı izleyen verilerin sayısal kodlaması.

Bu, MQWIIH yapısını izleyen verilerin sayısal kodlamasını belirtir; bu, MQWIIH 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 değeri WINONE değeridir.

WIFMT (8 baytlık karakter dizilimi)

MQWIIH 'ı izleyen verilerin adını biçimle.

Bu, MQWIIH yapısından sonra gelen verilerin biçim adını belirler.

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 'dır.

WILEN (10 basamaklı işaretli tamsayı)

MQWIIH yapısının uzunluğu.

Değer şu olmalıdır:

WILEN1

version-1 iş bilgileri üstbilgisi yapısı uzunluğu.

Aşağıdaki değişmez, yürürlükteki sürümün uzunluğunu belirtir:

WILENC

İş bilgisi üstbilgisi yapısının geçerli sürümü uzunluğu.

Bu alanın ilk değeri WILEN1' dir.

WIRSV (32 baytlık karakter dizilimi)

Ayrılmış.

Bu ayrılmış bir alandır; boş bırakılmalıdır.

WISID (4 baytlık karakter dizilimi)

Yapı tanıtıcısı.

Değer şu olmalıdır:

WISIDV

İş bilgileri üstbilgi yapısına ilişkin tanıtıcı.

Bu alanın ilk değeri WISIDV ' dir.

WISNM (32 baytlık karakter dizilimi)

Hizmet adı.

Bu, iletiyi işlemek için kullanılan hizmetin adıdır.

Bu alanın uzunluğu LNSVNM tarafından verilir. Bu alanın ilk değeri 32 boş karakterdir.

WISST (8 baytlık karakter dizilimi)

Hizmet adımı adı.

Bu ad, iletinin ilişkilendirdiği *WISNM* adının adıdır.

Bu alanın uzunluğu LNSVST tarafından verilir. Bu alanın ilk değeri 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 uzunluğuna LNMTOK değeri verilir. Bu alanın ilk değeri MTKON değeridir.

WIVER (10 basamaklı işaretli 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 167. MQWIIH içindeki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
WISID	WISIDV	'WIIH~'
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* MQWIIH Structure
D*
D* Structure identifier
D  WISID          1      4    INZ('WIIH ')
D* Structure version number
D  WIVER          5      8I 0 INZ(1)
D* Length of MQWIIH structure
D  WILEN          9     12I 0 INZ(120)
D* Numeric encoding of data that followsMQWIIH
D  WIENC         13     16I 0 INZ(0)
D* Character-set identifier of data thatfollows MQWIIH
D  WICSI         17     20I 0 INZ(0)
D* Format name of data that followsMQWIIH
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'0000000000000000-
D                                0000000000000000')
```

MQXQH-İletim kuyruğu üstbilgisi

MQXQH yapısı, ileti kuyruklarında uygulama iletisi 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 özneliği 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 özneliği 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:

- Ayır 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. Ayır 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 iletisi verileri (varsa) sırası gelir. MQMDE aşağıdakilerden biri olabilir:

- Kuyruk yöneticisi tarafından üretilir (uygulama iletiyi koymak için bir version-2 MQMD kullanıyorsa) ya da
- Uygulama iletisi 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.

- [“Ayır ileti tanımlayıcısındaki alanlar” sayfa 1209](#)
- [“Katıştırılmış ileti tanımlayıcısındaki alanlar” sayfa 1210](#)
- [“İletileri Uzak Kuyruklara Koyma” sayfa 1210](#)
- [“İletilerin doğrudan iletim kuyruklarına koyma” sayfa 1210](#)
- [“İletim kuyruklarından ileti alınması” sayfa 1211](#)
- [“Alanlar” sayfa 1211](#)
- [“Başlangıçtaki değerler” sayfa 1212](#)
- [“RPG bildirimi” sayfa 1212](#)

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.

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 <i>CodedCharSetId</i> 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	İletin iletim kuyruğuna konduğu tarih.
MDPT	İletin 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 iletise, *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 öneki 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 dizisi)

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 iletiyorsa, 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.

D	XQ1PT	417	424	INZ
D	XQ1AOD	425	428	INZ

İşlev çağrıları

IBM i programlamasındaki işlev çağrıları 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 967](#).

Parametrenin yönü şu şekilde olabilir:

Giriş

Siz (programcı) bu değiştireyi sağlamalısınız.

Çıkış

Çağrı bu parametreyi döndürür.

Giriş/çıkış

Bu parametreyi sağlamanız gerekir, ancak bu parametre çağrıyla değiştirilir.

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, çağrılarının 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 değişmez büyük-küçük boyutlu karakter dizgileri dizileridir. Bu parametrelerin açıklamalarında, küçük harf *n* bir sayısal değişmezi temsil eder. When you code the declaration for that parameter, replace the *n* with the numeric value you require.

MQBACK-Değişiklikleri geri al

MQBACK çağrısı, kuyruk yöneticisine tüm iletinin aldığı ve son eşitleme noktasının geriletilmesinden bu yana gerçekleşen tüm çağrılarının oluştuğunu gösterir. Bir iş biriminin bir parçası olarak gönderilen iletler silinir; iş biriminin bir parçası olarak alınan iletler kuyruktan geri alınır.

- IBM işletim sistemi, uyumluluk kipinde çalışan uygulamalar için bu çağrıyı desteklemez.

Önemli:  Uyumluluk kipini kullanmaya devam etseniz, uygulamanızı yeniden derlemeyi ve bunun yerine statik bağlı çağrıları kullanmayı düşünün. Bkz. [IBM üzerindeki RPG uygulamaları için uyumluluk kipi](#)

- “Sözdizimi” sayfa 1214
- “Kullanım notları” sayfa 1214
- “Parametreler” sayfa 1215
- “RPG Bildirimi” sayfa 1216

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 “MQDISC-Bağlantı kesme kuyruğu yöneticisi” sayfa 1253 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 bir iletiyi göz atlayan son başarılı MQGET çağrısı (bu *olamaz* bir iş biriminin parçası olabilir).

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 geriletilirse, grup ve bölüm bilgilerini geri yüklemeyiz.

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 seçenekleri” sayfa 1138’inde açıklanan PMLOGO seçeneğine ve “MQGMO-Get-message seçenekleri” sayfa 1044’ 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 çağrılar, 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 1239 içinde açıklanan HCONN parametresine bakın.
2. Bu çağrıdan etkilenecek, 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 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

MQBACK çağrısı aşağıdaki değıştirmeleri içerir:

HCONN (10 basamaklı işaretili tamsayı)-giriş

Bağlantı tanıtıcısı.

Bu tanıtıcı, kuyruk yöneticisiyle olan bağlantıyı gösterir. The value of HCONN was returned by a previous MQCONN or MQCONN call.

CMPCOD (10 basamaklı işaretili tamsayı)-çıkış

Tamamlanma kodu.

Bu, aşağıdakilerden biridir:

CCOK

İşlem başarıyla tamamlandı.

CCCFAIL

Arama başarısız oldu.

REASON (10 basamaklı işaretili tamsayı)-çıkış

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..
D*MQBACK          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

AIX, HP-UX, IBM i, Solaris, Windows S/390

MQBEGIN-İş birimini başlat

MQBEGIN çağrısı, kuyruk yöneticisi tarafından eşgüdümlü bir iş birimi başlatır ve bu, dış kaynak yöneticilerini içermeyebilir.

- Bu arama şu ortamlarda desteklenmektedir: AIX, HP-UX, IBM i, Solaris, Windows.
- “Sözdizimi” sayfa 1216
- “Kullanım notları” sayfa 1216
- “Parametreler” sayfa 1218
- “RPG Bildirimi” sayfa 1219

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 eşgüdümçüsü 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 eşgüdümçüsü olarak işlev görmez. Bunun yerine, kuyruk yöneticisinin işbirliği yaptığı bir iş birimi eşgüdümçüsü 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 [“MQDISC-Bağlantı kesme kuyruğu yöneticisi” sayfa 1253](#) 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 eşgüdümçüsü 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 değıştirmeler vardır:

HCONN (10 basamaklı işaretli tamsayı)-giriş

Bağlantı tanıtıcısı.

Bu tanıtıcı, kuyruk yöneticiyle olan bağlantıyı gösterir. The value of *HCONN* was returned by a previous MQCONN or MQCONNX call.

BEGOP (MQBO)-giriş/çıkış

MQBEGIN işlemini denetleyen seçenekler.

Ayrıntılar için bkz. “MQBO-Başlat seçenekleri” sayfa 989.

Herhangi bir seçenek gerekmiyorsa, C ya da S/390 çevirici programında yazılan programlar, bir MQBO yapısının adresini belirtmek yerine, boş değeri bir parametre adresi belirtebilir.

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:

RC2121

(2121, X'849 ') Katılan kaynak yöneticisi kayıtlı değil.

RC2122

(2122, X'84A') Katılan kaynak yöneticisi kullanılamıyor.

CMPCOD CCCFAIL ise:

RC2134

(2134, X'856 ') Begin-options yapısı geçerli değil.

RC2219

(2219, X'8AB') MQI çağrısı, önceki arama tamamlanmadan önce yeniden girildi.

RC2009

(2009, X'7D9') Kuyruk yöneticisine yönelik bağlantı kaybedildi.

RC2012

(2012, X'7DC') Ortamda arama geçerli değil.

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
```

MQBUFMMH-Arabelleği ileti tutamaçına dönüştür

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 1219
- “Kullanım notları” sayfa 1219
- “Parametreler” sayfa 1219
- “RPG Bildirimi” sayfa 1221

Sözdizimi

MQBUFMMH (*Hconn, Hmsg, BufMsgHOpts, MsgDesc, Buffer, BufferLength, DataLength, CompCode, Reason*)

Kullanım notları

MQBUFMMH çağrıları API çıkışlarıyla algılanamaz; arabellek, uygulama alanındaki bir ileti tanıtıcısı içine dönüştürülür; çağrı kuyruk yöneticisine ulaşmaz.

Parametreler

MQBUFMMH çağrısında şu değişirgeler vardır:

HCONN (10 basamaklı işaretli tamsayı)-giriş

Bu tanıtıcı, kuyruk yöneticisiyle olan bağlantıyı gösterir. *HCONN* değeri, *Hmsg* değiştirgesinde belirtilen ileti tanıtıcısını yaratmak için kullanılan bağlantı tanıtıcısı ile eşleşmelidir.

İleti tanıtıcısı HCUNAS kullanılarak yaratıldıysa, bir arabelleği ileti tutamaçına dönüştüren iş parçacığıda geçerli bir bağlantı kurulmalıdır. Geçerli bir bağlantı kurulmamışsa, arama RC2009 ile başarısız olur.

HMSG (20 basamaklı işaretli tamsayı)-giriş

Bu tanıtıcı, bir arabelleğin doldurulması gereken ileti tanıtıcısıdır. Değer önceki bir MQCRTMH çağrısı tarafından döndürüldü.

BMHOPT (MQBMHO)-giriş

MQBMHO yapısı, uygulamaların, ileti tutamaçlarının arabelleklerden nasıl üretildiğini denetleyen seçenekleri belirlemesine olanak sağlar.

Ayrıntılar için bkz. [“MQBMHO-Arabelleğiyle ileti tanıtıcısı seçenekleri” sayfa 988.](#)

MSGDSC (MQMD)-giriş/çıkış

MSGDSC yapısı, ileti tanımlayıcı özelliklerini içerir ve arabellek alanının içeriğini açıklar.

Çağırdan 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.

BUFLEN (10 basamaklı işaretli tamsayı)-giriş

BUFLEN , arabellek alanının bayt cinsinden uzunluğudur.

Sıfır baytlık bir BUFLEN değeri geçerlidir ve arabellek alanının veri içermediği anlamına gelir.

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 , özelliklerin kaldırıldığı, arabelleğin bayt cinsinden uzunluğudur.

CMPCOD (10 basamaklı işaretli tamsayı)-çıkış

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 kullanılamıyor.

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 çağrısı.

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 ayrıştırılamadı.

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
```

MQCB-Geri çağırma yönet

MQCB çağrısı, belirtilen nesne tanıtıcısı için geri çağırma yeniden kaydettirir ve geri bildirmeye ilişkin etkinleştirme ve değişiklikleri denetler.

Geri çağrı, belirli olaylar oluştuğunda IBM MQ tarafından çağrılan 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 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 tüketicisi için belirli koşullar (örneğin, RC2016;) için çağrılmaz, ancak geri çağırma işlevi olağan şekilde bitmezse çağrılır.

- “Sözdizimi” sayfa 1222
- “MQCB için kullanım notları” sayfa 1222
- “MQCB için değişirgeler” sayfa 1223
- “RPG Bildirimi” sayfa 1230

Sözdizimi

`MQCB (HCONN, OPERATN, HOBJ, CBDSC, MSGDSC, GMO, CMPCOD, REASON)`

MQCB için 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.

İleti tüketicisi geri çağırma sırası

Tüketiciyi, 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 belirtik olarak ya da örtük olarak silindiğinde.

Çizelge 169. 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 parçacığında çağrılır.

START

Her zaman MQCTL (START) komutu ile zamanuyumlu olarak çağrılır.

- 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 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. 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 MQCB ya da MQCTL komutunu veren uygulamanın sonucu olarak, uygulama geri çağrıldığında, uygulamanın başka bir MQI çağırısı yayınlamaya izin verilir. Bu yönetim ortamı, CBCTRC tipi CBCTRC ile çağrıldığında, tüketici işlevinde bir MQOPEN çağırısı (örneğin, bir MQOPEN çağırısı) yayınlatabileceğiniz anlamına gelir. MQDISC dışında herhangi bir MQI çağırısına izin verilir.

MQCB için değiştirgeler

MQCB çağırısında aşağıdaki değiştirgeler vardır:

HCONN (10 basamaklı işaretli tamsayı)-giriş

Geri bildirme işlevini yönet-HCONN parametresi.

Bu tanıtıcı, kuyruk yöneticisiyle olan bağlantıyı gösterir. The value of *HCONN* was returned by a previous MQCONN or MQCONNX call.

On IBM i for applications running in compatibility mode, the MQCONN call can be omitted, and you can specify the following special value for *HCONN* :

HCDEFH

Varsayılan bağlantı tanıtıcısı.

Önemli:  Uyumluluk kipini kullanmaya devam etseniz, uygulamanızı yeniden derlemeyi ve bunun yerine statik bağlı çağrılar kullanmayı düşünün. Bkz. [IBM üzerindeki RPG uygulamaları için uyumluluk kipi](#)

OPERATN (10 basamaklı işaretli tamsayı)-giriş

Geri bildirme işlevini yönet-OPERATN parametresi.

Belirtilen nesne tanıtıcısı için tanımlanan geri bildirme işleminde işlem yapılıyor. Aşağıdaki seçeneklerden birini belirlemeniz gerekir; birden fazla seçenek gerekliyse, değerler eklenebilir (aynı değişmezi bir kereden fazla eklemeyin) ya da bit ya da işlemi kullanarak (programlama dili, bit işlemleri destekliyorsa).

Geçerli olmayan birleşimler dikkat edilir; diğer tüm birleşimler geçerli olur.

KESME

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 değiştirilir. Geri çağırma değiştirilirken bir hata saptanırsa, işlev kayıttan kaldırılır.

Bir geri çağrı, daha önce kaydı kaldırıldığı aynı geri çağırma işleminde kaydedilirse, bu bir değiştirme işlemi olarak işlem 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.

İlişkili 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 işlem kayıtlı bir tüketicisi olmayan bir *Hobj* için verilirse, çağrı RC2448 ile döner.

CTLSU

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.

CCTLRE

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.

CBDSC (MQCBD)-giriş

Geri bildirme işlevini 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-Callback tanımlayıcısı” sayfa 286 .

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 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 (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
- Oobrw

MSGDSC (MQMD)-giriş

Geri bildirme işlevini yönet -MSGDSC parametresi.

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 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 yaptırsa, 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 “MQGMO-Get-message seçenekleri” sayfa 1044, 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 tüketiciye 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 RC2033 ile 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 bildirimi ya da toplu iş tipi işleme gerçekleştirmesini sağlar.

GMWT ve WIULIM ' dan GMWI

Bu, RC2033 döndürmeden önce sonsuz bekleme değerini belirtir. Tüketici hiçbir zaman RC2033 ile 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 kullanılamıyor.

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') İntilendirme tanıtıcısı hatası.

- RC2010**
(2010, X'7DA') Veri uzunluğu parametresi geçerli değil.
- RC2016**
(2016, X'7E0') Kuyruğun engellenmesini sağlar.
- RC2351**
(2351, X'92F') Genel iş çıkışmaları 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 çıkışıyor.
- RC2247**
(2247, X'8C7') Eşleştirme seçenekleri geçerli değil.
- RC2485**
(2485, X'9B4') Yanlış *MaxMsgLength* 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, 64 bit değil ya da geçerli bir dinamik bağlantı kitaplığı.
- RC2495**
(2495, X'9BF') Modül, arama yolunda bulunamadı ya da yükleme yetkisine sahip değil.
- RC2250**
(2250, X'8CA') İleti sıra numarası geçerli değil.
- RC2331**
(2331, X'91B') İleti simgesinin kullanımı 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.

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 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.

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 kullanılabilecek 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
```

MQCLOSE-Nesneyi kapat

MQCLOSE çağırısı bir nesneye yeniden bağlantı verir ve MQOPEN çağırısının tersi olur.

- [“Sözdizimi” sayfa 1230](#)
- [“Kullanım notları” sayfa 1230](#)
- [“Parametreler” sayfa 1231](#)
- [“RPG Bildirimi” sayfa 1236](#)

Sözdizimi

MQCLOSE (HCONN, HOBJ, OPTS, CMPCOD, REASON)

Kullanım notları

1. Bir uygulama MQDISC çağırısını yayınlarken ya da olağan ya da olağan dışı sona erdirildiğinde, uygulama tarafından açılmış olan ve hala açık olan nesneler 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 çağırısı sırasında bu tanıtıcı için şu anda bir ileti kilitliyse, kilit serbest bırakılır ([“MQGMO-Get-message seçenekleri” sayfa 1044](#) ' 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 çağırısında belirlenen seçeneklerden bağımsız olarak CODEL ya da COPURG seçenekleri belirtilebilir.

- Dinamik bir kuyruk silindiğinde, kuyruğa karşı üstün olan GMWT seçeneğiyle tüm MQGET çağrıları iptal edilir ve RC2052 neden kodunun döndürülmesini sağlar. “MQGMO-Get-message seçenekleri” sayfa 1044’ünde 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 *değil*, bir denetim, MQOPER çağrısının geçerliliğini denetlemek için kullanılan kullanıcı kimliğinin kuyruğu silme yetkisi olduğunu doğrulayın. MQOPEN çağrısında OOALTU seçeneği belirtildiyse, denetlenen kullanıcı kimliği *ODA*U adını işaretir.

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 iletişi 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.

5. IBM üzerinde, uygulama ilk MQOP çağrısı yayınlandığında örtük olarak bağlandıysa, son MQCLOSE komutu verildiğinde örtük bir MQDISC oluşur.

Yalnızca uyumluluk kipinde çalışan uygulamalar örtük olarak bağlanabilir; diğer uygulamalar da, kuyruk yöneticisine belirttik olarak bağlanmak için MQCONN ya da MQCONNx çağrısını yayınlamalıdır.

Önemli:  Uyumluluk kipini kullanmaya devam etseniz, uygulamanızı yeniden derlemeyi ve bunun yerine statik bağlı çağrıları kullanmayı düşünün. Bkz. [IBM üzerindeki RPG uygulamaları için uyumluluk kipi](#)

Parametreler

MQCLOSE çağrısında şu değiştirgeler var:

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*X call.

On IBM i for applications running in compatibility mode, the *MQCONN* call can be omitted, and the following value specified for *HCONN* :

HCDEFH

Varsayılan bağlantı tanıtıcısı.

Önemli:  Uyumluluk kipini kullanmaya devam etseniz, uygulamanızı yeniden derlemeyi ve bunun yerine statik bağlı çağrılar kullanmayı düşünün. Bkz. [IBM üzerindeki RPG uygulamaları için uyumluluk kipi](#)

HOBJ (10 basamaklı işaretli tamsayı)-giriş/çıkış

Nesne tanıtıcısı.

Bu tanıtıcı, kapatılmakta olan nesneyi gösterir. Nesne herhangi bir tipte olabilir. The value of *HOBJ* 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 1331 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 kaldırdığında, 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, kuyruktaki bulunan tüm iletiler temizlenir.

Diğer tüm durumlarda, *Hob*j MQSUB çağrısında döndürüldüğü durum da içinde olmak üzere, çağrı neden kodu RC2045 ile 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, *Hob*j MQSUB çağrısında döndürüldüğü durum da içinde olmak üzere, çağrı neden kodu RC2045 ile başarısız olur ve nesne silinmez.

Sonraki çizelge, hangi kapanış seçeneklerinin geçerli olduğunu ve nesnenin alıkonulup tutulmadığını ve silinip silinmeyeceğini gösterir.

<i>Çizelge 170. Alıkonan ya da silinen nesnelerle kullanmak için geçerli kapanış seçenekleri</i>			
Nesne ya da kuyruk tipi	CONONE	CODEL	KOPÇA
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:

Görev	Abonelik kapatma seçeneği
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:

Görev	Abonelik kapatma seçeneği
Yayınları MQOPENed tanıtıcısında tut	CORMSB
Bir MQOPENed 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 MQCLOSE 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, MQCLOSE 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 MQGET çağrısıyla ve COIMM kullanıldıysa atılacak olan MQCLOSE 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 MQCLOSE 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.

Bu *gerekir* için şu öge belirtilmeli:

- Kuyruktan başka nesneler
- Önceden Tanımlı Kuyruk
- Geçici dinamik kuyruklar (ancak, yalnızca *HOBJ* , kuyruğu yaratan MQOPER çağrısının döndürdüğü tanıtıcıyı *değil*).
- 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 MQOPEN 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 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.

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
```

MQCMIT-Kesinleştirme değişiklikleri

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 kılınsın; bir iş biriminin bir parçası olarak alınan iletiler silinir.

- IBM işletim sistemi, uyumluluk kipinde çalışan uygulamalar için bu çağrıyı desteklemez.

Önemli: **IBM i** Uyumluluk kipini kullanmaya devam etmeniz, uygulamanızı yeniden derlemeyi ve bunun yerine statik bağlı çağrılar kullanmayı düşünün. Bkz. [IBM üzerindeki RPG uygulamaları için uyumluluk kipi](#)

- “Sözdizimi” sayfa 1237
- “Kullanım notları” sayfa 1237
- “Parametreler” sayfa 1238
- “RPG Bildirimi” sayfa 1239

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 “MQDISC-Bağlantı kesme kuyruğu yöneticisi” sayfa 1253 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 seçenekleri” sayfa 1138’inde açıklanan PMLOGO seçeneğine ve “MQGMO-Get-message seçenekleri” sayfa 1044’ 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 etkilenecek, 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 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.

COMCOD (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 *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-Bağlantı kuyruğu yöneticisi

MQCONN çağrısı bir uygulama programını kuyruk yöneticisine bağlar. Sonraki ileti kuyruklama çağrılarında uygulama tarafından kullanılan bir kuyruk yöneticisi bağlantı tanıttıcısı sağlar.

- IBM işletim sistemi, uyumluluk kipinde çalışan uygulamaların bu çağrıyı yayınlamaması gerekmez. Bu uygulamalar, ilk MQOPEN çağrısını yayınlarken kuyruk yöneticisine otomatik olarak bağlanır. Ancak, MQCONN ve MQDISC çağrıları hala IBM i uygulamalarından kabul edilir.

Diğer uygulamalar (yani, uyumluluk kipinde çalışmayan uygulamalar), kuyruk yöneticisine bağlanmak için MQCONN ya da MQCONNX çağrısını ve kuyruk yöneticisinden bağlantıyı kesmek için MQDISC çağrısını kullanmalıdır. Bu tarz bir programlama kullanmayı düşünün.

Önemli:  Uyumluluk kipini kullanmaya devam etmeniz, uygulamanızı yeniden derlemeyi ve bunun yerine statik bağlı çağrıları kullanmayı düşünün. Bkz. [IBM üzerindeki RPG uygulamaları için uyumluluk kipi](#)

IBM MQ for Windows, UNIXve IBM üzerinde, 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 1239
- “Kullanım notları” sayfa 1239
- “Parametreler” sayfa 1240
- “RPG Bildirimi” sayfa 1243

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 üzerinde, kuyruk yöneticisinin MQSeries V5.1 öncesi yayın düzeyleri için yazılan uygulamalar, yeniden derleme gerekmeden çalışabilir.
6. Bu *bir uyumluluk kipi*. Bu işlem kipi, dinamik bağlantı kullanılarak yazılmış uygulamalar için uyumlu bir yürütme ortamı sağlar. Bu, aşağıdaki özellikleri içerir:

- QMQM kitaplığında bulunan AMQZSTUB hizmet programı.

AMQZSTUB, önceki yayın düzeyleriyle aynı genel arabirimi sağlar ve aynı imzayı alır. Bu hizmet programı, bağlı yordam çağrıları yoluyla MQI ' ye erişmek için kullanılabilir.

- QMQM, QMQM kitaplığında bulunan QMQM programı.

QMQM, devingen program çağrıları yoluyla MQI ' ye erişilmesini sağlar.

- MQCLOSE, MQCONN, MQDISC, MQGET, MQINQ, MQOPEN, MQPUT, MQPUT1, ve QMQM kitaplığında bulunan MQSET programları programlar.

Bu programlar, dinamik program çağrılarına, ancak IBM MQ çağrılarının standart açıklamalarına karşılık gelen bir parametre listesiyle birlikte MQI ' ye erişim olanağı da sağlar.

Önemli:  Uyumluluk kipini kullanmaya devam etmeniz, uygulamanızı yeniden derlemeyi ve bunun yerine statik bağlı çağrıları kullanmayı düşünün. Bkz. [IBM üzerindeki RPG uygulamaları için uyumluluk kipi](#)

These three interfaces do not include capabilities that were introduced in version 5.1. Örneğin, MQBACK, MQCMIT ve MQCONNX çağrıları desteklenmez. Bu arabirimler tarafından sağlanan destek, yalnızca tek iş parçacıklı uygulamalar içindir.

Tek iş parçacıklı uygulamalarda ve çok iş parçacıklı uygulamalarda tüm IBM MQ çağrıları için statik bağlı IBM MQ çağrıları için destek, LIBMQM ve LIBMQM_R hizmet programları aracılığıyla sağlanır.

7. 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 değişirgeler vardır:

QMNAME (48 baytlık karakter dizilimi)-giriş

Kuyruk yöneticisinin adı.

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 (/), 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 paylaşımı grupları: Birkaç kuyruk yöneticisinin var olduğu ve bir kuyruk paylaşım grubu oluşturmak üzere yapılandırıldığı sistemlerde, kuyruk paylaşım grubunun adı, kuyruk yöneticisi adının yerine *QMNAME* için belirtilebilir. Bu, uygulamanın, kuyruk paylaşımı grubunda bulunan *any* kuyruk yöneticisine bağlanmasını sağlar. Sistem, boş bir *QMNAME* ' un varsayılan kuyruk yöneticisi yerine kuyruk paylaşım grubuyla bağlantı oluşmasına neden olması için de yapılandırılabilir.

QMNAME , kuyruk paylaşım grubunun adını belirtiyorsa, ancak sistemde o adı taşıyan bir kuyruk yöneticisi de varsa, ikinciye tercihte eski olan bir kuyruk yöneticisi de vardır. 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 bağlantısı 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, *QMqrName* kuyruk yöneticisi özneteliğini 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.

On IBM i for applications running in compatibility mode, the value returned is:

HCDEFH

Varsayılan 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 CCWARN ise:

RC2002

(2002, X'7D2') Uygulaması zaten bağlı.

CMPCOD CCCFAIL 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
```

MQCONN-Connect kuyruk yöneticisi (genişletilmiş)

MQCONN çağrısı bir uygulama programını kuyruk yöneticisine bağlar. Bu, sonraki IBM MQ çağrılarında uygulama tarafından kullanılan bir kuyruk yöneticisi bağlantı tanıtıcısı sağlar.

MQCONN çağrısı, MQCONN çağrısının, çağrıların çalışma şeklini denetlemek için belirtilmesine izin vermesi dışında, MQCONN çağrısına benzer.

- IBM işletim sistemi, uyumluluk kipinde çalışan uygulamalar için bu çağrıyı desteklemez.

Önemli:  Uyumluluk kipini kullanmaya devam etmeniz, uygulamanızı yeniden derlemeyi ve bunun yerine statik bağlı çağrılar kullanmayı düşünün. Bkz. [IBM üzerindeki RPG uygulamaları için uyumluluk kipi](#)

IBM MQ for Windows, UNIX ve IBM üzerinde, bir uygulamadaki her iş parçası farklı kuyruk yöneticilerine bağlanabilir. Diğer sistemlerde, bir işlem içindeki koşut zamanlı tüm bağlantıların aynı kuyruk yöneticisi olması gerekir.

- “Sözdizimi” sayfa 1244
- “Parametreler” sayfa 1244
- “RPG Bildirimi” sayfa 1245

Sözdizimi

`MQCONN (QMNAME, CNOPT, HCONN, CMPCOD, REASON)`

Parametreler

MQCONN çağrısında şu değişirler vardır:

QMNAME (48 baytlık karakter dizilimi)-giriş

Kuyruk yöneticisinin adı.

Ayrıntılar için “MQCONN-Bağlantı kuyruğu yöneticisi” sayfa 1239 içinde açıklanan *QMNAME* parametresine bakın.

CNOPT (MQCNO)-giriş/çıkış

MQCONN 'in işlemini denetleyen seçenekler.

Ayrıntılar için bkz. “MQCNO-Bağlantı seçenekleri” sayfa 1017.

HCONN (10 basamaklı işaretli tamsayı)-çıkış

Bağlantı tanıtıcısı.

Ayrıntılar için “MQCONN-Bağlantı kuyruğu yöneticisi” sayfa 1239 içinde açıklanan *HCONN* parametresine bakın.

CMPCOD (10 basamaklı işaretli tamsayı)-çıkış

Tamamlanma kodu.

Ayrıntılar için “MQCONN-Bağlantı kuyruğu yöneticisi” sayfa 1239 içinde açıklanan *CMPCOD* parametresine bakın.

REASON (10 basamaklı işaretli tamsayı)-çıkış

Neden kodu ön eleme *CMPCOD*.

Olası neden kodlarına ilişkin ayrıntılar için “MQCONN-Bağlantı kuyruğu yöneticisi” sayfa 1239 içinde açıklanan *REASON* parametresine bakın.

İzleyen ek neden kodları, MQCONN çağrısı tarafından döndürülebiliyor:

CMPCOD CCFAIL ise:

RC2278

(2278, X'8E6') İstemci bağlantı alanları geçerli değil.

RC2139

(2139, X'85B') Connect-options yapısı geçerli değil.

RC2046

(2046, X'7FE') Seçenekler geçerli değil ya da tutarlı değil.

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* 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
```

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 iletin özellikleriyle ilişkilendirir. Benzer şekilde, MQGET çağrısında bir ileti tanıtıcısı belirleyerek, MQGET çağrısı tamamlandığında ileti tanıtıcısı kullanılarak, alınmakta olan iletin özelliklerine erişilebilir.

İleti tanıtıcısını silmek için [MQDLTMH](#) ögesini kullanın.

- “Sözdizimi” sayfa 1245
- “Parametreler” sayfa 1245
- “RPG Bildirimi” sayfa 1247

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 parçasısında bulunan bir kuyruk yöneticisine en az bir geçerli bağlantı olmalıdır; tersi durumda, arama RC2018 ile başarısız olur.

On IBM i for applications running in compatibility mode, the MQCONN call can be omitted, and you can specify the following value for *HCONN* :

HCDEFH

Varsayılan bağlantı tanıtıcısı

Önemli:  Uyumluluk kipini kullanmaya devam etseniz, uygulamanızı yeniden derlemeyi ve bunun yerine statik bağlı çağrılar kullanmayı düşünün. Bkz. [IBM üzerindeki RPG uygulamaları için uyumluluk kipi](#)

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 çağrıları kullanarak 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
- MQMHBUFF
- 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 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.

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 1388](#) .

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
```

MQCTL-Geri çağırma denetimi

MQCTL çağrısı, bağlantı için açılan nesne tanıtıcılarında denetleme işlemlerini gerçekleştirir.

- [“Sözdizimi” sayfa 1248](#)
- [“Kullanım notları” sayfa 1248](#)

- “Parametreler” sayfa 1248
- “RPG Bildirimi” sayfa 1253

Sözdizimi

MQCTL (*Hconn, Operation, ControlOpts, CompCode, Reason*)

Kullanım notları

1. Geri bildirme yordamları, çağırdıkları tüm hizmetlerden gelen yanıtları denetlemeli ve yordam çözilemeyen bir koşul saptarsa, geri çağırma yordamlarına yinelenen çağrılarının önlenmesi için bir MQCB (CBREG) komutu yayınlamalıdır.

Parametreler

MQCTL çağırısı aşağıdaki parametrelere sahiptir:

HCONN (10 basamaklı işaretli tamsayı)-giriş

Bu tanıttıcı, kuyruk yöneticisiyle olan bağlantıyı gösterir. The value of *HCONN* was returned by a previous MQCONN or MQCONNX call.

On IBM i for applications running in compatibility mode, the MQCONN call can be omitted, and you can specify the following special value for *HCONN* :

HCDEFH

Varsayılan bağlantı tanıttıcısı.

Önemli:  Uyumluluk kipini kullanmaya devam etseniz, uygulamanızı yeniden derlemeyi ve bunun yerine statik bağlı çağrılarını kullanmayı düşünün. Bkz. [IBM üzerindeki RPG uygulamaları için uyumluluk kipi](#)

OPERATN (10 basamaklı işaretli tamsayı)-giriş

Belirtilen nesne tanıttıcısı için tanımlanan geri bildirme işleminde işlem yapılıyor. Aşağıdaki seçeneklerden birini ve yalnızca birini belirtmeniz gerekir:

CTLSR

Belirtilen bağlantı tanıttıcısı 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 tanıttıcısı denetimini verir. Tüketici iş parçacığı dışındaki bir iş parçacığı tarafından yayınlanabilen yalnızca MQI çağrıları şunlardır:

- CTLSP İşlemi ile MQCTL
- CTLSU İşlemi ile MQCTL
- MQDISC-HConn bağlantısı kesilmeden önce, bu işlem CTLSP işlemi ile MQCTL ' yi gerçekleştirir.

Bağlantı tanıttıcısı başlatıldığında bir IBM MQ API çağırısı 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 çağırısı 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 CTLSP 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 tütütmesini başlatın.

İleti tüketicileri aynı iş parçacığı üzerinde çalışır ve denetim MQCTL ' nin çağrısına döndürülünceye kadar döndürülmez:

- MQCTL CTLSW 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 CTLSW işlemi yayınlanır.

Bu seçenek, yürürlükteki bağlantı tanıtıcısı ya da başka bir bağlantı tanıtıcısı için, geri bildirme yordamından kullanılamaz. Arama girişiminde bulunulursa, RC2012 ile birlikte geri döndürülür.

Bir CTLSW işlemi sırasında herhangi bir zamanda kayıtlı olmayan, askıya alınmış olmayan tüketicilere arama başarısız olur ve bir RC2446 neden kodu ile başarısız olur.

Bir CTLSW işlemi sırasında bağlantı askıya alınırsa, MQCTL çağrısı RC2521; için bir uyarı neden kodu döndürür; bağlantı 'başlatıldı' kalır.

Uygulama CTLSW 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 (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 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 çağrılanmaz.

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 kullanılamıyor.

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 çıkışmaları 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 çıkışıyor.

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 işlev 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-tanıtıcısı 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 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.

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 kullanılabilecek 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
```

MQDISC-Bağlantı kesme kuyruğu yöneticisi

MQDISC çağrısı, kuyruk yöneticisi ile uygulama programı arasındaki bağlantıyı keser ve MQCONN ya da MQCONNx çağrısının tersi olur.

- IBM işletim sistemi, uyumluluk kipinde çalışan uygulamaların bu çağrıyı yayınlamaya gerek yoktur. Ek bilgi için [“MQCONN-Bağlantı kuyruğu yöneticisi” sayfa 1239](#) başlıklı konuya bakın.

Önemli:  Uyumluluk kipini kullanmaya devam etmeniz, uygulamanızı yeniden derlemeyi ve bunun yerine statik bağlı çağrıları kullanmayı düşünün. Bkz. [IBM üzerindeki RPG uygulamaları için uyumluluk kipi](#)

- [“Sözdizimi” sayfa 1253](#)
- [“Kullanım notları” sayfa 1253](#)
- [“Parametreler” sayfa 1254](#)
- [“RPG Bildirimi” sayfa 1255](#)

Sözdizimi

MQDISC (HCONN, CMPCOD, REASON)

Kullanım notları

1. Bir MQDISC çağrısı, uygulama hala nesneleri açıkken yayınlanırsa, bu nesneler 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.

3. IBM işletim sistemi, uyumluluk kipinde çalışan uygulamaların bu çağrışı yayınlamaması gerekmez; daha fazla ayrıntı için MQCONN çağrısına bakın.

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.

On IBM i for applications running in compatibility mode, the MQCONN call can be omitted, and the following value specified for *HCONN*:

HCDEFH

Varsayılan bağlantı tanıtıcısı.

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
```

MQDLTMH-İleti tanıtıcısını sil

MQDLTMH çağrısı bir ileti tanıtıcısını siler ve MQCRTMH çağrısının tersi olur.

- “Sözdizimi” sayfa 1255
- “Kullanım notları” sayfa 1255
- “Parametreler” sayfa 1257
- “RPG Bildirimi” sayfa 1258

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. “MQBEGIN-İş birimini başlat” sayfa 1216.

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 çağrılar 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 [“MQDISC-Bağlantı kesme kuyruğu yöneticisi” sayfa 1253](#) 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 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 yedeklediye, 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 yüklemeyiz.

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 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 \(10 basamaklı işaretli tamsayı\)](#)-çıkış başlıklı konuya bakın.
6. Bu çağrıdan etkilenen, 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 gerileme çağrısını hiçbir zaman yayınlamayan, uzun süredir çalışan bir uygulama, diğer uygulamaların kullanımına sunulmayan iletilerle kuyrukları doldurabilir. Bu olasılığa karşı koruma sağlamak için, yöneticinin *MaxUncommittedMsgs* queue-manager öznelikliğini, kuyrukları dolduran

kaçak uygulamaları önleyecek kadar düşük bir değere ayarlaması gerekir, ancak beklenen ileti sistemi uygulamalarının doğru şekilde çalışmasına izin verecek kadar yüksek.

Parametreler

MQDLTMH çağrısı aşağıdaki deęiřtirgeleri ierir:

HCONN (10 basamaklı iřaretli tamsayı)-giriř

Bu tanıtıcı, kuyruk yöneticiyle olan baęlantıyı gösterir.

Deęerin, *HMSG* parametresinde belirtilen ileti tanıtıcısını yaratmak iin kullanılan baęlantı tanıtıcısı ile eřleşmesi gerekir.

İleti tanıtıcısı HCUNAS kullanılarak yaratıldıysa, ileti tanıtıcısını silmeye iliřkin iř paracısında geerli bir baęlantı oluřturulmalıdır; tersi durumda, aęrı RC2009 ile bařarısız olur.

HMSG (20 basamaklı iřaretli tamsayı)-giriř/ıkıř

Bu, silinmek üzere ileti tanıtıcısıdır. Deęer önceki bir MQCRTMH aęrısı tarafından döndürüldü.

aęrıyı bařarıyla tamamlamaya iliřkin tanıtıcı, ortam iin geersiz bir değere ayarlıdır. Bu değeri:

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 iin bkz. [MQDMHO](#) .

CMPCOD (10 basamaklı iřaretli tamsayı)-ıkıř

Tamamlanma kodu; bu kod aşağıdakilerden biridir:

CCOK

İřlem bařarıyla tamamlandı.

CCCFAIL

Arama bařarısız oldu.

REASON (10 basamaklı iřaretli tamsayı)-ıkıř

Neden kodu niteleyici *CMPCOD*.

CMPCOD CCOK ise:

YOK

(0, X'000 ') Raporlamak iin bir neden yok.

CMPCOD değeri CCFAIL ise:

RC2204

(2204, X'089C') Baędařtırıcı kullanılmıyor.

RC2130

(2130, X'852 ') Baędařtırıcı hizmeti modülü yüklenemiyor.

RC2157

(2157, X'86D') Birincil ve ana sunucu ASID değeri farklı.

RC2219

(2219, X'08AB') Önceki arama tamamlanmadan önce girilen MQI aęrısı.

RC2009

(2009, X'07D9') Kuyruk yöneticisine yönelik baęlantı kaybedildi.

RC2462

(2462, X'099E') İleti tanıtıcısı seeneklerini silme yapısı geerli deęil.

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 1388](#).

RPG Bildirimi

```
C*..1.....2.....3.....4.....5.....6.....7..
C                                CALLP      MQDLTMH(HCONN : HMSG : DLTPT :
                                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 DLTPT                      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 1258](#)
- [“Parametreler” sayfa 1258](#)
- [“RPG Bildirimi” sayfa 1260](#)

Sözdizimi

MQDLTMP (*Hconn, Hmsg, DltPropOpts, Name, CompCode, Reason*)

Parametreler

MQDLTMP çağrısı aşağıdaki değiştirgeleri içerir:

HCONN (10 basamaklı işaretli tamsayı)-Giriş

Bu tanıtıcı, kuyruk yöneticiyle olan bağlantıyı gösterir. Değerin, *HMSG* parametresinde belirtilen ileti tanıtıcısını yaratmak için kullanılan bağlantı tanıtıcısı ile eşleşmesi 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)-giriş

Bu, silinecek özelliği içeren ileti tanıtıcısıdır. Değer önceki bir MQCRTMH çağrısı tarafından döndürüldü.

DLTOPT (MQDMPO)-Giriş

Ayrıntılı bilgi için MQDMPO veri tipine bakın.

PRNAME (MQCHARV)-giriş

Silinecek özelliğin adı. Özellik adlarıyla ilgili ek bilgi için Özellik adları konusuna bakın.

Özellik adında genel arama karakterlerine izin verilmez.

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 *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 ayrıştırılamadı.

CMPCOD CCFAIL ise:

RC2204

(2204, X'089C') Bağdaştırıcı kullanılamıyor.

RC2130

(2130, X'0852 ') Bağdaştırıcı hizmeti modülü yüklenemiyor.

RC2157

(2157, X'086D') Birincil ve ana ASID değerleri farklı.

RC2219

(2219, X'08AB') Önceki arama tamamlanmadan önce girilen MQI çağırısı.

RC2009

(2009, X'07D9') Kuyruk yöneticisine yönelik bağlantı kaybedildi.

RC2481

(2481, X'09B1') İleti silme seçenekleri yapısı geçerli değil.

RC2460

(2460, X'099C') İleti tanıtıcısı 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 geçersiz.

RC2111

(2111, X'083F') Özellik adı kodlanmış karakter takımı tanıtıcısı geçerli değil.

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)
```

Çağrıya 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
```

MQGET-İletiyi al

MQGET çağırısı, MQOPEN çağırısı kullanılarak açılmış olan bir yerel kuyruktan bir ileti alır.

- [“Sözdizimi” sayfa 1260](#)
- [“Kullanım notları” sayfa 1260](#)
- [“Parametreler” sayfa 1263](#)
- [“RPG Bildirimi” sayfa 1268](#)

Sözdizimi

MQGet (*HCONN*, *HOBJ*, *MSGDSC*, *GMO*, *BUFLN*, *BUFFER*, *DATLEN*, *CMPCOD*, *REASON*)

Kullanım notları

1. Alınan ileti olağan durumda kuyruktan silinir. Bu silme, MQGET çağırısının 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 [“MQGMO-Get-message seçenekleri” sayfa 1044](#)).
2. Göz atma seçeneklerinden biri olan GMLK seçeneği belirtilirse, göz atma ileti yalnızca bu tutamaçla görünür olacak şekilde kilitlenir.

GMUNLK seçeneği belirtilirse, önceden kilitlenmiş bir iletinin kilidi açılır. Bu durumda hiçbir ileti alınmaz; *MSGDSC*, *BUFLN*, *BUFFER* ve *DATLEN* parametreleri denetlenmez ya da değiştirilmez.
3. MQGET çağırısını yayınlayan uygulama IBM MQ MQI client olarak çalışıyorsa, MQGET çağırısının 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 çağırısının 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* bir kuyruksa (yani, yerel bir kuyruksa), iş birimi içindeki tüm iletiler aynı anda kullanılabilir olur.
 - Kuyruk bir *paylaşılan* kuyruksa, iş birimi içindeki iletiler 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 RC2033 ile 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 *MDCID* belirtilerek, 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 başlıklara bakın:
 - MQMD ' de *MDMFL* 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-İleti tanımlayıcısı” sayfa 1076](#) içinde açıklanan *MDFB* alanına bakın.
7. *HOB*J ile tanımlan kuyruk OOSAVA seçeneğiyle açıldıysa ve MQGET çağrısından tamamlanma kodu CCOK ya da CCWARN ise, *HOB*J 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 iletilindiğinde) aktarılmasını sağlar. İleti bağlamına ilişkin daha fazla bilgi için bkz. [İleti bağlamı ve Bağlam bilgilerini denetleme](#).
8. DGMCONV seçeneği *GMO* parametresine eklenirse, uygulama iletileri verileri, veri *BUFFER* parametresine yerleştirilmeden önce, giriş uygulaması tarafından istenen gösteriye dönüştürülür:
 - İletideki denetim bilgilerindeki *MDFMT* alanı, uygulama verilerinin yapısını tanımlar ve iletideki denetim bilgilerindeki *MDCSI* ve *MDENC* alanları, karakter kümesi tanıtıcısını ve kodlamasını belirtir.
 - MQGET çağrısını yayınlayan uygulama, *MSGDSC* parametresindeki *MDCSI* ve *MDENC* alanlarında, uygulama iletileri verilerinin dönüştürülmesi gereken karakter kümesi tanıtıcısını ve kodlamasını belirtir.

İleti verilerinin dönüştürülmesi gerekiyorsa, 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
FMCICS	FMPCF
FMCMD1	FMRMH
FMCMD2	FNRFH
FBIDLH	FMRFH2
FMDH	FMSTR
FMEVNT	FMTM
FMIMS	FMXQH
FMIMVS	

- FMNONE biçim adı, iletteki verilerin niteliğin 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 niteliği dönüştürme gerektirmediği anlamına geldiğinde ya da gönderme ve alma uygulamaları, kendileri arasında ileti verilerinin gönderilmesi gereken form arasında anlaştığında 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 UCS-2, bu tür bir karakter kümesine bir örnektir.

MQGET ' den dönüştürme, 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: The interpretation of the reason code described in this example is true for conversions performed by user-written exits *yalnızca* if the exit conforms to the processing guidelines.

- Daha önce listelenen yerleşik biçimler için, kuyruk yöneticisi, GMCONV 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 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 dönüştürülmesi yalnızca aşağıdakilerin *tüm* değeri 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 belirlenmese 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.

On IBM i for applications running in compatibility mode, the MQCONN call can be omitted, and the following value specified for *HCONN* :

HCDEFH

Varsayılan bağlantı tanıtıcısı.

Önemli:  Uyumluluk kipini kullanmaya devam etseniz, uygulamanızı yeniden derlemeyi ve bunun yerine statik bağlı çağrılarını kullanmayı düşünün. Bkz. [IBM üzerindeki RPG uygulamaları için uyumluluk kipi](#)

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 nesnesi” sayfa 1284](#) 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-İleti tanımlayıcısı” sayfa 1076](#).

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 [“MQGMO-Get-message seçenekleri” sayfa 1044](#)).

Uygulama bir version-1 MQMD sağlıyorsa, döndürülen iletinin uygulama iletisi verilerine eklenmiş bir MQMDE öneki vardır; ancak, MQMDE içindeki bir ya da daha çok alanın varsayılan değeri olmayan bir değeri varsa, *yalnızca* . 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. [“MQGMO-Get-message seçenekleri” sayfa 1044](#).

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 1331](#).

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 [“MQGMO-Get-message seçenekleri” sayfa 1044](#) 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 [“MQGMO-Get-message seçenekleri” sayfa 1044](#) 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.

This is the length in bytes of the application data *İletide*. Bu ileti uzunluğu *BUFLLEN*değerinden büyükse, *BUFFER* parametresine yalnızca *BUFLLEN* byte 'ları döndürülür (yani, ileti kesilir). Değer sıfırsa, iletinin uygulama verisi içermediği anlamına gelir.

If *BUFLLEN* 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 [“MQGMO-Get-message seçenekleri” sayfa 1044](#) 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 *BUFFER*içine sığması çok uzun olursa, *DATLEN* için döndürülen değer şöyledir:

- 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 *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 çağrılırsa, *REASON* değiştirgesi için hangi değer döndürüleceğine karar veren çıkışıdır. Sonuç olarak, bu bölümde daha sonra belgelenen değerler dışındaki değerler 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 kilitlenmedi.

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 çağırısı, 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 kullanılabilecek 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
```

MQINQ-Nesne öznitelikleri ile ilgili bilgi

MQINQ çağırısı, 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 1268](#)
- [“Kullanım notları” sayfa 1268](#)
- [“Parametreler” sayfa 1269](#)
- [“RPG Bildirimi” sayfa 1276](#)

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 aşmanız bile geçerlidir.

Dinamik kuyruğun öznitelikleri (belirli kural dışı durumlarla), dinamik kuyruk yaratıldığı sırada, model kuyruklarıyla aynı olur. Bu kuyruktan sonra MQINQ çağırısını 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 kuyruğunsa, MQINQ çağrısının döndürdüğü öznitelik değerleri diğer ad kuyruğunun ve diğer adın çözümleneceği temel kuyruklardır.
4. Sorgulanmakta olan nesne bir küme kuyruğysa, 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) kullanılabileceği 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 1331
 - “Ad listelerine ilişkin öznitelikler” sayfa 1359
 - “Süreç tanımlamalarına ilişkin öznitelikler” sayfa 1360
 - “Kuyruk yöneticisine ilişkin öznitelikler” sayfa 1362
8. Yeni bir yerel kuyruk SYSTEM.ADMIN.COMMAND.EVENT , komutlar ne zaman yayınlandığında oluşturulan kuyruğa alma iletileri için kullanılır. CMDEV kuyruk yöneticisi özniteliğinin nasıl ayarlanırsa 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 değiştireleri 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 *MQCONN*X call.

On IBM i for applications running in compatibility mode, the *MQCONN* call can be omitted, and the following value specified for *HCONN* :

HCDEFH

Varsayılan bağlantı tanıtıcısı.

Önemli:  Uyumluluk kipini kullanmaya devam etseniz, uygulamanızı yeniden derlemeyi ve bunun yerine statik bağlı çağrılar kullanmayı düşünün. Bkz. [IBM üzerindeki RPG uygulamaları için uyumluluk kipi](#)

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. Döndürülemede 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, nesneden dışındaki tip ya da tip kuyruklarına *yalnızca* uygunsa, 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* değişmezleri, kuyruk yöneticisinin kabul ettiği en yüksek değeri tanımlar.

2. Tüm IA* seçicileri önce gerçekleşirse, aynı öge numaraları *SELS* ve *INTATR* dizilerinde karşılık gelen öğeleri adreslemek için kullanılabilir.

Sorgulanabilen öznitelikler aşağıdaki 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 171. Kuyruklar için MQINQ öznitelik seçicileri.

Notların açıklaması için çizelgenin alt kısmına bakın.

Seçici	Tanım	Not
CAALTD	En son değiştirilme tarihi (LNDATE).	1
CATTT	En son değiştirilme zamanı (LNTIME).	1
CABRQN	Aşırı yedek-istek adı (LNQN).	5
CABASQ	Diğer adın çözdüğü kuyruğun adı (LNQN).	
CACFSN	Bağlaşım-tesis yapısı adı (LNCFSN).	3
KILLER	Küme adı (LNCLUN).	1
CACLNL	Küme adlistesi (LNNLN).	1
CACRTD	Kuyruk yaratma tarihi (LNCRTD).	
CACRTT	Kuyruk yaratma süresi (LNCRTT).	
CAINIQ	Başlatma kuyruğu 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) bulunduğu gibi uzak kuyruğun adı.	
CATRGD	Tetikleme verileri (LNTRGD).	5
CAXQN	İletim kuyruğu adı (LNQN).	
IABTHR	Geriletme eşiği.	5
IACDEP	Kuyruktaki ileti sayısı.	
IADBND	Varsayılan bağ tanımı.	1
IADINP	Giriş için varsayılan açma seçeneği.	5
IADPER	Varsayılan ileti kalıcılığı.	
IADPRI	Varsayılan ileti önceliği.	5
KALDIRIM	Kuyruk tanımlaması tipi.	
IADIST	Dağıtım listesi desteği.	2
IAHGB	-Geri sayım mı?-Evet.	5
IAIGET	Alma işlemlerine izin verilip verilmediği.	
IAIPUT	Koyma işlemlerine izin verilip verilmediği.	
IAMLEN	İleti uzunluğu üst sınırı.	
IAMDEP	Kuyruktan izin verilen ileti sayısı üst sınırı.	
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.	

Çizelge 171. Kuyruklar için MQINQ öznitelik seçicileri.

Notların açıklaması için çizelgenin alt kısmına bakın.

(devamı var)

Seçici	Tanım	Not
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 paylaşılamayacağı.	
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. Bu sistemlere bağlı AIX, HP-UX, z/OS, IBM i, Solaris, Windows, artı IBM MQ MQI clients sistemlerinde desteklenir.
2. Bu sistemlere bağlı AIX, HP-UX, IBM i, Solaris, Windows, artı IBM MQ istemcilerinde desteklenir.
3. z/OSüzerinde desteklenir.
4. z/OSüzerinde desteklenmez.
5. VSE/ESAüzerinde desteklenmez.

Çizelge 172. Ad listeleri için MQINQ öznitelik seçicileri.

Notlara ilişkin açıklamalar için bkz. [notlar](#) .

Seçici	Tanım	Not
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	Ad listesindeki adlar (listede LNQN x ad sayısı)	1

Çizelge 172. Ad listeleri için MQINQ öznitelik seçicileri.

Notlara ilişkin açıklamalar için bkz. [notlar](#) .

(devamı var)

Seçici	Tanım	Not
IANAMC	Namelist 'teki ad sayısı	1
IAQSGD	Kuyruk paylaşımı grubu yok etme	3

Çizelge 173. Süreç tanımlamaları için MQINQ öznitelik seçicileri.

Notlara ilişkin açıklamalar için bkz. [notlar](#) .

Seçici	Tanım	Not
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

Çizelge 174. Kuyruk yöneticisine ilişkin MQINQ öznitelik seçicileri.

Notlara ilişkin açıklamalar için bkz. [notlar](#) .

Seçici	Tanım	Not
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
CAQMN	Yerel kuyruk yöneticisinin adı (LNQMN)	5
CAQSGN	Kuyruk paylaşım grubu adı (LNQSGN)	3
CARPN	Kuyruk yöneticisinin havuz hizmetleri sağladığı kümenin adı (LNQMN)	1

Çizelge 174. Kuyruk yöneticisine ilişkin MQINQ öznitelik seçicileri.

Notlara ilişkin açıklamalar için bkz. [notlar](#) .

(devamı var)

Seçici	Tanım	Not
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
IALCLE	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.

*HOB*J 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, *CHRA*TR 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.

*HOB*J bir kuyruğu temsil ediyorsa, ancak bir öznitelik seçici o kuyruk tipi için geçerli değilse, tam olarak yıldız işaretleri (*) içeren bir karakter dizgisi, *CHRA*TR' taki o özneliğin değeri olarak döndürülür.

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:

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 çağırısı, 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                                CALEN : CHRATR : CMPCOD :
C                                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

```

MQINQMP-Questery ileti özelliği

MQINQMP çağrısı, bir iletinin özellik değerini döndürür.

- “Sözdizimi” sayfa 1277
- “Parametreler” sayfa 1277
- “RPG Bildirimi” sayfa 1281

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 yöneticiyle 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 RC2471ile 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. 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, "JMSExpiration", "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 *IPITYP* 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 "[MQIMPO-Sorgula ileti özelliği seçenekleri](#)" sayfa 1069 ' in IPCTYP seçeneğinin açıklamasına bakın.

Tip dönüştürme isteğinde bulunmuyorsanız, giriş sırasında aşağıdaki değeri kullanabilirsiniz:

TÜRÜ

Özellik 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 dizgisi x VALLEN)-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 IPRETCSI 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 arabellenin 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 ayrıştırılamadı.

CMPCOD CCFAIL ise:

RC2204

(2204, X'089C') Bağdaştırıcı kullanılamıyor.

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 İletiler ve neden kodları](#)

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 tanıtıcısını arabelleğe dönüştür

MQMHBUF, bir ileti tanıtıcısını arabellek içine dönüştürür ve MQBUFMH çağrısının tersi olur.

- [“Sözdizimi” sayfa 1281](#)
- [“Kullanım notları” sayfa 1282](#)
- [“Parametreler” sayfa 1282](#)
- [“RPG Bildirimi” sayfa 1284](#)

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 değişirgeleri vardır:

HCONN (10 basamaklı işaretli tamsayı)-giriş

Bu tanıtıcı, kuyruk yöneticiyle olan bağlantıyı gösterir.

HCONN değeri, *HMSG* değiştirgesinde belirtilen ileti tanıtıcısını yaratmak için kullanılan bağlantı tanıtıcısı ile eşleşmelidir.

İleti tanıtıcısı HCUNAS kullanılarak yaratıldıysa, ileti tanıtıcısını silmeye ilişkin iş parçacığıda geçerli bir bağlantı kurulmalıdır. Geçerli bir bağlantı kurulmıyışsa, arama RC2009 ile başarısız olur.

HMSG (20 basamaklı işaretli tamsayı)-giriş

Bu tanıtıcı, bir arabelleğin doldurulması gereken ileti tanıtıcısıdır.

Değer önceki bir MQCRTMH çağrısı tarafından döndürüldü.

MHBOPT (MQMHBO)-giriş

MQMHBO yapısı, uygulamaların, ileti tanıtıcılarından arabelleklerin nasıl üretildiğini denetleyen seçenekleri belirlemesine olanak sağlar.

Ayrıntılar için bkz. [“MQBMHO-Arabelleğiyle ileti tanıtıcısı seçenekleri” sayfa 988](#).

PRNAME (MQCHARV)-giriş

Arabelleğe yerleştirilecek özelliğin ya da özelliklerin adı.

Adla eşleşen bir özellik bulunamazsa, arama RC2471 ile başarısız olur.

jokerler

Arabelleğe birden çok özellik koymak için genel arama karakteri kullanabilirsiniz. Bunu yapmak için, özellik adının sonunda yüzde işaretini (%) kullanın. Bu genel arama karakteri, nokta (.) karakteri de içinde olmak üzere, sıfır ya da daha fazla karakterle eşleşir.

Özellik adlarının kullanımına ilişkin ek bilgi için [Özellik adları](#) ve [Özellik adı kısıtlamaları](#) başlıklı konuya bakın.

MSGDSC (MQMD)-giriş/çıkış

MSGDSC 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.

BUFLEN (10 basamaklı işaretli tamsayı)-giriş

BUFLEN, 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      MQMHBUFF(HCONN : HMSG : MHBOPT :
                                           PRNAME : MSGDSC : BUFLN :
                                           BUFFER : DATLEN :
                                           CMPCOD : REASON)
```

Çağrıya ilişkin prototip tanımı:

```
DMQMHBUFF      PR              EXTPROC('MQMHBUFF')
D* Connection handle
D HCONN              10I 0 VALUE
D* Message handle
D HMSG              20I 0 VALUE
D* Options that control the action of MQMHBUFF
D MHBOPT              12A
D* Property name
D PRNAME              32A
D* Message descriptor
D MSGDSC              364A
D* Length in bytes of the Buffer area
D BUFLN              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
```

MQOpen-Open nesnesi

MQOPEN çağrısı bir nesneye erişim oluşturur.

Aşağıdaki nesne tipleri geçerlidir:

- Kuyruk (dağıtım listeleri de içinde olmak üzere)
- Ad Listesi
- Süreç tanımlaması
- Kuyruk yöneticisi
- Konu
- [“Sözdizimi” sayfa 1285](#)
- [“Kullanım notları” sayfa 1285](#)

- “Parametreler” sayfa 1289
- “RPG Bildirimi” sayfa 1295

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 kuyruğusa, 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 çağrılılarıyla 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. Çözümleme 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 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 model kuyruğuysa, 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 ("[MQOD-Nesne tanımlayıcısı](#)" sayfa 1124 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, kuyrukta 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. OOBW 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* değiştirilmesinde yalnızca aşağıdaki açık seçenekler geçerlidir:
 - *OOOUT*
 - *OOPAS**
 - *OOSET**
 - *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 değiştirilme aşağıdaki gibi ayarlanır:
 - Dağıtım listesindeki kuyruklara ilişkin açık işlemler başarılı olursa ya da aynı şekilde başarısız olursa, tamamlanma kodu ve neden kodu değiştirilme ortak sonucu tanımlamaya ayarlanır. Bu durumda, MQRR 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 hiçbirisi 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 RC2136olarak 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 *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 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 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 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 RC2017ile 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 çözülsürse, 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 RC2189ile 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:

- OOBROW
- 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 hiçbirini 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. IBM işletim sistemi üzerinde, uyumluluk kipinde çalışan uygulamalar, uygulama tarafından yayınlanan ilk MQOPEN çağrısıyla otomatik olarak kuyruk yöneticisine bağlanır (uygulama, MQCONN çağrısını kullanarak kuyruk yöneticisine önceden bağlanmadıysa).

Önemli:  Uyumluluk kipini kullanmaya devam etseniz, uygulamanızı yeniden derlemeyi ve bunun yerine statik bağlı çağrılarını kullanmayı düşünün. Bkz. [IBM üzerindeki RPG uygulamaları için uyumluluk kipi](#)

Uyumluluk kipinde çalışmayan uygulamalar, bir nesneyi açmak için MQOPED çağrısını kullanmadan önce, kuyruk yöneticisine belirttik olarak bağlanmak için MQCONN ya da MQCONNX çağrısını yayınlamalıdır.

12. 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 kuyrukta 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 MQCONNX call.

On IBM i for applications running in compatibility mode, the MQCONN call can be omitted, and the following value specified for *HCONN* :

HCDEFH

Varsayılan bağlantı tanıtıcısı.

Önemli:  Uyumluluk kipini kullanmaya devam etseniz, uygulamanızı yeniden derlemeyi ve bunun yerine statik bağlı çağrılarını kullanmayı düşünün. Bkz. [IBM üzerindeki RPG uygulamaları için uyumluluk kipi](#)

OBJDSC (MQOD)-giriş/çıkış

Nesne tanımlayıcısı.

Bu yapı, açılacak nesneyi tanımlayan bir yapıdır; ayrıntılar için [“MQOD-Nesne tanımlayıcısı” sayfa 1124](#) ' e bakın.

OBJDSC parametresindeki *ODON* alanı, bir model kuyruğunun adı ise, dinamik bir yerel kuyruk Model kuyruğunun öznelikleri ile 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 çağrılarını için bile geçerlidir. *OBJDSC* parametresindeki model kuyruğunun adı, yaratılan

dinamik kuyruğun adıyla değiştirilir. Dinamik kuyruğun tipi, model kuyruğunun *DefinitionType* özneliğinin değerine göre belirlenir (bkz. “Kuyruklara ilişkin öznitelikler” sayfa 1331). 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:

- OOBROW
- OOINP* (bunlardan yalnızca biri)
- OOINQ
- OOUT
- OOSSET
- OORLQ

Diğer seçenekler, zorunlu olarak belirtilebilir. Birden fazla seçenek gerekliyse, değerler eklenebilir (aynı değişmezi 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 özneliğinin değerine bağlı olarak paylaşılan ya da dışlayıcı bir tiptir; ayrıntılar için “[Kuyruklara ilişkin öznitelikler](#)” sayfa 1331 ' e bakın.

Bu seçenek yalnızca yerel, diğer ad ve model kuyrukları için geçerlidir; uzak kuyruklar, dağıtım listeleri ve kuyruklar olmayan nesneler 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 nesneler 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 nesneler için geçerli değildir.

Aşağıdaki notlar bu seçenekler için geçerlidir:

- Bu seçeneklerden yalnızca biri belirlenebilir.
- *InhibitGet* kuyruk özneliği QAGETI olarak ayarlansa bile, bu seçeneklerden biriyle MQOPER çağrısı başarılı olabilir (öznitelik bu değere ayarlandığında sonraki MQGET çağrıları başarısız olur).
- Kuyruk paylaşılamaz olarak tanımlandıysa (yani, *Shareability* kuyruk özneliğinin 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 çözümleyeceği 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 “MQGMO-Get-message seçenekleri” sayfa 1044 ' ta açıklanan *GMOPT* alanına bakın.

Bu seçenek yalnızca yerel, diğer ad ve model kuyrukları için geçerlidir; uzak kuyruklar, dağıtım listeleri ve kuyruklar olmayan nesneler 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 dizesi.

Kuyruk, sonraki MQPUT çağrılarıyla kullanılmak üzere açılır.

InhibitPut kuyruk özneliği QAPUTİ olarak ayarlansa bile, bu seçenekle birlikte bir MQOP çağrısı başarılı olabilir (öznitelik bu değere ayarlandığında sonraki MQPUT çağrıları 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:

OQBNDO

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 çağrıları, 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* değiş 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 çağrıları 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 yoksayılır.

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 yerleştirecek bir iletiye geçirilebilir. [“MQPMO-put-message seçenekleri” sayfa 1138](#) 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 nesneler 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:

OALU

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, MQOPEN çağrısını başarısız olacak şekilde zorlar.

Bu seçenek, tüm nesne tipleri için geçerlidir.

OORLQ

Açılan yerel kuyruğun adını girin.

Bu seçenek, MQOD yapısındaki ResolvedQName ' in (varsa), açılan yerel kuyruğun adıyla girileceğini belirtir. ResolvedQMgrAdı benzer şekilde, yerel kuyruk bulunduran yerel kuyruk yöneticisinin adıyla da girilir.

Her kuyruk tipi için geçerli MQOPEN seçenekleri

Seçenek	Diğer Ad ("1" sayfa 1295)	Yerel ve Model	Uzak	Yerel Olmayan Küme	Dağıtım listesi	Konu
OOINPQ	✓	✓	-	-	-	-
OOINPS	✓	✓	-	-	-	-
OOINPX	✓	✓	-	-	-	-
OOBRW	✓	✓	-	-	-	-
OOOUT	✓	✓	✓	✓	✓	✓
OOINQ	✓	✓	"2" sayfa 1295	✓	-	-
OOSET	✓	✓	"2" sayfa 1295	-	-	-
OOBNDQ ("3" sayfa 1295)	✓	✓	✓	✓	✓	-
OOBNDN ("3" sayfa 1295)	✓	✓	✓	✓	✓	-
OOBNDQ ("3" sayfa 1295)	✓	✓	✓	✓	✓	-
OOSAVA	✓	✓	-	-	-	-
OOPASI	✓	✓	✓	✓	✓	"5" sayfa 1295
OOPASA	✓	✓	✓	✓	✓	"5" sayfa 1295
OOSETI	✓	✓	✓	✓	✓	"5" sayfa 1295
OOSETA	✓	✓	✓	✓	✓	"5" sayfa 1295
OOALTU	✓	✓	✓	✓	✓	✓
OOFIQ	✓	✓	✓	✓	✓	✓

Seenek	Diğer Ad ("1" sayfa 1295)	Yerel ve Model	Uzak	Yerel Olmayan Küme	Dağıtım listesi	Konu
OORLQ	✓	✓	✓	✓	-	-

Notlar:

1. Diğer adlara ilişkin seçeneklerin geçerliliği, diğer adın özdüğü 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)-ıkış

Nesne tanıtıcısı.

Bu tanıtıcı, nesne için kurulmuş olan erişimi gösterir. Nesne üzerinde işlem yapan sonraki ileti kuyruklama çağrılarında bu değeri belirtilmelidir. MQCLOSE çağrısı yayınlandığında ya da tutama işleminin kapsamını tanımlayan işlem birimi sonlandığında geçerli olmak üzere sona erer.

Tutama kapsamı, en küçük birimle sınırlıdır Uygulamanın alıştığı altyapı tarafından desteklenen koşut işleme; tanıtıcı, MQOPEN çağrısının yayınlandığı koşut işleme birimi dışında geçerli değil:

- IBM üzerinde, tutama kapsamı, aramayı yayınlayan işidir.

CMPCOD (10 basamaklı işaretli tamsayı)-ıkış

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
```

MQPUT-İleti koy

MQPUT çağırısı bir ileti, dağıtım listesi ya da bir konuya ileti koyar. Kuyruk, dağıtım listesi ya da konu zaten açık olmalıdır.

- “Sözdizimi” sayfa 1296
- “Kullanım notları” sayfa 1296
 - “Konular” sayfa 1296
 - “MQPUT ve MQPUT1” sayfa 1297
 - “Hedef kuyruklar” sayfa 1297
 - “Dağıtım listeleri” sayfa 1298
 - “Üst Bilgiler” sayfa 1299
 - “Arabellek” sayfa 1300
- “Parametreler” sayfa 1300
- “RPG Bildirimi” sayfa 1305

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 hiçbirisi teslim edilmezse, MQPUT RC2502 ile dönecektir. Bu durum aşağıdaki durumlarda oluşabilir:
 - Bir ileti, PMSGDLV ya da NPMSGDLV (iletinin kalıcılığına 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 kalıcılığına bağlı olarak) değeri ALLUR ile bir TOPIC 'e yayınlanır ve kalıcı bir abonelik, yayını alamayacak bir kuyruğa sahiptir.

Yayınların bazı abonelere aşağıdaki durumlarda teslim edilememesine rağmen, MQPUT RCT ile geri dönebiliyor:

 - Bir ileti, PMSGDLV ya da NPMSGDLV (iletinin kalıcılığına 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 kalıcılığına 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 saklanması ilişkin bir sorun yoktur.

MQPUT ve MQPUT1

İletileri bir kuyruğa koymak için hem MQPUT hem de MQPUT1 çağrılarını 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 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 hiçbirisi 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*J kullanılarak 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 *NPFAST* değerine ayarlıysa, kalıcı olmayan iletiler, kalıcı iletilerin önüne atlayabilir ve kalıcı iletilerin, kalıcı olmayan iletilere göre korunmamasına neden olur. Ancak, birbiriyle göreceli kalıcı iletilerin sırası ve birbiriyle göreceli kalıcı olmayan iletilerin sırası korunur.

Bu koşullar yerine getirilmezse, ileti grupları ileti sırasını korumak için kullanılabilir; 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 ' deMDMFL 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 1331 ' de açıklanan *DistLists* kuyruğu özniteliğine 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 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 özneliliğine 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, 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 iletileri 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 iletilinde yapıların uzunluklarının toplamı, MQPUT ya da MQPUT1 çağrısında *BUFLen* parametresiyle belirtilen uzunluğa eşit olmalıdır. Bir PCF iletili, 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 değişirgeler 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.

On IBM i for applications running in compatibility mode, the MQCONN call can be omitted, and the following value specified for *HCONN* :

HCDEFH

Varsayılan bağlantı tanıtıcısı.

Önemli:  Uyumluluk kipini kullanmaya devam etseniz, uygulamanızı yeniden derlemeyi ve bunun yerine statik bağlı çağrılar kullanmayı düşünün. Bkz. [IBM üzerindeki RPG uygulamaları için uyumluluk kipi](#)

HOBJ (10 basamaklı işaretli tamsayı)-giriş

Nesne tanıtıcısı.

Bu tanıtıcı, iletinin eklendiği kuyruğu ya da iletinin yayınlandığı konuyu gösterir. The value of *HOBJ* was returned by a previous MQOPEN call that specified the OOOUT option.

MSGDSC (MQMD)-giriş/çıkış

İleti tanımlayıcısı.

Bu yapı, gönderilmekte olan iletinin özniteliklerini tanımlar ve koyma isteği tamamlandıktan sonra iletiyle ilgili bilgileri alır. Ayrıntılar için bkz. [“MQMD-İleti tanımlayıcısı” sayfa 1076](#).

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-İleti tanımlayıcı uzantısı” sayfa 1118](#) .

PMO (MQPMO)-giriş/çıkış

MQPUT işleminin denetlenmesini denetleyen seçenekler.

Ayrıntılar için bkz. [“MQPMO-put-message seçenekleri” sayfa 1138](#).

BUFLEN (10 basamaklı işaretli tamsayı)-giriş

*BUFFER*içindeki iletinin uzunluğu.

Sıfır geçerli ve iletinin herhangi bir uygulama verisi içermediği anlamına gelir. *BUFLEN* için üst sınır çeşitli etmenlere bağlıdır:

- Hedef kuyruk paylaşılan bir kuyruksa, üst sınır 63 KB ' dir (64 512 bayt).
- Hedef yerel bir kuyruksa ya da yerel bir kuyruğa çözüyorsa (ancak, paylaşılan bir kuyruk değilse), 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 MFSEGA 'dır ve bir version-2 MQMD' de ya da version-1 MQMD ile kullanılan bir MQMDE içinde belirtilebilir.

Bu koşulların her ikisi de karşılanırsa, *BUFLEN* , MQMD ' deki *MDOFF* alanının değeri ile 999 999 999 'u geçemez. Koyulabilecek en uzun mantıksal ileti, bu nedenle 999 999 bayttır (*MDOFF* sıfır olduğunda). Ancak, uygulamanın çalıştırıldığı işletim 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 istendiyse 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 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.

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') 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.
- RC2219**
(2219, X'8AB') MQI çağrısı, önceki arama tamamlanmadan önce yeniden girildi.
- RC2241**
(2241, X'8C1') İleti grubu tamamlanmadı.
- RC2242**
(2242, X'8C2') Mantıksal ileti tamamlanmadı.
- RC2245**
(2244, X'8C5') Tutarsız iş birliği-iş belirtimi.
- RC2248**
(2248, X'8C8') İleti tanımlayıcısı uzantısı geçersiz.
- 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.
- RC2258**
(2258, X'8D2') Grup tanıtıcısı geçerli değil.
- 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 MQPUT çağrısı yayınlandı, ancak ileti verileri geçerli olmayan bir MQPH yapısı içeriyor.

RC2479

(2479, X'9AF') Yayın saklanamadı.

RC2480

(2480, X'9B0') Hedef tipi değişti: Diğer ad kuyruğu bir kuyruğa gönderme yapıyor, ancak şimdi bir konuya başvuruyor.

RC2502

(2502, X'9C6') Yayınlama başarısız oldu ve yayın herhangi bir aboneye teslim edilmedi

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      MQPUT(HCONN : HOBJ : MSGDSC : PMO :
C                               BUFLN : 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 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
```

MQPUT1 -Bir ileti girin

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 1305](#)
- [“Kullanım notları” sayfa 1305](#)
- [“Parametreler” sayfa 1306](#)
- [“RPG Bildirimi” sayfa 1311](#)

Sözdizimi

MQPUT1 (HCONN, OBJDSC, MSGDSC, PMO, BUFLN, 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 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, 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.

On IBM i for applications running in compatibility mode, the MQCONN call can be omitted, and the following value specified for *HCONN* :

HCDEFH

Varsayılan bağlantı tanıtıcısı.

Önemli:  Uyumluluk kipini kullanmaya devam etseniz, uygulamanızı yeniden derlemeyi ve bunun yerine statik bağlı çağrılar kullanmayı düşünün. Bkz. [IBM üzerindeki RPG uygulamaları için uyumluluk kipi](#)

OBJDSC (MQOD)-giriş/çıkış

Nesne tanımlayıcısı.

Bu yapı, iletinin eklendiği kuyruğu tanımlayan bir yapıdır. Ayrıntılar için bkz. [“MQOD-Nesne tanımlayıcısı” sayfa 1124.](#)

Kullanıcı, çıkış için kuyruğu açma yetkisine sahip olmalıdır. Kuyruk, bir model kuyruğu **olmamalıdır** .

MSGDSC (MQMD)-giriş/çıkış

İleti tanımlayıcısı.

Bu yapı, gönderilmekte olan iletinin özniteliklerini tanımlar ve koyma isteği tamamlandıktan sonra geri bildirim bilgilerini alır. Ayrıntılar için bkz. [“MQMD-İleti tanımlayıcısı” sayfa 1076.](#)

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-İleti tanımlayıcı uzantısı” sayfa 1118 .](#)

PMO (MQPMO)-giriş/çıkış

MQPUT1işleminin denetlenmesini denetleyen seçenekler.

Ayrıntılar için bkz. [“MQPMO-put-message seçenekleri” sayfa 1138.](#)

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 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: 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 CCCFAIL 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 ayrıştırılamadı.

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

MQSET-Nesne özniteliklerinin ayarlanması

MQSET çağrısı, bir tanıtıcı tarafından temsil edilen bir nesnenin özniteliklerini değiştirmek için kullanılır. Nesne bir kuyruk olmalıdır.

- “Sözdizimi” sayfa 1312
- “Kullanım notları” sayfa 1312
- “Parametreler” sayfa 1312
- “RPG Bildirimi” sayfa 1316

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 dizgileri 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 öznitelikleriyle ilgili bilgi” sayfa 1268 konusuna bakın.
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 kuyruğunsa, açılacak açık için küme kuyruğunun yerel bir yönetim ortamı olmalıdır.

 Nesne özniteliklerine ilişkin ek bilgi için aşağıdaki başlıklara bakın:

- “Kuyruklara ilişkin öznitelikler” sayfa 1331
- “Ad listelerine ilişkin öznitelikler” sayfa 1359
- “Süreç tanımlamalarına ilişkin öznitelikler” sayfa 1360
- “Kuyruk yöneticisine ilişkin öznitelikler” sayfa 1362

Parametreler

MQSET çağrısı aşağıdaki değiştiricileri 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.

On IBM i for applications running in compatibility mode, the MQCONN call can be omitted, and the following value specified for HCONN :

HCDEFH

Varsayılan bağlantı tanıtıcısı.

Önemli:  Uyumluluk kipini kullanmaya devam etmeniz, uygulamanızı yeniden derlemeyi ve bunun yerine statik bağlı çağrılar kullanmayı düşünün. Bkz. [IBM üzerindeki RPG uygulamaları için uyumluluk kipi](#)

HOBJ (10 basamaklı işaretli tamsayı)-giriş

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 işaretli tamsayı)-giriş

Seçicilerin sayısı.

Bu, SELS dizisinde sağlanan seçicilerin sayısıdır. Bu, ayarlanacak 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) ayarlanacak bir değerle tanımlar.

Her seçici, HOBJ ' in temsil ettiği kuyruk türü için geçerli olmalıdır. Yalnızca belirli IA* ve CA* değerlerine izin verilir; bu değerler 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 göreceli 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 dizinin bayt cinsinden uzunluğunu tanımlayan sabit parantez içinde sağlanır.

Çizelge 175. Kuyruklar için MQSET öznitelik seçicileri		
Seçici	Tanım	Not
CATRGD	Tetikleme verileri (LNTRGD).	“2” sayfa 1314
IADIST	Dağıtım listesi desteği.	“1” sayfa 1314

Çizelge 175. Kuyruklar için MQSET öznitelik seçicileri (devamı var)		
Seçici	Tanım	Not
IAIGET	Alma işlemlerine izin verilip verilmediği.	
IAIPUT	Koyma işlemlerine izin verilip verilmediği.	
IATRGC	Tetik kontrolü.	“2” sayfa 1314
ALRIGI	Tetik derinliği.	“2” sayfa 1314
IATRGP	Tetikleyiciler için eşik iletisi önceliği.	“2” sayfa 1314
İLAÇ	Tetikleyici tipi.	“2” sayfa 1314

Notlar:

1. Yalnızca AIX, HP-UX, IBM i, Solaris, Windowsve bu sistemlere bağlı IBM MQ istemcilerinde desteklenir.
2. VSE/ESAüzerinde 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 tamsayı x IACNT)-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 bitleştirilir. 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 çağırısı, 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.

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 kullanılabılır 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..
DMQSET                                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
```

MQSETMP-İleti tanıtıcısı özelliğini ayarla

MQSETMP çağrı kümeleri ya da bir ileti tanıtıcısının bir özelliğini değiştirir.

- [“Sözdizimi” sayfa 1317](#)
- [“Kullanım notları” sayfa 1317](#)
- [“Parametreler” sayfa 1318](#)
- [“RPG Bildirimi” sayfa 1321](#)

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. [“MQBEGIN-İş birimini başlat” sayfa 1216.](#)

- 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 çağrılar 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 [“MQDISC-Bağlantı kesme kuyruğu yöneticisi” sayfa 1253](#) içindeki kullanım notlarına bakın.
- 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 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 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.

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 gerileme çağrısını hiçbir zaman yayınlamayan, uzun süredir çalışan bir uygulama, diğer uygulamaların kullanımına sunulmayan iletilerle kuyrukları doldurabilir. Bu olasılığa karşı koruma sağlamak için, yöneticinin *MaxUncommittedMsgs* queue-manager öznelikliğini, kuyrukları dolduran kaçak uygulamaları önleyecek kadar düşük bir değere ayarlaması gerekir, ancak beklenen ileti sistemi uygulamalarının doğru şekilde çalışmasına izin verecek kadar yüksek.

Parametreler

MQSETMP çağrısında şu değiştirgeler vardır:

HCONN (10 basamaklı işaretli tamsayı)-giriş

Bu tanıtıcı, kuyruk yöneticisiyle olan bağlantıyı gösterir.

Değerin, *HMSG* parametresinde belirtilen ileti tanıtıcısını yaratmak için kullanılan bağlantı tanıtıcısı ile eşleşmesi gerekir.

İleti tanıtıcısı HCUNAS kullanılarak yaratıldıysa, ileti tutamacının bir özelliği ayarlanırken iş parçacısında geçerli bir bağlantı oluşturulmalıdır; tersi durumda, çağrı neden kodu RC2009 ile başarısız olur.

HMSG (20 basamaklı işaretli tamsayı)-giriş

Bu, değiştirilecek ileti işidir. Değer önceki bir MQCRTMH çağrısı tarafından döndürüldü.

SETOPT (MQSMPO)-giriş

İleti özelliklerinin nasıl ayarlanacağını denetler.

Bu yapı, uygulamaların ileti özelliklerinin nasıl ayarlanmasını denetleyen seçenekleri belirlemesine olanak sağlar. Yapı, MQSETMP çağrısında bir giriş değiştirgisi. Ek bilgi için MQSMPO başlıklı konuya bakın.

PRNAME (MQCHARV)-giriş

Ayarlanacak özelliğin adı.

Özellik adlarının kullanımına ilişkin ek bilgi için Özellik adları ve Özellik adı kısıtlamaları başlıklı konuya bakın.

PRPDSC (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 bağlamı
- malın akındığı gibi kopyalandığı mesajlar

Bu yapıyla ilgili ek bilgi için MQPD başlıklı konuya bakın.

TYPE (10 basamaklı işaretli tamsayı)-giriş

Ayarlanmakta olan özelliğin veri tipi. Aşağıdakilerden biri olabilir:

TYPEYÜKLEME

Bir boole. *ValueLength* değeri 4 olmalıdır.

TYPBST

Bayt dizilimi. *ValueLength* değeri sıfır ya da daha büyük olmalıdır.

TYPI8

8 bitlik işaretli tamsayı. *ValueLength* değeri 1 olmalıdır.

TYPI16

16 bitlik bir işaretli tamsayı. *ValueLength* değeri 2 olmalıdır.

TYPI32

32 bitlik işaretli tamsayı. *ValueLength* değeri 4 olmalıdır.

TYPI64

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 ayrıştırılamadı.

CMPCOD değeri 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.

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.

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.

RC2482

(2482, X'09B2') Özellik tanımlayıcı yapısı geçerli değil.

RC2442

(2442, X'098A') Özellik adı geçersiz.

RC2473

(2473, X'09A9') Özellik veri tipi geçersiz.

RC2472

(2472, X'09A8') Değer verilerinde sayı biçimi hatası saptandı.

RC2463

(2463, X'099F') İleti özelliği seçenekleri yapısını ayarlama geçerli değil.

RC2111

(2111, X'083F') Özellik adı kodlanmış karakter takımı tanıtıcısı geçerli 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 1388](#).

RPG Bildirimi

```
C*..1.....2.....3.....4.....5.....6.....7..  
C                                CALLP      MQSETMP(HCONN : HMSG : SETOPT :  
                                PRNAME : PRPDSC :  
                                TYPE : VALLEN : VALUE :  
                                CMPCOD : REASON)
```

Çağrıya 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
```

MQSTAT-Durum bilgilerini al

Durum bilgilerini almak için MQSTAT çağrısını kullanın. Döndürülen durum bilgisi tipi, çağrıda belirtilen STYPE değeri tarafından belirlenir.

- [“Sözdizimi” sayfa 1321](#)
- [“Kullanım notları” sayfa 1321](#)
- [“Parametreler” sayfa 1322](#)
- [“RPG Bildirimi” sayfa 1323](#)

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 olduğunda) bir uyarı ya da başarısız olduğunda, bu sayaçlar artırılır (not, dağıtım listesi başına bir kez dağıtım listesi sayım işlemi için bir kez olduğunda).
4. MQSTAT 'a başarılı bir çağrı, önceki hata bilgilerinde ya da sıfırlanmakta olan sayılarla sonuçlanıyor.

Parametreler

MQSTAT çağrısında aşağıdaki değişirgeler vardır:

Hconn (MQHCONN)-giriş

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)-giriş

İstenilen durum bilgilerinin tipi. Geçerli tek değer şunlardır:

STATAPT

Önceki zamanuyumsuz put işlemleriyle ilgili bilgi döndürür.

STS (MQSTS)-giriş/çıkış

Durum bilgisi yapısı. Ayrıntılar için bkz. [“MQSTS-Durum raporlama yapısı” sayfa 1193.](#)

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 işlem geçerli ortamda yok.

Bu kodlarla ilgili ayrıntılı bilgi için aşağıdaki başlara bakın:

- [İletiler ve neden kodları](#)

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
```

MQSUB-Aboneliği Kaydet

MQSUB çağrısı, uygulama aboneliğini belirli bir konuya kaydeder.

- [“Sözdizimi” sayfa 1323](#)
- [“Kullanım notları” sayfa 1323](#)
- [“Parametreler” sayfa 1325](#)
- [“RPG Bildirimi” sayfa 1329](#)

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 dizgelerinin kullanılması” sayfa 560](#)’inde açıklandığı şekilde, iki kısımin bitişirmesiyle oluşturulmuş bir konuya eklenir.
- 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, çağrıya kısa bir ad verilmesiyle ya da uzun bir

ad sağlanırsa, konu sıradüzenindeki en yakın kısa ad nesnesine yeridir. Bu konu nesnesinde, abone olma yetkisi, çıkış için yetki denetimi ve hedef kuyruğunda, çıkış için yetki tanımlandığından emin olmak üzere yapılır. SDMAN seçeneği kullanılırsa, bu, bu konu nesnesiyle ilişkili yönetilen kuyruk adında bir yetki denetimi yapıldığında ve yönetilmeyen bir kuyruk sağlandıysa, bu, *HOBJ* parametresiyle gösterilen kuyruğun üzerinde bir yetki denetimi yapıp yapılmadığı anlamına gelir.

- SOMAN seçeneği kullanıldığında, MQSUB çağrısında döndürülen *HOBJ* , Backout eşiği ve Excessive backout requirue adı gibi öznelilikleri bulmak için sorgu ile sorgulanabilir. Yönetilen kuyruğun adını da sorgulayabilirsiniz, ancak bu kuyruğu doğrudan açma girişiminde bulunmamalısınız.
- Abonelikler, grubun birden fazla yayınının yayınla eşleştigi durumlarda bile yalnızca tek bir yayının abonelikleri grubuna teslim edilmesine olanak tanıyabilir. Abonelikler SOGRP seçeneği kullanılarak gruplanır ve abonelikleri gruplamak için aşağıdakileri yapmak zorunda oldukları abonelikleri gruplandırılır:
 - MQSUB çağrısındaki *HOBJ* parametresiyle gösterilen aynı kuyruk yöneticisine aynı adlandırılmış kuyruğu (SOMAN seçeneği kullanmıyor) kullanın.
 - aynı *SDCID*' yi paylaş
 - aynı *SDSL* olmalıdır

Bu öznelilikler, grupta yer alınacak olan aboneliklerin kümesini tanımlar ve bir abonelik gruplandırılırsa, değiştirilemeyen öznelilikler de vardır. *SDSL* sonuçlarının RC2512'de değiştirilmesi ve diğer kişilerin (abonelik gruplanmaması durumunda değiştirilebilecek) herhangi birinin değiştirilmesi RC2515' de sonuçlandırılır.

- MQSD ' deki alanlar, SORES seçeneğini kullanan bir MQSUB çağrısından dönmekte tamamlanır. Döndürülen MQSD, SOALT seçeneğini kullanan bir MQSUB çağrısına doğrudan aktarılabilir. Bu çağrı, MQSD ' ye uygulanan abonelik ile ilgili olarak yapmanız gereken değişikliklerle birlikte SOALT seçeneğini kullanır. Bazı alanlarda, çizelgede belirtildiği şekilde dikkat edilmesi gereken noktalar vardır.

MQSUB ' dan MQSD çıkışı	
MQSD ' de alan adı	Özel noktalar
Erişim ya da yaratma seçenekleri	Bu seçeneklerin hiçbiri, MQSUB çağrısından döndürülebileceği ayarlanmaz. Daha sonra bir MQSUB çağrısındaki MQSD ' yi yeniden kullanırsanız, gerek duyduğunuz seçeneğin belirtik olarak ayarlanması gerekir.
Dayanıklılık seçenekleri, Hedef seçenekleri, Kayıt Seçenekleri ve Genel Arama Karakteri seçenekleri	Bu seçenekler uygun şekilde ayarlanacaktır
Yayın seçenekleri	Bu seçenekler, yalnızca SOCRE için geçerli olan SONEWP dışında, uygun şekilde ayarlanacaktır.
Diğer seçenekler	Bu seçenekler, bir MQSUB çağrısından geri dönmeden değiştirilmez. Bunlar, API çağrısının yayınlanmalarını ve aboneliklerle 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>SDRO</i> alanında kullanılan tam konu adı döndürülür.

MQSUB ' dan MQSD çıkışı (devamı var)	
MQSD ' de alan adı	Özel noktalar
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	SORES seçeneğini kullanarak bir MQSUB çağrısından geri dönüş sırasında, bu alan aboneliğin özgün süre bitimi olarak ayarlanacak ve kalan süre bitimi değil. Daha sonra, SOALT seçeneğini kullanarak bir MQSUB çağrısında MQSD ' yi yeniden kullanırsanız, aboneliğin süre bitimini yeniden sayımı başlatacak şekilde ilk durumuna getireceksiniz.
SubName	Bu alan, bir MQSUB çağrısına ilişkin bir giriş alanıdır ve çıkışta değiştirilmez.
SubUserVerileri ve SelectionString	Bu değişken uzunluk alanları, bir arabellek sağlanırsa, SORES seçeneği kullanılarak bir MQSUB çağrısından çıkışa döndürülecek ve VCHRPiçinde de artı bir arabellek uzunluğuna neden olur. Hiçbir arabellek sağlanmıyorsa, sağlanan arabellek MQCHARV.If alanının VCHRL alanında döndürülecek. Bu alan, alanı döndürmek için gereken alandan küçükse, sağlanan arabellete yalnızca VCHRP byte ' ı döndürülür. Daha sonra SOALT seçeneğini kullanarak bir MQSUB çağrısında MQSD ' yi yeniden kullanırsanız ve arabellek sağlanmaz; ancak sıfır dışında bir VCHRL sağlanırsa, bu uzunluk alanın var olan uzunluğuna uygun olursa, bu alanda bir değişiklik yapılmayacaktır.
SubCorrelTanıtıcısı ve PubAccountingSimgesi	SOSCID ' yi kullanmayacaksanız, kuyruk yöneticisi SDCID tarafından oluşturulur. SOSETI kullanmayacaksanız, kuyruk yöneticisi SDACC tarafından oluşturulur. Bu alanlar, SORES seçeneğini kullanarak bir MQSUB çağrısından MQSD ' ye döndürülecektir. Kuyruk yöneticisi tarafından oluşturulduysa, SOCRE ya da SOALT seçeneğini kullanarak, oluşturulan değer MQSUB çağrısında döndürülecek.
PubPriority, SubLevel & PubApplIdentityData	Bu alanlar MQSD ' de döndürülecektir.
ResObjectDizgisi	Bir arabellek sağlanırsa, bu çıkış MQSD ' de yalnızca bu alan döndürülür.

Parametreler

MQSUB çağrısında şu değiştirgeler vardır:

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.

On IBM i for applications running in compatibility mode, the MQCONN call can be omitted, and the following value specified for *HCONN* :

HCDEFH

Varsayılan bağlantı tanıtıcısı.

Önemli:  Uyumluluk kipini kullanmaya devam etseniz, uygulamanızı yeniden derlemeyi ve bunun yerine statik bağlı çağrılar kullanmayı düşünün. Bkz. [IBM üzerindeki RPG uygulamaları için uyumluluk kipi](#)

SUBDSC (MQSD)-giriş/çıkış

Bu yapı, uygulama tarafından kaydedilmekte olan nesneyi kullanan bir yapıdır. Ek bilgi için “MQSD-Abonelik tanımlayıcısı” sayfa 1176 başlıklı konuya bakın.

HOBJ (10 basamaklı işaretli tamsayı)-giriş/çıkış

Bu tanıtıcı, bu aboneliğe gönderilen iletileri almak için kurulmuş olan erişimi gösterir. Bu iletiler belirli bir kuyrukta saklanabilir ya da kuyruk yöneticisinde, belirli bir kuyruğa gerek duymadan depolanmalarını yönetmeleri istenebilir.

Nesne tanıtıcısı.

Belirli bir kuyruk kullanılacaksa, bu kuyruğun yaratılma zamanındaki abonelik ile ilişkilendirilmesi gerekir. Bu işlem iki şekilde yapılabilir:

- Bu tanıtıcı değeri, SDCRT seçeneğiyle MQSUB çağrılırken sağlanır. Çağrıya bir giriş parametresi olarak bu tanıtıcı sağlanırsa, en az bir OOINP*, OOUT (örneğin, uzak bir kuyruk) ya da OOBW seçeneği kullanılarak, kuyruğun önceki bir MQOUT çağrısından döndürülen geçerli bir nesne tanıtıcısı olmalıdır. Böyle bir durum yoksa, arama RC2019 ile başarısız olur. Bir konu nesnesine çözülen bir diğer ad kuyruğunda nesne tanıtıcısı olamaz. Böyle bir durumda, arama RC2019 ile başarısız olur
- DEFINE SUB MQSC komutunu kullanarak ve bu komutu bir kuyruk nesnesinin adıyla birlikte sağlamış olabilir.

Kuyruk yöneticisi, bu aboneliğe gönderilen iletilerin depolanmasını yönetiyorsa, SOMAN seçeneğini kullanarak ve parametre değerini HONONE olarak ayarlayarak, abonelik yaratıldığında bunu belirtmeniz gerekir. Kuyruk yöneticisi, çağrıdaki çıkış değiştirgesi olarak tanıtıcı değerini döndürür ve döndürülen tanıtıcı, yönetilen tanıtıcı olarak bilinir. Honone belirtilirse ve SOMAN da belirtilmediyse, arama RC2019 ile başarısız olur.

Kuyruk yöneticisi tarafından döndürülen, bir MQGET ya da MQCB çağrısında, bir MQINQ çağrısında ya da MQCLOSE üzerinde, bir MQGET ya da MQCB çağrısında kullanılabilecek bir yönetilen tanıtıcı kullanılabilir. MQPUT, MQSET ya da sonraki bir MQSUB üzerinde kullanılamaz; bu nedenle, MQPUT için RC2039 , MQSET için RC2040 ya da MQSUB için RC2038 ile başarısız olma girişiminde bulunamaz.

Bu aboneliği sürdürmek için, MQSD yapısındaki OPTS alanındaki SORES seçeneği kullanılırsa, HONONE belirtilirse, tanıtıcı bu parametredeki uygulamaya geri döndürülebilir. Bu seçeneği, aboneliğin yönetilen bir tanıtıcı kullanıp kullanmadığını kullanabilirsiniz. DEFINE SUB komutunda tanımlanan abonelik kuyruğuna ilişkin tanıtıcı değeri istiyorsanız, DEFINE SUB kullanılarak yaratılan abonelikler için kullanışlı olabilir. Yönetimsel olarak bir aboneliğin oluşturulması durumunda, kuyruğun OOINPQ ve OOBW ile açıldığı durumlarda. Diğer seçenekler gerekiyorsa, uygulama, abonelik kuyruğunu açık bir şekilde açmalı ve çağrıdaki nesne tanıtıcısını sağlamalıdır. Kuyruğun açılması sırasında bir sorun varsa, çağrı RC2522 ile başarısız olur. HOBJ sağlandıysa, özgün MQSUB çağrısındaki HOBJ ile eşdeğer olmalıdır. This means if an object handle returned from an MQOPEN call is being provided, the handle must be to the same queue as previously used or the call fails with RC2019.

Bu abonelik değiştiriliyorsa, MQSD yapısındaki OPTS alanında SOALT seçeneğini kullanarak, farklı bir HOBJ sağlanabilir. Önceden bu parametre aracılığıyla tanıtılan kuyruğa teslim edilmiş olan yayınlar o

kuyruğun üzerinde kalır ve *HOBJ* parametresi artık farklı bir kuyruğu gösteriyorsa, bu iletileri almak için uygulamanın sorumluluğundadır.

Bu parametrenin çeşitli abonelik seçenekleri ile kullanımı aşağıdaki tabloda özetlenmiştir:

Seçenekler	Hobj	Tanım
SOCRT + SOMAN	Girişte yoksayıldı	Kuyruk yöneticisi tarafından yönetilen iletilerin deposuyla bir abonelik yaratır.
SOCRT	Geçerli nesne tanıtıcısı	İletiler için hedef olarak belirli bir kuyruk sağlayan bir abonelik oluşturur.
SORES	HONONE	Önceden yaratılmış bir aboneliği sürdürür (yönetilen ya da değil) ve kuyruk yöneticisinin, uygulama tarafından kullanılmak üzere nesne tanıtıcısını döndürmesini sağlar.
SORES	Geçerli, eşleştirme, nesne tanıtıcısı	İletilere ilişkin hedef olarak belirli bir kuyruğu kullanan, önceden yaratılmış bir aboneliği sürdürür ve belirli açık seçeneklerle bir nesne tanıtıcısı kullanır.
SOALT + SOMAN	HONONE	Daha önce belirli bir kuyruk kullanan var olan bir aboneliği, şimdi yönetilecek olan var olan bir abonelik ile değiştirir.
SOALT	Geçerli nesne tanıtıcısı	Belirli bir kuyruğu kullanmak için var olan bir aboneliği değiştirir (yönetimden ya da farklı bir kuyruktan).

Whether it was provided or returned, *HOBJ* must be specified on subsequent MQGET calls that you need to receive the publications.

The *HOBJ* handle ceases to be valid when the MQCLOSE call is issued on it, or when the unit of processing that defines the scope of the handle terminates. Döndürülen nesne tanıtıcısı kapsamı, çağrıda belirtilen bağlantı tanıtıcısı ile aynı. Kapsam kapsamına ilişkin bilgi için [HCONN](#) başlıklı konuya bakın. An MQCLOSE of the *HOBJ* handle has no effect on the *HSUB* handle.

HSUB (10 basamaklı işaretli tamsayı)-çıkış

Bu tanıtıcı, yapılan aboneliği temsil eder. Diğer iki işlem için de kullanılabilir:

- Sonraki bir MQSUBRQ çağrısında, abonelik yapılırken SOPUBR 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. An MQCLOSE of the *HSUB* handle has no effect on the *HOBJ* handle.

Bu tanıtıcı bir MQGET ya da MQCB çağrısına geçirilemez. *HOBJ* parametresini kullanmanız gerekir. Bu tanıtıcı, RC2019 içindeki diğer herhangi bir IBM MQ çağrısı sonuçlarıyla geçirmektedir.

CMPCOD (10 basamaklı işaretli tamsayı)-çıkış

Tamamlanma kodu; bu kod aşağıdakilerden biridir:

CCOK

Başarıyla tamamlandı

CCWARN

Uyarı (kısmi tamamlama)

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:

RC2019

(2019 X'07E3') Nesne tanıtıcısı geçerli değil

RC2046

(2046 X'07FE') Seçenekler geçerli değil ya da tutarlı değil

RC2085

(2085 X'0825 ') Nesne saptanamayan 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 aboneliklerle eşleşiyor

RC2440

(2440 X'0988 ') SubName alanı geçerli değil

RC2441

(2441 X'0989 ') Objectstring alanı geçerli değil

RC2435

(2435 X'0983 ') Öznitelik SDALT kullanılarak değiştirilemez ya da SDIMM' le abonelik yaratıldı.

RC2436

(2436 X'0984 ') SODUR seçeneği geçerli değil

RC2459

(2459, X'99B') Seçim dizgisi sözdizimi hatası.

RC2503

(2503 X'09C7') Abone olunan konular için MQSUB çağrılar engellenmektedir.

RC2519

(2519, X'9D7') Bu seçim dizgisi, MQCHARV yapısının nasıl kullanılacağını tanımlamaya ilişkin bir değer olarak belirlenmez.

RC2551

(2551, X'9F7') Belirtilen seçim dizgisi kullanılamıyor.

RPG Bildirimi

```
C*..1.....2.....3.....4.....5.....6.....7..  
C                                CALLP      MQSUB(HCONN : SUBDSC : HOBJ :  
C                                HSUB : CMPCOD : REASON)
```

Çağrıya ilişkin prototip tanımı:

```
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
```

MQSUBRQ-Abonelik İsteği

MQSUBRQ çağrısı bir abonelik ile ilgili istekte bulundur.

- “Sözdizimi” sayfa 1329
- “Kullanım notları” sayfa 1329
- “Parametreler” sayfa 1330
- “RPG Bildirimi” sayfa 1331

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 HOBJ 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, SBROPT 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 çağrışı 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 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 değişirgeler vardır:

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.

On z/OS for CICS applications, and on IBM i for applications running in compatibility mode, the MQCONN call can be omitted, and the following value specified for *HCONN* :

HCDEFH

Varsayılan bağlantı tanıtıcısı.

Önemli:  Uyumluluk kipini kullanmaya devam etseniz, uygulamanızı yeniden derlemeyi ve bunun yerine statik bağlı çağrıları kullanmayı düşünün. Bkz. [IBM üzerindeki RPG uygulamaları için uyumluluk kipi](#)

HSUB (10 basamaklı işaretli tamsayı)-giriş

Bu tanıtıcı, bir güncellemenin isteneceği aboneliği temsil eder. *HSUB* değeri önceki bir MQSUB çağrısından döndürüldü.

İŞLEM (10 basamaklı işaretli tamsayı)-giriş

Bu değıştirge, abonelikte istenmekte olan işlemi denetler. Aşğıdakilerden biri (ve yalnızca bir tanesi) belirtilmelidir:

SSRAPUB

Bu işlem, belirtilen konu için bir güncelleme yayınının gönderilmesini ister. Bu durum, abonenin, aboneliği 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 özelliği 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 isteği seçenekleri” sayfa 569](#) ' e bakın.

CMPCOD (10 basamaklı işaretli tamsayı)-çıkış

Tamamlanma kodu; bu kod aşğıdakilerden biridir:

CCOK

Başarıyla tamamlandı

CCWARN

Uyarı (kısmi tamamlama)

CCCFAIL

Arama başarısız oldu

Neden (10 basamaklı işaretli tamsayı)-çıkış

Neden kodu niteleyici *CMPCOD*.

CPMCOD CCOK ise:

YOK

(0, X'000 ') Raporlamak için bir neden yok.

CPMCOD 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 değil.

RPG Bildirimi

```
C*..1.....2.....3.....4.....5.....6.....7..  
C                                CALLP      MQSUBRQ(HCONN : HSUB : ACTION :  
C                                SBROPT : CMPCOD : REASON)
```

Çağrıya ilişkin prototip tanımı:

```
D*..1.....2.....3.....4.....5.....6.....7..  
DMQSUBRQ      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
```

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 paylaşımı 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 kuyruktaki 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 diğer adı

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 tipleri, [Çizelge 176 sayfa 1333](#) ve sonraki tablolardaki ✓ simgesi ile gösterilir.

[Çizelge 176 sayfa 1333](#) , kuyruklara özgü öznitelikleri özetler. Öznitelikler alfabetik sırayla açıklanır.

Not: Bu kısımda gösterilen özniteliklerin adları, MQINQ ve MQSET çağrıları ile kullanılan adlardır. Öznitelikleri tanımlamak, değiştirmek ya da görüntülemek için MQSC komutları kullanıldığında, diğer kısa adlar kullanılır; ayrıntılar için [Script \(MQSC\) Commands](#) (Komut Dosyası (MQSC) Komutları) konusuna bakın.

Çizelge 176. Kuyruklara ilişkin öznitelikler. Kolonlar 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.

Öznitelik	Tanım	Yerel	Model	Diğer Ad	Uzak	Küme
AlterationDate	Tanımlamanın son olarak değiştirildiği tarih	✓		✓	✓	
AlterationTime	Tanımlamanın son olarak değiştirildiği saat	✓		✓	✓	
BackoutRequeueQName	Aşırı geriletme yeniden kuyruğa alma kuyruğu adı	✓	✓			
BackoutThreshold	Geriletme eşiği	✓	✓			
BaseQName	Diğer adın çözümleyicileceği kuyruk adı			✓		
ClusterChannelAd	Küme-gönderen kanal adı	✓	✓			
ClusterName	Kuyruğun ait olduğu kümenin adı	✓		✓	✓	
ClusterNamelist	Kuyruğun ait olduğu kümelerin adlarını içeren ad listesi nesnesinin adı	✓		✓	✓	
CreationDate	Kuyruğun yaratıldığı tarih	✓				
CreationTime	Kuyruğun yaratıldığı saat	✓				
CurrentQDepth	Yürürlükteki kuyruk derinliği	✓				
DefBind	Varsayılan bağ tanımı	✓		✓	✓	✓
DefinitionType	Kuyruk tanımlaması tipi	✓	✓			
DefInputOpenOption	Varsayılan giriş açma seçeneği	✓	✓			
DefPersistence	Varsayılan ileti kalıcılığı	✓	✓	✓	✓	✓
DefPriority	Varsayılan ileti önceliği	✓	✓	✓	✓	✓

Çizelge 176. Kuyruklara ilişkin öznitelikler. Kolonlar 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.

(devamı var)

Öznitelik	Tanım	Yerel	Model	Diğer Ad	Uzak	Küme
DistLists	Dağıtım listesi desteği	✓	✓			
HardenGetGeri çekişi	Doğru bir geriletme sayısının sağlanıp korunmayacağı	✓	✓			
InhibitGet	Kuyruğa ilişkin alma işlemlerinin izin verilip verilmediğini denetler	✓	✓	✓		
InhibitPut	Kuyruğa ilişkin koyma işlemlerine izin verilip verilmediğini denetler	✓	✓	✓	✓	✓
InitiationQName	Başlatma kuyruğunun adı	✓	✓			
MaxMsgUzunluğu	İleti uzunluğu üst sınırı (byte)	✓	✓			
MaxQDepth	Kuyruk derinliği üst sınırı	✓	✓			
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	✓	✓			
OpenInputSayı	Giriş için açma sayısı	✓				
OpenOutputSayısı	Çıkış için açma sayısı	✓				
ProcessName	İşlem adı	✓	✓			
QDepthHighOlayı	Kuyruk Derinliği Yüksek olaylarının oluşturulup oluşturulmayacağını denetler	✓	✓			
QDepthHighSınırı	Kuyruk derinliği üst sınırı	✓	✓			

Çizelge 176. Kuyruklara ilişkin öznitelikler. Kolonlar 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.

(devamı var)

Öznitelik	Tanım	Yerel	Model	Diğer Ad	Uzak	Küme
QDepthLowOlayı	Kuyruk Derinliği Düşük olaylarının oluşturulup oluşturulmayacağını denetler	✓	✓			
QDepthLowSınırı	Kuyruk derinliği için alt sınır	✓	✓			
QDepthMaxOlayı	Kuyruk Dolu olaylarının oluşturulup oluşturulmayacağını denetler	✓	✓			
QDesc	Kuyruk tanımlaması	✓	✓	✓	✓	✓
QName	Kuyruk adı	✓		✓	✓	✓
QServiceInterval	Kuyruk hizmet aralığı hedefi	✓	✓			
QServiceIntervalOlayı	Hizmet Aralığı Yüksek ya da Hizmet Aralığı Tamam olaylarının oluşturulup oluşturulmayacağını denetler	✓	✓			
QType	Kuyruk tipi	✓		✓	✓	✓
RemoteQMgrAd	Uzak kuyruk yöneticisinin adı				✓	
RemoteQName	Uzak kuyruğun adı				✓	
RetentionInterval	Alıkoyma aralığı	✓	✓			
Kapsam	Kuyruğun bir girişinin bir hücre dizininde de olup olmadığını denetler	✓		✓	✓	
Paylaşılabilirlik	Kuyruk paylaşılabilirliği	✓	✓			
TriggerControl	Tetikleyici denetimi	✓	✓			

Çizelge 176. Kuyruklara ilişkin öznitelikler. Kolonlar 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.

(devamı var)

Öznitelik	Tanım	Yerel	Model	Diğer Ad	Uzak	Küme
TriggerData	Verileri tetikle	✓	✓			
TriggerDepth	Tetikleyici derinliği	✓	✓			
TriggerMsgÖnceliği	Tetikleyiciler için eşik iletisi önceliği	✓	✓			
TriggerType	Tetikleme Tipi	✓	✓			
Kullanım	Kuyruk kullanımı	✓	✓			
XmitQName	İletim kuyruğu adı				✓	

AlterationDate (12 baytlık karakter dizgisi)

Tanımlamanın en son değiştirildiği tarih.

Yerel	Model	Diğer Ad	Uzak	Küme
✓		✓	✓	

Bu, tanımın en son değiştirildiği tarihtir. Tarihin biçimi şöyledir: YYYY-MM-DD, uzunluğun 12 baytı (örneğin, 1992-09-23↵↵, burada ↵↵ iki boş karakteri temsil eder) içeren iki boşlukla doldurulabilir.

Kuyruk yöneticisi olarak, belirli özniteliklerin değerleri (örneğin, *CurrentQDepth*) değişir. Bu özniteliklerde yapılan değişiklikler *AlterationDate* işlemini etkilemez.

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.

Yerel	Model	Diğer Ad	Uzak	Küme
✓		✓	✓	

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 'dır. 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.

BackoutRequeueQName (48 byte 'lık karakter dizgisi)

Gereğinden fazla geriletme yeniden kuyruğa alma kuyruğu adı.

Yerel	Model	Diğer Ad	Uzak	Küme
✓	✓			

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 özniteliğinin değerini temel olarak bir işlem gerçekleştirmez.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla CABRQN seçicisini kullanın. Bu özniteliğin uzunluğu LNQN tarafından verilir.

BackoutThreshold (10 basamaklı işaretli tamsayı)

Geriletme eşiği.

Yerel	Model	Diğer Ad	Uzak	Küme
✓	✓			

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 özniteliğinin değerini temel olarak bir işlem gerçekleştirmez.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IABTHR seçiciyi kullanın.

BaseQName (48 byte 'lık karakter dizgisi)

Diğer adın çözümleyicilerin bulunduğu kuyruk adı.

Yerel	Model	Diğer Ad	Uzak	Küme
		✓		

Bu ad, yerel kuyruk yöneticisinde tanımlı olan bir kuyruğun adıdır. (Kuyruk adlarıyla ilgili daha fazla bilgi için, MQOD 'daki ODOM 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.

BaseType (tamsayı değiştirgesi yapısı)

Diğer adın çözdüğü nesne tipi.

Yerel	Model	Diğer Ad	Uzak	Küme
		✓		

Bu öznitelik aşağıdaki değerlerden birine sahip olabilir:

OTQ

Temel nesne tipi bir kuyruktur

OTTOP

Temel nesne tipi bir konudur

CFStrucName (12 baytlık karakter dizilimi)

Bağlaşım-tesis yapısı adı.

Yerel	Model	Diğer Ad	Uzak	Küme
✓	✓			

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 yoksayılır.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla CACFSN seçiciyi kullanın. Bu özneliğin uzunluğu LNCFSN tarafından verilir.

Bu öznitelik yalnızca z/OSüzerinde desteklenir.

ClusterChannelAd (20 baytlık karakter dizgisi)

ClusterChannelName , bu kuyruğu bir iletim kuyruğu olarak kullanan, kümenin gönderici kanallarının genel adıdır. Öznitelik, hangi küme gönderen kanallarının bu küme iletim kuyruğundan bir küme alıcı kanalına ileti gönderdiğini belirtir.

Yerel	Model	Diğer Ad	Uzak	Küme
✓	✓			

Varsayılan kuyruk yöneticisi yapılandırması, tek bir iletim kuyruğundan ileti göndermek için tüm küme gönderici kanallarının *SYSTEM.CLUSTER.TRANSMIT.QUEUE* 'dir. Varsayılan yapılanış, kuyruk yöneticisi özneliği (**DefClusterXmitQueueType**) değiştirilerek değiştirilebilir. Özneliğin varsayılan değeri *SCTQ* 'dur. Değeri *KANAL* olarak değiştirebilirsiniz. **DefClusterXmitQueueType** özneliğini *KANAL* olarak ayarlıyorsanız, her bir küme gönderen kanalı varsayılan olarak belirli bir küme iletim kuyruğunu kullanacak şekilde varsayılan değer olarak *SYSTEM.CLUSTER.TRANSMIT.ChannelName* kullanılır.

Ayrıca, iletim kuyruğu özneliği *ClusterChannelName* özneliğini bir küme gönderen kanalına el ile de ayarlayabilirsiniz. Küme gönderen kanalının bağlı olduğu kuyruk yöneticisine gönderilen iletiler, küme gönderici kanalını tanımlayan iletim kuyruğunda saklanır. Bunlar varsayılan küme iletim kuyruğunda saklanmaz. *ClusterChannelName* özneliğini boşluklara ayarlıyorsanız, kanal yeniden başlatıldığında, kanal varsayılan küme iletim kuyruğuna geçer. Varsayılan kuyruk, kuyruk yöneticisi *DefClusterXmitQueueType* özneliğinin değerine bağlı olarak *SYSTEM.CLUSTER.TRANSMIT.ChannelName* ya da *SYSTEM.CLUSTER.TRANSMIT.QUEUE* olur.

Yıldız işaretlerini ("*" , **ClusterChannelName**) belirterek, bir iletim kuyruğunu bir küme gönderici kanalları kümesiyle ilişkilendirebilirsiniz. Yıldız imleri, kanal adı dizgisinin başında, başında ya da

ortasındaki herhangi bir sayıda yere olabilir. **ClusterChannelName** , 20 karakterden oluşan bir uzunlula sınırlıdır: MQ_CHANNEL_NAME_LENGTH.

ClusterName (48 baytlık karakter dizgisi)

Kuyruğun ait olduğu kümenin adı.

Yerel	Model	Diğer Ad	Uzak	Küme
✓		✓	✓	

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.

ClusterNameList (48 byte 'lık karakter dizgisi)

Kuyruğun ait olduğu kümelerin adlarını içeren ad listesi nesnesinin adı.

Yerel	Model	Diğer Ad	Uzak	Küme
✓		✓	✓	

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 LNNLN tarafından verilir.

CreationDate (12 baytlık karakter dizilimi)

Kuyruk yaratıldığı tarih.

Yerel	Model	Diğer Ad	Uzak	Küme
✓				

Bu, kuyruğun oluşturulduğu tarihtir. Tarihin biçimi şöyledir: YYYY-MM-DD, uzunluğun 12 baytı (örneğin, 1992-09-23-- , burada -- iki boş karakteri temsil eder) içeren iki boşlukla doldurulabilir.

- 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.

CreationTime (8 baytlık karakter dizilimi)

Kuyruğun oluşturulduğu saat.

Yerel	Model	Diğer Ad	Uzak	Küme
✓				

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.

CurrentQDepth (10 basamaklı işaretli tamsayı)

Yürürlükteki kuyruk derinliği.

Yerel	Model	Diğer Ad	Uzak	Küme
✓				

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 alınmamalarına 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-İleti tanımlayıcısı” sayfa 1076’inde 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.

DefBind (10 basamaklı işaretli tamsayı)

Varsayılan bağ tanımı.

Yerel	Model	Diğer Ad	Uzak	Küme
✓		✓	✓	✓

Bu öznitelik, 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 özniteliğin değerini saptamak için, MQINQ çağrısıyla IADBND seçicisini kullanın.

DefinitionType (10 basamaklı imzalı tamsayı)

Kuyruk tanımlaması tipi.

Yerel	Model	Diğer Ad	Uzak	Küme
✓	✓			

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 “[Kuyruk yöneticisine ilişkin öznitelikler](#)” sayfa 1362).

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* özniteliği için QDPERM değerine sahiptir.

Bu kuyruk tipi, MQCLOSE çağrısını kullanarak silinebilir. Daha ayrıntılı bilgi için bkz. “[MQCLOSE- Nesneyi kapat](#)” sayfa 1230 .

Kalıcı dinamik kuyruğa ilişkin *QSGDisp* özniteliğinin 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 alır.

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- Nesneyi kapat](#)” sayfa 1230 .

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 IADEFT seçiciyi kullanın.

DefInputOpenOption (10 basamaklı işaretli tamsayı)

Varsayılan giriş açma seçeneği.

Yerel	Model	Diğer Ad	Uzak	Küme
✓	✓			

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.

DefPersistence (10 basamaklı işaretli tamsayı)

Varsayılan ileti kalıcılığı.

Yerel	Model	Diğer Ad	Uzak	Küme
✓	✓	✓	✓	✓

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 bulunsa bile geçerlidir.

Paylaşılan kuyrukların özel durumunda, kalıcı olmayan iletiler *do* kuyruk-paylaşım grubundaki kuyruk yöneticilerinin yeniden başlatılmasını sağlar, ancak paylaşılan kuyruklarda iletileri saklamak için kullanılan bağlaşım tesisinin başarısızlıklarını atlamaz.

Hem kalıcı hem de kalıcı olmayan iletiler aynı kuyruğun üzerinde var olabilir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IADPER seçiciyi kullanın.

DefPriority (10 basamaklı işaretli tamsayı)

Varsayılan ileti önceliği.

Yerel	Model	Diğer Ad	Uzak	Küme
✓	✓	✓	✓	✓

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 kuyrukta, 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 1331 içinde açıklanan *MsgDeliverySequence* özniteliğine bakın.

Priorities are in the range zero (lowest) through *MaxPriority* (highest); see the *MaxPriority* attribute described in “Kuyruk yöneticisine ilişkin öznitelikler” sayfa 1362.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IADPRI seçiciyi kullanın.

DefReadİleri (10 basamaklı işaretli tamsayı)

İstemciye teslim edilen kalıcı olmayan iletiler için varsayılan okuma tamamlama davranışını belirtir.

Yerel	Model	Diğer Ad	Uzak	Küme
✓	✓	✓		

DefReadİleriye göre aşağıdaki değerlerden birine ayarlanabilir:

RAHNO.

Kalıcı olmayan iletiler, bir uygulama tarafından istekte bulunmadan önce istemciye gönderilmez. İstemci olağandışı şekilde sona ererse, en çok bir kalıcı olmayan ileti kaybedilebilir.

RAHYES

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.

RAHDIS

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 IADRAH seçiciyi kullanın.

DefPResp (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.

Yerel	Model	Diğer Ad	Uzak	Küme
✓	✓	✓	✓	✓

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.

DistLists (10 basamaklı işaretli tamsayı)

Dağıtım listesi desteği.

Yerel	Model	Diğer Ad	Uzak	Küme
✓	✓			

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 iletiyi koyarsa ve bu ileti bu kuyruğa konacaksa, kuyruk yöneticisi dağıtım listesi iletiyi 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.

HardenGetGeri Al (10 basamaklı işaretli tamsayı)

Doğru bir geriletme sayımının sağlanıp korunmayacağı.

Yerel	Model	Diğer Ad	Uzak	Küme
✓	✓			

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.

InhibitGet (10 basamaklı imzalı tamsayı)

Bu kuyruğa ilişkin alma işlemlerine izin verilip verilmediğini denetler.

Yerel	Model	Diğer Ad	Uzak	Küme
✓	✓	✓		

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.

InhibitPut (10 basamaklı imzalı tamsayı)

Bu kuyruğa ilişkin koyma işlemlerine izin verilip verilmediğini denetler.

Yerel	Model	Diğer Ad	Uzak	Küme
✓	✓	✓	✓	✓

Kuyruk-adı 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 engellememektedir.

QAPUTA

Put işlemlerine izin verilir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IAIPUT seçiciyi kullanın. Bu özniteliğin değerini değiştirmek için MQSET çağrısını kullanın.

InitiationQName (48 byte 'lık karakter dizgisi)

Başlatma kuyruğunun adı.

Yerel	Model	Diğer Ad	Uzak	Küme
✓	✓			

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.

MaxMsgUzunluğu (10 basamaklı işaretli tamsayı)

Byte cinsinden ileti uzunluğu üst sınırı.

Yerel	Model	Diğer Ad	Uzak	Küme
✓	✓			

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* 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 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 'ta ileti uzunluğu üst sınırı 100 MB' dir (104 857 600 bayt).

Daha fazla bilgi için, “MQPUT-İleti koy” sayfa 1296’inde açıklanan *BUFLen* parametresine bakın.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IAMLEN seçiciyi kullanın.

MaxQDepth (10 basamaklı işaretli tamsayı)

Kuyruk derinliği üst sınırı.

Yerel	Model	Diğer Ad	Uzak	Küme
✓	✓			

Bu, herhangi bir zamanda kuyrukta bulunabilecek fiziksel ileti sayısı için tanımlanan üst sınırdır.

MaxQDepth iletilerini içeren bir kuyruğa ileti koyma girişi, neden kodu RC2053 ile başarısız olur.

İş biriminin işlenmesi ve iletilerin bölünmesinin 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.

MediaLog (10 basamaklı işaretli tamsayı)

Günlük kapsamının tanıtıcısı (ya da IBM 'i üzerindeki günlük nesnesi) Belirli bir kuyruğun ortam kurtarma işlemi için gereklidir.

Yerel	Model	Diğer Ad	Uzak	Küme
✓	✓			

Dairesel günlük kaydının kullanımda olduğu kuyruk yöneticilerinde, değer boş bir dizgi olarak döndürülür.

MsgDeliverySırası (10 basamaklı işaretli tamsayı)

Mesaj teslim sırası.

Yerel	Model	Diğer Ad	Uzak	Küme
✓	✓			

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.

OpenInputSayı (10 basamaklı işaretli tamsayı)

Giriş için açılan açma sayısı.

Yerel	Model	Diğer Ad	Uzak	Küme
✓				

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, kuyruk için, yerel kuyruk yöneticisinin ait olduğu kuyruk paylaşım grubundaki diğer kuyruk yöneticilerindeki kuyruk için gerçekleştirilen giriş için sayı 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.

OpenOutputSayı (10 basamaklı işaretli tamsayı)

Çıkış için açılan açma sayısı.

Yerel	Model	Diğer Ad	Uzak	Küme
✓				

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, kuyruk için, 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 IAQOC seçicisini kullanın.

ProcessName (48 baytlık karakter dizgisi)

Süreç adı.

Yerel	Model	Diğer Ad	Uzak	Küme
✓	✓			

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.

QDepthHighOlayı (10 basamaklı imzalı tamsayı)

Kuyruk Derinliği Yüksek olaylarının oluşturulup oluşturulmayacağını denetler.

Yerel	Model	Diğer Ad	Uzak	Küme
✓	✓			

Kuyruk Derinliği Yüksek olayı, bir uygulamanın, kuyruktaki ileti sayısının kuyruk derinliği üst eşiğinden 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.

QDepthHighSınırı (10 basamaklı işaretli tamsayı)

Kuyruk derinliği üst sınırı.

Yerel	Model	Diğer Ad	Uzak	Küme
✓	✓			

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 eşiğinden büyük ya da ona eşit olmasına neden olduğunu gösterir. See the *QDepthHighEvent* attribute.

Değer, kuyruk derinliği üst sınırının bir yüzdesi olarak ifade edilir (*MaxQDepth* özniteliği) ve 0 ile 100 aralığında yer alıyor. Varsayılan değer 80 'dir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IAQDHL seçicisini kullanın.

QDepthLowOlayı (10 basamaklı imzalı tamsayı)

Kuyruk Derinliği Düşük olaylarının oluşturulup oluşturulmayacağını denetler.

Yerel	Model	Diğer Ad	Uzak	Küme
✓	✓			

Kuyruk Derinliği Düşük olayı, bir uygulamanın kuyruktaki ileti sayısının, kuyruk derinliği alt eşiğinden küçük ya da ona eşit olmasına neden olan bir kuyruktan ileti alındığını belirtir (*QDepthLowLimit* özniteliğine bakın).

Not: Bu özniteliğin değeri dinamik olarak değişebilir.

QDepthLowOlayı aşağıdaki değerlerden birine sahip olabilir:

EVARDIS

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 IAQDLE seçiciyi kullanın.

QDepthLowSınırı (10 basamaklı işaretli tamsayı)

Kuyruk derinliği için alt sınır.

Yerel	Model	Diğer Ad	Uzak	Küme
✓	✓			

Kuyruk Derinliği Düşük bir olay oluşturmak için kuyruk derinliğinin karşılaştırıldığı eşik budur. Bu olay, bir uygulamanın kuyruktan ileti aldığına ve kuyruğun kuyruk derinliği düşük eşiğinden 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* özniteliği) ve 0 ile 100 aralığında yer alıyor. Varsayılan değer 20'dir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IAQDLL seçicisini kullanın.

QDepthMaxOlayı (10 basamaklı imzalı tamsayı)

Kuyruk Dolu olaylarının oluşturulup oluşturulmayacağını denetler.

Yerel	Model	Diğer Ad	Uzak	Küme
✓	✓			

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:

EVARDIS

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.

QDesc (64 baytlık karakter dizilimi)

Kuyruk tanımlaması.

Yerel	Model	Diğer Ad	Uzak	Küme
✓	✓	✓	✓	✓

Bu alan, tanımlayıcı açıklamalar için kullanılabilecek 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 özneliğiyle 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.

QName (48 byte 'lık karakter dizisi)

Kuyruk adı.

Yerel	Model	Diğer Ad	Uzak	Küme
✓		✓	✓	✓

Bu ad, yerel kuyruk yöneticisinde tanımlı olan bir kuyruğun adıdır. Kuyruk adlarıyla ilgili ek bilgi için [IBM MQ nesnelerinin adlandırılmasına ilişkin kurallar](#) başlıklı konuya bakın. Bir kuyruk yöneticinde tanımlı olan tüm kuyruklar, aynı kuyruk ad alanını paylaşır. Bu nedenle, QTLOC kuyruğu ve QTALS kuyruğu aynı ada sahip olamaz.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla CAQN seçiciyi kullanın. Bu özneliğin uzunluğu LNQN tarafından verilir.

QServiceInterval (10 basamaklı işaretli tamsayı)

Kuyruk hizmet aralığı hedefi.

Yerel	Model	Diğer Ad	Uzak	Küme
✓	✓			

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.

QServiceIntervalOlayı (10 basamaklı imzalı tamsayı)

Hizmet Aralığı Yüksek ya da Hizmet Aralığı Tamam olaylarının oluşturulup oluşturulmayacağını denetler.

Yerel	Model	Diğer Ad	Uzak	Küme
✓	✓			

- 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.

QSGDisp (10 basamaklı işaretli tamsayı)

Kuyruk paylaşımı grubu yok etme.

Yerel	Model	Diğer Ad	Uzak	Küme
✓		✓	✓	

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 bilindiği anlamına gelir; tanımlama, kuyruk paylaşım grubundaki diğer kuyruk yöneticilerine bilinmez.

Kuyruk-paylaşım grubundaki her bir kuyruk yöneticisi için, yürürlükteki nesneyle aynı ada ve tipe sahip bir nesne olması mümkündür, ancak bunlar ayrı nesnelerdir 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, kuyruk-paylaşım grubundaki tüm kuyruk yöneticileri tarafından bilinen nesnenin tek bir somut örneğinin bulunduğu anlamına gelir. Gruptaki bir kuyruk yöneticisi nesneye eriştiğinde, nesneye ilişkin tek paylaşılan eşgörünüme erişir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IAQSGD seçicisini kullanın.

Bu öznitelik yalnızca z/OSüzerinde desteklenir.

QType (10 basamaklı işaretli tamsayı)

Kuyruk tipi.

Yerel	Model	Diğer Ad	Uzak	Küme
✓		✓	✓	✓

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.

RemoteQMgrAd (48 byte 'lık karakter dizgisi)

Uzak kuyruk yöneticisinin adı.

Yerel	Model	Diğer Ad	Uzak	Küme
			✓	

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.

RemoteQName (48 byte 'lık karakter dizgisi)

Uzak kuyruğun adı.

Yerel	Model	Diğer Ad	Uzak	Küme
			✓	

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.

RetentionInterval (10 basamaklı işaretli tamsayı)

Alıkoyma aralığı.

Yerel	Model	Diğer Ad	Uzak	Küme
✓	✓			

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 özneteliğe 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 öznetelik önceden tanımlanmış kuyruklarla da kullanılabilir.

Bu özneteliğin değerini saptamak için, MQINQ çağrısıyla IARINT seçiciyi kullanın.

Kapsam (10 basamaklı işaretli tamsayı)

Bu kuyruğa ilişkin bir girişin bir hücre dizininde de bulunup bulunmadığını denetler.

Yerel	Model	Diğer Ad	Uzak	Küme
✓		✓	✓	

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 kullandığını unutmayın.

Model ve dinamik kuyrukların hücre kapsamı olamaz.

Bu değer yalnızca, bir hücre dizinini destekleyen bir ad hizmeti yapılandırıldıysa geçerlidir.

Bu özneteliğin değerini saptamak için, MQINQ çağrısıyla IASCOPI seçicisini kullanın.

Bu özneteliğe ilişkin destek aşağıdaki kısıtlamalara tabidir:

- IBM üzerinde öznetelik desteklenir, ancak yalnızca SCOQM geçerli olur.

Paylaşılabilirlik (10 basamaklı işaretli tamsayı)

Kuyruğun giriş için paylaşılıp paylaşılamayacağı.

Yerel	Model	Diğer Ad	Uzak	Küme
✓	✓			

Bu, kuyruğun koşutzamanlı 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 özneliğin değerini saptamak için, MQINQ çağrısıyla IASHAR seçiciyi kullanın.

TriggerControl (10 basamaklı işaretli tamsayı)

Tetik kontrolü.

Yerel	Model	Diğer Ad	Uzak	Küme
✓	✓			

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 özneliğin değerini saptamak için, MQINQ çağrısıyla IATRGC seçiciyi kullanın. Bu özneliğin değerini değiştirmek için MQSET çağrısını kullanın.

TriggerData (64 baytlık karakter dizgisi)

Verileri tetikler.

Yerel	Model	Diğer Ad	Uzak	Küme
✓	✓			

Bu, kuyruk yöneticisinin, bu kuyruğa gelen bir ileti bir tetikleme iletilerinin başlatma kuyruğuna yazılmasına neden olduğunda, tetikleme iletilerine 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.

TriggerDepth (10 basamaklı işaretli tamsayı)

Tetik derinliği.

Yerel	Model	Diğer Ad	Uzak	Küme
✓	✓			

Bu, bir tetikleme iletisi 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 öznitelik, başka bir şekilde kullanılmamaktadır.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IAATGD seçiciyi kullanın. Bu özniteliğin değerini değiştirmek için MQSET çağrısını kullanın.

TriggerMsgÖnceliği (10 basamaklı imzalı tamsayı)

Tetikleyiciler için eşik iletisi önceliği.

Yerel	Model	Diğer Ad	Uzak	Küme
✓	✓			

Bu, iletilerin tetikleme iletilerinin oluşturulmasına katkıda bulunmadığı (yani, kuyruk yöneticisi, tetikleme iletilerinin oluşturulup oluşturulmayacağını belirlerken bu iletileri yoksaydığı) ileti önceliğidir. *TriggerMsgPriority* , sıfır (en düşük) ile *MaxPriority* (en düşük) arasında olabilir (highest; see “Kuyruk yöneticisine ilişkin öznitelikler” sayfa 1362); sıfır değeri tüm iletilerin, tetikleyici iletileri oluşturulmasına katkıda bulunmasına neden olur.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IAATGP seçiciyi kullanın. Bu özniteliğin değerini değiştirmek için MQSET çağrısını kullanın.

TriggerType (10 basamaklı işaretli tamsayı)

Tetikleyici tipi.

Yerel	Model	Diğer Ad	Uzak	Küme
✓	✓			

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 özniteliğin değerini saptamak için, MQINQ çağrısıyla IAATRGT seçiciyi kullanın. Bu özniteliğin değerini değiştirmek için MQSET çağrısını kullanın.

Kullanım (10 basamaklı işaretli tamsayı)

Kuyruk kullanımı.

Yerel	Model	Diğer Ad	Uzak	Küme
✓	✓			

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 aracılığıyla 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. “MQXQH-İletim kuyruğu üstbilgisi” sayfa 1208) 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.

XmitQName (48 byte 'lık karakter dizisi)

İletim kuyruğu adı.

Yerel	Model	Diğer Ad	Uzak	Küme
			✓	

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 çağrılarını 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 kullanılabilecek 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 nesnelerinin 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 nesnelerini adlandırmabaşlıklı](#) konuya bakın.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla CANAMS seçiciyi kullanın.

Listedeki her bir adın uzunluğu LNOBJN tarafından verilir.

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.

ApplId (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 *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.

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.

ApplType (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.

ApplType 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 kullanılabilecek 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 nesnelerinin 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*içeren) oluşur, ardından bir boşluk, ardından *EnvData* sonunda boşluklar kaldırılmış olarak bulunur.

Karakter dizgisi boş boş (nulls) içeremez. Gerekirse boşluklarla doldurulabilir.

Bu özneliğin değerini saptamak için, MQINQ çağrısıyla CAUSRD seçicisini kullanın. Bu özneliğin uzunluğu LNPROU tarafından verilir.

Kuyruk yöneticisine ilişkin öznelikler

Kuyruk yöneticisi özneliklerinin bir özeti.

Bazı kuyruk yöneticisi öznelikleri belirli uygulamalar için sabitlenirken, diğer kullanıcılar da ALTER QMGRMQSC komutu kullanılarak değiştirilebilir. Öznelikler, DISPLAY QMGRkomutu kullanılarak da görüntülenebilir. Çoğu kuyruk yöneticisi öznelikleri özel bir OTQM nesnesi açarak ve döndürülen tanıtıcı ile MQINQ çağrısını kullanarak sorgulanabilir.

Aşağıdaki çizelge, kuyruk yöneticisine özgü öznelikleri özetlemektedir. Öznelikler alfabetik sırayla açıklanır.

Not: Bu kısımda gösterilen özneliklerin adları, MQINQ ve MQSET çağrıları ile kullanılan adlardır. Öznelikleri tanımlamak, değiştirmek ya da görüntülemek için MQSC komutları kullanıldığında, diğer kısa adlar kullanılır; ek bilgi için [Script \(MQSC\) Commands](#) (Komut Dosyası (MQSC) Komutları) konusuna bakın.

Çizelge 177. Kuyruk yöneticisine ilişkin öznelikler	
Öznelik	Tanım
AlterationDate	Tanımlamanın son olarak değiştirildiği tarih
AlterationTime	Tanımlamanın son olarak değiştirildiği saat
AuthorityEvent	Yetkilendirme (Yetkili Değil) olaylarının oluşturulup oluşturulmayacağını denetler
BridgeEvent	IMS köprüsü olaylarının oluşturulup oluşturulmayacağını denetler
ChannelAutoDef	Otomatik kanal tanımlamasının kullanılmasına izin verilip verilmediğini denetler
ChannelAutoDefEvent	Kanal otomatik tanımlama olaylarının oluşturulup oluşturulmayacağını denetler
ChannelAutoDefExit	Otomatik kanal tanımlaması için kullanıcı çıkışının adı
ChannelEvent	Kanal olaylarının oluşturulup oluşturulmayacağını denetler
ClusterCacheTipi	Küme önbelleğinin boyutlarda mı, yoksa dinamik olarak boyutlandırılıp düzeltilmediğini denetler
ClusterWorkloadVerileri	Küme iş yükü çıkışa ilişkin kullanıcı verileri
ClusterWorkloadÇıkış	Küme iş yükü yönetimi için kullanıcı çıkışının adı
ClusterWorkloadUzunluğu	Küme iş yükü çıkışa aktarılan ileti verisi uzunluğu üst sınırı
CodedCharSetId	Kodlanmış karakter takımı tanıtıcısı
CommandEvent	Komut olay iletilerinin kuyruğa gönderilip gönderilmediğini denetler
CommandInputQName	Komut giriş kuyruğu adı
CommandLevel	Komut düzeyi
ConfigurationEvent	Yapılandırma olayı

Çizelge 177. Kuyruk yöneticisine ilişkin öz nitelikler (devamı var)

Öznitelik	Tanım
DeadLetterQName	Ölü-mektup kuyruğunun adı
DefClusterXmitQueueTipi	Varsayılan küme iletim kuyruğu tipi
DefXmitQName	Varsayılan iletim kuyruğu adı
DistLists	Dağıtım listesi desteği
InhibitEvent	Engelleyici (Inhibit Get ve Inhibit Sput) olaylarının oluşturulup oluşturulmayacağını denetler
LocalEvent	Yerel hata olaylarının oluşturulup oluşturulmadığını denetler
LoggerEvent	Kurtarma günlüğü olaylarının oluşturulup oluşturulmayacağını denetler
MaxHandles	Tanıtıcı sayısı üst sınırı
MaxMsgUzunluğu	İleti uzunluğu üst sınırı (byte)
MaxPriority	En yüksek öncelik
MaxUncommittedMsgs	Bir iş birimi içinde kesinleştirilmemiş ileti sayısı üst sınırı
PerformanceEvent	Başarımla ilgili olayların oluşturulup oluşturulmayacağını denetler
Altyapı	Kuyruk yöneticisinin çalışmakta olduğu altyapı
PubSubKipi	Yayınlama/abone olma motoru ve kuyruğa alınmış yayınlama/abone olma arabiriminin çalıştırılıp çalıştırılmayacağı
QMgrDesc	Kuyruk yöneticisi tanımı
QMgrIdentifier	Kuyruk yöneticisinin iç sistem tarafından oluşturulan benzersiz tanıtıcısı
QMgrName	Kuyruk yöneticisi adı
RemoteEvent	Uzak hata olaylarının oluşturulup oluşturulmadığını denetler
RepositoryName	Bu kuyruk yöneticisinin havuz hizmetleri sağladığı kümenin adı
RepositoryNamelist	Bu kuyruk yöneticisinin havuz hizmetleri sağladığı kümelerin adlarını içeren ad listesi nesnesinin adı
SSLCRLNamelist	Kimlik doğrulama bilgileri nesnelerinin adlarını içeren ad listesi nesnesinin adı (Bkz. Not 1)
SSLEC	SSL olaylarının oluşturulup oluşturulmayacağını denetler
SSLKeyRepository	SSL anahtar havuzunun yeri (Bkz. Not 1)
SSLKeyResetSayı	Şifreleme anahtarı yeniden anlaşılmadan önce, SSL etkileşimi içinde gönderilen ve alınan şifrelenmemiş bayt sayısını belirler.
StartStopOlayı	Başlatma ve durdurma olaylarının oluşturulup oluşturulmayacağını denetler
SyncPoint	Eşitleme noktası kullanılabilirliği
TraceRouteKaydı	İletilere ilişkin izleme rotası bilgilerinin kaydedilmesini denetler
TreeLifeSaati	Yönetimle ilgili olmayan konuların ömrü (saniye)
TriggerInterval	Tetikleme-ileti aralığı

Çizelge 177. Kuyruk yöneticisine ilişkin öznitelikler (devamı var)

Öznitelik	Tanım
Notlar: 1. Bu öznitelik, MQINQ çağrısını kullanarak sorgulanamaz ve bu bölümde anlatılmaz. Bu özniteliğe ilişkin ek bilgi için Kuyruk Yöneticisini Değiştirme başlıklı konuya bakın.	

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.

AuthorityEvent (10 basamaklı imzalı tamsayı)

Yetkilendirme (Yetkili Değil) olaylarının oluşturulup oluşturulmayacağını denetler.

AuthorityEvent özniteliğinin şu değerlerden birine ayarlanması gerekir:

EVARDIS

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.

BridgeEvent (karakter dizgisi)

Bu öznitelik, IMS köprüsü olay iletilerinin SYSTEM.ADMIN.CHANNEL.EVENT kuyruğu. Yalnızca z/OS üzerinde desteklenir.

ChannelAutoDef (10 basamaklı işaretli tamsayı)

Otomatik kanal tanımlamasına izin verilip verilmediğini denetler.

Bu öznitelik, CTCRVR 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.

ChannelAutoDefEvent (10 basamaklı imzalı tamsayı)

Kanal otomatik tanımlama olaylarının oluşturulup oluşturulmayacağını denetler.

Bu, CTCRVR, CTSVCN ve CTCLSD tipindeki kanallar için geçerlidir. Bu, aşağıdaki değerlerden birine sahip olabilir:

EVARDIS

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.

ChannelAutoDefExit (20 baytlık karakter dizilimi)

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.
- Kanalin oluşturulmasını tamamen engelle.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla CACADX seçiciyi kullanın. Bu özniteliğin uzunluğu LNEXN tarafından verilir.

ChannelEvent (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 özniteliğin 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 SSL 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 özniteliğine 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).
- EVARDIS (hiçbir kanal olayı oluşturulmadı; bu, kuyruk yöneticisi ilk varsayılan değeridir).

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IACHNE seçiciyi kullanın.

ClusterCacheTip (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 özniteliğin değerini saptamak için, MQINQ çağrısıyla CALWD seçiciyi kullanın.

ClusterWorkloadVerileri (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 özniteliğin değerini saptamak için, MQINQ çağrısıyla CALWD seçiciyi kullanın.

ClusterWorkloadExit (20 baytlık karakter dizgisi)

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 özniteliğin değerini saptamak için, MQINQ çağrısıyla CACLWX seçiciyi kullanın. Bu özniteliğin uzunluğu LNEXTN tarafından verilir.

ClusterWorkloadUzunluğu (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* özniteliği.
- *ClusterWorkloadLength* özniteliği.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IACLWL seçiciyi kullanın.

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.

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 bildirimi 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.

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 kuyruğudur (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.

CommandLevel (10 basamaklı imzalı tamsayı)

Komut Düzeyi. Bu, kuyruk yöneticisi tarafından desteklenen sistem denetimi komutlarının düzeyini gösterir.

Düzey, aşağıdaki değerlerden biridir:

CML700

Sistem denetimi komutlarının 700. düzeyi.

Bu değer aşağıdaki uygulamalar tarafından döndürülür:

- IBM MQ for IBM i Sürüm 7 Yayın 0

CML701

Sistem denetimi komutlarının 701. düzeyi.

Bu değer aşağıdaki uygulamalar tarafından döndürülür:

- IBM MQ for IBM i Sürüm 7 Yayın 0 Değişiklik 1

CML71

Sistem denetimi komutlarının 71. düzeyi.

Bu değer aşağıdaki uygulamalar tarafından döndürülür:

- IBM MQ for IBM i Sürüm 7 Yayın 1

CML800

Sistem denetimi komutlarının 800. düzeyi.

Bu değer aşağıdaki uygulamalar tarafından döndürülür:

- IBM MQ for IBM i
 - Sürüm 8.0

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 çağırısıyla IACMDL seçiciyi kullanın.

ConfigurationEvent

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 BİLGİLERİNİ SİL
- KANAL SİLME

- ADI SIL
- Süreci Sil
- QLOCAL ' ı DELETE (geçici bir dinamik kuyruk olmadığı sürece)
- QMODELİ/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, QSVCIEV 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.

DeadLetterQName (48 byte 'lık karakter dizisi)

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-İleti tanımlayıcısı” sayfa 1076 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. “MQDLH -Dead-letter üstbilgisi” sayfa 1032 .

Bu kuyruk, USNORM ' un Usage özniteliği olan bir yerel kuyruk olmalıdır.

Bir kuyruk yöneticisi tarafından ya da bir kuyruk yöneticisi tarafından desteklenmeyen bir kuyruk (teslim edilemeyen ileti) desteklenmiyorsa, ad boşluk Yekilariya yanŸlanmŸtir. 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 özniteliğin değerini saptamak için, MQINQ çağrısıyla CADLQ seçiciyi kullanın. Bu özniteliğin uzunluğu LNQN tarafından verilir.

DefClusterXmitQueueTipi (10 basamaklı işaretli tamsayı)

DefClusterXmitQueueType özniteliği, ileti almak için varsayılan olarak küme gönderen kanalları tarafından hangi iletim kuyruğunun seçildiğini, iletileri küme alıcılı kanallara göndermesini sağlar.

DefClusterXmitQueueType değeri, MQCLXQ_SCTQ ya da MQCLXQ_CHANNEL' dir.

MQCLXQ_SCTQ

All cluster-sender channels send messages from SYSTEM.CLUSTER.TRANSMIT.QUEUE. İletim kuyruğuna yerleştirilen iletilerin correlID , iletinin hangi küme gönderici kanalına gönderildiğini belirler.

SCTQ is set when a queue manager is defined. Bu davranış, Sürüm 7.5sürümünden önceki IBM WebSphere MQsürümlerinde örtük bir davranışa neden olur. Önceki sürümlerde, kuyruk yöneticisi özniteliği DefClusterXmitQueueTürü mevcut değildi.

MQCLXQ_CHANNEL

Her bir küme gönderici kanalı, iletileri farklı bir iletim kuyruğundan gönderir. Each transmission queue is created as a permanent dynamic queue from the model queue SYSTEM.CLUSTER.TRANSMIT.MODEL.QUEUE.

If the queue manager attribute, DefClusterXmitQueueTürü, is set to KANAL, varsayılan yapılandırma, tek tek küme iletim kuyruklarıyla ilişkilendirilmekte olan küme gönderici kanallarına çevrilir. The transmission queues are permanent-dynamic queues created from the model queue SYSTEM.CLUSTER.TRANSMIT.MODEL.QUEUE. Her iletim kuyruğu bir küme gönderici kanalı ile ilişkilendirilir. Bir küme gönderici kanal hizmetleri bir küme iletim kuyruğu olarak, iletim kuyruğu tek bir kümede yalnızca tek bir kuyruk yöneticisine ilişkin iletileri içerir. Kümeleri yapılandırabilir; böylece, bir kümedeki her bir kuyruk yöneticisinin tek bir küme kuyruğu içermesi gerekir. Bu durumda, kuyruk yöneticisinden her küme kuyruğuna gelen ileti trafiği, iletilerden diğer kuyruklara ayrı ayrı 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

“MQINQ-Nesne öznitelikleriyle ilgili bilgi” sayfa 1268

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

İlgili bilgiler

[Kuyruk Yöneticisini Değiştir](#)

[Sorgu Kuyruğu Yöneticisi](#)

DefXmitQName (48 byte 'lık karakter dizgisi)

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 özniteliğin ilk değeri boştur.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla CADXQN seçiciyi kullanın. Bu özniteliğin uzunluğu LNQN tarafından verilir.

DistLists (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 özniteliğin değerini saptamak için, MQINQ çağrısıyla IADIST seçiciyi kullanın.

InhibitEvent (10 basamaklı imzalı 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.

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.

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](#).

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, *MaxHandle* 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 'tır.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IAMHND seçicisini kullanın.

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 destekliorsa, 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 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.

MaxPriority (10 basamaklı işaretli tamsayı)

Maksimum öncelik.

Bu, kuyruk yöneticisi tarafından desteklenen ileti önceliği üst sınırıdır. Öncelikler, sıfır (en düşük) ile *MaxPriority* (en yüksek) arasında değişiyor.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IAMPRI seçiciyi kullanın.

MaxUncommittedMsgs (10 basamaklı işaretli tamsayı)

Bir iş birimi içinde kesinleştirilmemiş ileti sayısı üst sınırı.

Bu, bir iş birimi içinde bulunabilecek kesinleştirilmemiş ileti sayısı üst sınırıdır. Kesinleştirilmemiş ileti sayısı, yürürlükteki iş biriminin başlangıcından bu yana aşağıdaki toplamın toplamını içerir:

- Uygulama tarafından PMSYP seçeneği ile gönderilen iletiler
- Uygulama tarafından GMSYP seçeneği ile alınan iletiler
- PMSYP seçeneği içeren iletiler için kuyruk yöneticisi tarafından oluşturulan iletileri ve COA rapor iletilerini tetikler
- GMSYP seçeneğiyle alınan iletiler için kuyruk yöneticisi tarafından oluşturulan COD raporu iletileri

Aşağıdakiler, kesinleştirilmemiş iletiler olarak sayılan *değil* ' dir:

- 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 (sona erme raporu iletisinin GMSYP ' yi belirtmesine neden olsa bile)
- Kuyruk yöneticisi tarafından oluşturulan olay iletileri (olay iletisine neden olarak PMSYP ya da GMSYP olmasına neden olsa bile)

Not:

1. Kural dışı durum rapor iletileri Message Channel Agent (MCA) ya da uygulama tarafından oluşturulur ve bu nedenle, uygulama tarafından konulan ya da alınan sıradan iletiler ile aynı şekilde işlem görür.

2. Bir ileti ya da bölüm, PMSYP seçeneğiyle birlikte konulduğunda, kesinleştirilmemiş iletilerin 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 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.

Bu özneteliğin değerini saptamak için, MQINQ çağrısıyla IAMUNC seçicisini kullanın.

PerformanceEvent (10 basamaklı imzalı 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.

Altyapı (10 basamaklı işaretli tamsayı)

Kuyruk yöneticisinin çalışmakta olduğu altyapı.

Kuyruk yöneticisinin üzerinde çalışmakta olduğu işletim sistemini gösterir. Değer:

PL400

IBM i.

PubSubKipi (10 basamaklı işaretli tamsayı)

Yayınlama/abone olma motorunun ve kuyruğa alınan yayınlama/abone olma arabiriminin çalışıp çalışmadığı, bu nedenle uygulama programlama arabirimi 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.

QMGrDesc (64 baytlık karakter dizgisi)

Kuyruk yöneticisi tanımlaması.

Bu alan, tanımlayıcı açıklamalar için kullanılabilecek 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 öznitelemesiyle tanımlandığı gibi), bu alan başka bir kuyruk yöneticisine gönderilirse, bu karakterler yanlış çevrilebilir.

IBM i' ta varsayılan değer boşluktur.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla CAQMD seçiciyi kullanın. Bu özniteliğin uzunluğu LNQMMD tarafından verilir.

QMGrIdentifier (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.

QMGrName (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-İleti tanımlayıcısı](#)" sayfa 1076 ' 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.

RemoteEvent (10 basamaklı imzalı 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.

RepositoryName (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.

RepositoryNameList (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 LNNLN tarafından verilir.

SSLEVat (karakter dizgisi)

SSL olaylarının oluşturulup oluşturulmayacağını belirler.

Değer aşağıdakilerden biridir:

- EVRENA (MQINQ/PCF/config olayı) ENABLED (MQSC): SSL olayları oluşturulur (yani RC2371 olayı oluşturulur).
- EVRDIS (MQINQ/PCF/config olayı) DISABETLE (MQSC): SSL olayları oluşturulmaz. Kuyruk yöneticisinin ilk varsayılan değeri budur.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IASSLE seçiciyi kullanın.

SSLKeyResetSayı (tamsayı)

Gizli anahtar yeniden anlaşılmadan önce, bir SSL etkileşimi içinde gönderilen ve alınan toplam şifrelenmemiş baytların sayısını belirler. Byte sayısı, ileti kanalı aracısı (MCA) tarafından gönderilen denetim bilgilerini içerir.

Bu değer yalnızca, bu kuyruk yöneticisinden iletişimi başlatan SSL kanalı MCA ' lar tarafından kullanılır (yani, gönderici ve alıcı kanal çiftindeki gönderen kanalı MCA).

Bu özniteliğin değeri 0 'dan büyükse ve kanal kalp atışları bir kanal için etkinleştirilmişse, bir kanal sağlıklı işletim bildirimi 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 öznitelik için 0 değeri, gizli anahtarın hiçbir zaman yeniden anlaşma vermediğini gösterir. 1 bayt-32 KB arasında bir SSL/TLS gizli anahtar sıfırlama sayısı belirtirseniz, SSL/TLS kanalları 32 KB gizli anahtar sıfırlama sayısını kullanır. Bu, küçük SSL/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 bildirimi hemen sonra yeniden ilişki gerçekleşir.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IASSRC seçiciyi kullanın.

StartStopOlayı (10 basamaklı işaretli tamsayı)

Başlatma ve durdurma olaylarının oluşturulup oluşturulmayacağını denetler.

Bu öznitelik 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 özniteliğin değerini saptamak için, MQINQ çağrısıyla IASSE seçicisini kullanın.

SyncPoint (10 basamaklı işaretli tamsayı)

Uyumluluk noktası kullanılabilirliği.

Bu durum, yerel kuyruk yöneticisinin, MQGET, MQPUT ve MQPUT1 çağrılarına sahip iş birimlerini destekleyip desteklemediğini gösterir.

SPAVL

İş birimleri ve syncpoint'leme kullanılabilir.

SPNAVL

İş birimi ve uyumluluk noktası kullanılamıyor.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IASync seçiciyi kullanın.

TraceRouteKaydı (10 basamaklı imzalı tamsayı)

Bu, iletilere ilişkin bilgilerin bir kuyruk yöneticisiyle akış olarak kaydedilip kaydedilmeyeceğini denetler.

Değer aşağıdakilerden biridir:

- RECD: 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 [“Rapor seçenekleri ve ileti işaretleri” sayfa 1395](#) başlıklı konuya bakın.

Bu özniteliğin değerini saptamak için, MQINQ çağrısıyla IATRGI seçiciyi kullanın.

TreeLifeSüre (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 dizisine 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 değer 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.

TriggerInterval (10 basamaklı işaretli tamsayı)

Tetikleme iletileri aralığı.

Tetikleme iletilerinin sayısını sınırlamak için kullanılan bir zaman aralığı (milisaniye cinsinden). Bu, yalnızca *TriggerType* TTFRST 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 TTFRST 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 değer aralığında yer alıyor. Varsayılan değer 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 aşina olmalısınız; isimleri ve içerikleri hakkında kısa bir açıklama aşağıdaki metinde yer almalıdır.

IBM MQ dosyaları kopyala

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 son eki var. Adlandırılmış değişmezleri içeren ayrı kopya dosyaları ve her bir yapı için bir dosya vardır. Kopyalama dosyaları [“Dil bilgileri” sayfa 980](#) içinde listelenir.

Not: ILE RPG/400 için, dosya üyeleri olarak sağlar. QMQM kitaplığındaki QRPGLSRC.

Yapı bildirimleri DS deyimlerini içermez. Bu, uygulamanın DS deyimini kodlayarak ve bildirimin geri kalan kısmında kopyalanacak /COPY 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 980](#).

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 ileti 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 çağrıları 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) ' ın 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 örneklerle 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 çağrılarını kullanan programlar (statik bağlı çağrılar)

Kaynak QMQMSAMP/QRPGLESRC içinde var. Üyeler AMQ3xxx4adını taşır; burada xxx , örnek işlevi belirtir. Kopyalama üyeleri QMQM/QRPGLESRC içinde var olabilir. Her üye adı, G ya da Hsonekine sahiptir.

Çizelge 178 sayfa 1378 , 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.

Çizelge 178. Örnek programların adları	
	RPG (ILE)
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 kullanılabilecek 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 1379.](#)

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.

Çizelge 179. MQI ' nin (MQI) kullanımını gösteren örnek programlar	
Teknik	RPG (ILE)
MQCONN ve MQDISC çağrılarının kullanılması	AMQ3ECH4 ya da AMQ3INQ4
Örtük olarak bağlanma ve bağlantı kesme	AMQ3PUT4
MQPUT çağrısını kullanarak ileti koyma	AMQ3PUT4

Çizelge 179. MQI ' nin (MQI) kullanımını gösteren örnek programlar (devamı var)	
Teknik	RPG (ILE)
MQPUT1 çağrısını kullanarak tek bir ileti koyma	AMQ3ECH4 ya da AMQ3INQ4
İstek iletiyi yanıtlatma	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
Yanıtlatma 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. CHGMQMPRC komutunu ([Change MQ Process \(CHGMQMPRC\)](#) 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 *gerekir* 48 karakter uzunluğunda olabilir; Queue_Name ve Queue_Manager_Name değerini gereken sayıda boşlukla doldurarak elde edin.

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 ileti 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 ileti 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 *gerekir* 48 karakter uzunluğunda olabilir; Queue_Name ve Queue_Manager_Name değerini gereken sayıda boşlukla doldurarak elde edin. 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 ileti 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 büyüklerin 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 çağrıları 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')
```

where Queue_Name and Queue_Manager_Name *gerekir* be 48 characters in length, which you achieve by padding the Queue_Name and Queue_Manager_Name with the required number of blanks. 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 iletisi yazar.

Kuyrukta 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 kuyrukta 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 iletisi 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, kuyrukta 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 iletiğine 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.

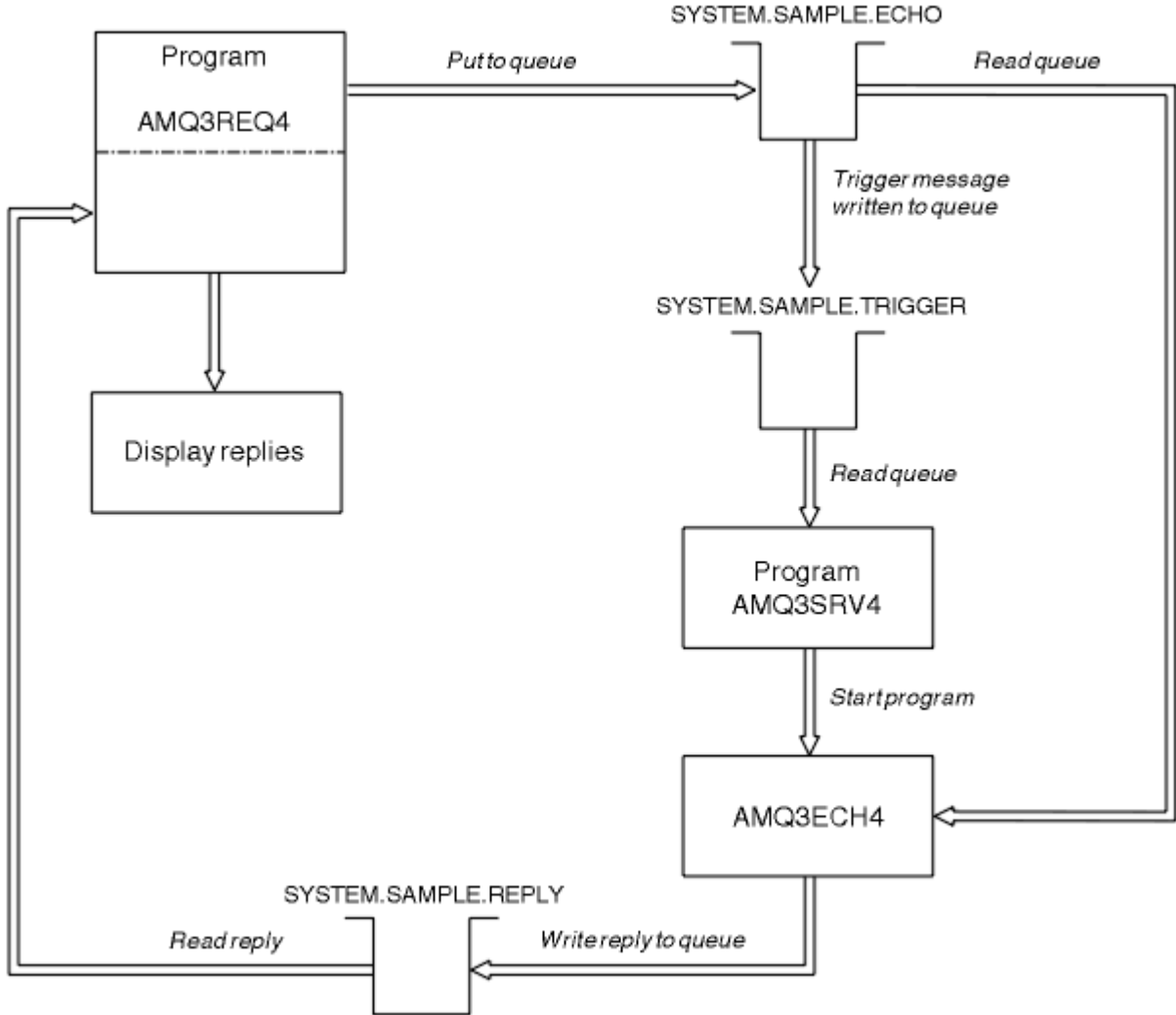
Daha sonra, program, MQCLOSE çağrısını kullanarak hem sunucu kuyruğunu, hem de yanıtlamayı kuyruğa kapatır. [Çizelge 180 sayfa 1382](#) , Sorgula ve Ayarla örnek programlarını çalıştırmak için gerekli olan Echo örnek programında yapılan değişiklikleri gösterir.

Not: Echo örnek programına ilişkin ayrıntılar başvuru amaçlı olarak eklenmiştir.

Çizelge 180. İstemci/Sunucu örnek programı ayrıntıları		
Program adı	SYSTEM/örnek kuyruğu	Program başlatıldı
Yankı	YANKI	AMQ3ECH4

Çizelge 180. İstemci/Sunucu örnek programı ayrıntıları (devamı var)

Program adı	SYSTEM/örnek kuyruğu	Program başlatıldı
Sor	INQ	AMQ3INQ4
Belirle	Belirle	AMQ3SET4



Şekil 9. Örnek İstemci/Sunucu (Echo) programı akış şeması

İstek örneğiyle tetiklemeyi kullanma

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:

- 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.
- 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 CHGMQMPCRC command (see [Change MQ Process \(CHGMQMPCRC\)](#) 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 1383 içinde 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')
```

Çünkü kuyruk adı ve kuyruk yöneticisi adı 48 karakter uzunluğunda olmalıdır *olmalıdır* .

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 çeşitlendirmeleri 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 olmalı *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 AMQ3ECH4 adını 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 *ApplId* 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 iletisinde, istek iletisinde belirtilen yanıt kuyruğuna veri içeren bir yanıt iletisi gönderir.

Echo örnek programının tasarımı

Program tetiklendiğinde, MQCONN çağrısını kullanarak varsayılan kuyruk yöneticisine belirtik olarak bağlanır. IBM MQ for IBM için bu 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 iletileri kaldırmak için MQGET çağrısını kullanır. Bu çağrı, 5 saniye bekleme süresi ile GMATM ve GMWT seçeneklerini kullanır. Program, bir istek iletilisi olup olmadığını görmek için her iletinin tanımlayıcısını sınar; değilse, program iletiyi atar ve bir uyarı iletilisi görüntüler.

İstek kuyruğundan kaldırılan her istek iletilisi için, program, yanıtlama kuyruğuna yanıt iletilisi koymak için MQPUT çağrısını kullanır. Bu ileti, istek iletilisinin 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 MQ for IBM i, although no sample is supplied for this situation. ECHO program çalışmasını yapmak için aşağıdaki işlemi gerçekleştirdiniz:

- Metin isteği iletileri göndermek için *Format, Encodingve 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 iletilisi) yapısıdır. Bu yapı, sorgulanacak özniteliklere sahip bir hedef kuyruğun adını içerir.

Tetikleme işleminin çalışması için, SYSTEM.SAMPLE.INQ. To do ao, specify the name of the Inquire sample program in the *AppLId* field of the SYSTEM.SAMPLE.INQPROCESS process definition. (Bunun için, [Change MQ Process \(CHGMQMPPRC\)](#) içinde açıklanan CHGMQMPPRC 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. AMQ3SRV4yerine 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 iletilisi için, Sorgula örnek programı, istek iletilisinde belirtilen kuyruğa ilişkin bilgileri içeren bir yanıt iletilisi gönderir. Yanıtlar, istek iletilisinde belirtilen yanıtlama kuyruğuna gönderilir.

Sorgulamak için örnek program tasarımı

Program tetiklendiğinde, MQCONN çağrısını kullanarak varsayılan kuyruk yöneticisine belirtik olarak bağlanır. IBM MQ for IBM için gerekli olmasa da, bu tasarım özelliği kaynak kodu değiştirmeden diğer platformlarda aynı programı kullanabilmenize yardımcı olur.

Program daha sonra, başlatıldığı sırada tetikleme iletilisi 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 iletilisi olup olmadığını görmek için her iletinin tanımlayıcısını sınar; değilse, program iletiyi atar ve bir uyarı iletilisi görüntüler.

İstek kuyruğundan kaldırılan her istek iletilisi 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, CurrentQDepthve 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 *ApplId* 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 iletilerini 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 MQ for IBM için bu 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* değerini 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 değiştirgeleri 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 olmalıdır , 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 *girmelisiniz* :

```
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 başvurusu](#) listesinde yer alıyor.
- C++ çağrılarını [C++ kullanılması](#) 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), çağırının, çağrıların 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 [API tamamlama ve neden kodları](#) 'nda yer alıyor.

Bu listede IBM i neden kodunuzu bulmak için öndeki "RC" yi kaldırın; örneğin, RC2002 , 2002 olur. Ayrıca, diğer platformlarda olduğu gibi gösterilen tamamlama kodları da gösterilir:

Çizelge 181.	
IBM i	Diğer platformlar
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:*
 - OOBW

- OOINPQ
- OOINPX
- OOINPS
- OOINQ
- OOUT
- OOSSET
- Aşağıdakilerin yalnızca *bir* ' e izin verilir:
 - OOINPQ
 - OOINPX
 - OOINPS
- Aşağıdakilerin yalnızca *bir* ' e izin verilir:
 - OOBNDQ
 - 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.
- OOSSET* 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
- GMPSYP, 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-İleti tanımlayıcısı” sayfa 1076.](#)

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.

GIRIŞ MSK

Ayrılmış bitler için maske.

Bu alt alan, *MDENC* alanı içinde 0-19 arasındaki konumları bit olarak kaplar.

İkili 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.

Paketlenmiş-ondalık-tamsayı kodlaması

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.

Kayan noktalı kodlama

Kayan noktalı kodlama için geçerli değerler

Kayan noktalı kodlama için geçerli olan değerler şunlardır:

GIRIş

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 ayarlanmışlarla 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.

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.

Kodlamalar 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 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 değer için eşitlik için test edilebilir.

Makine mimarisi kodlamalarının özeti

Makine mimarileri için kodlamaları özetleyen bir tablo.

Makine mimarilerine ilişkin kodlamalar [Çizelge 182 sayfa 1394](#) içinde gösterilir.

Çizelge 182. Makine mimarilerine ilişkin kodlamaların özeti			
Makine mimarisi	İkili tamsayı kodlaması	Paketli-onlu tamsayı kodlaması	Kayan noktalı kodlama
IBM i	normal	normal	IEEE olağan
Intel x86	Tersine Çevrilmiş	Tersine Çevrilmiş	IEEE ters

Çizelge 182. Makine mimarilerine ilişkin kodlamaların özeti (devamı var)

Makine mimarisi	İkili tamsayı kodlaması	Paketli-onlu tamsayı kodlaması	Kayan noktalı kodlama
PowerPC	normal	normal	IEEE olağan
System/390	normal	normal	System/390

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-İleti tanımlayıcısı” sayfa 1076. 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 tanıımıyorsa 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 unutulmadır. 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.

Rapor alanının çözülmesi

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.

İletinin yapısı-işaretler alanı

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 çağrısının 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.

Veri dönüştürme

Bu konuda, veri dönüştürme çıkışa ilişkin arabirim ve veri dönüştürme gerektiğinde kuyruk yöneticisi tarafından gerçekleştirilen işleme anlatılır.

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 GMCONV seçeneğinin belirtilmesini gerektirir.

Aşağıdaki açıklamalar açıklanmaktadır:

- GMCONV seçeneğine yanıt olarak kuyruk yöneticisi tarafından gerçekleştirilen işleme; bkz. [“Dönüştürme işlemi” sayfa 1399](#).
- Yerleşik bir biçim işlenirken kuyruk yöneticisi tarafından kullanılan işleme kuralları; kullanıcı tarafından yazılan çıkışlar için de bu kurallar önerilir. Bkz. [“İşleme kuralları” sayfa 1400](#).
- Rapor iletilerinin dönüştürülmesine özel dikkat edilmesi gereken noktalar; bkz. [“Rapor iletilerinin dönüştürülmesi” sayfa 1403](#).
- Veri dönüştürme çıkışa aktarılan parametreler; bkz. [“MQCONVX-Veri dönüştürme çıkışı” sayfa 1414](#).
- Karakter verilerini farklı gösterimler arasında dönüştürmek için çıkışta kullanılabilecek bir çağrı; bkz. [“MQXCNCV-Karakterleri dönüştür” sayfa 1409](#).
- Çıkışa özel veri yapısı parametresi; bkz. [“MQDXP-Data-conversion çıkış değiştirgesi” sayfa 1404](#).

Dönüştürme işlemi

Bu bilgilerde, kuyruk yöneticisi tarafından GMCONV seçeneğine yanıt olarak gerçekleştirilen işleme anlatılır.

MQGET çağrısında GMCONV seçeneği belirtilirse ve uygulamaya döndürüleniz 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 değıştirgesinin uzunluğu sıfır.

Bu durumda, MQGET çağrısını yayınlayan uygulamaya dönüştürme yapılmadan ileti döndürölür; MSGDSC parametresindeki MDCSI ve MDENC değeri, iletteki denetim bilgilerindeki değeri 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 değeri farklıysa ve dönüştürölcek veriler varsa gerçekleştirilir:

1. İletideki denetim bilgilerindeki MDFMT alanı FMNONE değeri sahipse, ileti dönüştürölmeden, tamamlanma kodu CCWARN ve neden kodu RC2110döndürölür.

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 yapılır:

- GMATM seçeneği değil belirtilmişse, ileti dönüştürölmeden, tamamlanma kodu CCWARN ve neden kodu RC2080olan bir ileti döndürölür.
- GMATM seçeneği belirtilmişse , tamamlanma kodu CCWARN olarak ayarlanır, neden kodu RC2079olarak ayarlanır ve dönüştürme işlemi devam eder.

4. İleti, kesilmeden arabelleğe sığabiliyorsa ya da GMATM seçeneği belirtilmişse, aşağıdaki yapılır:

- 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 RC2110ile çevrilmemiş olarak döndürölür.

Hiçbir hata oluşmazsa, veri dönüştürme hizmetindeki ya da kullanıcı tarafından yazılan çıkıştan alınan çıkış, dönüştürölülen iletidir ve MQGET çağrısını yayınlayan uygulamaya döndürölcek tamamlanma kodu ve neden kodu.

5. Dönüştürme başarılı olursa, kuyruk yöneticisi dönüştürölülen iletiyi uygulamaya geri döndürölür. Bu durumda, MQGET çağrısının döndürölüğü 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 iletii uygulamaya geri döndürür.

İşleme 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 şunlardır:

Yerleşik biçimler
FMADMN
FMMDE
FMCIICS
FMPCF
FMCMO1
FMRMH
FMCMO2
FNRFH
FBIDLH
FMRFH2
FMDH
FMSTR
FMEVNT
FMTM
FMIMS
FMXQH
FMIMVS

- İleti dönüştürme sırasında genişler ve *BUFFER* parametresinin boyutunu aşarsa, aşağıdaki yapılır:
 - GMATM seçeneği *değil* belirtilmişse, ileti dönüştürülmeden, tamamlanma kodu CCWARN ve neden kodu RC2120 olan bir ileti döndürülür.
 - GMATM seçeneği *belirtilmişse*, ileti kısaltılır, tamamlanma kodu CCWARN olarak ayarlanır, neden kodu RC2079 olarak ayarlanır ve dönüştürme işlemi devam eder.
- Kesme gerçekleşirse (dönüştürme öncesinde ya da sırasında), *BUFFER* değıştirgesinde 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 genişletebilir 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 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 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 kayan noktalı veri içeriyorsa (kaynak ve hedef kayar noktalı kodlamalar özdeş olduğu için), hata ortaya konabilir ya da saptanmayabilir.

Hata tanılınırsa, 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 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, *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 dışında:

- 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 UCS-2 , kuyruk adlarında geçerli olan karakterler için SBCS karakterleri olmayan bir karakter takımı örneğidir.

2. If the message data for a built-in format is truncated, fields within the message which contain lengths of strings, or counts of elements or structures, are *değil* adjusted to reflect the length of the data returned to the application; the values returned for such fields within the message data are the values applicable to the message *kesmeden önce*.

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 MQDLH yapısıyla başlar ve bu, sıfır ya da daha fazla sayıda uygulama ileti verisi izlemiş olabilir. Uygulama ileti verilerinin biçimi, karakter kümesi ve kodlaması, iletinin başlangıcındaki MQDLH yapısındaki *DLFMT*, *DLCS* ve *DLENC* alanları tarafından tanımlanır. MQDLH yapısı ve uygulama ileti verilerinin farklı karakter kümeleri ve kodlamalar olabildiği için, MQDLH yapısının biri ya da her ikisi ya da her ikisi için de dönüştürme gerektirecek veriler olabilir.

Kuyruk yöneticisi, önce MQDLH yapısını gerektiği gibi dönüştürür. Dönüştürme başarılı olursa ya da MQDLH yapısı dönüştürme gerektirmiyorsa, kuyruk yöneticisi, MQDLH yapısındaki *DLCS* ve *DLENC* alanlarını, uygulama ileti verilerinin dönüştürülmesinin gerekli olup olmadığını görmek için denetler. Dönüştürme *gerekliyse*, kuyruk yöneticisi kullanıcı tarafından yazılan çıkışı, MQDLH yapısındaki *DLFMT* alanı tarafından verilen adda 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:

- MQDLH yapısı dönüştürülemedi. Bu durumda, uygulama ileti verileri de dönüştürülmez.
- MQDLH yapısı dönüştürüldü, ancak uygulama ileti verileri dönüştürülmedi.

The application can examine the values returned in the *MDCSI* and *MDENC* fields in the *MSGDSC* parameter, and those in the MQDLH 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 verile izlenebilir. Bu ek veriler tipik olarak, uygulama ileti 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 içindeki *içinde* bulunan MQMD yapısındaki *MDFMT*, *MDCSI* ve *MDENC* alanları tarafından verilir. Sonraki her IBM MQ üstbilgi yapısı için, yapıdaki *MDFMT*, *MDCSI* ve *MDENC* alanları, o yapıyı izleyen verileri açıklar; bu veriler, başka bir IBM MQ üstbilgi yapısı ya da uygulama ileti veridir.

Bir FMXQH ileti için GMCONV seçeneği belirtilirse, uygulama ileti verileri ve MQ üstbilgi yapılarının belirli 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 *değil* the application message data; the values will therefore *değil* 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*, *MDCSI* ve *MDENC* alanlarının değerleri, uygulama ileti verilerini açıklar. Başka bir IBM MQ üstbilgi yapısı yoksa, uygulama ileti 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 iletiyse, 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 ileti 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 aracılığı tarafından iletim kuyruklarından ileti aldıklarında kullanılmak üzere tasarlanmıştır.

Rapor iletilerinin dönüştürülmesi

Bir rapor ileti, özgün iletinin göndericisinin belirttiği rapor seçeneklerine göre değişen miktarlarda uygulama ileti verisi içerebilir.

Belirli bir durumda, bir rapor ileti 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 olasılıklar nedeniyle, veri dönüştürme çıkışı, aktarılan verilerin uzunluğunu bir biçimde ifade etmek için biçim adını *kullanmamalıdır* ; bunun yerine, çıkış, sağlanan verilerin uzunluğunu denetleyerek, biçim adının belirttiği uzunluktan *daha az* veri dönüştürmeye hazırlıklı olmalıdır. 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.

MQDXP-Data-conversion çıkış değiştirgesi

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 özneliği 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 *olamaz* değiştirilir.

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 1405
- “RPG bildirimi (CMQDXPH dosyasını kopyala)” sayfa 1409

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 döndürülebilmek 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 verdiğinden 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 dönüştürülebilecek, 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 döndürülebilecek neden kodunu içerir. Olası değerler arasında, iletinin, uygulamanın sağladığı arabelleğe sığması için kesildiğini belirten RC2079 ve 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 dönüştürülemez iletiyi döndürmesi için, çıkışta *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 dizgisi)

Yapı tanıttıcısı.

Değer şu olmalıdır:

DXSIDV

Veri dönüştürme çıkış parametresi yapısıyla ilgili tanıttıcı.

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 bildirimi (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
```

MQXCNCV-Karakterleri dönüştür

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 1410](#)
- [“Parametreler” sayfa 1410](#)
- [“RPG çağrısı \(ILE\)” sayfa 1414](#)

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 değişmezi bir kereden fazla eklemeyin).

Default-conversion option(Varsayılan dönüştürme seçeneği): Aşağıdaki seçenek, varsayılan karakter dönüştürmenin kullanımını denetler:

DCCDEF

Varsayılan dönüştürme.

Bu seçenek, çağrıda belirtilen karakter kümelerinden biri ya da her ikisi desteklenmiyorsa, varsayılan karakter dönüştürmenin kullanılabilirliğini 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.

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 seçeneği: Aşağıdaki seçenek, dönüştürülen dizginin hedef arabelleğe sığmasını sağlamak için, kuyruk yöneticisinin dönüştürülen dizeyi boşluklarla doldurmasına ya da önemsiz sondaki karakterleri atmasına olanak sağlar.

DCCFIL

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 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 (UCS-2 gibi) olduğunda bu durum oluşabilir; ancak *TGTLEN* , tek sayıda byte olan bir uzunluk belirtir.
- *TGTLEN* , *SRCLLEN* değerinden 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* , *SRCLLEN* değerinden küçük ya da daha büyük olabilir. MQXCNCV ' den dönüşte *DATLEN* , *TGTLEN* değerinden küçük ya da ona eşit.

Kodlama seçenekleri: Kaynak ve hedef dizgilerinin tamsayı kodlamasını belirtmek için aşağıdaki seçenekler kullanılabilir. The relevant encoding is used *yalnızca* when the corresponding character set identifier indicates that the representation of the character set in main storage is dependent on the encoding used for binary integers. Bu, yalnızca bazı çok baytlı karakter kümelerini (example, UCS-2 karakter kümeleri için) 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.

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 “Kodlamalar çözümleniyor” sayfa 1394 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 UCS-2 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 SRCSLEN)-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. *SRCSLEN* değerinden 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 1388.](#)

RPG çağırısı (ILE)

```
C*..1.....2.....3.....4.....5.....6.....7..  
C                                CALLP      MQXCNCV(HCONN : OPTS : SRCCSI :  
C                                SRCLN : SRCBUF : TGTCSE :  
C                                TGTLEN : TGTBUF : DATLEN :  
C                                CMPCOD : REASON)
```

Çağrıya ilişkin prototip tanımı:

```
D*..1.....2.....3.....4.....5.....6.....7..  
DMQXCNCV      PR      EXTPROC('MQXCNCV')  
D* Connection handle  
D HCONN      10I 0 VALUE  
D* Options that control the action of MQXCNCV  
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 TGTCSE      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
```

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 1416](#)).

This definition is part of the IBM MQ Data Conversion Interface (DCI), which is one of the IBM MQ framework interfaces.

- [“Sözdizimi” sayfa 1414](#)
- [“Kullanım notları” sayfa 1414](#)
- [“Parametreler” sayfa 1416](#)
- [“RPG çağırısı \(ILE\)” sayfa 1417](#)

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 işlev, çıkışa ilişkin sağlayıcı tarafından tanımlanır; ancak çıkış, burada açıklanan kurallara ve MQDXP ' nin ilişkili değiştirgeyle uyumlu olmalıdır.

Veri dönüştürme çıkışı için kullanılabilecek programlama dilleri ortam tarafından belirlenir.

2. Çıkış yalnızca aşağıdakilerin *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, çıkış 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. "[Rapor iletilerinin dönüştürülmesi](#)" sayfa 1403).

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 *değil* 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 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. *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 RC2219 ya da öngörülemez diğer hatalar ile başarısız olur.
11. Kuyruk yöneticisi tarafından sağlanan bir MQCONVX giriş noktası sağlanmadı. Çıkışa ilişkin ad, tüm ortamlarda gerekmeseyse de, bu adın biçim adıyla (MQMD ' de *MDFMT* alanında bulunan ad) aynı olması gerekir.

Parametreler

MQCONVX çağrısında şu değişirler var:

MQDXP (MQDXP)-giriş/çıkış

Veri dönüştürme çıkış parametresi öbeği.

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 1404 .

MQMD (MQMD)-giriş/çıkış

İleti tanımlayıcısı.

Çıkışa giriş sırasında, dönüştürme gerçekleştirilmezse, uygulamaya döndürüleceğini bildiren ileti tanımlayıcısı budur. Bu nedenle, *INBUFI* içinde yer alan dönüştürülemez iletinin *MDFMT*, *MDENC* ve *MDCSI* değerini içerir.

Not: Çıkışa aktarılan *MQMD* parametresi her zaman, çıkışı 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, çıkış bir version-2 *MQMD* geçerilir.

Çıkışta, çıkış *MDENC* ve *MDCSI* alanlarını uygulama tarafından istenen değerlere değiştirmeli, dönüştürme başarılı olursa, bu değişiklikler uygulamaya geri yansıtılır. Çıkışta yapılan diğer değişiklikler yoksayılır; bu değişiklikler uygulamaya geri yansıtılmaz.

Çıkış, *MQDXP* yapısının *DXRES* alanında *XROK* değerini döndürürse, ancak ileti tanımlayıcısındaki *MDENC* ya da *MDCSI* alanlarını değiştirmezse, kuyruk yöneticisi bu alanlar için, *MQDXP* yapısındaki ilgili alanların çıkışa girişte sahip olduğu değerleri 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 yoksayılır.

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

SOAP başvurusu

Alfabetik olarak düzenlenmiş SOAP başvuru bilgileri içinIBM MQ iletimi.

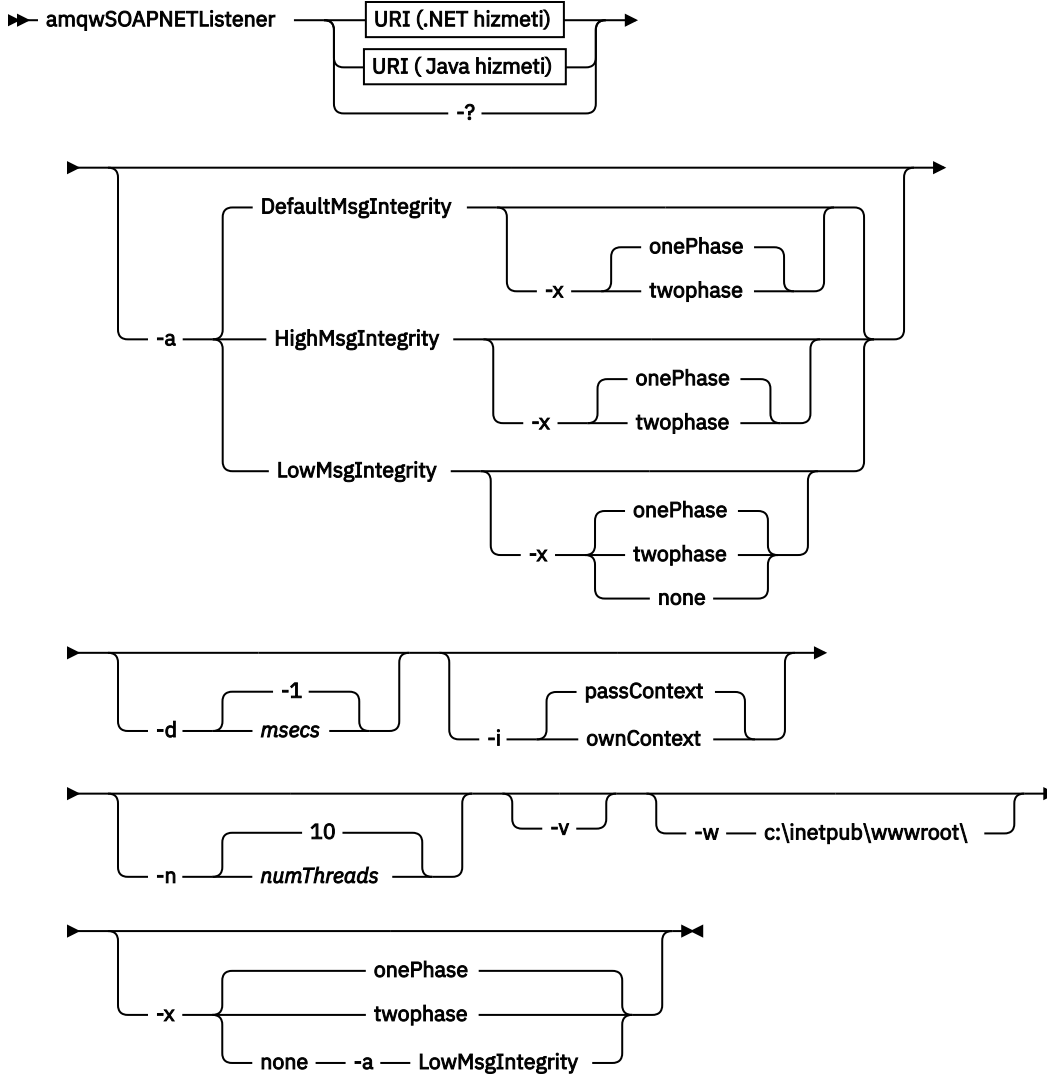
amqwSOAPNETListener: IBM MQ SOAP dinleyicisi for .NET Framework 1 ya da 2

IBM MQ SOAP dinleyicisi için .NET Framework 1 ya da 2 için sözdizimi ve parametreler.

Amaç

Starts the IBM MQ SOAP listener for .NET Framework 1 or 2.

.NET



Gerekli parametreler

URI altyapı

Bkz. [“Web hizmeti konuşlandırması için URI sözdizimi ve değıştirmeleri”](#) sayfa 1455.

-?

Komutun nasıl kullanıldığını açıklayan yardım metni yazdırın.

İsteğe bağlı parametreler

-a integrityOption

integrityOption , başarısız bir istek iletisini ölü mektup kuyruğuna koymanın mümkün değilse, IBM MQ SOAP dinleyicilerinin davranışını belirtir. *integrityOption* , aşağıdaki değerlerden birini alabilir:

DefaultMsgIntegrity

Kalıcı olmayan iletiler için, dinleyici bir uyarı iletisi görüntüler ve atılmakta olan özgün iletiyle yürütülmeye devam eder. Kalıcı iletiler için, bu ileti bir hata iletisi görüntüler, istek iletisini yedekler, böylece istek kuyruğunda kalır ve çıkılır. DefaultMsgBütünlüğü applies if the -a option is omitted, or if *integrityOption* is not specified.

LowMsgIntegrity

Hem kalıcı hem de kalıcı olmayan iletiler için dinleyici bir uyarı görüntüler ve iletiyi yürütmeye, yürütmeye devam eder.

HighMsgIntegrity

Hem kalıcı hem de kalıcı olmayan iletiler için, dinleyici bir hata iletisi görüntüler, istek iletisini yedekler, böylece istek kuyruğunda kalır ve çıkılır.

Devreye alma yardımcı programı, -x ve -a işaretlerinin uyumluluklarını denetler. -x none belirtilirse, -a LowMsgIntegrity (Bütünlük Ayarı) belirtilmelidir. İşaretler uyumsuz olursa, konuşlandırma yardımcı programı bir hata iletisiyle çıkar ve herhangi bir devreye alma adımı atılmıyorsa.

-d msec

msec , herhangi bir iş parçacığında istek iletileri alındıysa, IBM MQ SOAP dinleyicisinin canlı olarak kalabileceği süreyi (milisaniye) belirtir. *msec* için -1 değeri belirlendiyse, dinleyici süresiz olarak etkin kalır.

-i Bağlam

Bağlam , dinleyicilerin kimlik bağlamını geçip geçmeyeceğini belirtir. *Bağlam* aşağıdaki değerleri alır:

passContext

Özgün istek iletisinin kimlik bağlamını yanıt iletisine ayarlayın. SOAP dinleyici, bağlamı istek kuyruğundan kaydetme ve yanıt kuyruğuna aktarmaya ilişkin yetkiye sahip olduğunu denetler. Bağlam, bağlam kaydetmek için istek kuyruğunu açarken ve bağlam geçirmek için yanıt kuyruğu açılırken çalıştırma zamanında çekleri yapar. Gereken yetkiye sahip değilse ya da MQOPEN çağrısı başarısız olursa ve yanıt iletisi işlenmez. Yanıt iletisi, başarısız olan MQOPEN' den dönüş kodunu içeren, ölü harf üstbilgisiyle birlikte gönderilen iletiler kuyruğuna konmaktadır. Daha sonra dinleyici, sonraki gelen iletileri olağan şekilde işlemeye devam eder.

ownContext

SOAP dinleyicisi bağlam geçirmiyor. Döndürülen bağlam, özgün istek iletisini yaratan kullanıcı kimliği yerine dinleyici tarafından çalışmakta olan kullanıcı kimliğini yansıtır.

Kaynak bağlamdaki alanlar, kuyruk yöneticisi tarafından ayarlanır ve SOAP dinleyicisine göre değil.

-n numThreads

numThreads , IBM MQ SOAP dinleyicisine ilişkin oluşturulan başlangıç komut kütüklerindeki iş parçacıklarının sayısını belirtir. Varsayılan değer 10'dur. Yüksek ileti verimi varsa, bu sayıyı artırmayı düşünün.

-v

-v , dış komutlardan ayrıntılı çıkışı ayarlar. Hata iletileri her zaman görüntülenir. Uyarlanmış konuşlandırma komut dosyaları yaratmak için uyarlayabileceğiniz çıkış komutlarını -v kullanın.

-w serviceDirectory

serviceDirectory , web hizmetini içeren dizindir.

-x transaktiflik

transactionality , dinleyici için işlemsel denetimin tipini belirtir. *transactionality* , aşağıdaki değerlerden birine ayarlanabilir:

onePhase

IBM MQ bir fazlı destek kullanılır. Sistem işleme sırasında başarısız olursa, istek iletisine yeniden teslim edilir. IBM MQ işlemleri, yanıt iletilerinin tam olarak bir kez yazıldığından emin olur.

twoPhase

İki aşamalı destek kullanılır. Hizmet uygun şekilde yazılmışsa, ileti tam olarak bir kez teslim edilir, diğer kaynaklarla koordine edilir, hizmetin tek bir taahhüt halinde yürütülmesini sağlar. Bu seçenek yalnızca sunucu bağ tanımları bağlantıları için geçerlidir.

none

İşlem desteği yok. Sistem işleme sırasında başarısız olursa, istek iletisi, kalıcı olsa bile kaybedilebilir. Hizmet yürütülmeyebilir ya da yürütülmemiş olabilir ve yanıt, rapor ya da ölü harf iletileri yazılamayabilir ya da olmayabilir.

Devreye alma yardımcı programı, -x ve -a işaretlerinin uyumluluklarını denetler. Ayrıntılar için -a işaretinin açıklamalarına bakın.

.NET Örnek

```
amqwSOAPNETListener
-u "jms:/queue?destination=myQ&connectionFactory=()
&targetService=myService&initialContextFactory=com.ibm.mq.jms.Nojndi"
-w C:/wmqsoap/demos
-n 20
```

amqswsdl: .NET Framework 1 ya da 2 hizmeti için WSDL oluşturur

amqswsdl , .NET Framework 1 ya da 2 için yazılan bir Web hizmetini alır ve sınıf için WSDL 'yi oluşturur; oluşturulan WSDL' ye SOAP for SOAP olanağına ilişkin IBM MQ iletimi için sağladığınız URI ' yi yerleştirir.

Amaç

IBM MQ'a konuşlandırılan hizmetin URI 'sını içeren WSDL' yi oluşturmak için **amqswsdl** kullanın. İstemci yetkili sunucuları oluşturmak için WSDL ' yi kullanın.

➡ amqswsdl — *escapedUri* — *className* — .asmx — *className* — .wsdl ➡

Parametreler

escapedUri (Giriş)

Tüm "&" ile hizmet URI 'si "&." a kaçtı. Örneğin:

```
"jms:/queue?destination=REQUESTDOTNET
&amp.initialContextFactory=com.ibm.mq.jms.Nojndi
&amp.connectionFactory=(connectQueueManager(QM1)binding(server))
&amp.targetService=Quote.asmx"
```

className.asmx (Giriş)

Hizmet sınıfı.

className.wsdl (Çıkış)

Hizmet WSDL.

Tanım

Sınıf, kod ardında programlama modeli kullanılarak uygulandıysa, *className.dll* oluşturmalı ve bunu ./bin içinde saklamanız gerekir.

amqwclientconfig: SOAP için IBM MQ iletimi için Axis 1.4 Web hizmetleri istemcisi konuşlandırma tanımlayıcısı yaratın

amqwclientconfig , client-config.wsdd Axis 1.4 istemci konuşlandırma tanımlayıcısı dosyasını yaratır.

Amaç

Tanımlayıcısına jms: / iletimi ekler ve java:com.ibm.mq.soap.transport.jms.WMQSender iletimi, jms: iletimi için SOAP isteklerini işlemek için sınıf olarak registers' i kayda geçirir.

Sözdizimi

➤ amqwclientconfig ➤

Tanım

amqwclientconfig calls **amqwsetcp** to set the CLAS and runs the command:

```
java org.apache.axis.utils.Admin client "%WMSOAP_HOME%\bin\amqwclientTransport.wsdd"
```

amqwdeployWMQService: web hizmeti yardımcı programını konuşlandır

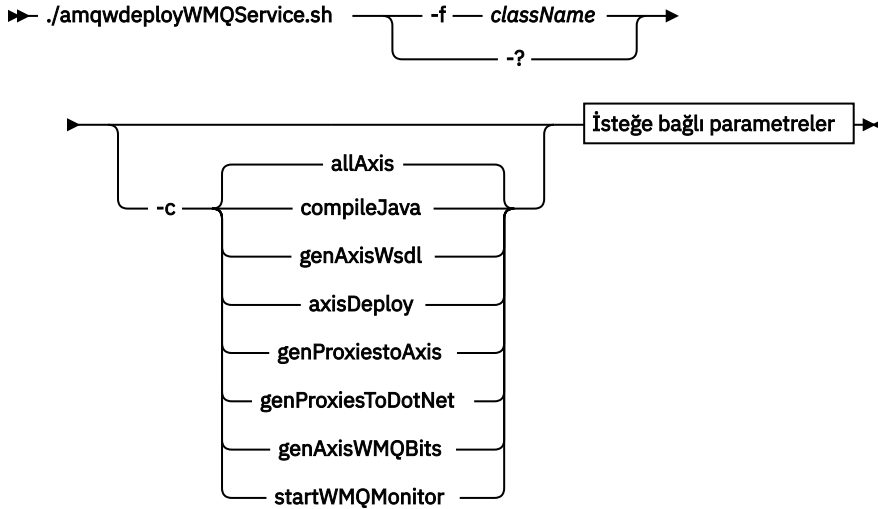
Konuşlandırma yardımcı programı, iletim olarak IBM MQ komutunu kullanarak bir Web hizmeti olarak kullanılmak üzere bir hizmet sınıfı hazırlar.

Amaç

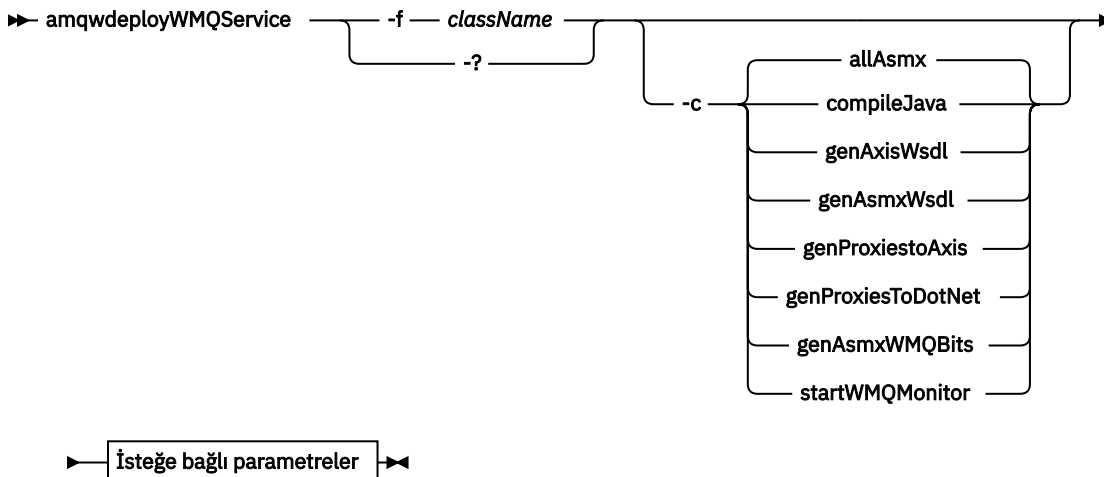
Bir Axis 1.4, .NET Framework 1 ya da .NET Framework 2 hizmeti konuşlandırmak için gereken dosyaları oluşturmak için devreye alma yardımcı programını kullanın. Use the files to deploy a service invoked by IBM MQ. **amqwdeployWMQService** tarafından oluşturulan dosyalar, "[amqwdeployWMQService' tan çıkış dosyaları](#)" sayfa 1425 içinde gösterilir.

Sözdizimi şeması

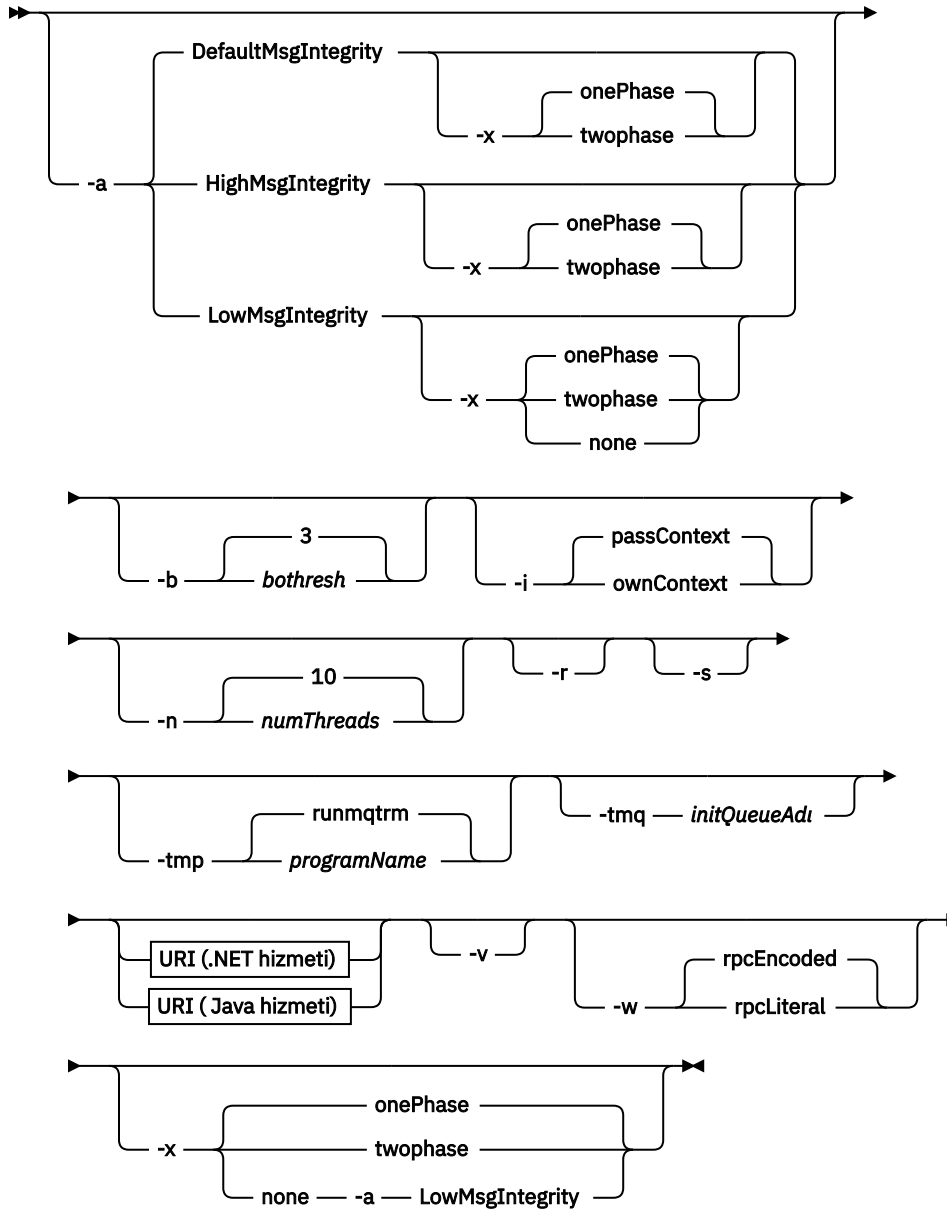
UNIX ve Linux sistemleri



Windows



İsteğe bağlı parametreler



Gerekli parametreler

-f className

`className`, konuşlandırılacak sınıfın adıdır. Axis hizmetleri için `className`, Java kaynak dosyasıdır ve .NET hizmetleri için `.asmx` dosyası. Şekil 10 sayfa 1422, bir Axis hizmetinin konuşlandırılmasını ve .NET hizmetinin Şekil 11 sayfa 1422 'inin konuşlandırılmasını gösterir.

```
amqwdployWMQService -f javaDemos/service/StockQuoteAxis.java
```

Şekil 10. Axis hizmetini devreye alma örneği

```
amqwdployWMQService -f StockQuoteDotNet.asmx
```

Şekil 11. .NET hizmetini örnek olarak devreye alma

Java için, *className* tam olarak paket adı ile nitelenmelidir. Bu, dizin ayırıcıları olan bir yol adı olarak ya da nokta ayırıcıları olan bir sınıf adı olarak belirtilebilir. Oluşturulan sınıf, *./generated/client/remote/path name* içinde bulunur. Bir .NET hizmeti için, dizin belirtilmesine rağmen, oluşturulan Java yetkili sunucuları her zaman *./generated/client/remote/dotNetService*'de bulunur.

-u seçeneği ve URI değeri *targetService* olarak bir URI belirtirseniz, konuşlandırma yardımcı programı *className* ögesini denetler. *className* , *targetService* ile eşleşmelidir. Sınıf ve hizmet eşleşmezse, devreye alma yardımcı programı bir hata iletisi görüntüler ve çıkar.

-?

Komutun nasıl kullanıldığını açıklayan yardım metni yazdırın.

İsteğe bağlı parametreler

-a *integrityOption*

integrityOption , başarısız bir istek iletisini ölü mektup kuyruğuna koymanın mümkün değilse, IBM MQ SOAP dinleyicilerinin davranışını belirtir. *integrityOption* , aşağıdaki değerlerden birini alabilir:

DefaultMsgIntegrity

Kalıcı olmayan iletiler için, dinleyici bir uyarı iletisi görüntüler ve atılmakta olan özgün iletiyle yürütülmeye devam eder. Kalıcı iletiler için, bu ileti bir hata iletisi görüntüler, istek iletisini yedekler, böylece istek kuyruğunda kalır ve çıkarılır. *DefaultMsgBütünlüğü* applies if the -a option is omitted, or if *integrityOption* is not specified.

LowMsgIntegrity

Hem kalıcı hem de kalıcı olmayan iletiler için dinleyici bir uyarı görüntüler ve iletiyi yürütmeye, yürütmeye devam eder.

HighMsgIntegrity

Hem kalıcı hem de kalıcı olmayan iletiler için, dinleyici bir hata iletisi görüntüler, istek iletisini yedekler, böylece istek kuyruğunda kalır ve çıkarılır.

Devreye alma yardımcı programı, -x ve -a işaretlerinin uyumluluklarını denetler. -x none belirtilirse, -a LowMsgIntegrity (Bütünlük Ayarı) belirtilmelidir. İşaretler uyumsuz olursa, konuşlandırma yardımcı programı bir hata iletisiyle çıkar ve herhangi bir devreye alma adımı atılmıyorsa.

-b *bothresh*

bothresh , istek kuyruğuna ilişkin geriletme eşiği ayarını belirtir. Varsayılan 3'tür.

-c *işlem*

işlem , konuşlandırma işleminin hangi kısmının yürütüleceğini belirtir. *işlem* aşağıdaki seçeneklerden biridir:

allAxis

Bir Axis ya da Java hizmeti için tüm derleme ve ayar adımlarını gerçekleştirin.⁵

compileJava

Java hizmetini derleyin: *.java - .class*.

genAxisWsd1

WSDL oluşturun: *.class - .wsdl*.

axisDeploy

Deploy the class file: *.wsdl to .wsdd, apply .wsdd*.

genProxiestoAxis

Yetkili sunucular oluşturun: *.wsdl - .java ve .class*.

genAxisWMQBits

IBM MQ kuyruklarını, IBM MQ SOAP dinleyicilerini ve bir Axis hizmeti için tetikleyicileri ayarlayın.

allAsmx

.NET hizmeti için tüm kuruluş adımlarını gerçekleştirme⁶.

⁵ *className* ' un bir .java uzantısı varsa varsayılan değeri

⁶ *className* bir .asmx uzantısına sahipse varsayılan değer.

genAsmxWsd1

WSDL oluştur: .asmx - .wsdl.

genProxiesToDotNet

Yetkili sunucular oluşturun: .wsdl - .java, .class, .cs ve .vb.

genAsmxWMQBits

IBM MQ kuyrukları, IBM MQ SOAP dinleyicileri ve tetikleyicileri ayarlama

startWMQMonitor

Start the trigger monitor for IBM MQ SOAP services.

Not: runmqtrm, mqm kullanıcı kimliği altında çalışır. Güvenlik bir sorunsu, dinleyicilerin uygun kullanıcı kimlikleri altında başlatıldığından emin olmanız gerekir.

-i Bağlam

Bağlam, dinleyicilerin kimlik bağlamını geçip geçmeyeceğini belirtir. *Bağlam* aşağıdaki değerleri alır:

passContext

Özgün istek iletisinin kimlik bağlamını yanıt iletisine ayarlayın. SOAP dinleyici, bağlamı istek kuyruğundan kaydetme ve yanıt kuyruğuna aktarmaya ilişkin yetkiye sahip olduğunu denetler. Bağlam, bağlam kaydetmek için istek kuyruğunu açarken ve bağlam geçirmek için yanıt kuyruğu açılırken çalıştırma zamanında çekleri yapar. Gereken yetkiye sahip değilse ya da MQOPEN çağrısı başarısız olursa ve yanıt iletisi işlenmez. Yanıt iletisi, başarısız olan MQOPEN' den dönüş kodunu içeren, ölü harf üstbilgisiyle birlikte gönderilen iletiler kuyruğuna konmaktadır. Daha sonra dinleyici, sonraki gelen iletileri olağan şekilde işlemeye devam eder.

ownContext

SOAP dinleyicisi bağlam geçirmiyor. Döndürülen bağlam, özgün istek iletisini yaratan kullanıcı kimliği yerine dinleyici tarafından çalışmakta olan kullanıcı kimliğini yansıtır.

Kaynak bağlamdaki alanlar, kuyruk yöneticisi tarafından ayarlanır ve SOAP dinleyicisine göre değil.

-n numThreads

numThreads, IBM MQ SOAP dinleyicisine ilişkin oluşturulan başlangıç komut kütüklerindeki iş parçacıklarının sayısını belirtir. Varsayılan değer 10'dur. Yüksek ileti verimi varsa, bu sayıyı artırmayı düşünün.

-r

-r varolan bir istek ya da tetikleyici izleme kuyruğu tanımlarının değiştirildiğini belirtir. Tetikleme izleme kuyrukları, yalnızca -tmq belirtilirse değiştirilir. Kuyruklar standart varsayılan özniteliklerle yeniden yaratılır ve kuyruklardaki var olan iletiler silinir. -r seçeneği kullanılmazsa, var olan kuyruk tanımları değiştirilmez ve var olan iletiler silinmez. -rbelirtmeyerek, özelleştirilmiş kuyruk özniteliklerinin korunduğundan emin olun.

-s

İletişimci, IBM MQ hizmeti olarak çalışacak şekilde yapılandırın. Her ikisi de -s ve -tmq belirtilirse, konuşlandırma yardımcı programı bir hata iletisi görüntüler ve çıkar.

-tmp programName

programName, bir tetikleme izleme programının adını belirtir. **runmqtrm**ürününü kullanmaya alternatif olarak bir UNIX ya da Linux ortamında -tmp *programName* ögesini kullanın. Bu programı başlatan programlar mqm yetkisi altında çalışır.

Örneğin:

```
amqwdployWMQService -f javaDemos/service/StockQuoteAxis.java  
-tmq trigger.monitor.queue -tmp trigmon
```

-tmq queueName

queueName, bir tetikleyici izleme kuyruğu adını belirtir. IBM MQ process definitions are created to configure automatic triggering of IBM MQ SOAP listeners with the associated trigger monitor queue name. Seçenek belirtilmezse, devreye alma yardımcı programı tarafından herhangi bir tetikleme yapılandırması tanımlanmaz. Her ikisi de -s ve -tmq belirtilirse, konuşlandırma yardımcı programı bir hata iletisi görüntüler ve çıkar.

URI altyapı

Bkz. "Web hizmeti konuşlandırması için URI sözdizimi ve deęiřtirgeleri" sayfa 1455.

-v

-v , dıř komutlardan ayrıntılı çıkıřı ayarlar. Hata iletileri her zaman görüntülenir. Uyarlanmış konuşlandırma komut dosyaları yaratmak için uyarlayabileceğiniz çıkıř komutlarını -v kullanın.

-w

-w oluřturmak için WSDL ' nin stilini denetler. Varsayılan deęer, SOAP için IBM MQ iletimi önceki yayınlarıyla uyumluluk açısından rpcEnclosed' dır. Axis2 istemcisi yetkili sunucu üretimiyle uyumlu bir WSDL yaratmak için rpcLiteral öęesini kullanın. rpcEncoded , WS-I önerileriyle uyumlu deęildir.

-x transaktiflik

transactionality , dinleyici için işlemsel denetimin tipini belirtir. *transactionality* , ařaęıdaki deęerlerden birine ayarlanabilir:

onePhase

IBM MQ bir fazlı destek kullanılır. Sistem işleme sırasında başarısız olursa, istek iletisine yeniden teslim edilir. IBM MQ işlemleri, yanıt iletilerinin tam olarak bir kez yazıldıęından emin olur.

twoPhase

İki aşamalı destek kullanılır. Hizmet uygun şekilde yazılmışsa, ileti tam olarak bir kez teslim edilir, dięer kaynaklarla koordine edilir, hizmetin tek bir taahhüt halinde yürütölmesini saęlar. Bu seęenek yalnızca sunucu baę tanımları baęlantıları için geęerlidir.

none

İşlem desteęi yok. Sistem işleme sırasında başarısız olursa, istek iletisi, kalıcı olsa bile kaybedilebilir. Hizmet yürütölmebilir ya da yürütölmemiş olabilir ve yanıt, rapor ya da ölü harf iletileri yazılamayabilir ya da olmayabilir.

Devreye alma yardımcı programı, -x ve -a işaretlerinin uyumluluklarını denetler. Ayrıntılar için -a işaretinin açıklamalarına bakın.

Hatalar

On Windows, if errors are reported from **amqswsdl**, try issuing the following command to register .asmx files as services.

```
%windir%/Microsoft.NET/Framework/ version number /aspnet_regiis.exe -ir
```

Sorun genellikle, IIS 'nin kurulu olmadıęı sistemlerde ya da NET' ten sonra IIS ' nin kurulu olduęu sistemlerde ortaya çıkar. The problem is encountered when **amqswsdl** generates the .wsdl files.

Not: Dinleyici, dinleyiciye hizmetleri çağırmasına izin vermek için de gereklidir. Kendi özelleřtirilmiş devreye alma yordamlarınızı kullanırsanız, çalıştırma zamanına kadar sorunla karřılařmayabilirsiniz.

amqwdeployWMQService' tan çıkıř dosyaları

amqwdeployWMQService' un dizinlerinin ve dosya çıkıřlarının listesi

Çizelge 183. amqwdeployWMQService' tan çıkıř dosyaları			
Çıkıřlar	Tanım	Çıkıř dizini	Dosya adı
.class	Derlenmiş Java kaynak dosyası	./generated/server/server package	classname.class
.wsdl	Hizmet Tanımı	./generated	className Axis_Wmq.wsdl className DotNet_Wmq.wsdl

Çizelge 183. amqwdployWMQService 'tan çıkış dosyaları (devamı var)			
Çıkışlar	Tanım	Çıkış dizini	Dosya adı
.wsdd	Axis istemcisi ve hizmet konuşlandırma dosyaları	./	client-config.wsdd server-config.wsdd
		./generated/server/server package	className _deploy.wsdd className _undeploy.wsdd
İstemci kaynağı (.vb, .cs, .java)	.Net istemci sınırlı kod öbekleri Eksen hizmetine	./generated/client	classname AxisService.cs classname AxisService.vb
	.Net istemci sınırlı kod öbekleri. Net hizmet	./generated/client	classname DotNet.cs classname DotNet.vb
İstemci yardımcısı (.java ve .class)	Java istemcisi yetkili sunucuları .Net hizmet	./generated/server/soap/ client/ remote/dotnetService	className DotNet.class className DotNet.java className DotNetLocator.class className DotNetLocator.java className DotNetSoap12Stub.class className DotNetSoap12Stub.java className DotNetSoap_BindingStub.class className DotNetSoap_BindingStub.java className DotNetSoap_PortType.class className DotNetSoap_PortType.java
	Java client proxies to Axis service	./generated/server/soap/ client/ remote/client package	SoapServer className AxisBindingSoapStub.class SoapServer className AxisBindingSoapStub.java className Axis.class className Axis.java className AxisService.class className AxisService.java className AxisServiceLocator.class className AxisServiceLocator.java
Komut dosyaları (.cmd ve .sh)	Dinleyici komut dosyaları	/generated/server	startWMQJListener.cmd startWMQJListener.sh startWMQNListener.cmd endWMQJListener.cmd endWMQJListener.sh endWMQNListener.cmd

amqwdployWMQService için kullanım notları

amqwdployWMQService tarafından gerçekleştirilen görevleri açıklar.

Konuşlandırma yardımcı programı aşağıdaki işlemleri gerçekleştirir.

1. Yolları aşağıdaki dosyaları denetler:

- `axis.jar`.
- `WMQSOAP_HOME/java/lib/com.ibm.mq.soap.jar`.
- Windowsüzerinde, `csc.exe`

2. On Windows, uses either `%SystemRoot%\Microsoft.NET\Framework\v1.1.432` or, if the C# compiler is installed, the path to `csc.exe` as the path to the .NET Framework.

Not: Microsoft Visual Studio 2008 kurulu (Sürüm 9) varsa, `wsdl.exe`, `csc.exe` yolunda değil. Yol değişkeninize .NET çerçevesinin yolunu eklemeniz gerekir; örneğin:

```
Set Path=C:\WINDOWS\Microsoft.NET\Framework\v2.0.50727;%Path%
```

3. Yoksa, `./generated` dizinini ve gerekli alt dizinleri yaratır.

4. Java hizmetleri için, kaynağı `className.class` olarak derler.

5. WSDL oluşturur.

6. For Java services, creates deployment descriptor files `className_deploy.wsdd` and `className_undeploy.wsdd`

7. Java hizmetleri için, Axis konuşlandırma tanımlayıcısı dosyasını (`server-config.wsdd`) yaratır ya da günceller.

8. WSDL 'den Java, C# ve Visual Basic için istemci yetkili sunucuları oluşturur.

Not: Windowsüzerinde, konuşlandırma yardımcı programı, hizmetin yazıldığı dilden bağımsız olarak Visual Basic ve C# için yetkili sunucular oluşturur. WSDL ve bu WSDL 'den oluşturulan yetkili sunucular, hizmeti çağırmak için uygun URI' yi içerir:

a.

```
jms:/queue?destination=SOAPN.demos@WMQSOAP.DEMO.QM
&connectionFactory=(connectQueueManager(WMQSOAP.DEMO.QM))
&initialContextFactory=com.ibm.mq.jms.Nojndi
&targetService=StockQuoteDotNet.asmx
&replyDestination=SYSTEM.SOAP.RESPONSE.QUEUE
```

Şekil 12. Example URI in generated .NET client to call .NET service

b.

```
jms:/queue?destination=SOAPJ.demos@WMQSOAP.DEMO.QM
&connectionFactory=(connectQueueManager(WMQSOAP.DEMO.QM))
&initialContextFactory=com.ibm.mq.jms.Nojndi
&targetService=soap.server.StockQuoteAxis.java
&replyDestination=SYSTEM.SOAP.RESPONSE.QUEUE
```

Şekil 13. Üretilen .NET istemcisinde Axis 1 hizmetini çağırmak için URI örneği

9. Java yetkili sunucularını derler.

10. Hizmet isteklerini tutmak için bir IBM MQ kuyruğu, `requestQueue` oluşturur. Varsayılan kuyruk adı `SOAPJ.directory` biçimidir ya da `-u` URI seçeneğinde `requestQueue` değerini belirtebilirsiniz.

11. İstek kuyruğunu işleyen IBM MQ SOAP dinleyicilerini başlatmak için komut ve kabuk komut dosyaları yaratır.

12. `-tmq` seçeneği kullanıldıysa, konuşlandırma yardımcı programı IBM MQ SOAP dinleyicisi işlemlerini otomatik olarak tetiklemek için IBM MQ tanımlamaları yaratır.

- Konuşlandırma yardımcı programı, dinleyiciyi başlatmak için bir komut içerecek `runmqsc` DEFINE PROCESS komutunun APPLICID öznelikliğini kullanır. Komut, yerleştirmede yerleşik konuşlandırma dizininin adını içerir. APPLICID alanında en çok 256 olan uzunluk üst sınırı, konuşlandırma dizininin uzunluk üst sınırını sınırlandırıyor. Java hizmetlerine ilişkin izin sınırı aşağıdaki gibidir:

- UNIX ve Linux sistemleri: 218
- Windows: 197 eksi istek kuyruğu adının uzunluğuna sahip.

.NET hizmetleri için, dizin sınırı aşağıdaki gibidir:

- Windows: 209 minus length of the service name, minus the .asmx extension.
- Konuşlandırma yardımcı programı, APPLICID için sınırın aşıp aşılmadığını denetler. Sınır aşılsa, yardımcı program tetikleme işlemini tanımlamayı denemez. Bir hata iletisi görüntüler ve konuşlandırma işlemi gerçekleştirilmeden devreye alma işlemi başarısız olur.

Aşağıdaki örneklerde, konuşlandırma yardımcı programı tarafından oluşturulan yapılandırma ve başlatma komutları bir IBM MQ SOAP dinleyicisi başlatmak için gösterilmektedir.

```
DEFINE PROCESS(requestQueue) APPLICID(applicIDStr) REPLACE
ALTER QLOCAL (requestQueue) TRIGTYPE(FIRST) TRIGGER
PROCESS(requestQueue) INITQ(initQueueName) TRIGMPRI(0)
```

Şekil 14. Bir SOAP dinleyicisini tetikleyecek IBM MQ yapılandırma komutları.

```
applicIDStr = start "Java WMQSoapListener -requestQueue"
/min .\generated\server\startWMQJListener.cmd;
```

Şekil 15. Windows üzerinde Axis SOAP dinleyicisi başlatılıyor

```
applicIDStr = start "WMQAsmxListener -className\
/min .\generated\server\startWMQNListener.cmd;
```

Şekil 16. Starting .NET SOAP listener on Windows

```
applicIDStr = xterm -iconic -T \"Java WMQSoapListener_requestQueue\"
-e ./generated/server/startWMQJListener.sh & #
```

Şekil 17. UNIX ve Linux sistemlerinde Eksen SOAP dinleyicisini başlatma

amqwRegisterdotNet: SOAP for SOAP to .NET(SOAP için IBM MQ iletimi)

SOAP için IBM MQ iletimi .NET üzerindeki genel derleme önbellegine kaydettirin.

Amaç

amqwRegisterdotNet , IBM MQ SOAP gönderen, SOAP dinleyicisini ve WSDL işlemcisini .NET Framework 1 ya da 2 ile kaydeder.

Sözdizimi

➤ amqwRegisterdotNet ➤

Tanım

amqwRegisterdotNet , kuruluş sırasında otomatik olarak çalıştırılır. Kullandığınız .NET Framework, SOAP için IBM MQ iletimi öncesinde kurulduysa, bu işlemi yeniden çalıştırmaz gerekir. İstedik kadar çalıştırabilirsiniz. Farklı .NET Framework sürümlerine sahip SOAP için IBM MQ iletimi yeniden kaydetmek için bu programı kullanın.

Not: On Windows 2003 Server, you must also run the **aspnet_regiis** utility, even if you are not deploying to Internet Information Server (IIS). **aspnet_regiis.exe** yardımcı programının konumu, Microsoft .NET Framework ürününün farklı sürümlerine göre değişiklik gösterebilir, ancak genellikle şu konumda bulunur: %SystemRoot%\Microsoft.NET\Framework\version number\

aspnet_regiis. Birden çok sürüm kuruluysa, kullanmakta olduğunuz .NET Framework sürümü için **aspnet_regiis** 'i kullanın.

Apache yazılım lisansı

Apache License Sürüm 2.0, Ocak 2004 <http://www.apache.org/licenses/>
<https://www.apache.org/licenses/>

Apache Lisansı
Sürüm 2.0, Ocak 2004
<http://www.apache.org/licenses/>

KULLANIM, ÖLÇME VE DAĞITIM KOŞULLARI

1. Tanımlar.

"Lisans", kullanım, çoğaltma için kayıt ve koşullar anlamına gelir.
ve bu belgenin 1-9 arasındaki Bölümler tarafından tanımlanan dağıtım.

"Lisans Veren Taraf", telif hakkı sahibi veya tüzel kişi tarafından yetkilendirilmiş olan
Lisansa izin veren telif hakkı sahibi.

"Tüzel Kişi", bu tüzel kişiliğin ve tüm bunların birleşmesi anlamına gelir.
kontrol eden, kontrol edilen ya da ortak olan diğer varlıklar
o varlık ile kontrol edin. Bu tanımın amaçları açısından,
"kontrol", (i) gücü, doğrudan veya dolaylı, amaca neden olan
sözleşmeye göre veya sözleşmeye göre bu tür bir varlığın yönü veya yönetimi
aksi takdirde, veya (ii) yüzde ellisinin (%50) veya daha fazlasının mülkiyetinin
üstün paylaşımlar, ya da (iii) bu tür bir varlığın faydalı sahipliği.

"You" (ya da "Yoo"), bir kişi veya tüzel kişi anlamına gelir
Bu Lisans tarafından verilen izinleri kullanın.

"Kaynak" formu, değişiklik yapmak için tercih edilen form anlamına gelir,
Yazılım kaynak kodu, belgeler de dahil, ancak tümü bunlarla sınırlı değildir
kaynak ve yapılandırma dosyaları.

"Nesne" formu, mekanik bir biçimden kaynaklanan herhangi bir form anlamına gelir
Bir Kaynak biçiminin dönüştürülmesi ya da çevirisi de dahil olmak üzere,
derlenmiş nesne kodu, oluşturulan belgeler ile sınırlı değildir,
ve diğer ortam tiplerine dönüştürmeler.

"Çalış", ister kaynak ister, ister kaynak olsun, yetki verme işini,
Bir nesne tarafından belirtildiği şekilde, Lisans kapsamında sağlanan nesne formu
çalışmalara dahil olan veya bağlı olduğu telif hakkı bildirimi
(Ek, Ek 'te sağlanmıştır).

"Türetilen İşler", Kaynak veya Nesne içinde olsun, herhangi bir çalışma anlamına gelir
(ya da türetilen) (ya da bunlardan türetilen),
yazı düzeltmeleri, ek açıklamalar, ayrıştırılmalar ya da diğer değişiklikler
temsil etmek, bir bütün olarak, bir otoritenin orijinal bir eseri. Amaçlar için
Bu Lisansın, Türetilen İşlerin kapsadığı eserler arasında yer almayacaktır.
ayrılabilir, ya da yalnızca bağlantı (ya da ada göre bağlan) arabirimlerine,
Çalışma ve Türetilen İşler.

"Katkı" da dahil olmak üzere, herhangi bir yetki verme
İşin özgün sürümü ve yapılan değişiklikler ya da eklemeler
Bu Çalışma ya da Türetilen İşlere İlişkin Olarak Kasıtlı Olarak

Telif hakkı sahibi tarafından İş 'e dahil edilmesi için Lisans Veren Tarafa gönderildi Veya kendi adına teslim etme yetkisine sahip olan bir kişi veya tüzel kişi tarafından telif hakkı sahibi. Bu tanımın amaçları açısından, "gönderilen" herhangi bir elektronik, sözel ya da yazılı iletişim anlamına gelir Dahil ancak tümü bunlarla sınırlı olmamak üzere, Lisans Veren 'e veya temsilcilerine elektronik posta listeleri, kaynak kod kontrol sistemleri ile ilgili iletişim, ve tarafıma göre yönetilen, veya adına yönetilen takip sistemleri Çalışmayı tartışmak ve geliştirmek amacıyla lisans veren taraf, ancak belirgin biçimde işaretlenmiş veya başka bir şekilde iletişim hariç olmak Telif hakkı sahibi tarafından "Katkı Değil" olarak yazılı olarak belirlenmiştir.

"Katkıda Bulunan", Lisans Veren Taraf ve herhangi bir kişi veya Tüzel Kişi anlamına gelir Lisans veren ve Lisans Alan Taraf adına bir katkıda bulunan kişi adına daha sonra İş içinde birleştirilen.

2. Telif Hakkı Lisansı Ver. Bu koşulların kayıt ve koşullarına tabi Bu Lisans, her Contributor bu belgeyle size sürekli olarak kabul edilen bir dünya çapında, münhasır olmayan, ücretsiz, royaltisiz, geri dönülemez Çoğaltmak için telif hakkı lisansı, Türetilen İşlerin hazırlanması, genel olarak gösteri, alt lisans, ve dağıtım dağıtma İş ve bu tür Türetilen İşler Kaynak ya da Nesne formunda.

3. Patent Lisansı Verilmesi. Bu koşulların kayıt ve koşullarına tabi Bu Lisans, her Contributor bu belgeyle size sürekli olarak kabul edilen bir dünya çapında, münhasır olmayan, ücretsiz, royaltisiz, geri dönülemez (bu bölümde belirtilenler hariç) patent lisansı yapmak, yapmak, kullanımı, satmak, satmak, ithalat yapmak ve başka bir şekilde transferi yapmak, bu tür bir lisans yalnızca bu patent talep edilen patent talepleri için geçerlidir Bu tür bir Contributor tarafından, kendi adlarına doğru bir şekilde ihlal Katkı (lar) tek başına veya Katkıda Bulunan (ler) in birleşimiyle Bu tür katkıların gönderileceği İş ile birlikte gönderilmişti. Eğer herhangi bir varlıkla ilgili (a) institute patent davaları İş üzerinde hak iddia ya da karşı talep), bu işin devam ettiğini iddia eder. veya İş içinde birleştirilen bir katkı, doğrudan veya patent ihlaline katkıda bulunan, sonra herhangi bir patent lisansı Size, bu Lisans kapsamında size verilen bu Lisans kapsamında sona erecektir. bu tür bir dava açıldığı tarihten itibaren.

4. Yeniden dağıtım. Kopyaların kopyalarını çoğaltabilir ve dağıtabilirsiniz Çalışma ya da Türetilen İşlerin herhangi bir ortamda, herhangi bir ortamda ya da modifikasyonlar ve Kaynak ya da Nesne biçiminde, aşağıdaki koşulları karşıla:

(a) İşin diğer alıcılarını da vermelisiniz.

Türetilen İşler, bu Lisansın bir kopyasını içerir; ve

(b) Değiştirilen herhangi bir dosyanın öne çıkan bildirimleri taşımasına neden olmanız gerekir Dosyaları değiştirdiğini; ve

(c) İşitsel İşlerin Kaynak biçiminde, bu şekilde korunmanız gerekir.

dağıtım, tüm telif hakkı, patent, ticari marka, ve

İşin Kaynak biçiminden atıf bildirimleri,

herhangi bir parçası ile ilgili olmayan bu bildirimleri kapsam dışı bırakma

Türetilen İşler; ve

(d) İş, bu dosyanın bir parçası olarak bir "NOTICE" metin dosyası içeriyorsa

dağıtım, daha sonra dağıtmanız gereken Türetilen İşlerin

bulunan attribution bildirimlerinin okunabilir bir kopyasını içerir

Bu tür bir NOTICE dosyası içinde, bu tür bildirimler hariç
En az bir (en az) Türetilen İşlerin herhangi bir bölümüne ilişkin
Aşağıdaki yerler: NOTECE metin dosyası içinde dağıtılmış
Türetilen İşlerin bir parçası olarak; Kaynak form içinde veya
belgeler, Türetilen İşler ile birlikte sağlandıysa; ya da
Türetilen İşler tarafından oluşturulan bir görüntü biriminde, eğer ve
Bu tür üçüncü taraf bildirimleri her yerde görüntülenir. İçindekiler
NOTICE dosyası yalnızca bilgilendirme amaçlıdır ve
Lisansı değiştirmeyin. Kendi atfınızı ekleyebilirsiniz
Dağıttığınız Türetilen İşler içindeki bildirimler,
ya da NOTICE metnine ek olarak, İsteki
bu tür ek öznitelik bildirimlerinin yorumlanamaması
Lisansı değiştirdiği gibi.

Kendi telif hakkı bildiriminizi değişikliklerinize ekleyebilir ve
ek veya farklı lisans kayıt ve koşulları sağlayabilir
Değişikliklerinizin kullanılması, çoğaltılması ya da dağıtılması ya da
Kullanımınız sağlandığı gibi, tüm bu tür Türetilen İşlerin tümü için
çoğaltma ve İşin Dağıtımı, aksi takdirde
bu Lisansın içinde belirtilen koşullar.

5. Katkıların Gönderilmesi. Aksi halde aksini belirtmediysen,
Herhangi Bir Katkı, özellikle işe dahil edilmek üzere sunulmuş olan herhangi bir
Sizin tarafınızdan Lisans Veren 'e göre, bu koşullar altında
Bu Lisans, herhangi bir ek kayıt ya da koşul olmaksızın
Yukarıda belirtilenler dikkate alınmaksızın, burada hiçbir şey yerine geçemez veya değiştirilemez
Yürütmüş olabileceğiniz ayrı bir lisans sözleşmesinin koşulları
Bu tür katkılara ilişkin Lisans Veren ile birlikte.

6. Ticari markalar. Bu Lisans, ticareti kullanmak için izin vermez
Lisans Veren Tarafın adları, ticari markaları, hizmet markaları ya da ürün adları,
anlatımında makul ve adetli kullanım için gerekli olduğu durumlar dışında
Çalışmanın kaynağı ve NOTICE dosyasının içeriğini yeniden üretmenin.

7. Garantinin Reddi. Geçerli yasalar tarafından zorunlu olmadıkça veya
yazılı olarak kabul edilen Lisans Verici, İş (ve her biri) sağlar
Katkıda bulunan) bir "OLDUĞU GİBİ" ESASIYLA, "OLDUĞU GİBİ" ESASIYLA,
HERHANGİ BİR GARANTİ VEYA KOŞUL OLMAKSIZIN AÇIK YA DA
zımni, herhangi bir sınırlama olmaksızın, herhangi bir garanti ya da koşul da dahil olmak
TITLE, HAK İHLALİ YAPILMAYACAĞINA DAİR ZİMNİ GARANTİLERLE TİCARİLİK VEYA BELİRLİ BİR
AMACA UYGUNLUK İÇİN ZİMNİ GARANTİLER DE,
AMAÇLI. Yalnızca,
İşin kullanılması ya da yeniden dağıtılıp yeniden dağıtıma uygunluğunun
Bu Lisans kapsamındaki izinlerin alımınızla ilişkili riskler.

8. Sorumlulukların Sınırı. Hiçbir şekilde ve hiçbir yasal teori altında,
tort (ihmal dahil), sözleşme ya da başka bir şekilde,
uygulanabilir yasalar tarafından gerekmedikçe (kasti ve grossly gibi
ihmalkar eylemler) veya yazılı olarak kabul edilen herhangi bir Katkıda Bulunan
Herhangi bir doğrudan, dolaylı, özel de dahil olmak üzere, zararlardan sorumlu olarak size karşı sorumlu
tutu
arızı ya da sonuçta ortaya çıkan herhangi bir karakterin sonuçta ortaya çıkan zararları
Bu Lisansın sonucu ya da kullanımın sonucu ya da kullanılamaması
İş (iyi niyet kaybı için zararlar da dahil olmak, ancak tümü bunlarla sınırlı olmamak üzere)
iş durdurma sayfası, bilgisayar arızası ya da arıza, ya da herhangi bir
diğer ticari zararlar veya kayıplar), bu tür bir katkıda bulunsa bile
bu tür zararların ortaya çıkma olasılığı hakkında tavsiyede bulunmuştur.

9. Garanti veya Ek Sorumluluk Kabul Ediliyor. Yeniden dağıtılırken Çalışma ya da Türetilen İşleri, teklif etmeyi seçebilirsiniz. ve ücreti, desteği, garantisi, tazminatı kabul, veya bu konuda tutarlı olan diğer sorumluluk yükümlülükleri ve/veya hakları Lisans. Ancak, bu tür yükümlülüklerin kabul edilmesi için, yalnızca Kendi adınıza ve tek sorumluluğunuzda, sizin adınıza değil Diğer herhangi bir Contributor için ve yalnızca tazelik etmeyi kabul etmeniz de, savunmasını ve her Contributor 'ı herhangi bir yükümlülük için zararsız bir nedeni temelinde, bu tür bir Katkıda Bulunan tarafının uğradığı ya da iddia edilen iddiaların Bu tür bir garanti ya da ek yükümlülüğü kabul ettiğiniz için.

KAYIT VE KOŞULLAR

APPENDIX: How to apply the Apache License to your work.

To apply the Apache License to your work, attach the following boilerplate notice, parantez içine alınmış alanlarla "[]" kendi tanımlayıcı bilgilerinizle değiştirin. (İçerme köşeli ayraçlar!) Metin uygun şekilde yerleştirilmelidir. Dosya biçimi için açıklama sözdizimi. bunu da tavsiye ediyoruz. dosyanın dosya veya sınıf adı ve tanımı da dahil olmak üzere daha kolay için telif hakkı duyurusu olarak aynı "basılı sayfa" Üçüncü taraf arşivlerinde tanımlama.

Copyright [yyyy] [telif hakkı sahibinin adı]

Licensed under the Apache License, Version 2.0 (the "License");
bu dosyayı, Lisansın koşullarına uymadıkça kullanamazsınız.
Lisansın bir kopyasını şu adreste bulabilirsiniz:

<http://www.apache.org/licenses/LICENSE-2.0>

Geçerli yasalar tarafından zorunlu olmadıkça veya yazılı olarak kabul edilmedikçe, yazılım "OLDUĞU GİBİ" ESASIYLA, Lisans kapsamında dağıtılan "OLDUĞU GİBİ" ESASIYLA dağıtılır. AÇIK YA DA SÖZCÜKLERİ HERHANGİ BİR TÜRDEN Belirli dil yönetme izinlerine ilişkin Lisansa bakın ve Lisans kapsamındaki sınırlamalar.

MQMD SOAP ayarları

IBM MQ SOAP göndericisi ve IBM MQ SOAP dinleyicisi bir ileti tanımlayıcısı yaratır (**MQMD**). Bu konuda, kendi SOAP göndericinizi ya da dinleyicinizi yaraladığınızda, MQMD ' de ayarlamamız gereken alanlar açıklanır.

Amaç

The values set in the **MQMD** control the exchange of messages between the IBM MQ SOAP sender, the IBM MQ SOAP listener, and the SOAP client program. If you create your own SOAP sender or listener, follow the rules in [Çizelge 184 sayfa 1433](#).

Tanım

[Çizelge 184 sayfa 1433](#) , **MQMD** alanlarının IBM MQ SOAP göndericisi ve IBM MQ SOAP dinleyicisi tarafından nasıl ayarlanmakta olduğunu açıklar. Kendi göndericinizi ya da dinleyicinizi yazarsanız, bu alanları ileti alışverişi kurallarına uygun olarak ayarlamamız gerekir. IBM MQ SOAP dinleyicisi tipik IBM MQ ileti alışverişi protokollerine uygundur. If you write your own sender to work with the IBM MQ SOAP listeners, you can set different **MQMD** values.

Çizelge 184 sayfa 1433' ta, Setting (Ayar) sütunundaki değerler aşağıdaki gibi düzenlenmiştir:

İstek, bir yol

IBM MQ SOAP göndericisi tarafından yapılan ayarlar.

Yanıt, Rapor

Settings made by IBM MQ SOAP listener in response to IBM MQ SOAP sender request.

TÜMÜ

Hem IBM MQ SOAP göndericisi hem de IBM MQ SOAP dinleyicisi tarafından yapılan ayarlar.

Uyarlanmış gönderen

Kendi göndereni yazabilirsin. Tipik olarak, uyarlanmış bir gönderen standart rapor seçeneklerini geçersiz kılar.

Çizelge 184. MQMD SOAP ayarları		
Alan adı	Ayar	Değerler
<i>StrucId</i>	TÜMÜ MQMD_STRUC_ID	'MD-1' ¹
<i>Version</i>	TÜMÜ MQMD_VERSION_2	2
<i>Report</i>	TÜMÜ MQRO_NONE + MQRO_NEW_MSG_ID + MQRO_COPY_MSG_ID_TO_CORREL_ID + MQRO_EXCEPTION + MQRO_EXPIRY + MQRO_DISCARD Uyarlanmış gönderen Bkz. "Özelleştirilmiş rapor seçenekleri" sayfa 1437	52428800
<i>MsgType</i>	İstek MQMT_REQUEST Yanıt MQMT_REPLY Rapor MQMT_REPORT Tek Yönlü MQMT_DATAGRAM	MQMT_REQUEST 1 MQMT_REPLY 2 MQMT_REPORT 4 MQMT_DATAGRAM 8
<i>Expiry</i>	İstek, bir yol URI 'deki Süre bitimi seçeneği ile belirtilir. Varsayılan olarak MQEI_UNLIMITEDdeğerine ayarlanır. Yanıt İstek iletilsinde Süre bitimi değeri Rapor MQEI_UNLIMITED	MQEI_UNLIMITED -1

Çizelge 184. MQMD SOAP ayarları (devamı var)

Alan adı	Ayar	Değerler
<i>Feedback</i>	İstek, Yanıt, Tek yönlü MQFB_NONE. Rapor - Kuyruk yöneticisi tarafından oluşturulan değer, olağan kurallara göre ayarlanır. - IBM MQ SOAP Listener tarafından oluşturuldu: MQRC_BACKOUT_THRESHOLD_REACHED Birden çok deneme sayısı için geriletme eşiği aşıldı. MQRCCF_MD_FORMAT_ERROR İleti, MQRFH2 üstbilgisine sahip olarak tanınmıyor. MQRC_RFH_PARM_MISSING A required parameter, for example, SoapAction, in MQRFH2 is missing. MQRC_RFH_FORMAT_ERROR MQRFH2 temel bütünlük denetimi başarısız oldu; örneğin, iç uzunluklar bozuk. MQRC_RFH_ERROR MQRFH2 bir bütünlük denetimi geçirdi, ancak iletinin gövdesi MQFMT_NONE olarak ayarlı değil.	MQFB_NONE 0 MQRC_BACKOUT_THRESHOLD_REACHED 2362 MQRCCF_MD_FORMAT_ERROR 3023 MQRC_RFH_PARM_MISSING 2339 MQRC_RFH_FORMAT_ERROR 2421 MQRC_RFH_ERROR 2334
<i>Encoding</i>	TÜMÜ MQENC_NATIVE	Ortama göre değişir
<i>CodedCharSetId</i>	TÜMÜ UTF-8 olarak ayarla	1208
<i>Format</i>	İstek, Yanıt, Tek yönlü MQFMT_RF_HEADER_2 Rapor Kuyruk yöneticisi raporları Follows IBM MQ rules IBM MQ SOAP dinleyici raporları Özgün istek iletinin biçimi.	MQFMT_RF_HEADER_2 "MQRFH2 "
<i>Priority</i>	İstek, bir yol URI 'deki Priority (Öncelik) seçeneği ile belirtilir. Varsayılan olarak MQPRI_PRIORITY_AS_Q_DEF değerine ayarlanır. Yanıt, Rapor İstek iletinde Öncelik değeri.	MQPRI_PRIORITY_AS_Q_DEF -1

Çizelge 184. MQMD SOAP ayarları (devamı var)

Alan adı	Ayar	Değerler
<i>Persistence</i>	İstek, bir yol MQPER_PERSISTENCE_AS_Q_DEF. Yanıt, Rapor İstek iletilisinde Persistence değeri.	MQPER_PERSISTENCE_AS_Q_DEF 2
<i>MsgId</i>	İstek, bir yol Kuyruk yöneticisi tarafından oluşturuldu. Yanıt, Rapor IBM MQ SOAP gönderen kümeleri MQRO_NEW_MSG_ID ve <i>MsgId</i> üretilir.	Oluşturuldu Kuyruk yöneticisi tarafından üretilen benzersiz değer
<i>CorrelId</i>	İstek, Bir yönlü, Rapor MQCI_NONE Yanıt, Rapor IBM MQ SOAP gönderen kümeleri MQRO_COPY_MSG_ID_TO_CORREL_ID ve dinleyici, istek iletilisinden <i>MsgId</i> kopyalarını kopyalar.	MQCI_NONE 0
<i>BackoutCount</i>	TÜMÜ Kullanılmadı	0
<i>ReplyToQ</i>	İstek URI 'deki replyDestination seçeneği tarafından belirtilir. Varsayılan olarak SYSTEM.SOAP.RESPONSE.QUEUEdeğer ine ayarlanır. Yanıt, Bir yönlü, Rapor Boş bırakıldı	
<i>ReplyToQMgr</i>	TÜMÜ Alan boş kaldı	Kuyruk yöneticisi tarafından üretilir; bkz. Yanıtla kuyruk ve kuyruk yöneticisi .
<i>UserIdentifier</i>	İstek, Bir yönlü, Rapor Boş bırakıldı Yanıt Dinleyiciye sağlanan -i <i>passContext</i> seçeneğine bağlıdır ve dinleyici, dinleyici altında çalışmakta olan yetkiyi içerir.	İstek, Bir yönlü, Rapor Kuyruk yöneticisi tarafından oluşturuldu; bkz. “UserIdentifier (MQCHAR12)” sayfa 447 Yanıt <i>Değişken</i>
<i>AccountingToken</i>	TÜMÜ MQACT_NONE	MQACT_NONE Dizgi ya da boşluk boş değerli Kuyruk yöneticisi tarafından ayarlanır; bkz. “AccountingToken (MQBYTE32)” sayfa 407
<i>ApplIdentityData</i>	TÜMÜ Yok	Boş değerli dize ya da boşluk ²

Çizelge 184. MQMD SOAP ayarları (devamı var)

Alan adı	Ayar	Değerler
<i>PutApplType</i>	TÜMÜ MQAT_NO_CONTEXT	MQAT_NO_CONTEXT 0 Kuyruk yöneticisi tarafından oluşturulan değer; bkz. “PutApplTipi (MQUZE)” sayfa 435.
<i>PutApplName</i>	TÜMÜ Yok	Kuyruk yöneticisi tarafından oluşturulan değer; bkz. “PutApplAd (MQCHAR28)” sayfa 434.
<i>PutDate</i>	TÜMÜ Yok	Kuyruk yöneticisi tarafından oluşturulan değer; bkz. “PutDate (MQCHAR8)” sayfa 436.
<i>PutTime</i>	TÜMÜ Yok	Kuyruk yöneticisi tarafından oluşturulan değer; bkz. “PutTime (MQCHAR8)” sayfa 437.
<i>ApplOriginData</i>	TÜMÜ Yok	Boş değerli dize ya da boşluk ²
<i>GroupId</i>	İstek, Bir yönlü, Rapor MQGI_NONE Yanıt Alan istek iletisinden kopyalanır	Boş Değerler
<i>MsgSeqNumber</i>	İstek, Bir yönlü, Rapor Kullanılmadı Yanıt Alan istek iletisinden kopyalanır	Kuyruk yöneticisi tarafından oluşturulur; bkz. Kuyruğun fiziksel sırası.
<i>Offset</i>	İstek, Bir yönlü, Rapor Kullanılmadı Yanıt Alan istek iletisinden kopyalanır	0
<i>MsgFlags</i>	İstek, Bir yönlü, Rapor MQMF_NONE Yanıt Alan istek iletisinden kopyalanır	MQMF_NONE 0 Bkz. “MsgFlags (MQUZE)” sayfa 424
<i>OriginalLength</i>	İstek, Bir yönlü, Yanıt MQOL_UNDEFINED Rapor Özgün istek iletisinin uzunluğu	MQOL_UNDEFINED -1

Çizelge 184. MQMD SOAP ayarları (devamı var)

Alan adı	Ayar	Değerler
Notlar: 1. ~ simgesi tek bir boş karakteri temsil eder. 2. Boş değerli dizgi ya da boşluk değeri, C içindeki boş değerli dizeyi ve diğer programlama dillerindeki boş karakterleri gösterir.		

Özelleştirilmiş rapor seçenekleri

Kendi SOAP göndericinizi yazabilir ve sağlanan dinleyicilerle kullanabilirsiniz. Genellikle, rapor seçeneklerinin seçimini değiştirmek için bir gönderici yazabilirsiniz. IBM MQ SOAP dinleyicileri, aşağıdaki listelerde açıklandığı gibi, rapor seçeneklerinin çoğu birleşimlerini destekler.

- IBM MQ SOAP dinleyicileri tarafından desteklenen rapor seçenekleri:
 - MQRO_EXCEPTION
 - MQRO_EXCEPTION_WITH_DATA
 - MQRO_EXCEPTION_WITH_FULL_DATA
 - MQRO_DEAD_LETTER_Q
 - MQRO_DISCARD_MSG
 - MQRO_NONE
 - MQRO_NEW_MSG_ID
 - MQRO_PASS_MSG_ID
 - MQRO_COPY_MSG_ID_TO_CORREL_ID
 - MQRO_PASS_CORREL_ID
- Kuyruk yöneticisi tarafından desteklenen rapor seçenekleri:
 - MQRO_COA
 - MQRO_COA_WITH_DATA
 - MQRO_COA_WITH_FULL_DATA
 - MQRO_COD
 - MQRO_COD_WITH_DATA
 - MQRO_COD_WITH_FULL_DATA
 - MQRO_EXPIRATION
 - MQRO_EXPIRATION_WITH_DATA
 - MQRO_EXPIRATION_WITH_FULL_DATA
- Aşağıdaki rapor seçenekleri, IBM MQ SOAP dinleyicileri tarafından desteklenmez.
 - MQRO_PAN
 - MQRO_NAN

The behavior of IBM MQ SOAP listeners in response to combinations of MQRO_EXCEPTION_* and MQRO_DISCARD is described in [Çizelge 185 sayfa 1438](#).

MQRO_EXCEPTION_* gösterimi, MQRO_EXCEPTION, MQRO_EXCEPTION_WITH_DATA ya da MQRO_EXCEPTION_WITH_FULL_DATA' nin kullanımını belirtir.

Çizelge 185. Listener behavior resulting from MQRO_EXCEPTION_* and MQRO_DISCARD settings

	MQRO_DISCARD etkinleştirildi	MQRO_DISCARD Etkin değil
MQRO_EXCEPTION_* etkinleştirildi	Varsayılan davranış. Gerekliyse, rapor iletileri otomatik olarak oluşturulur ve iptal edilen özgün istek atılır. Bir rapor iletilisi yanıt kuyruğuna iade edilemediyse, rapor iletilisi ölü mektup kuyruğuna gönderilir.	Gerekliyse, rapor iletileri otomatik olarak oluşturulur ve özgün ileti, ölü mektup kuyruğuna gönderilir. Rapor iletilisi yanıt kuyruğuna iade edilemediyse, bu ileti de ölü mektup kuyruğuna gönderilir. Bu durumda, başarısız olan istek için iki adet ölü harf kuyruğu girişi vardır.
MQRO_EXCEPTION_* Etkin değil	Gelen biçim tanınmadığında ya da geri çıkış denemelerinin sayısı aşıldığında, rapor iletileri otomatik olarak oluşturulmaz. İleti, ölü mektup kuyruğuna gönderilmez. İstemcinin inceleyebileceği ve özgün istek iletilisinin kaybolduğu bir bildirim döndürülmez.	Gelen biçim tanınmadığında ya da geri çıkış denemelerinin sayısı aşıldığında, rapor iletileri otomatik olarak oluşturulmaz. Ancak, bir rapor başka bir şekilde oluşturulabildiğinde, özgün istek iletilisi, ölü mektup kuyruğuna yazılır.

MQRFH2 SOAP ayarları

IBM MQ SOAP göndericileri ve dinleyicileri, aşağıdaki ayarlarla bir MQRFH2 almayı ya da almayı bekler.

Amaç

IBM MQ SOAP göndericileri, özellikleri IBM MQ JMStarafından oluşturulan <usr> klasörüne ekler. Özellikler, hedef ortamdaki SOAP taşıyıcısının gerektirdiği bilgileri içerir. “Özellik sözdizimi” sayfa 1438 , bir MQRFH2' ye eklendiklerinde özelliklerin sözdizimini açıklar. Açıklama bir MQRFH2 üstbilgisi için bkz. MQRFH2 -Kural ve biçimlendirme üstbilgisi 2.

Özellik sözdizimi

```

➤ <usr> — <contentType> — text/xml; charset=utf-8 — </contentType> — <endpointURL> — URI ➤

➤ </endpointURL> ————— <targetService> — = — serviceName 1 — </targetService> ————— ➤

➤ ————— <soapAction> — = — serviceName 2 — </soapAction> ————— <transportVersion> — 1 ➤

➤ </transportVersion> — </usr> ➤

```

Notlar:

¹ targetService is required for .NET Framework 1 or 2, and not used on Axis 1.4.

² soapAction is optional for .NET Framework 1 or 2, and not used on Axis 1.4.

Parametreler

contentType

contentType her zaman text/xml; charset=utf-8 dizisini içerir.

endpointURL

Bkz. “Web hizmeti konuşlandırması için URI sözdizimi ve değiştirgeleri” sayfa 1455.

targetService

⁷Axis 'te, *serviceName* , bir Java hizmetinin tam olarak nitelenmiş adıdır, örneğin: `targetService=javaDemos.service.StockQuoteAxis`. `targetService` belirtilmemişse, varsayılan Eksen mekanizması kullanılarak bir hizmet yüklenir.

⁸.NET üzerinde, *serviceName* , konuşlandırma dizininde bulunan bir .NET hizmetinin adıdır; örneğin: `targetService=myService.asmx`. .NET ortamında, `targetService` parametresi tek bir IBM MQ SOAP dinleyicinin birden çok hizmete ilişkin istekleri işleyebilmesini olanaklı kılar. Bu hizmetler, aynı dizinden devreye alınmalı.

soapAction

transportVersion

transportVersion her zaman 1olarak ayarlanır.

Örnek

Örnekte, bir MQRFH2 ve aşağıdaki SOAP iletisi gösterilmektedir. Klasörlerin uzunlukları ondalık olarak gösterilir.

Not: URI 'deki& , & olarak kodlanır

```

52464820 00000002 0000002B0 00000001 RFH/ 0002 1208 0001
000004B8 20202020 20202020 00000000 1208 ? ? ? ? ? ? ? 0000
000004B8 1208
32 <mcd>
<Msd>jms_bytes</Msd>
</mcd>?
208 <jms>
<Dst>queue://queue://SOAPJ.demos</Dst>
<Rto>queue://WMQSOAP.DEMO.QM/SYSTEM.SOAP.RESPONSE.QUEUE</Rto>
<Tms>1157388516465</Tms>
<Cid>ID:0000000000000000000000000000000000000000000000000000000000000000</Cid>
<Dlv>1</Dlv>
</jms>
400 <usr>
<contentType>text/xml; charset=utf-8</contentType>
<transportVersion>1</transportVersion>
<endpointURL>
jms:/queue?destination=SOAPJ.demos@WMQSOAP.DEMO.QM
&amp;connectionFactory=connectQueueManager(WMQSOAP.DEMO.QM)
clientConnection(localhost%25289414%2529)
clientChannel(TESTCHANNEL)
&amp;replyDestination=SYSTEM.SOAP.RESPONSE.QUEUE
&amp;initialContextFactory=com.ibm.mq.jms.Nojndi
</endpointURL>
</usr>
<?xml version="1.0" encoding="UTF-8"?>
<soapenv:Envelope
xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsd="https://www.w3.org/2001/XMLSchema"
xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance">
<soapenv:Body>
<ns1:getQuote
soapenv:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
xmlns:ns1="soap.server.StockQuoteAxis_Wmq">
<in0 xsi:type="xsd:string">XXX</in0>
</ns1:getQuote>
</soapenv:Body>
</soapenv:Envelope>

```

runivt: SOAP kurulumu doğrulama sınaması için IBM MQ iletimi

SOAP için IBM MQ iletimi ile bir kuruluş doğrulama test takımı (IVT) sağlanıyor. **runivt**, bir dizi gösterim uygulamasını çalıştırır ve kuruluştan sonra ortamın doğru bir şekilde ayarlanmasını sağlar.

⁷ YalnızcaJava hizmeti

⁸ Yalnızca.NET hizmeti

Amaç

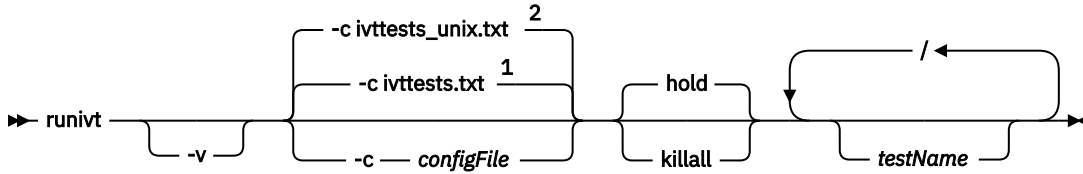
runivt komutu, istemcilerden hizmetlere Web hizmeti istekleri göndermek için SOAP için IBM MQ iletimi ile birlikte verilen örnek programları kullanır. Axis 1.4, .NET Framework 1 ve .NET Framework 2 için sınamalar çalıştırır. Sınamalar, bir test komut dosyası dosyasında yapılandırılır. Windows için varsayılan test komut dosyası, Java ile .NET istemcileri ve hizmetleri arasındaki testlerin bir birleşimini çalıştırır.

Tanım

runivt , kendi dizininden çalıştırılmalıdır.

Komut, dinleyicileri farklı bir komut penceresinde başlatır. Bu nedenle, komutu UNIX ve Linux sistemlerindeki bir X Window System oturumundan çalıştırmanız gerekir.

runivt sözdizimi



Notlar:

¹ Windows' ta varsayılan

² UNIX ve Linux sistemlerinde varsayılan değer

runivt parametreler

-v

Ayrıntı kipi. Konsola fuller hata iletileri yazın.

-c configFile

Çalıştırılacak sınamaları tanımlayan bir yapılanış dosyası. Varsayılan olarak, Windows, UNIX ya da Linux sistemleriyle birlikte sağlanan varsayılan yapılandırma dosyası kullanılır.

hold

Sınamalar tamamlandıktan sonra dinleyiciyi çalıştıktan sonra bırak

killall

Sınamalar tamamlanınca dinleyiciyi sona erdir

testName

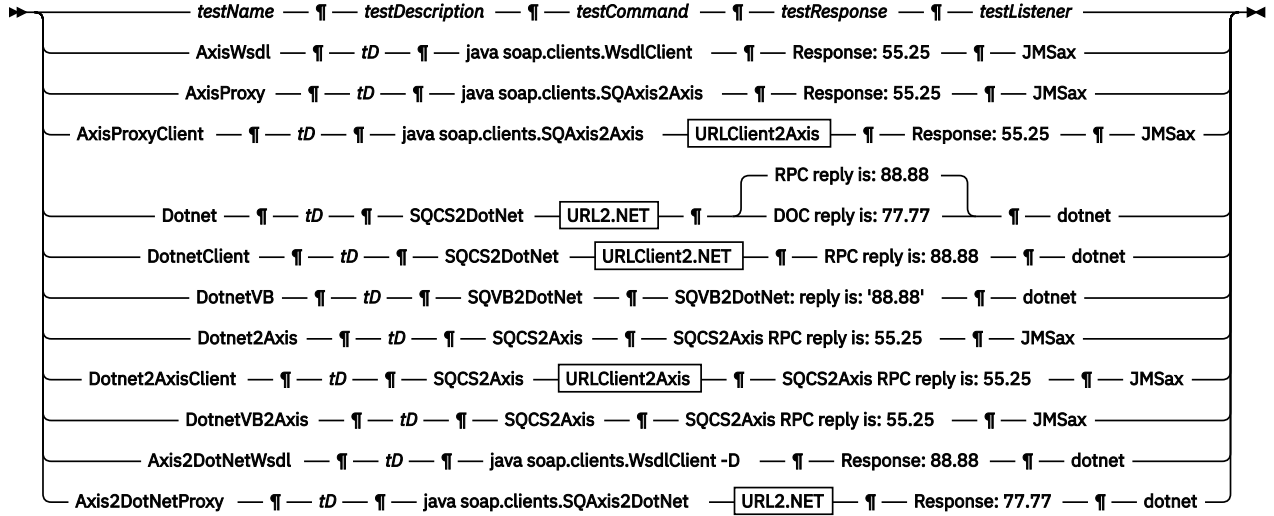
Çalıştırılacak testlerin boşlukla ayrılmış listesi. Sınama adları yapılanış kütüğünden seçilir. Ad belirtilmezse, yapılandırma dosyasındaki tüm sınamalar çalıştırılır.

Yapılandırma dosyası

Her yapılanış dosyası parametresi, dosyanın ayrı bir satırdır. Her parametre grubu arasında boş bir satır bırakın.

ivttests.txt parametre dosyasındaki parametreler listelenir.

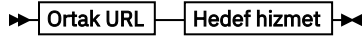
configFile sözdizimi



URLClient2Axis



URL2.NET



URLClient2.NET



Ortak URL

► jms:/queue?destination=SOAPJ.demos@WMQSOAP.DEMO.QM — & — initialContextFactory — = —>

► com.ibm.mq.jms.Nojndi — & — connectionFactory — = —>

► connectQueueManager — (— WMQSOAP.DEMO.QM —) —>

İstemci bağlantısı

► clientConnection — (— localhost%25289414WMQSOAP.DEMO.QM%2529 —) — clientChannel —>

► (— TESTCHANNEL —) —>

Hedef hizmet

► & — targetService — = — StockQuoteDotNet.asmx —>

configFile parametreleri

testName

Testin adı. **runivt** komutundaki *testName* komutunu kullanın

testDescription

Testle ilgili Documentation

testCommand

İstemci isteğini yapmak için **runivt** komutu tarafından yürütülen komut.

testResponse

Konsola istemci isteği tarafından döndürülen tam yanıt dizgisi. Başarılı olmak için test için *testResponse* , gerçek yanıtı uygun olmalıdır.

testListener

SOAP isteğini işlemek için **runivt** tarafından başlatılan IBM MQ SOAP dinleyicisinin adı. **dotnet** ve **JMSax** , sağlanan dinleyicilerin eş anlamlıdır, **amqwSOAPNETListener** ve **SimpleJavaListener**.

Örnekler

```
runivt
```

Şekil 18. tüm varsayılan sınamaları çalıştır

```
runivt dotnet
```

Şekil 19. Varsayılan testlerden belirli bir testi çalıştırma

```
runivt -c mytests.txt
```

Şekil 20. özel sınamalar kümesi çalıştırma

İlgili bilgiler

[SOAP için IBM MQ iletimi doğrulanıyor](#)

SOAP için IBM MQ iletimi üzerinden Web hizmetleri güvenliğini sağlayın

SOAP için IBM MQ iletimi kullanan Web hizmetlerini iki yoldan biriyle güvenli hale getirebilirsiniz. İstemci ile sunucu arasında bir SSL kanalı yaratın ya da Web hizmetleri güvenliğini kullanın.

SOAP için SSL ve IBM MQ iletimi

SOAP için IBM MQ iletimi, SSL kipinde çalışmak üzere yapılandırılmış istemci kanalıyla kullanılmak üzere belirlenebilecek birçok SSL seçeneği sağlar. Seçenekler, .NET ile Java ortamları arasında farklılık gösterir. IBM MQ SOAP göndericileri ve dinleyicileri, yalnızca kendi ortamları için geçerli olan SSL seçeneklerini işler. Uygulanamayacak seçenekleri yoksayar.

.NET istemcileri için `sslCipherSpec` seçeneğinin varlığı ya da yokluğu ve Java istemcileri için `sslCipherSuite` seçeneği, SSL 'nin kullanılıp kullanılmadığını belirler. URI içinde seçenek belirtilmezse, varsayılan olarak SSL kullanılmıyorsa ve diğer tüm SSL seçenekleri yoksayılır. Belirtilenler dışında tüm SSL seçenekleri isteğe bağlıdır.

IBM MQ istemcileri için, URI ya da kanal tanımlama çizelgesinde SSL özniteliklerini ayarlayın. On the server, set the attributes using the facilities of IBM MQ.

Varsayılan olarak, kanal üzerinde SSL etkinleştirilirken standart IBM MQ SSL seçeneği `SSLCAUTH`Hayarlanır. SSL iletişiminin başlayabilmesi için, istemcilerin kendilerini doğrulaması gerekir. `SSLCAUTH` belirlenmezse, istemci kimlik doğrulaması olmadan SSL iletişimleri kurulur.

İstemcilerin kimliklerini doğrulamak için, anahtar havuzunda, kuyruk yöneticisi tarafından kabul edilebilir bir sertifikaya sahip olması gerekir. Ek güvenlik için IBM MQ kanalları, yalnızca sınırlı bir listeden sertifikaları kabul edecek şekilde yapılandırılabilir. Bu liste, kanalın eş adı özniteliğine ilişkin sertifikana ilişkin ayırt edici ad denetlenerek kısıtlanır.

Javakullanırsanız, bir IBM MQ SOAP istemcisinden gelen ilk SSL bağlantısı, aşağıdaki SSL parametrelerinin düzeltilmesine neden olur. Aynı istemci işlemi kullanılarak, sonraki bağlantılarda da aynı değerler kullanılır:

- `sslKeyDeposu`
- `sslKeyStorePassword`
- `sslTrustDeposu`
- `sslTrustStorePassword`
- `sslFipsGerekli`

- sslLDAPCRLservers

Bu deęiřtirgelerin bu istemciden gelen sonraki baęlantılarda yrrlęe girmesinin etkisi tanımsız.

.NETkullanırsanız, bir IBM MQ SOAP istemcisinden gelen ilk SSL baęlantısı, ařaęıdaki SSL parametrelerinin dzeltilmesine neden olur. Aynı istemci iřlemi kullanılarak, sonraki baęlantılarda da aynı deęerler kullanılır:

- sslKeyHavuzu
- sslCryptoDonanımı
- sslFipsGerekli
- sslLDAPCRLservers

Bu deęiřtirgelerin bu istemciden gelen sonraki baęlantılarda yrrlęe girmesinin etkisi tanımsız. Tm SSL baęlantıları etkin deęilse ve yeni bir SSL baęlantısı yapılırsa bu parametreler sıfırlanır.

Ařaęıdaki zellikler sistem zellikleri olarak da belirtilebilir:

- sslKeyDeposu
- sslKeyStorePassword
- sslTrustDeposu
- sslTrustStorePassword

Bunlar hem sistem zellikleri, hem de URI ' de belirtilirse ve deęerler farklı olursa, konuřlandırma yardımcı programı bir uyarı grntler. URI deęerleri ncelikli olarak uygulanır.

İlgili bařvurular

IBM MQ Web hizmetleri URI 'sındaki SSL baęlantı reticisi deęiřtirgeleri

IBM MQ Web hizmetleri URI 'sındaki baęlantı reticisi seęenekleri listesine SSL seęenekleri ekleyin.

İlgili bilgiler

MQI istemcisinde alıřtırma sırasında yalnızca FIPS onaylı CipherSpecs ' in kullanıldığını belirtme

[Federal Information Processing Standards \(FIPS\) for UNIX, Linux and Windows](#)

IBM MQ Web hizmetleri URI 'sındaki SSL baęlantı reticisi deęiřtirgeleri

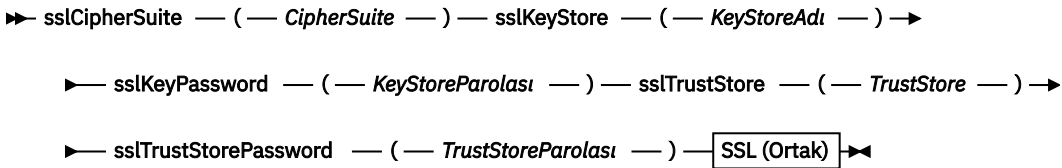
IBM MQ Web hizmetleri URI 'sındaki baęlantı reticisi seęenekleri listesine SSL seęenekleri ekleyin.

Ama

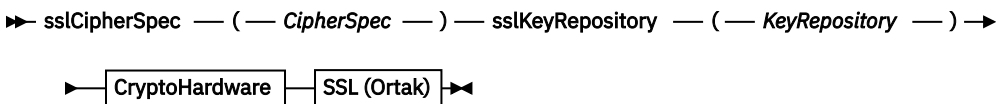
Bir IBM MQ Web hizmetleri istemcisi ile web hizmetini barındıran kuyruk yneticisi arasında gvenli bir baęlantı kullanabilirsiniz. SSL seęenekleri, SSL ' nin IBM MQ MQI client-server kanal baęlantısında nasıl yapılandırıldığı denetler.

Szdizimi řeması

SSL (Java)

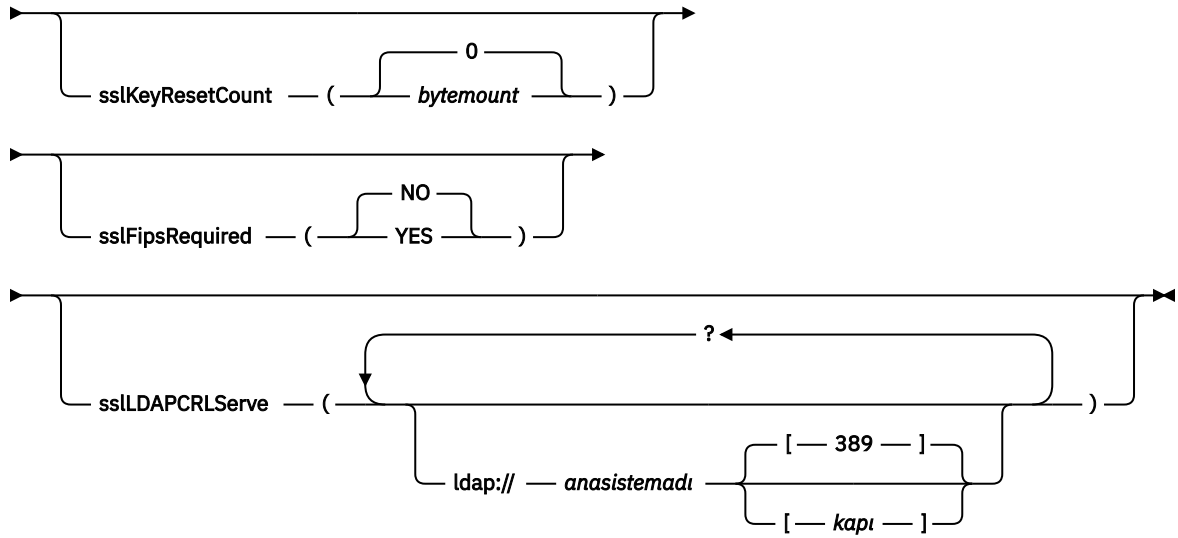


SSL (.NET)



SSL (Ortak)

➡ sslCipherPeerName — (— *PeerName* —) →



CryptoHardware

➡ sslCryptoHardware — = — *PKCS #11 Yolu ve dosya adı* — ; — *PKCS #11 belirteç etiketi* — ; →

➡ *PKCS #11 belirteç parolası* — ; — *simetrik şifre ayarı* — ; ➡

Gerekli SSL parametreleri (Ortak)

sslPeerName (*peerName*)

peerName , kanalda kullanılan sslPeerName ' i belirtir.

Gerekli SSL parametreleri (Java)

sslCipherSuite (*CipherSuite*)

CipherSuite , kanalda kullanılan sslCipherSuite ' i belirtir. İstemcinin belirttiği CipherSuite , sunucu bağlantı kanalında belirtilen CipherSuite ile eşleşmelidir.

sslKeyStore (*KeyStoreAdı*)

KeyStoreAdı , kanalda kullanılan sslKeyStoreName ' i (StoreName) belirtir. Anahtar deposu, istemciyi sunucu için doğrulamak için kullanılan istemcinin özel anahtarını tutar. SSL bağlantısı anonim istemci bağlantılarını kabul edecek şekilde yapılandırıldıysa, anahtar deposu isteğe bağlıdır.

sslKeyStorePassword (*KeyStoreParolası*)

KeyStoreParolası , kanalda kullanılan sslKeyStorePassword değerini belirtir.

sslTrustStore (*TrustStoreAdı*)

TrustStoreAdı , kanalda kullanılan sslTrustStoreName ' i belirtir. Güvenilir depo, sunucunun genel sertifikasını ya da sunucuyu istemcide kimlik doğrulaması yapmak için temel zincirini tutar. Sunucunun kimliğini doğrulamak için bir sertifika yetkilisinin kök sertifikası kullanılırsa, güvenilirlik deposu isteğe bağlıdır. Java' ta kök sertifikalar JRE sertifika deposunda tutulur, cacerts.

sslTrustStorePassword (*TrustStoreParolası*)

TrustStoreParolası , kanalda kullanılan sslTrustStorePassword değerini belirtir.

Gerekli SSL parametreleri (.NET)

sslCipherSpec (*CipherSpec*)

CipherSpec , kanalda kullanılan sslCipherSpec ' i belirtir. Seçenek belirlenirse, istemci kanalında SSL kullanılır.

sslKeyRepository (KeyRepository)

KeyRepository , SSL anahtarlarının ve sertifikalarının depolandığı kanalda kullanılan sslCipherBelirtimi değerini belirtir. *KeyRepository* kök biçiminde belirtilir; yani, dosya adı ile birlikte tam yol, ancak dosya uzantısı atlanır. sslKeyRepository ayarının etkisi, MQCONNX çağrısında bulunan **MQSCO** yapısındaki KeyRepository alanının ayarlandığı şekilde aynıdır.

İsteğe bağlı SSL parametreleri (.NET)

sslCryptoHardware (CryptoHardware)

CryptoHardware , kanalda kullanılan sslCryptoDonanımı ' nın kullanılacağını belirtir. Bu alan için olası değerler ve ayarın etkisi, MQCONNX üzerindeki **MQSCO** yapısının CryptoHardware alanı için aynıdır.

İsteğe bağlı SSL parametreleri (Ortak)

sslKeyResetCount (bytcount)

bytcount , SSL güvenlik dizgisi anahtarının yeniden görüşülebilmesi için önce bir SSL kanalından geçirilen bayt sayısını belirtir. SSL anahtarlarının yeniden anlaşılmasını devre dışı bırakmak için alanı kaldırın ya da sıfırın değerine ayarlayın. Sıfır, bazı ortamlarda desteklenen tek değerdir, bkz. [IBM MQ classes for Java](#) içindeki gizli anahtarın yeniden anlaşılması. sslKeyResetCount ayarının etkisi, MQCONNX çağrısındaki **MQSCO** yapısındaki KeyResetCount alanının ayarlandığı şekilde aynıdır.

sslFipsRequired (fipsCertified)

fipsCertified specifies whether *CipherSpec* or *CipherSuite* must use FIPS-certified cryptography in IBM MQ on the channel. *fipsCertified* ayarının etkisi, MQCONNX çağrısındaki **MQSCO** yapısının FipsRequired alanının ayarlandığı şekilde aynıdır.

sslLDAPCRLServers (LDAPServerList)

LDAPServerList , Sertifika İptal Listesi denetimi için kullanılacak LDAP sunucularının bir listesini belirtir.

SSL etkin istemci bağlantıları için, *LDAPServerList* , CRL (Sertifika İptal Listesi) denetimi için kullanılacak LDAP sunucularının bir listesidir. Kuyruk yöneticisi tarafından sağlanan sertifika, listelenen LDAP CRL sunucularından birine yönelik olarak denetlenir; bulunduysa, bağlantı başarısız olur. Her bir LDAP sunucusu, biri için bağlantı kurulumuna kadar sırayla denetlenir. Sunuculardan herhangi birine bağlanmak olanaksızsa, sertifika reddedilir. Bu sertifikalardan biri için bağlantı başarıyla kurulduktan sonra, bu LDAP sunucusunda bulunan CRL ' lere bağlı olarak sertifika kabul edilir ya da reddedilir.

LDAPServerList boşsa, kuyruk yöneticisine ait olan sertifika, bir Sertifika İptal Listesi 'ne göre denetlenmez. Sağlanan LDAP URI ' leri listesi geçerli değilse, bir hata iletisi görüntülenir. The effect of setting this field is the same as that of including MQAIR records and accessing them from an **MQSCO** structure on an MQCONNX.

İlgili başvurular

[SOAP için SSL ve IBM MQ iletimi](#)

SOAP için IBM MQ iletimi, SSL kipinde çalışmak üzere yapılandırılmış istemci kanalıyla kullanılmak üzere belirlenebilecek birçok SSL seçeneği sağlar. Seçenekler, .NET ile Java ortamları arasında farklılık gösterir. IBM MQ SOAP göndericileri ve dinleyicileri, yalnızca kendi ortamları için geçerli olan SSL seçeneklerini işler. Uygulanamayacak seçenekleri yoksayar.

İlgili bilgiler

[MQI istemcisinde çalıştırma sırasında yalnızca FIPS onaylı CipherSpecs ' in kullanıldığını belirtme](#)
[Federal Information Processing Standards \(FIPS\) for UNIX, Linux and Windows](#)

SimpleJavaListener: IBM MQ SOAP Listener for Axis 1.4

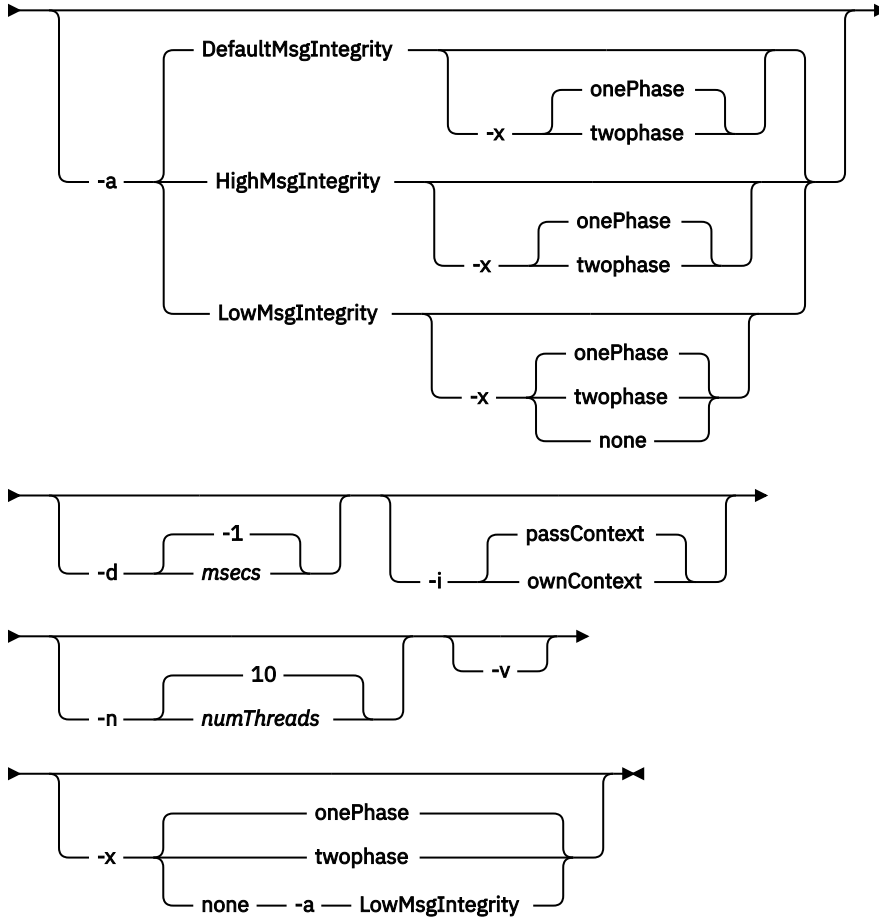
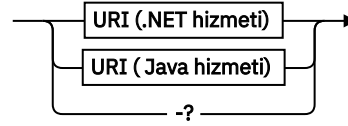
Syntax and parameters for the IBM MQ SOAP listener for Axis 1.4.

Amaç

IBM MQ SOAP dinleyicisi for Axis 1.4' i başlatır.

Java

➔ java — com.ibm.mq.soap.transport.jms.SimpleJavaListener



Gerekli parametreler

URI altyapı

Bkz. “Web hizmeti konuşlandırması için URI sözdizimi ve deęiřtirgeleri” sayfa 1455.

-?

Komutun nasıl kullanıldığını açıklayan yardım metni yazdırın.

İsteęe baęlı parametreler

-a integrityOption

integrityOption , başarısız bir istek iletisini ölü mektup kuyruęına koymanın mümkün deęilse, IBM MQ SOAP dinleyicilerinin davranışını belirtir. *integrityOption* , ařaęıdaki deęerlerden birini alabilir:

DefaultMsgIntegrity

Kalıcı olmayan iletiler için, dinleyici bir uyarı iletisi görüntüler ve atılmakta olan özgün iletiyle yürütölmeye devam eder. Kalıcı iletiler için, bu ileti bir hata iletisi görüntüler, istek iletisini yedekler, böylece istek kuyruęunda kalır ve çıkılır. DefaultMsgBütönlüęü applies if the -a option is omitted, or if *integrityOption* is not specified.

LowMsgIntegrity

Hem kalıcı hem de kalıcı olmayan iletiler için dinleyici bir uyarı görüntüler ve iletiyi yürötmeye, yürötmeye devam eder.

HighMsgIntegrity

Hem kalıcı hem de kalıcı olmayan iletiler için, dinleyici bir hata iletisi görüntüler, istek iletisini yedekler, böylece istek kuyruğunda kalır ve çıkılır.

Devreye alma yardımcı programı, -x ve -a işaretlerinin uyumluluklarını denetler. -x none belirtilirse, -a LowMsgIntegrity (Bütünlük Ayarı) belirtilmelidir. İşaretler uyumsuz olursa, konuşlandırma yardımcı programı bir hata iletisiyle çıkar ve herhangi bir devreye alma adımı atılmıyorsa.

-d msec

msecs , herhangi bir iş parçacığıda istek iletileri alındıysa, IBM MQ SOAP dinleyicisinin canlı olarak kalabileceği süreyi (milisaniye) belirtir. *msecs* için -1değeri belirlendiyse, dinleyici süresiz olarak etkin kalır.

-i Bağlam

Bağlam , dinleyicilerin kimlik bağlamını geçip geçmeyeceğini belirtir. *Bağlam* aşağıdaki değerleri alır:

passContext

Özgün istek iletisinin kimlik bağlamını yanıt iletisine ayarlayın. SOAP dinleyici, bağlamı istek kuyruğundan kaydetme ve yanıt kuyruğuna aktarmaya ilişkin yetkiye sahip olduğunu denetler. Bağlam, bağlam kaydetmek için istek kuyruğunu açarken ve bağlam geçirmek için yanıt kuyruğu açılırken çalıştırma zamanında çekleri yapar. Gereken yetkiye sahip değilse ya da MQOPEN çağrısı başarısız olursa ve yanıt iletisi işlenmez. Yanıt iletisi, başarısız olan MQOPEN' den dönüş kodunu içeren, ölü harf üstbilgisiyle birlikte gönderilen iletiler kuyruğuna konmaktadır. Daha sonra dinleyici, sonraki gelen iletileri olağan şekilde işlemeye devam eder.

ownContext

SOAP dinleyicisi bağlam geçirmiyor. Döndürülen bağlam, özgün istek iletisini yaratan kullanıcı kimliği yerine dinleyici tarafından çalışmakta olan kullanıcı kimliğini yansıtır.

Kaynak bağlamdaki alanlar, kuyruk yöneticisi tarafından ayarlanır ve SOAP dinleyicisine göre değil.

-n numThreads

numThreads , IBM MQ SOAP dinleyicisine ilişkin oluşturulan başlangıç komut kütüklerindeki iş parçacıklarının sayısını belirtir. Varsayılan değer 10'dur. Yüksek ileti verimi varsa, bu sayıyı artırmayı düşünün.

-v

-v , dış komutlardan ayrıntılı çıkışı ayarlar. Hata iletileri her zaman görüntülenir. Uyarlanmış konuşlandırma komut dosyaları yaratmak için uyarlayabileceğiniz çıkış komutlarını -v kullanın.

-w serviceDirectory

serviceDirectory , web hizmetini içeren dizindir.

-x transaktiflik

transactionality , dinleyici için işlemsel denetimin tipini belirtir. *transactionality* , aşağıdaki değerlerden birine ayarlanabilir:

onePhase

IBM MQ bir fazlı destek kullanılır. Sistem işleme sırasında başarısız olursa, istek iletisine yeniden teslim edilir. IBM MQ işlemleri, yanıt iletilerinin tam olarak bir kez yazıldığından emin olur.

twoPhase

İki aşamalı destek kullanılır. Hizmet uygun şekilde yazılmışsa, ileti tam olarak bir kez teslim edilir, diğer kaynaklarla koordine edilir, hizmetin tek bir taahhüt halinde yürütülmesini sağlar. Bu seçenek yalnızca sunucu bağ tanımları bağlantıları için geçerlidir.

none

İşlem desteği yok. Sistem işleme sırasında başarısız olursa, istek iletisi, kalıcı olsa bile kaybedilebilir. Hizmet yürütülmeyebilir ya da yürütülmemiş olabilir ve yanıt, rapor ya da ölü harf iletileri yazılamayabilir ya da olmayabilir.

Devreye alma yardımcı programı, -x ve -a işaretlerinin uyumluluklarını denetler. Ayrıntılar için -a işaretinin açıklamalarına bakın.

Java Örnek

```
java com.ibm.mq.soap.transport.jms.SimpleJavaListener
-u "jms:/queue?destination=myQ&connectionFactory=()
&initialContextFactory=com.ibm.mq.jms.NoJndi"
-n 20
```

IBM MQ SOAP dinleyicileri

IBM MQ SOAP dinleyicisi, URI 'de hedef olarak belirtilen kuyruktan gelen bir SOAP isteğini okur. Bu, istek iletisinin biçimini denetler ve daha sonra Web hizmetleri altyapısını kullanarak bir Web hizmetini çağırır. IBM MQ SOAP dinleyicisi, URI 'deki yanıt hedefi kuyruğunu kullanarak bir Web hizmetinden herhangi bir yanıt ya da hata döndürür. Yanıt kuyruğuna IBM MQ raporları döndürür.

Terim dinleyicisi, standart Web hizmetleri algısında burada kullanılır. Bu, **runmq1sr** komutu tarafından çağrılan standart IBM MQ dinleyicisinden farklıdır.

Tanım

Java SOAP dinleyicisi bir Java sınıfı olarak uygulanır ve 1.4Axis 'i kullanarak hizmetler çalıştırılır. .NET dinleyici, bir konsol uygulamasıdır ve .NET Framework 1 ya da .NET Framework 2 hizmetlerini çalıştırır. For .NET Framework 3 services, use the IBM MQ custom channel for Microsoft Windows Communication Foundation (WCF).

Konuşlandırma yardımcı programı, Java ya da .NET SOAP dinleyicilerini otomatik olarak başlatmak için komut dosyaları yaratır. Bir SOAP Listener, **amqSOAPNETListener** komutu kullanılarak ya da SimpleJavaListener sınıfı çağrılarak el ile başlatılabilir. You can configure the IBM MQ SOAP listener to be started as an IBM MQ service by setting the -S option in the deployment utility. Diğer bir seçenek olarak, dinleyicileri tetiklemeyi kullanarak başlatın ya da konuşlandırma yardımcı programı tarafından oluşturulan başlangıç ve bitiş dinleyicisi komut dosyalarını kullanın. Tetiklemeyi otomatik olarak yapılandırmak için el ile tetiklemeyi yapılandırabilir ya da -tmq ve -tmp konuşlandırma seçeneklerini kullanabilirsiniz. Bir dinleyiciyi, istek kuyruğunu GET (DISABLED) olarak ayarlayarak sona erdirebilirsiniz.

Çizelge 186. Konuşlandırma yardımcı programı tarafından oluşturulan komut dosyaları			
Web Hizmeti Altyapısı	UNIX ve Linux sistemleri	Windows Java	Windows.NET
Dinleyici başlat	startWMQJListener.sh	startWMQJListener.cmd	startWMQNListener.cmd
Dinleyiciyi durdur	endWMQJListener.sh	endWMQJListener.cmd	endWMQNListener.cmd
Dinleyici hizmetini tanımla	defineWMQJListener.sh	defineWMQJListener.cmd	defineWMQNListener.cmd

IBM MQ SOAP dinleyicisi, SOAP iletisinden SOAP altyapısına endpointURL ve soapAction alanlarını geçirir. Dinleyici, Web Hizmetleri altyapısı aracılığıyla hizmeti çağırır ve yanıt için bekler. Dinleyici endpointURL ve soapActionögesini doğrulamaz. The fields are set by the IBM MQ SOAP sender from the data that is provided in the URI set by a SOAP client.

Dinleyici yanıt iletisini oluşturur ve bunu istek iletisi URI 'sında sağlanan yanıt hedefine gönderir. Ayrıca, dinleyici yanıt iletisinde ilinti tanıtıcısını istek iletisinde bulunan rapor seçeneğine göre ayarlar. İstek iletisinden süre bitimi, kalıcılık ve öncelik ayarlarını döndürür. Dinleyici, bazı durumlarda istemcilere rapor iletileri de gönderir.

SOAP isteğinde biçim hataları varsa, dinleyici yanıt hedef kuyruğunu kullanarak istemciye bir rapor iletisi döndürür. Bir rapor istendiye, kuyruk yöneticisi yanıt hedef kuyruğunu kullanarak istemciye rapor iletilerini de döndürür. Bir dizi olaya yanıt olarak, yanıt kuyruğuna tam rapor iletileri yazılır:

- Bir istisna.
- İleti süre bitimi.
- İstek iletinin biçimi tanınmadı.
- **MQRFH2** üstbilgisinin bütünlük denetimi hatası.
- Ana ileti gövdesinin biçimi MQFMT_NONEdeğil.
- IBM MQ SOAP dinleyicisi isteği işlerken backout/yeniden deneme eşiği aşılr.

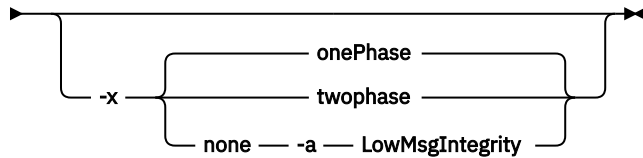
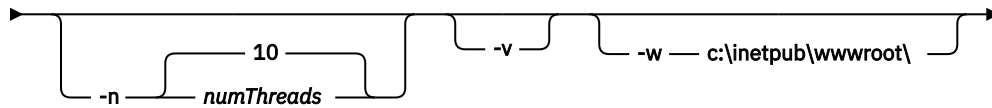
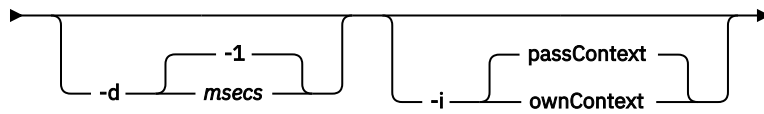
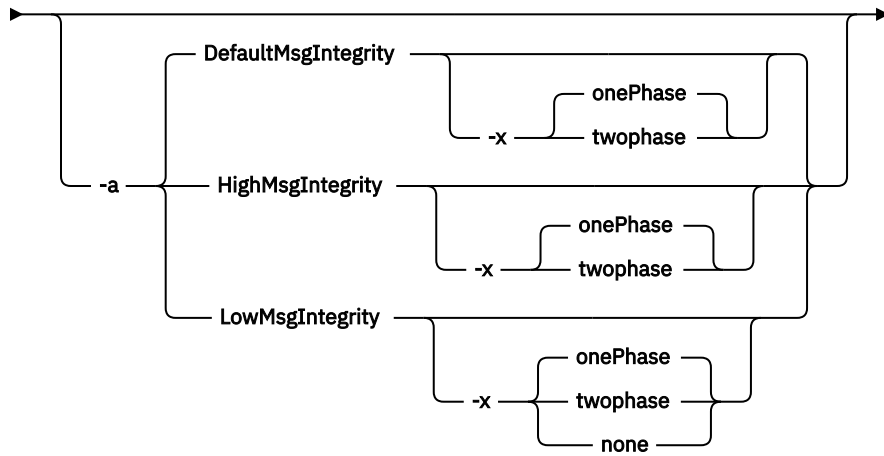
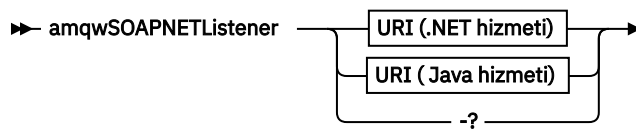
IBM MQ SOAP gönderen kümeleri MQRO_EXCEPTION_WITH_FULL_DATA ve MQRO_EXPIRATION_WITH_FULL_DATA rapor seçeneklerini ayarlar. IBM MQ SOAP göndericisi tarafından ayarlanan rapor seçenekleri sonucunda, rapor ileti, kaynak istek iletinin tamamını içerir. IBM MQ SOAP göndericisi, bir rapor ileti döndürüldükten sonra iletinin atılmasına neden olan MQRO_DISCARD seçeneğini de ayarlar. Rapor seçenekleri gereksinimlerinizi karşılamazsa, farklı MQRO_EXCEPTION ve MQRO_DISCARD rapor seçeneklerini kullanmak için kendi göndericilerinizi yazın. SOAP isteği, MQRO_DISCARD ayarlanmamış farklı bir gönderici tarafından gönderilirse, başarısız olan ileti çıkış kuyruğuna (DLQ) yazılır.

Dinleyici bir rapor ileti oluşturursa, ancak raporu gönderme işleminde başarısız olursa, rapor ileti DLQ 'ya gönderilir. DLQ işleyicinizin bu iletileri doğru şekilde işlediğinden emin olun.

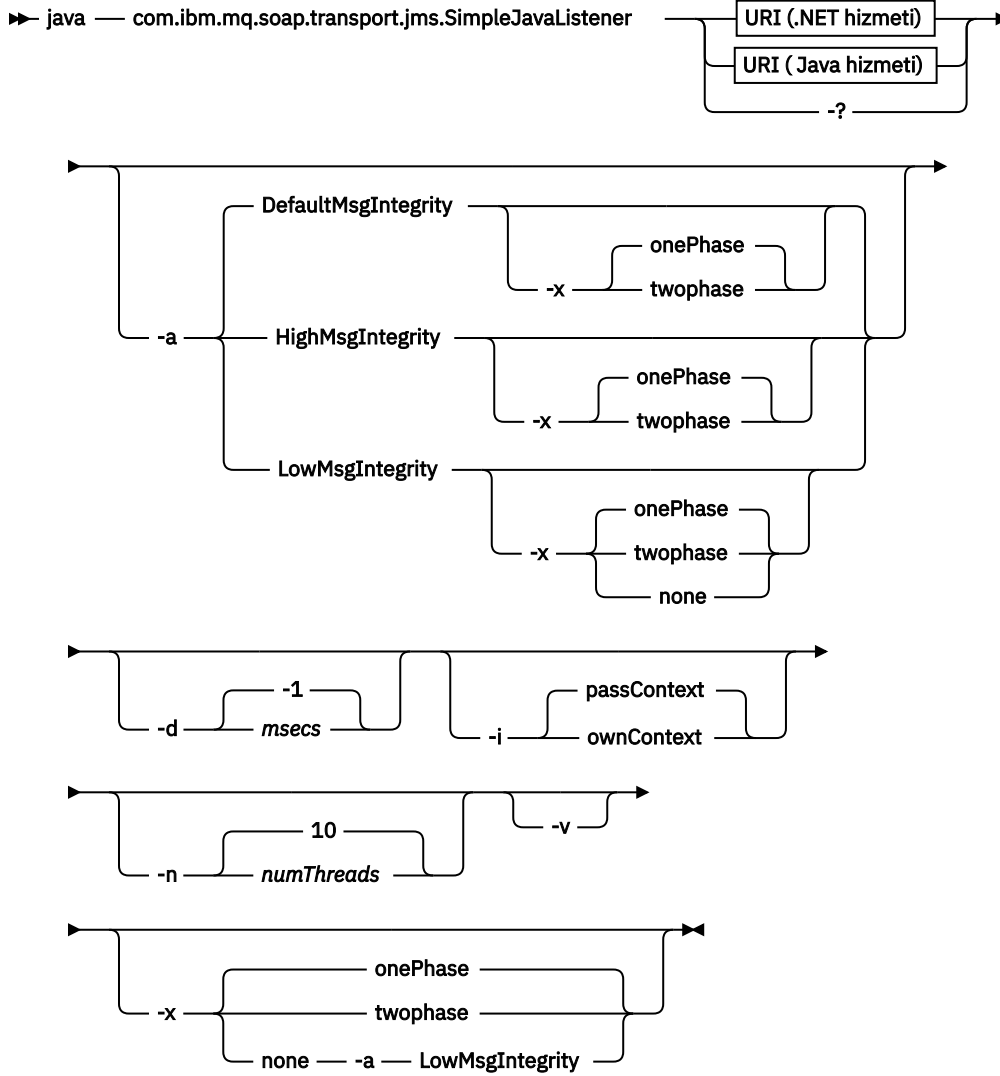
Bir hata, ölü-mektup kuyruğuna yazma girişimi sırasında IBM MQ hata günlüğüne bir ileti yazıyorsa gerçekleşir. Dinleyicinin daha fazla iletiyi işlemeye devam edip etmeyeceğini, hangi ileti kalıcılığı ve işlem seçeneklerinin seçilmesine bağlı olarak değişir. Dinleyici tek aşamalı işlemsel kipte çalışıyorsa ve kalıcı olmayan bir istek iletinin işliyorsa, özgün ileti atılır. IBM MQ SOAP dinleyicisi yürütülmeye devam eder. İstek ileti kalıcıysa, istek ileti istek kuyruğuna ve dinleyici çıkışlarına geriiletir. İstek kuyruğu, yanlışlıkla tetiklenen bir yeniden başlatma işlemini önlemek için, get-inhicted to get-inhibid

Sözdizimi şeması

.NET



Java



Gerekli parametreler

URI altyapı

Bkz. “Web hizmeti konuşlandırması için URI sözdizimi ve değıştirmeleri” sayfa 1455.

-?

Komutun nasıl kullanıldığını açıklayan yardım metni yazdırın.

İsteğe bağlı parametreler

-a integrityOption

integrityOption , başarısız bir istek iletisini ölü mektup kuyruğuna koymanın mümkün değilse, IBM MQ SOAP dinleyicilerinin davranışını belirtir. *integrityOption* , aşağıdaki değerlerden birini alabilir:

DefaultMsgIntegrity

Kalıcı olmayan iletiler için, dinleyici bir uyarı iletisi görüntüler ve atılmakta olan özgün iletiyle yürütülmeye devam eder. Kalıcı iletiler için, bu ileti bir hata iletisi görüntüler, istek iletisini yedekler, böylece istek kuyruğunda kalır ve çıkılır. DefaultMsgBütünlüğü applies if the -a option is omitted, or if *integrityOption* is not specified.

LowMsgIntegrity

Hem kalıcı hem de kalıcı olmayan iletiler için dinleyici bir uyarı görüntüler ve iletiyi yürütmeye, yürütmeye devam eder.

HighMsgIntegrity

Hem kalıcı hem de kalıcı olmayan iletiler için, dinleyici bir hata iletisi görüntüler, istek iletisini yedekler, böylece istek kuyruğunda kalır ve çıkılır.

Devreye alma yardımcı programı, -x ve -a işaretlerinin uyumluluklarını denetler. -x none belirtilirse, -a LowMsgIntegrity (Bütünlük Ayarı) belirtilmelidir. İşaretler uyumsuz olursa, konuşlandırma yardımcı programı bir hata iletisiyle çıkar ve herhangi bir devreye alma adımı atılmıyorsa.

-d msec

msecs , herhangi bir iş parçacığıda istek iletileri alındıysa, IBM MQ SOAP dinleyicisinin canlı olarak kalabileceği süreyi (milisaniye) belirtir. *msecs* için -1değeri belirlendiyse, dinleyici süresiz olarak etkin kalır.

-i Bağlam

Bağlam , dinleyicilerin kimlik bağlamını geçip geçmeyeceğini belirtir. *Bağlam* aşağıdaki değerleri alır:

passContext

Özgün istek iletisinin kimlik bağlamını yanıt iletisine ayarlayın. SOAP dinleyici, bağlamı istek kuyruğundan kaydetme ve yanıt kuyruğuna aktarmaya ilişkin yetkiye sahip olduğunu denetler. Bağlam, bağlam kaydetmek için istek kuyruğunu açarken ve bağlam geçirmek için yanıt kuyruğu açılırken çalıştırma zamanında çekleri yapar. Gereken yetkiye sahip değilse ya da MQOPEN çağrısı başarısız olursa ve yanıt iletisi işlenmez. Yanıt iletisi, başarısız olan MQOPEN' den dönüş kodunu içeren, ölü harf üstbilgisiyle birlikte gönderilen iletiler kuyruğuna konmaktadır. Daha sonra dinleyici, sonraki gelen iletileri olağan şekilde işlemeye devam eder.

ownContext

SOAP dinleyicisi bağlam geçirmiyor. Döndürülen bağlam, özgün istek iletisini yaratan kullanıcı kimliği yerine dinleyici tarafından çalışmakta olan kullanıcı kimliğini yansıtır.

Kaynak bağlamdaki alanlar, kuyruk yöneticisi tarafından ayarlanır ve SOAP dinleyicisine göre değil.

-n numThreads

numThreads , IBM MQ SOAP dinleyicisine ilişkin oluşturulan başlangıç komut kütüklerindeki iş parçacıklarının sayısını belirtir. Varsayılan değer 10'dur. Yüksek ileti verimi varsa, bu sayıyı artırmayı düşünün.

-v

-v , dış komutlardan ayrıntılı çıkışı ayarlar. Hata iletileri her zaman görüntülenir. Uyarlanmış konuşlandırma komut dosyaları yaratmak için uyarlayabileceğiniz çıkış komutlarını -v kullanın.

-w serviceDirectory

serviceDirectory , web hizmetini içeren dizindir.

-x transaktiflik

transactionality , dinleyici için işlemsel denetimin tipini belirtir. *transactionality* , aşağıdaki değerlerden birine ayarlanabilir:

onePhase

IBM MQ bir fazlı destek kullanılır. Sistem işleme sırasında başarısız olursa, istek iletisine yeniden teslim edilir. IBM MQ işlemleri, yanıt iletilerinin tam olarak bir kez yazıldığından emin olur.

twoPhase

İki aşamalı destek kullanılır. Hizmet uygun şekilde yazılmışsa, ileti tam olarak bir kez teslim edilir, diğer kaynaklarla koordine edilir, hizmetin tek bir taahhüt halinde yürütülmesini sağlar. Bu seçenek yalnızca sunucu bağ tanımları bağlantıları için geçerlidir.

none

İşlem desteği yok. Sistem işleme sırasında başarısız olursa, istek iletisi, kalıcı olsa bile kaybedilebilir. Hizmet yürütülmeyebilir ya da yürütülmemiş olabilir ve yanıt, rapor ya da ölü harf iletileri yazılamayabilir ya da olmayabilir.

Devreye alma yardımcı programı, -x ve -a işaretlerinin uyumluluklarını denetler. Ayrıntılar için -a işaretinin açıklamalarına bakın.

.NET Örnek

```
amqwSOAPNETlistener
-u "jms:/queue?destination=myQ&connectionFactory=()
&targetService=myService&initialContextFactory=com.ibm.mq.jms.Nojndi"
-w C:/wmqsoap/demos
-n 20
```

Java Örnek

```
java com.ibm.mq.soap.transport.jms.SimpleJavaListener
-u "jms:/queue?destination=myQ&connectionFactory=()
&initialContextFactory=com.ibm.mq.jms.Nojndi"
-n 20
```

SOAP gönderen için IBM MQ iletimi

Axis ve .NET Framework 1 ve .NET Framework 2 için gönderen sınıfları sağlanır. Gönderen bir SOAP isteğini oluşturur ve onu bir kuyruğa koyar, daha sonra yanıt kuyruğundan bir yanıt okuuncaya kadar öbekler. Bir SOAP istemcisinden farklı URI ' leri aktararak sınıfların davranışını değiştirebilirsiniz. For .NET Framework 3 use IBM MQ custom channel for Microsoft Windows Communication Foundation (WCF).

Amaç

IBM MQ SOAP gönderici, bir SOAP isteğini Web hizmetini bir IBM MQ istek kuyruğuna çağırmak için yerleştirir. Gönderen, **MQRFH2** üstbilgisindeki alanları URI ' de belirtilen seçeneklere göre ya da varsayılan değerlere göre ayarlar.

URI seçeneklerini kullanarak mümkün olan bir gönderenin davranışını değiştirmeniz gerekiyorsa, kendi göndericinizi yazın. Your sender can work with the IBM MQ transport for SOAP listeners, or with other SOAP environments. Your sender must construct SOAP messages in the format defined by IBM MQ. Biçim, IBM MQ SOAP dinleyicisi tarafından desteklenir ve WebSphere Application Server ve CICStarafından sağlanan SOAP dinleyicileridir. Gönderenin IBM MQ istekçisine ilişkin kuralları izlemesi gerekir. IBM MQ SOAP dinleyicisi, yanıt ve rapor iletilerini döndürür. See "MQMD SOAP ayarları" sayfa 1432 for details how to set the report options in the **MQMD**. Rapor seçenekleri, IBM MQ SOAP dinleyicisinin döndürdüğü rapor iletilerini denetler.

Tanım

IBM MQ SOAP Java gönderen, jms: URI öneki için Axis anasistem ortamına kayıtlı. Gönderen, org.apache.axis.handlers.BasicHandlers sınıfından türetilen com.ibm.mq.soap.transport.jms.WMQSenders sınıfında uygulanmaktadır. If the Axis host environment detects a jms: URI prefix it invokes the com.ibm.mq.soap.transport.jms.WMQSender class. İleti, yanıt kuyruğundan bir yanıt okuyana kadar, iletiyi yerleştirdikten sonra sınıf öbekleri. Zamanaşımı süresi içinde yanıt alınmazsa, gönderen bir kural dışı durum yayınlar. Zamanaşımı süresi içinde bir yanıt alınır, Axis çerçevesini kullanarak istemciye yanıt iletilisi döndürülür. İstemci uygulamanızın bu yanıt iletilerini işleyebilmesi gerekir.

Microsoft.NET Framework 1 ve .NET Framework 2 hizmetleri için, IBM MQ SOAP gönderen, System.Net.WebRequest ve System.Net.WebRequestCreate' den türetilen IBM.WMQSOAP.MQWebRequests sınıfında uygulanır. If .NET Framework 1 or .NET Framework 2 detects a jms: URI prefix it invokes the IBM.WMQSOAP.MQWebRequest class. Gönderen, yanıt kuyruğundan yanıt iletilisini okumak ve istemciye geri döndürmek için bir MQWebResponse nesnesi oluşturur.

com.ibm.mq.soap.transport.jms.WMQSender , bir son sınıftır ve IBM.WMQSOAP.MQWebRequest mihlülüdür. Alt sınıflar oluşturarak bunların davranışını değiştiremezsiniz.

Parametreler

Bir Web hizmeti SOAP istemcisinde IBM MQ SOAP göndericisinin davranışını denetleyebilmek için URI 'yi ayarlayın. Konuşlandırma yardımcı programı, konuşlandırma yardımcı programına sağlanan URI seçeneklerini içeren Web hizmeti istemcisi sınırlı kod öbekleri yaratır.

SOAP göndereni için IBM MQ SOAP iletimi ile bir kanal tanımlama çizelgesi kullanın

İstemci bağlantı kanalı tanımlaması, Web hizmeti URI 'sının ConnectionFactory öznitelikteki bağlantı özelliklerini ayarlamaya yönelik bir alternatiftir. Bağlantı özellikleri şunlardır: clientChannel, clientConnection, ve SSL parametreleri.

Tanım

İstemci bağlantıları tanımlayarak istemci kanal tanımlama çizelgesini yaratın. Bir Web hizmeti istemcisi farklı kuyruk yöneticilerine bağlıysa bile, bağlantı çizelgesindeki tüm bağlantıları tek bir kuyruk yöneticisinde yaratın. Bağlantı çizelgesinin varsayılan adı ve yeri *queue manager directory/@ipcc/AMQCLCHL.TAB* olur.

`com.ibm.mq.soap.transport.jms.mqchlurl` sistem özelliğini ayarlayarak, bağlantı tablolarının konumunu bir Java istemcisine geçirin.

MQCHLIB ve MQCHLTAB ortam değişkenlerini ayarlayarak, bağlantı tablolarının konumunu bir .NET istemcisine geçirin.

Web hizmeti URI 'sinin ConnectionFactory özniteliğinde hem kanal bağlantısı çizelgesi hem de kanal bağlantısı değiştiricileri sağlayabilirsiniz. ConnectionFactory içinde ayarlanan değerler, kanal tanımlama çizelgesindeki değerlere göre öncelikli olarak uygulanır.

Using a channel definition table in Java

```
java -Dcom.ibm.msg.client.config.location=file:/C:/mydir/myjms.config MyAppClass
```

Şekil 21. Starting Java client using a configuration file

```
com.ibm.mq.soap.transport.jms.mqchlurl=file:/C:/ibm/wmq/qmgrs/QM1/@ipcc/AMQCLCHL.TAB
```

Şekil 22. *myjms.config*

İşlemler

İletişimci başlatırken Web hizmetlerini farklı bir şekilde çalıştırmak için -x seçeneğini kullanın. Hizmet URI 'sında persistence (kalıcılık) seçeneğini ayarlayarak iletilerin bütünlüğünü seçin.

Web hizmetleri

İletişimci başlatırken Web hizmetlerini farklı bir şekilde çalıştırmak için -x seçeneğini kullanın. .NET Framework 1 ve 2 'de SOAP dinleyicisi Microsoft Transaction Coordinator (MTS) kullanır. 1.4 ekseninde, SOAP dinleyicisi kuyruk yöneticisi eşgüdümlü işlemleri kullanır.

Web hizmeti istemcileri

SOAP göndericileri hareket işleme değil.

IBM MQ Bağ Tanımları

SOAP gönderici için bağ tanımlama tipini ayarlayabilirsiniz. Bir IBM MQ sunucusu uygulaması olarak ya da bir istemci uygulaması olarak bağlanabilir. SOAP göndericiyi .NET üzerinde XA istemcisi olarak da bağlayabilirsiniz.

İleti kalıcılığı

URI 'deki Persistence (Persistence) seçeneğini ayarlayarak kalıcılık düzeyini seçin.

Web Hizmeti İşlemleri

SOAP göndereni işlem olmadığı için, Web hizmeti işlemlerini kullanabilirsiniz. Kendi SOAP göndericinizi yazarsanız ve Web hizmeti işlemlerini kullanmayı amaçlıyorsanız, işlemsel SOAP göndereni yaratmayın. İstek iletisini gönderemez ve aynı hareket içinde yanıt iletisini alabilirsiniz. Gönderme ve alma işlemi Web hizmeti işlemi tarafından eşgüdümlü olarak gerçekleştirilmemelidir.

Web hizmeti konuşlandırması için URI sözdizimi ve değiştirelileri

Bir IBM MQ Web hizmetini konuşlandırmak için kullanılan sözdizimi ve değiştireliler bir URI ' de tanımlanır. Konuşlandırma yardımcı programı, Web hizmetinin adına dayalı olarak bir varsayılan URI oluşturur. Konuşlandırma yardımcı programına değiştirge olarak kendi URI ' nizi tanımlayarak varsayılan değeri geçersiz kılabilirsiniz. Devreye alma yardımcı programı, oluşturulan Web hizmeti istemcisi sınırlı kod öbeklerindeki URI ' yi içerir.

Amaç

Bir web hizmeti URI (Universal Resource Identifier; Evrensel Kaynak Tanıtıcısı) kullanılarak belirtilir. Sözdizimi şemasında, SOAP için IBM MQ iletiminde desteklenen URI ' yi belirtir. URI, hedef hizmetlere erişmek için kullanılan IBM MQ ' e özgü SOAP parametrelerini ve seçeneklerini denetler. URI, .NET, Apache Axis 1, WebSphere Application Server, CICStarafından barındırılan Web hizmetleriyle uyumludur.

Tanım

URI, devreye alma yardımcı programı tarafından oluşturulan Web hizmeti istemcisi sınıflarına dahil edilir. İstemci, URI ' yı bir IBM MQ iletisinde IBM MQ SOAP Gönderen 'a iletir. URI, hem IBM MQ SOAP Gönderen, hem de IBM MQ SOAP dinleyicisi tarafından gerçekleştirilen işlemleri denetler.

Sözdizimi

URI sözdizimi aşağıdaki gibidir:

```
jms:/queue?name=value&name=value...
```

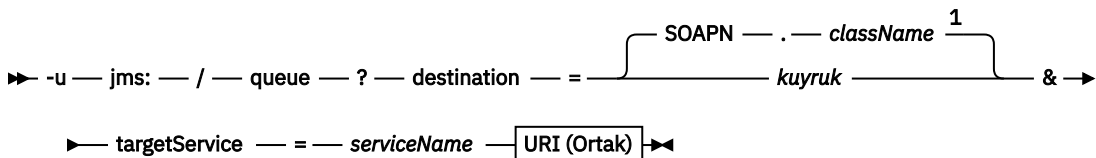
Burada name , bir parametre adı ve value uygun bir değerdir ve name = value ögesi, ikinci ve sonraki oluşumlardan önce ve işareti (&) ile herhangi bir sayıda yinelenebilir.

Parametre adları büyük/küçük harf duyarlıdır, IBM MQ nesnelerinin adları gibi. Herhangi bir parametre bir kereden fazla belirtilirse, parametrenin son oluşumu yürürlüğe girer. İstemci uygulamaları, parametrenin başka bir kopyasını URI ' ye ekleyerek, oluşturulmuş bir parametreyi geçersiz kılabilir. Herhangi bir ek tanınmayan parametre varsa, bunlar yoksayılır.

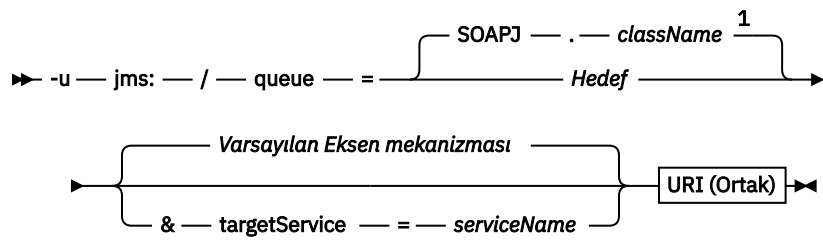
Bir URI ' yi bir XML dizisinde saklıyorsa, ve işareti karakterini & olarak göstermeniz gerekir. Benzer şekilde, bir URI bir komut dosyasında kodlandıysa, tersi durumda kabuk tarafından yorumlanacak & gibi çıkış karakterlerine özen gösteriniz.

Sözdizimi şeması

URI (.NET hizmeti)

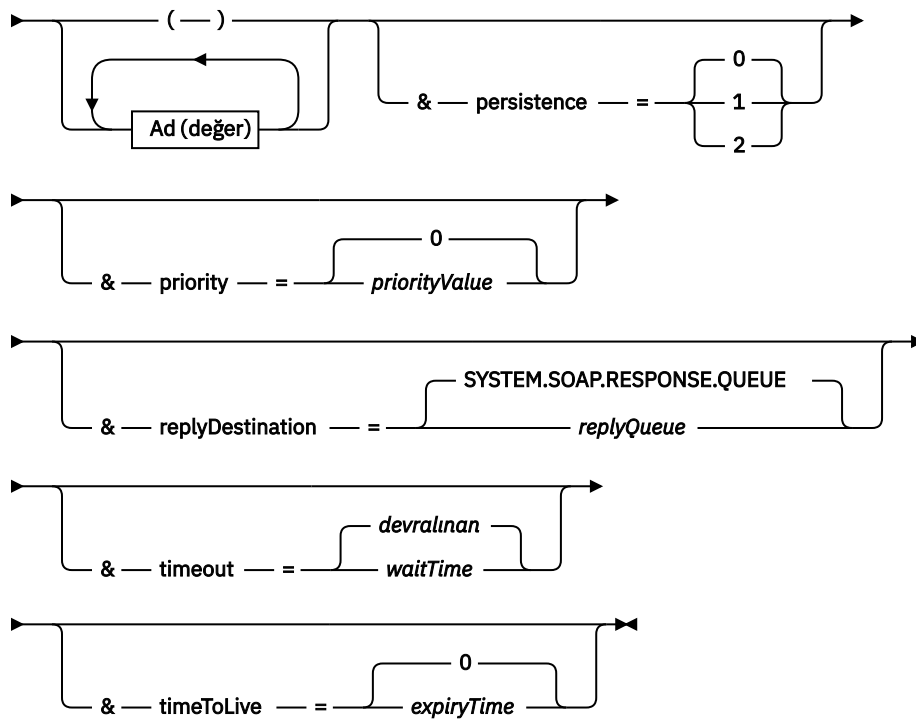


URI (Java hizmeti)

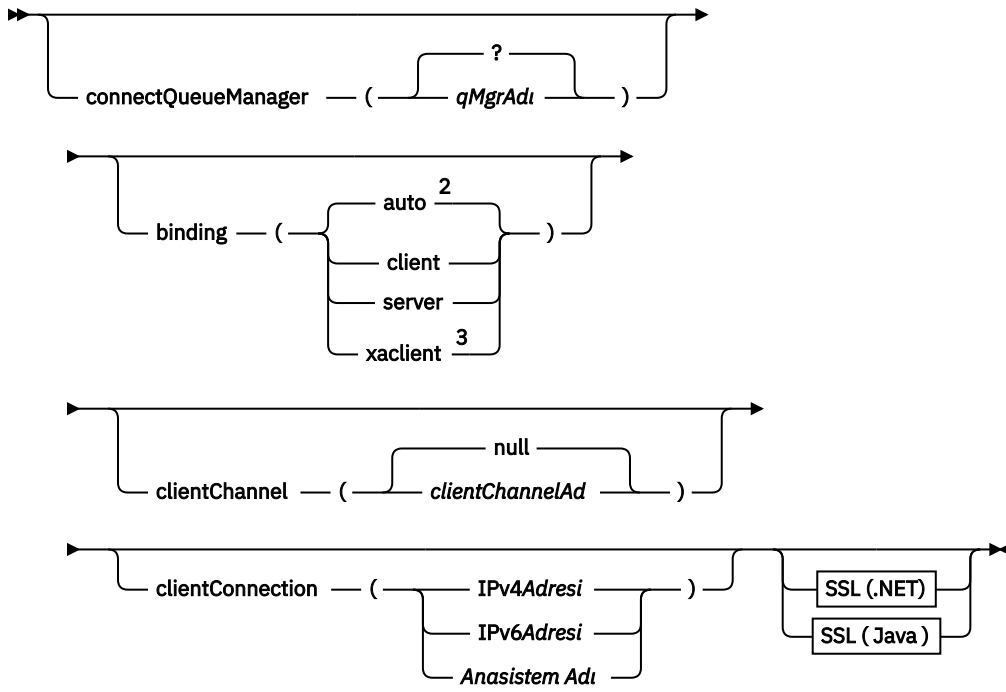


URI (Ortak)

►► & — initialContextFactory — = — com.ibm.mq.jms.Nojndi — & — connectionFactory — = — ►



Ad (değer)



Notlar:

¹ The queue manager transforms *className* to a queue name following the steps described in “Kuyruk adı dönüşümünü hedef alan hedef” sayfa 1457

² Bir istemci için uygun diğer seçenekler belirtilirse, *client* varsayılan değerdir; örneğin, *clientConnection*.

³ *xaclient* yalnızca .NET için geçerlidir

Kuyruk adı dönüşümünü hedef alan hedef

1. *className* is prefixed with SOAPJ. for Java services or with SOAPN. for .NET services.
2. Dosya uzantısı, *className* değiştirilmesinde verilen tam yol adından kaldırılır.
3. Sonuç dizgisi en çok 48 karakter olacak şekilde kısaltılır
4. Dizin ayırıcı karakterleri, nokta karakterleriyle değiştirilir.
5. Gömülü boşluklar, alt çizgi karakterleriyle değiştirilir.
6. Sürücü öneki harfinin ardından iki nokta üst üste, .NET hizmeti için bir nokta konur.

Not: Bazı ortamlarda, konuşlandırma yardımcı programı tarafından oluşturulan bir kuyruk adı benzersiz olmayabilir. Konuşlandırma yardımcı programı, kuyruğun yaratılıp yaratılmayacağını denetler. Konuşlandırma yardımcı programını, konuşlandırma dizini sıradüzenini yeniden yapılandırarak ya da sağlanan konuşlandırma işlemini uyarlayarak geçersiz kılamayı seçebilirsiniz.

Gerekli URI parametreleri

destination = kuyruk

queue, istek hedefinin adıdır. Kuyruk adı ya da kuyruk diğer adı olabilir. Kuyruk diğer adı söz konuyorsa, diğer ad bir konuya çözümleyebilir.

- -u parametresi atlanırsa, “Kuyruk adı dönüşümünü hedef alan hedef” sayfa 1457’inde açıklanan adımları kullanarak *classname* içinden *queue* (kuyruk) değeri oluşturulur.
- -u parametresi için *queue* değeri belirtilmişse ve ilk *jms: / queue ?* sonrasında URI 'nin ilk parametresi olması gerekir. Dizgi. Specify either an IBM MQ queue name, or a queue name and queue manager name connected by an @ symbol, for example SOAPN.trandemos@WMQSOAP.DEMO.QM.

- Konuşlandırma yardımcı programı kuyruk adının, üretilen ya da sağlanıp var olan bir kuyruğun adıyla eşleşip eşleşmediğini denetler. Alınan eylem, Çizelge 187 sayfa 1458’inde açıklanmaktadır.

Çizelge 187. kuyruk geçerlilik denetimi			
Dinleyici komut dosyası var mı?	Listener script exists in the ./generated/server directory		Dinleyici komut dosyası ./generated/server dizininde yok
Dinleyici komut dosyasında kuyruk kuyruk ile eşleşiyor mu?	kuyruk , dinleyici komut dosyasında kullanılmakta olan istek kuyruğunda eşleşmiyor	kuyruk , dinleyici komut dosyasında kullanılmakta olan istek kuyruklarıyla eşleşiyor	
kuyruk var	– Konuşlandırma, bir hatayla çıkar. – Hizmet zaten ./generated/serveriçinde konuşlandırıldı, ancak farklı bir kuyruk kullanılıyor.	– Konuşlandırma olağan şekilde devam eder. – Hizmet zaten ./generated/serveriçinde konuşlandırıldı.	– Konuşlandırma, bir hatayla çıkar. – Dinleyici başlatma komut dosyası ./generated/serverdosyasında bulunamadı, ancak kuyruk farklı bir hizmet ya da uygulama tarafından kullanılıyor.
kuyruk yok		– Devreye alma bir uyarıyla devam eder. – Önceki devreye alma başarısız olabilir; başlatma işlemi geçerli olduğundan, ancak queue (kuyruk) eksik.	– Konuşlandırma olağan şekilde devam eder. – Bu dizinden hizmet konuşlandırılmadı.

&connectionFactory = Ad (değer)

Ad aşağıdaki değıştirgelerden biridir:

- connectQueueYöneticisi (qMgrAdı)
- bağlama (bindingType)
- clientChannel(kanal)
- clientConnection(bağlantı)
- “Gerekli SSL parametreleri (Java)” sayfa 1444

Bu parametrelerin değerlerine ilişkin açıklamalar için bkz. “Bağlantı üreticisi değıştirgeleri” sayfa 1460 .

&targetService = serviceName

⁹.NETüzerinde, serviceName , konuşlandırma dizininde bulunan bir .NET hizmetinin adıdır; örneğin: targetService=myService.asmx. .NET ortamında, targetService parametresi tek bir IBM MQ SOAP dinleyicinin birden çok hizmete ilişkin istekleri işleyebilmesini olanaklı kılar. Bu hizmetler, aynı dizinden devreye alınmalı.

⁹ Yalnızca.NET hizmeti

İsteğe bağlı URI parametreleri

&initialContextFactory = contextFactory

contextFactory gereklidir ve `com.ibm.mq.jms.NoJndi` değerine ayarlanmalıdır. `NoJndi.jar` 'in bir WebSphere Application Server Web hizmetleri istemcisi için sınıf yolunda olduğundan emin olun. `NoJndi.jar`, bir dizine başvurarak değil, `connectionFactory` ve `destination` parametrelerinin içeriğini temel alan Java nesnelerini döndürür.

&targetService = serviceName

¹⁰*Axis* 'te, *serviceName*, bir Java hizmetinin tam olarak nitelenmiş adıdır, örneğin: `targetService=javaDemos.service.StockQuoteAxis`. `targetService` belirtilmemişse, varsayılan Eksen mekanizması kullanılarak bir hizmet yüklenir.

&persistence = messagePersistence

messagePersistence, aşağıdaki değerlerden birini alır:

0

Kalıcılık, kuyruk tanımlamasından devralınır.

1

İleti kalıcı değil.

2

İleti kalıcı

&priority = priorityValue

priorityValue, 0-9 aralığında yer alıyor. 0 değeri düşük öncelikli. Varsayılan değer ortama özgüdür; bu, IBM MQ durumunda 0 değeridir.

&replyDestination = replyToKuyruğu

Yanıt ileti için kullanılacak istemci tarafındaki kuyruk. Varsayılan yanıt kuyruğu `SYSTEM.SOAP.RESPONSE.QUEUE` 'dir.

- Varsayılan IBM MQ SOAP nesnelerini yaratmak için `setupWMQSOAP` komut dosyasını çalıştırın.
- Geçici ya da kalıcı bir dinamik yanıt kuyruğu yaratmak için *replyToKuyruğu* için bir model kuyruğu belirtin. Hem geçici hem de kalıcı dinamik yanıt kuyrukları için, her istek için ayrı bir dinamik kuyruk eşgörünümü yaratılır. Aşağıdaki olaylardan herhangi biri kuyruğa girerse, kuyruk silinir:

- Yanıt gelir ve işlenir.
- İstek zamanaşımına neden oldu.
- İstekte bulunan program sonlandırılır.

En iyi performans için, kalıcı dinamik kuyruklar yerine geçici dinamik kuyruklar kullanın. Geçici bir dinamik kuyruğa sahip bir URI 'ye kalıcı bir istek ileti göndermeyin. IBM MQ dinleyici SOAP, iletiyi işleyemiyor ve bir hata çıktı. Yanıt bekleyen istemci zaman aşımına neden olur.

- `setupWMQSOAP` komut kütüğü, `SYSTEM.SOAP.MODEL.RESPONSE.QUEUE` adlı varsayılan kalıcı bir dinamik model kuyruğu yaratır.

&timeout = waitTime

İstemcinin yanıt ileti için bekleyeceği süre (milisaniye olarak). *waitTime*, altyapı ya da istemci uygulaması tarafından ayarlanan değerleri geçersiz kılar. Belirtilmezse, uygulama değeri belirtildiyse ya da altyapının varsayılan değeri devralındıysa.

Not: Zamanaşımı ile `timeToLive` arasında bir ilişki uygulanmaz.

&timeToLive = expiryTime

expiryTime, iletinin süresi dolmadan önce milisaniye cinsinden belirtilen süredir. Varsayılan değer, sınırsız bir ömür boyu anlamına gelen sıfırdır.

Not: Zamanaşımı ile `timeToLive` arasında bir ilişki uygulanmaz.

¹⁰ Yalnızca Java hizmeti

Bağlantı üreticisi değiştirgeleri

connectQueueManager (*qMgrAdı*)

qMgrAd , istemcinin bağlanacağı kuyruk yöneticisini belirtir. Varsayılan değer boşluktur.

binding (*bindingType*)

bindingType , istemcinin *qMgrAdı* nı nasıl bağlandığı olduğunu belirtir. Varsayılan değer *auto* 'dur.

bindingType aşağıdaki değerleri alır:

auto

Gönderen, aşağıdaki bağlantı tiplerini sırayla dener:

1. İstemci bağlantısı için uygun diğer seçenekler belirtilirse, gönderen istemci bağ tanımı kullanır. Diğer seçenekler şunlardır: *clientConnection* ya da *clientChannel*.
2. Bir sunucu bağlantısı kullanın.
3. Bir istemci bağlantısı kullanın.

SOAP istemcisinde yerel kuyruk yöneticisi yoksa, *URI* 'da *binding* (*auto*) ögesini kullanın.

SOAP istemcisi için bir istemci bağlantısı oluşturuldu.

client

SOAP gönderene ilişkin bir istemci yapılandırması oluşturmak için *URI* 'da *binding* (*client*) ögesini kullanın.

server

SOAP gönderene ilişkin bir sunucu yapılandırması oluşturmak için *URI* 'deki *binding* (*server*) ögesini kullanın. Bağlantıda istemci tipi parametreleri varsa, bağlantı başarısız olur ve IBM MQ SOAP göndericisi tarafından bir hata görüntülenir. İstemci tipi değiştirgeleri şunlardır: *clientConnection*, *clientChannel* ya da *SSL* parametresidir.

xaclient

xaclient , Java istemcileri için değil, yalnızca .NET üzerinde geçerlidir. Bir XA istemcisi bağlantısı kullanın.

clientChannel (*kanal*)

SOAP istemcisi, bir IBM MQ istemci bağlantısı yapmak için *kanal* olanağını kullanır. *kanal* must match the name of a server connection channel, unless channel auto definition is enabled at the server. Client Connection Definition table (CCDT) sağlamıyorsanız, *clientChannel* zorunlu bir parametredir.

Provide a CCDT in Java by setting `com.ibm.mq.soap.transport.jms.mqchlurl`. .NET içinde, MQCHLLIB ve MQCHLTAB ortam değişkenlerini ayarlayın; bkz. [“SOAP göndereni için IBM MQ SOAP iletimi ile bir kanal tanımlama çizelgesi kullanın” sayfa 1454.](#)

clientConnection (*bağlantı*)

SOAP istemcisi, bir IBM MQ istemci bağlantısı yapmak için *bağlantı* kullanır. Varsayılan anasistem adı: *localhost* ve varsayılan kapı 1414' dir. *Bağlantı* bir TCP/IP adresiyse, üç biçimden birini alır ve bir kapı numarasıyla alt düzeltilebilir.

JMS clients can either use the format : `hostname:port` or 'escape' the brackets using the format `%X` where X is the hexadecimal value that represents the bracket character in the code page of the URI. For example, in ASCII, %28 and %29 for (and) respectively.

.Net istemcileri, köşeli ayraçları belirttik olarak : `hostname(port)` kullanabilir ya da 'kaçış ' biçimini kullanabilir.

IPv4 adres

Örneğin, 192.0.2.0.

IPv6 adres

Örneğin, 2001:DB8:0:0:0:0:0:0.

Anasistem adı

Örneğin, `www.example.com%281687%29`, `www.example.com:1687` ya da `www.example.com(1687)`.

SSL altyapı

Bkz. "Gerekli SSL parametreleri (Java)" sayfa 1444

Örnek URI 'ler

Not:

1. URI 'deki& , & olarak kodlanır
2. Listelenen tüm parametre istemciler için geçerli olur.
3. Yalnızca **destination**, **connectionFactory** ve **initialContextFactory** , WCF hizmeti için geçerlidir.

```
jms:/queue?  
destination=myQ&connectionFactory=()&initialContextFactory=com.ibm.mq.jms.Nojndi
```

Şekil 23. Yalnızca gerekli parametreleri tedarik eden bir Eksen hizmeti URI 'si

```
jms:/queue?destination=myQ&connectionFactory=()&targetService=MyService.asmx  
&initialContextFactory=com.ibm.mq.jms.Nojndi
```

Şekil 24. Yalnızca gerekli parametreleri tedarik eden bir .NET hizmeti URI 'si

```
jms:/queue?destination=myQ@myRQM&connectionFactory=connectQueueManager(myconnQM)  
binding(client)clientChannel(myChannel)clientConnection(myConnection)  
&initialContextFactory=com.ibm.mq.jms.Nojndi
```

Şekil 25. İsteğe bağlı connectionFactory değiştiricileri sağlayan bir Axis hizmeti URI 'si

```
jms:/queue?destination=myQ@myRQM&connectionFactory=connectQueueManager(myconnQM)  
binding(client)clientChannel(myChannel)clientConnection(myConnection)  
sslPeerName(CN=MQ Test 1,O=IBM,S=Hampshire,C=GB)  
&initialContextFactory=com.ibm.mq.jms.Nojndi
```

Şekil 26. connectionFactory parametresinin sslPeerAd seçeneğini sağlayarak bir Eksen hizmeti URI 'si

Nojndi mekanizması

The Nojndi mechanism enables JMS programs, which use JNDI interfaces, to use the same URI as IBM MQ programs, which do not use JNDI.

You can use the IBM MQ transport for SOAP to invoke Web services on WebSphere Application Server. WebSphere Application Server SOAP over JMS looks up the JMS resources using JNDI. Web hizmeti istemcisi .NET üzerinde çalışıyor ya da Web hizmetini çağırmak için 1.4Axis 'i kullanıyor olabilir ve JNDI kullanmıyor olabilir. İstemci ve sunucu için aynı URL ' yi kullanmak için, ortamın JNDI kullanıp kullanmadığına ilişkin bilgileri de sağlamalıdır.

Bir Web hizmeti istemcisi tarafından SOAP olanağına ilişkin IBM MQ iletisine geçirilen URI, belirli bir IBM MQ kuyruk yöneticisi ve kuyruk adı içerir. Bu adlar ayrıştırılır ve doğrudan IBM MQ SOAP desteği tarafından kullanılır.

Nojndi mekanizması, bir JMS programı tarafından kullanılan initialContextÜreticisi 'ı com.ibm.mq.jms.Nojndi' e yönlendirir. The com.ibm.mq.jms.Nojndi class is an implementation of the JNDI interface that returns the connectionFactory and hedef from the URL as ConnectionFactory and Kuyruk Java objects. If the JMS implementation is IBM MQ, MQConnectionFactory and MQQueue inherit from the ConnectionFactory and Kuyruk classes.

Nojndi mekanizmasını kullanarak, aynı URL ' yi kullanarak WebSphere Application Server ve .NET ile aynı bağlantı bilgilerini sağlayabilirsiniz.

W3C SOAP over JMS URI for the IBM MQ Axis 2 client

Define a W3C SOAP over JMS URI to call a Web service from an Axis 2 client using IBM MQ JMS as the SOAP transport. Web hizmeti, SOAP/JMS bağ tanımı için IBM MQ JMS ve W3C SOAP over JMS aday önerisine destek veren bir sunucu tarafından sağlanmalıdır.

Tanım

W3C aday önerisi, SOAP üzerinde SOAP bağ tanımı (SOAP over Java Message Service 1.0) tanımlar. Ayrıca, bunun örnekleri için de yararlı olur: [Java \(tm\) ileti hizmeti için URI şeması 1.0¹¹](#).

Use the syntax diagram to create W3C SOAP over JMS URIs that are syntactically correct, and are accepted by the IBM MQ Axis 2 client. Bu, IBM MQ Axis 2 istemcisi tarafından kabul edilen URI 'yi tanımlamaya sınırlıdır. Bu, iki bakımdan W3C önerisinin bir alt kümesidir:

1. jms-variant konusu desteklenmiyor ve IBM MQ Axis 2 istemcisine geçirilen bir URI 'de belirtilmemesi gerekir.
2. The following properties are omitted from the syntax diagram because they are JMS properties, and not part of the URI.
 - a. bindingVersion
 - b. contentType
 - c. soapAction
 - d. requestURI
 - e. isFault

JMS özellikleri, Axis 2 istemcisi ya da sunucu tarafından ayarlanır.

Çizge, özel bir parametre (connectionFactory) tanımlayarak W3C önerisini genişletir. connectionFactory , Axis 2 istemcisinin bir kuyruk kullanarak kuyruk yöneticisine nasıl bağlanacağını belirtmek için, JNDI için bir alternatif olarak kullanılır.

IBM MQ Axis 2 istemcisi yalnızca istemci uygulaması ya da ortam değişkenleri tarafından istemciye geçirilen URI 'nin bir parçası olarak özellikleri kabul eder. IBM MQ Axis 2 istemcisinin bir WSDL belgesini işleyebilme yeteneği yoktur. İstemci uygulaması ya da bir geliştirme aracı WSDL 'yi işleyebilir ve Axis 2 istemcisine geçmek için URI'yi yaratır. Bir IBM MQ Ekseni 2 istemci uygulaması, JMS ileti özelliklerini doğrudan ayarlayamaz.

Sözdizimi

W3C önerisine uygun olarak, ortam değişkenlerinden tüm parametreler elde edilebilir. Ortam değişkeni adları, parametre adı ile soapjms_ile önceden karşılaşılarak oluşturulur. Sözdizimi şöyledir: soapjms_ parameterName ; Örneğin,

```
set soapjms_targetServer=com.example.org.stockquote
```

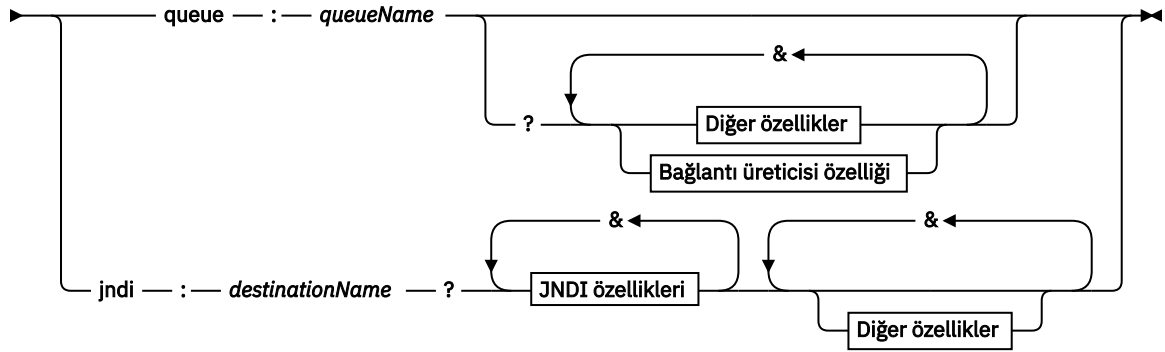
Bir değiştirge ortam değişkeni kullanılarak ayarlandıysa, bu değiştirge URI 'deki değer kümesini geçersiz kılar.

W3C önerisine uygun olarak, tüm parametreler yinelenebilir. Bir ortam değişkeni tarafından geçersiz kılınmadıkça, parametrenin son eşgörünümü kullanılır.

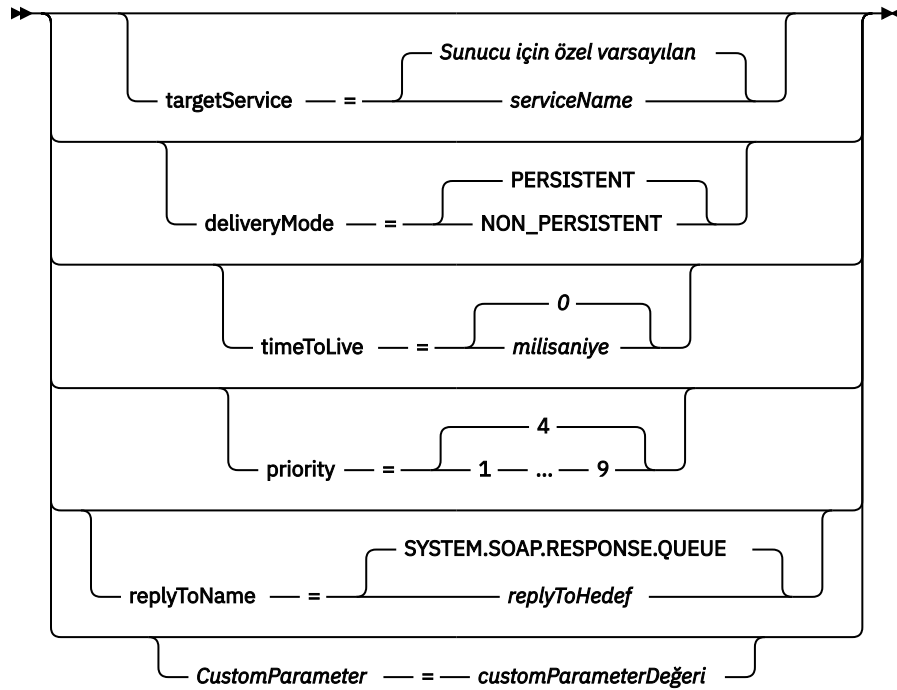
¹¹ En son taslağa ilişkin W3C belirtim başvurularında *JMS için URI Şeması*'na bakın.

jms-uri

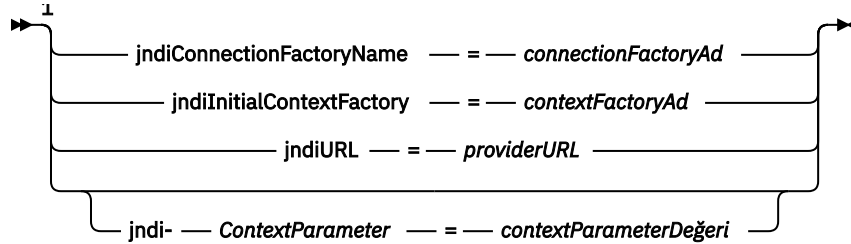
➡ jms: ➡



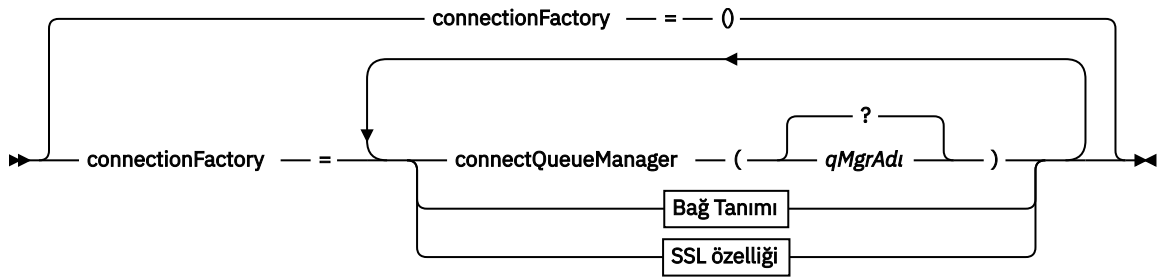
Diğer özellikler



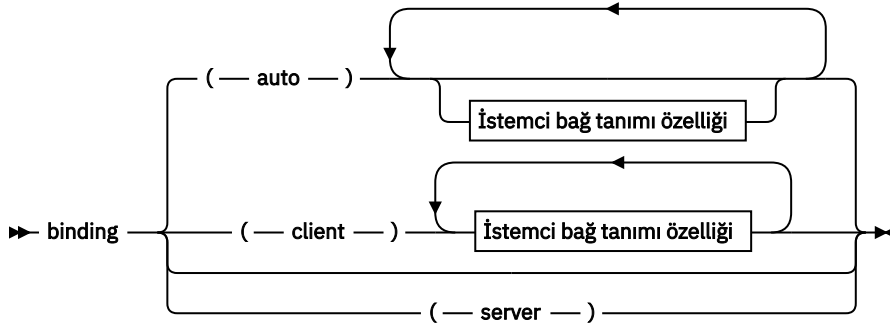
JNDI özellikleri



Bağlantı üreticisi özelliği

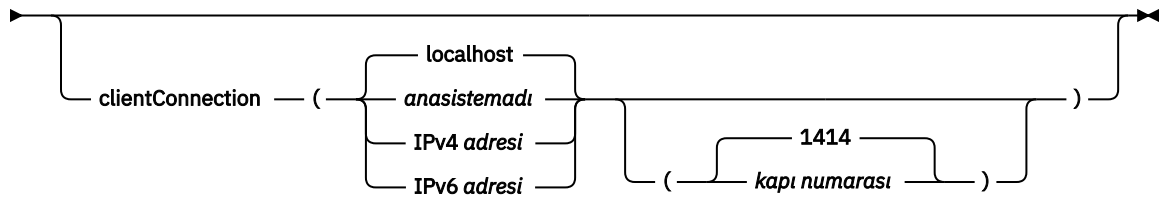


Bağ Tanımı



İstemci bağ tanımı özelliği

clientChannel — (— kanal —) →



SSL özelliği

sslCipherSuite — (— cipherSuite —) →
sslPeerName — (— peerName —)

sslKeyStore — (— KeyStoreAdı —) — sslKeyStorePassword — (— KeyStoreParolası —)

sslTrustStore — (— TrustStoreAdı —) — sslTrustPassword — (— TrustStoreParolası —)

sslKeyResetCount — (— byteCount —)

sslFipsRequired — (— fipsCertified —)

sslLDAPCRLServers — (— LDAPServerList —)

Notlar:

¹ **jndiConnectionFactoryName**, **jndiConnectionFactoryName** ve **jndiURL** gerekli tüm değiştirgelerdir. **jndi-** *ContextParameter* isteğe bağlıdır.

Parametreler

connectionFactory = connectionFactoryParameterList

connectionFactoryParameterList , hedef değişken kuyruk olduğunda, Axis 2 istemcisinin bir kuyruk yöneticisine nasıl bağlanacağını niteleyen parametredir.

connectionFactory , jndi hedef değişkeniyle belirtilmemelidir.

Parametreler, istek URI 'sındaki sunucuya iletilmiyor.

connectionFactory atlanırsa, kuyruk, Axis 2 istemciyle aynı sunucuda çalışan bir varsayılan kuyruk yöneticisine ait olmalıdır.

connectionFactoryParameterList:

binding (bindingType)

bindingType , istemcinin *qMGrAdın* nasıl bağlandığını belirtir. Varsayılan değer *auto* 'dur.

bindingType aşağıdaki değerleri alır:

auto

Gönderen, aşağıdaki bağlantı tiplerini sırayla dener:

1. İstemci bağlantısı için uygun diğer seçenekler belirtilirse, gönderen istemci bağ tanımı kullanır. Diğer seçenekler şunlardır: *clientConnection* ya da *clientChannel*.
2. Bir sunucu bağlantısı kullanın.
3. Bir istemci bağlantısı kullanın.

SOAP istemcisinde yerel kuyruk yöneticisi yoksa, *URI* 'da *binding (auto)* ögesini kullanın. SOAP istemcisi için bir istemci bağlantısı oluşturuldu.

client

SOAP gönderene ilişkin bir istemci yapılandırması oluşturmak için *URI* 'da *binding (client)* ögesini kullanın.

server

SOAP gönderene ilişkin bir sunucu yapılandırması oluşturmak için *URI* 'deki *binding (server)* ögesini kullanın. Bağlantıda istemci tipi parametreleri varsa, bağlantı başarısız olur ve IBM MQ SOAP göndericisi tarafından bir hata görüntülenir. İstemci tipi değiştirgelemleri şunlardır: *clientConnection*, *clientChannel* ya da *SSL* parametresidir.

xaclient

xaclient , Java istemcileri için değil, yalnızca .NET üzerinde geçerlidir. Bir XA istemcisi bağlantısı kullanın.

clientChannel (kanal)

SOAP istemcisi, bir IBM MQ istemci bağlantısı yapmak için *kanal* olanağını kullanır. *kanal* must match the name of a server connection channel, unless channel auto definition is enabled at the server. Client Connection Definition table (CCDT) sağlamıyorsanız, *clientChannel* zorunlu bir parametredir.

Provide a CCDT in Java by setting `com.ibm.mq.soap.transport.jms.mqchlurl`. .NET içinde, MQCHLLIB ve MQCHLTAB ortam değişkenlerini ayarlayın; bkz. “SOAP göndereni için IBM MQ SOAP iletimi ile bir kanal tanımlama çizelgesi kullanın” sayfa 1454.

clientConnection (bağlantı)

SOAP istemcisi, bir IBM MQ istemci bağlantısı yapmak için *bağlantı* kullanır. Varsayılan anasistem adı: *localhost* ve varsayılan kapı 1414' dir. *Bağlantı* bir TCP/IP adresiyse, üç biçimden birini alır ve bir kapı numarasıyla alt düzeltilebilir.

JMS clients can either use the format : `hostname:port` or 'escape' the brackets using the format `%X` where X is the hexadecimal value that represents the bracket character in the code page of the URI. For example, in ASCII, `%28` and `%29` for (and) respectively.

.Net istemcileri, köşeli ayraçları belirttik olarak : `hostname(port)` kullanabilir ya da 'kaçış ' biçimini kullanabilir.

IPv4 adres

Örneğin, 192.0.2.0.

IPv6 adres

Örneğin, 2001:DB8:0:0:0:0:0:0.

Anasistem adı

Örneğin, www.example.com%281687%29, www.example.com:1687ya da
www.example.com(1687).

sslCipherSuite (*CipherSuite*)

CipherSuite , kanalda kullanılan sslCipherSuite ' i belirtir. İstemcinin belirttiği CipherSuite , sunucu bağlantı kanalında belirtilen CipherSuite ile eşleşmelidir.

sslFipsRequired (*fipsCertified*)

fipsCertified specifies whether *CipherSpec* or *CipherSuite* must use FIPS-certified cryptography in IBM MQ on the channel. *fipsCertified* ayarının etkisi, MQCONNX çağrısındaki **MQSCO** yapısının FipsRequired alanının ayarlandığı şekilde aynıdır.

sslKeyStore (*KeyStoreAdı*)

KeyStoreAdı , kanalda kullanılan sslKeyStoreName ' i (StoreName) belirtir. Anahtar deposu, istemciyi sunucu için doğrulamak için kullanılan istemcinin özel anahtarını tutar. SSL bağlantısı anonim istemci bağlantılarını kabul edecek şekilde yapılandırıldıysa, anahtar deposu isteğe bağlıdır.

sslKeyResetCount (*bytecount*)

bytecount , SSL güvenlik dizgisi anahtarının yeniden görüşülebilmesi için önce bir SSL kanalından geçirilen bayt sayısını belirtir. SSL anahtarlarının yeniden anlaşılmasını devre dışı bırakmak için alanı kaldırın ya da sıfırın değerine ayarlayın. Sıfır, bazı ortamlarda desteklenen tek değerdir, bkz. [IBM MQ classes for Java içindeki gizli anahtarın yeniden anlaşılması](#). sslKeyResetCount ayarının etkisi, MQCONNX çağrısındaki **MQSCO** yapısındaki KeyResetCount alanının ayarlandığı şekilde aynıdır.

sslKeyStorePassword (*KeyStoreParolası*)

KeyStoreParolası , kanalda kullanılan sslKeyStorePassword değerini belirtir.

sslLDAPCRLServers (*LDAPServerList*)

LDAPServerList , Sertifika İptal Listesi denetimi için kullanılacak LDAP sunucularının bir listesini belirtir.

SSL etkin istemci bağlantıları için, *LDAPServerList* , CRL (Sertifika İptal Listesi) denetimi için kullanılacak LDAP sunucularının bir listesidir. Kuyruk yöneticisi tarafından sağlanan sertifika, listelenen LDAP CRL sunucularından birine yönelik olarak denetlenir; bulunduysa, bağlantı başarısız olur. Her bir LDAP sunucusu, biri için bağlantı kuruluncaya kadar sırayla denetlenir. Sunuculardan herhangi birine bağlanmak olanaksızsa, sertifika reddedilir. Bu sertifikalardan biri için bağlantı başarıyla kurulduktan sonra, bu LDAP sunucusunda bulunan CRL ' lere bağlı olarak sertifika kabul edilir ya da reddedilir.

LDAPServerList boşsa, kuyruk yöneticisine ait olan sertifika, bir Sertifika İptal Listesi 'ne göre denetlenmez. Sağlanan LDAP URI ' leri listesi geçerli değilse, bir hata iletisi görüntülenir. The effect of setting this field is the same as that of including MQAIR records and accessing them from an **MQSCO** structure on an MQCONNX.

sslPeerName (*peerName*)

peerName , kanalda kullanılan sslPeerName ' i belirtir.

sslTrustStore (*TrustStoreAdı*)

TrustStoreAdı , kanalda kullanılan sslTrustStoreName ' i belirtir. Güvenilir depo, sunucunun genel sertifikasını ya da sunucuyu istemcide kimlik doğrulaması yapmak için temel zincirini tutar. Sunucunun kimliğini doğrulamak için bir sertifika yetkilisinin kök sertifikası kullanılırsa, güvenilirlik deposu isteğe bağlıdır. Java' ta kök sertifikalar JRE sertifika deposunda tutulur, cacerts.

sslTrustStorePassword (*TrustStoreParolası*)

TrustStoreParolası , kanalda kullanılan sslTrustStorePassword değerini belirtir.

CustomParameter = customParameterDeğeri

CustomParameter , bir özel parametrenin kullanıcı tanımlı adıdır ve *customParameterValue* , parametrenin değeridir.

Axis 2 istemcisi tarafından kullanılmayan özel parametreler, Axis 2 istemcisi tarafından SOAP sunucusuna gönderilir. Sunucu belgelerine bakın. *connectionFactory* , Axis 2 istemcisi tarafından kullanılan ve sunucuya iletilmediği özel bir parametredir.

CustomParameter , var olan bir parametrenin adıyla eşleşmemelidir.

CustomParameter , bir JNDI hedefini aramak için *jndi* - dizgisiyle başlıyorsa, bkz. [jndi](#) -.

deliveryMode = deliveryMode

deliveryMode , ileti kalıcısını ayarlar. Varsayılan değer 'PERSISTENT' dir.

jndi : destinationName

destinationName , bir JMS kuyruğundan eşlenen bir JNDI hedef adıdır. *jndi* hedef değişkeni belirtilirse, bir *destinationNames* sağlamanız gerekir.

jndiConnectionFactoryName = connectionFactoryAd

connectionFactoryAd , bağlantı üreticisinin JNDI adını ayarlar. Hedef çeşitleme *jndi* ise, *connectionFactoryName* belirtilmelidir.

jndiInitialContextFactory = contextFactoryAd

contextFactoryName , ilk bağlam üreticisinin JNDI adını ayarlar. Hedef çeşitleme *jndi* ise, *contextFactoryName* belirtilmelidir. Bkz. [Bir JMS uygulamasında yönetilen nesneleri almak için JNDI kullanılması](#).

jndiURL = providerURL

jndiURL , JNDI sağlayıcısının URL adını ayarlar. Hedef çeşitleme *jndi* ise, *jndiURL* belirtilmelidir.

jndi- ContextParameter = contextParameterDeğeri

jndi- ContextParameter , JNDI sağlayıcısına bilgi geçirmek için kullanılan kullanıcı tanımlı bir parametrenin adıdır. *contextParameterDeğer* İletilen bilgidir.

priority = priorityValue

priorityValue , JMS ileti önceliğini ayarlar. 0 düşüktür, 9 yüksektir. Varsayılan değer 4'dür.

queue : queueName

queueName , SOAP isteğinin yerleştirileceği bir JMS kuyruğunun adıdır. Kuyruk çeşitlemesi belirtilirse, bir kuyruk adı sağlanmalıdır. Kuyruk, istemciyle aynı sunucudaki bir varsayılan kuyruk yöneticisine ait değilse, *connectionFactory* parametresini ayarlayın.

replyToName = replyToHedef

replyToHedefi , hedef kuyruk adını ayarlar. Hedef değişken *jndi* ise, ad bir kuyrukla eşlenmesi gereken bir JNDI adıdır. Değişken *queue* ise, ad bir JMS kuyruğudur. Varsayılan değer *SYSTEM . SOAP . RESPONSE . QUEUE* 'dir.

targetService = serviceName

Hedef Web hizmetini başlatmak için SOAP sunucusu tarafından kullanılan ad.

Axis 'te, *serviceName* , bir Java hizmetinin tam olarak nitelenmiş adıdır, örneğin:

targetService=*www.example.org.StockQuote*. *targetService* belirtilmemişse, varsayılan Eksen mekanizması kullanılarak bir hizmet yüklenir.

timeToLive = milisaniye

İletin süresi dolmadan önce *milisaniye* olarak ayarlayın. Varsayılan değer olan 0, meselenin süresi hiçbir zaman dolmaz.

Örnekler

```
jms:jndi:REQUESTQ
?jndiURL=file:/C:/JMSAdmin
&jndiInitialContextFactory=com.sun.jndi.fscontext.RefFSContextFactory
&jndiConnectionFactoryName=ConnectionFactory
&replyToName=RESPONSEQ
&deliveryMode(NON_PERSISTENT)
```

Şekil 27. SOAP/JMS isteği göndermek için jms: jndi ögesini kullanın.

```
jms:queue:SOAPJ.demos
?connectionFactory=connectQueueManager(QM1)
Bind(Client)
ClientChannel(SOAPClient)
ClientConnection(www.example.org(1418))
&deliveryMode(NON_PERSISTENT)
```

Şekil 28. SOAP/JMS isteği göndermek için jms: queue ögesini kullanın.

Desteklenen Web hizmetleri

Web hizmeti olarak çalıştırmak üzere yazılan kodların, SOAP için IBM MQ iletimi kullanacak şekilde değiştirilmesi gerekmez. You do need to deploy services differently to run with the IBM MQ transport for SOAP rather than using HTTP.

Tanım

V 8.0.0.2 IBM MQ 8.0.0, Düzeltme Paketi 2'tan SOAP' a ilişkin IBM MQ iletimi kullanımdan kaldırılmıştır. Daha fazla bilgi için bakınız: [Deprecations: IBM MQ transport SOAP client and HTTP Bridge](#).

The IBM MQ transport for SOAP provides a SOAP listener to run services for .NET Framework 1 and .NET 2, and for Axis 1.4. The IBM MQ custom channel for Microsoft Windows Communication Foundation runs services for .NET Framework 3. WebSphere Application Server ve CICS , SOAP için IBM MQ iletimi üzerinden hizmetlerin çalıştırılmasına ilişkin destek sağlar. WebSphere Enterprise Service Bus ya da WebSphere Process Serverolanağını kullanmak için özel bir dışa aktarma yaratın.

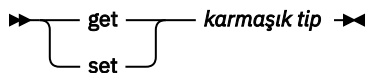
IBM MQ SOAP dinleyicisi, SOAP isteklerini ilişkisel olarak işleyebilir. **amqwdployWMQService** komutunu -x seçeneğini kullanarak çalıştırın. İki aşamalı seçeneği yalnızca sunucu bağ tanımlarını kullanan dinleyiciler için desteklenir. Diğer ortamlar, SOAP için IBM MQ iletimi için işlem desteği sağlayabilir. Belgelere bakın.

SOAP içinIBM MQ iletimi şu anda W3C' ye gönderilen, gelişmekte olan sektör standardı SOAP over JMS iletişim kuralı tarafından desteklenmiyor. You can distinguish a SOAP/JMS message written to the new standard by looking for the JMS BindingVersion property. SOAP içinIBM MQ iletimi, BindingVersion özelliğini ayarlamaz.

Eksen 1.4

Bir Java sınıfı genellikle değişiklik yapılmadan kullanılabilir. Web hizmetindeki yöntemlere ilişkin bağımsız değişkenlerin tipleri Axis motoru tarafından desteklenmelidir. Ek ayrıntılar için Axis belgelerine bakın. Hizmet bir bağımsız değişken olarak karmaşık bir nesne kullanıyorsa ya da bir değer döndürürse, o nesne Java Bean belirtimine uygun olmalıdır. See the examples in [Şekil 31 sayfa 1470](#), [Şekil 32 sayfa 1470](#), and [Şekil 33 sayfa 1471](#):

1. Genel bir değiştirge-daha az oluşturucuya sahip olmalıdır.
2. Bean 'in karmaşık tiplerinin, formun genel alıcıları ve ayarlayıcıları olması gerekir:



Prepare the service for deployment using the **amqwdeployWMQService** utility. Hizmet, hizmeti çalıştırmak için `axis.jar` 'i kullanan IBM MQ SOAP dinleyicisi tarafından çağrılır.

The only two-phase transaction manager supported for Axis 1.4 is IBM MQ.

Sağlanan devreye alma yardımcı programı, hizmetin farklı bir paketteki bir nesneyi hizmetin kendisine döndürdüğü durumu desteklemiyor. Farklı bir paket içinde döndürülen bir nesneyi kullanmak için kendi konuşlandırma yardımcı programını yazın. You can base your deployment utility on the supplied sample, or capture the commands it produces, using the `-v` option. İsteğe uyarlanmış bir komut dosyası oluşturmak için komutları sonlandırır.

Hizmet, Axis altyapısı ve IBM MQ SOAP çalıştırma zamanı ortamı dışındaki sınıfları kullanıyorsa, doğru CLASSPATH'yi ayarlamamız gerekir. CLASSPATH' u değiştirmek için, oluşturulan komut dosyasını, aşağıdaki yöntemlerden biriyle, gereken hizmetleri içerecek şekilde başlatan ya da tanımlayan komut dosyasını yeniden ekleyin:

- **amqwsetcp**'i aramadan sonra komut dosyasında doğrudan CLASSPATH 'i yeniden girin.
- Create a service-specific script to customize the CLASSPATH and invoke this script in the generated script after the call to **amqwsetcp**.
- Oluşturulan komut dosyasındaki CLASSPATH dosyasını otomatik olarak uyarlamak için uyarlanmış bir konuşlandırma işlemi yaratın.

.NET Framework 1 ve .NET Framework 2

HTTP Web hizmeti olarak önceden hazırlanmış bir hizmetin, IBM MQ Web hizmeti olarak kullanılmak üzere değiştirilmesine gerek yoktur. Bu, **amqwdeployWMQService** yardımcı programı kullanılarak devreye alınması gerekir.

.NET Framework 1 ve .NET 2 için desteklenen tek iki aşamalı hareket yöneticisi Microsoft Transaction Server 'dir (MTS).

Hizmet kodu HTTP Web hizmeti olarak hazırlanmadıysa, bu hizmeti bir Web hizmetine dönüştürmeniz gerekir. Sınıfı bir web hizmeti olarak bildirin ve her bir yöntemin parametrelerinin nasıl biçimlendirileceğini tanımlayın. Hizmetin yöntemlerine ilişkin bağımsız değişkenlerin ortamla uyumlu olup olmadığını denetlemeniz gerekir. Şekil 29 sayfa 1470 ve Şekil 30 sayfa 1470 , web hizmeti olarak hazırlanmış bir .NET sınıfını gösterir. Yapılan eklemeler kalın yazıyla gösterilir.

Şekil 29 sayfa 1470 , bir .NET Web hizmeti için kod arkasında kod arkasındaki programlama modelini kullanır. Kod arkasındaki modelde, hizmet kaynağı .asmx dosyasından ayrılır. The .asmx file declares the name of the associated source file with the Kod/kod keyword. IBM MQ içinde hem yerleşik hem de kod arkasında .NET Web hizmetleri örnekleri bulunur.

.NET Web services source must be compiled before deployment by the **amqwdeployWMQService** deployment utility. Hizmet bir kitaplığa derlenir (.dll). Kitaplık, konuşlandırma dizininin . /bin alt dizinine yerleştirilmelidir.

.NET Framework 3

.NET Framework 3 'e dağıtılan hizmetleri çağırmak için Microsoft Windows Communication Foundation (WCF) için bir IBM MQ özel kanalı oluşturun. WCF 'nin SOAP için IBM MQ iletimi kullanacak şekilde yapılandırılmasına ilişkin açıklamalar için [Developing WCF applications with IBM MQ](#) başlıklı konuya bakın.

WebSphere Application Server

SOAP için IBM MQ İletimi 'ni kullanarak WebSphere Application Server tarafından barındırılan web hizmetlerini çağırabilirsiniz. Bkz. [Web hizmetlerini aktarmak için JMS üzerinden SOAP kullanılması](#).

You need to modify the WSDL generated by deployment of a JMS service to WebSphere Application Server in order to generate a Web services client. The WSDL created by deployment to WebSphere Application Server includes a URI with a JNDI reference to the JMS InitialContextÜreticisi. JNDI başvurusunu Nojndi olarak değiştirmeniz ve bağlantı özniteliklerini “Web hizmeti konuşlandırması için URI sözdizimi ve değiştirelileri” sayfa 1455’inde açıklandığı gibi sağlamamız gerekir.

CICS

You can invoke CICS applications using the IBM MQ Transport for SOAP. Bkz. [CICS sisteminizi Web hizmetleri için yapılandırma](#).

WebSphere Enterprise Service Bus and WebSphere Process Server for Multiplatforms

WebSphere ESB ve WebSphere Process Server for Multiplatforms, yalnızca varsayılan WebSphere Application Server ileti alışverişi sağlayıcısını kullanırken hazır bir bağ tanımı ile JMSüzerinden SOAP 'ı destekler. SOAP için IBM MQ iletimi desteklemek üzere JMS için özel bir bağ tanımı yaratın. Bkz. [JMS veri bağ tanımları](#)..

Örnek

```
<%@ WebService Language="C#" CodeBehind="Quote.aspx.cs" Class="Quote.QuoteDotNet" %>
```

Şekil 29. .NET Framework 2 için hizmet tanımı: *Quote.aspx*

```
<%@ WebService Language="C#" CodeBehind="Quote.aspx.cs" Class="Quote.QuoteDotNet" %>
using System;
using System.Web;
using System.Web.Services;
using System.Web.Services.Protocols;

namespace Quote {
    [WebService(Namespace = "http://www.example.org/")]
    [WebServiceBinding(ConformsTo = WsiProfiles.BasicProfile1_1)]
    public class QuoteDotNet : System.Web.Services.WebService {
        [WebMethod]
        public string getQuote(String symbol){
            return symbol.ToUpper();
        }
    }
}
```

Şekil 30. .NET Framework 2 için hizmet uygulaması: *Quote.aspx.cs*

```
package org.example.www;
public interface CustomerInfoInterface extends java.rmi.Remote {
    public org.example.www.CustomerRecord
        getCustomerName(org.example.www.CustomerRecord request)
        throws java.rmi.RemoteException, org.example.www.GetCustomerName_faultMsg;
}
```

Şekil 31. Karmaşık tip kullananJava JAX-RPC hizmet arabirimi

```
package org.example.www;
public class CustomerInfoPortImpl implements org.example.www.CustomerInfoInterface{
    public org.example.www.CustomerRecord
        getCustomerName(org.example.www.CustomerRecord request)
        throws java.rmi.RemoteException, org.example.www.GetCustomerName_faultMsg {
        request.setName(request.getID().toString());
        return request;
    }
}
```

Şekil 32. Karmaşık tip kullananJava JAX-RPC hizmeti somutlaması

```

package org.example.www;
public class CustomerRecord {
    private java.lang.String name;
    private java.lang.Integer ID;
    public CustomerRecord() {}
    public java.lang.String getName() {
        return name;
    }
    public void setName(java.lang.String name) {
        this.name = name;
    }
    public java.lang.Integer getID() {
        return ID;
    }
    public void setID(java.lang.Integer ID) {
        this.ID = ID;
    }
}

```

Şekil 33. Java JAX-RPC hizmet bean somutlaması karmaşık tip

SOAP Web hizmeti istemcileri için IBM MQ iletimi

SOAP için IBM MQ iletimi ile HTTP istemcisi üzerinde var olan bir SOAP istemcisini yeniden kullanabilirsiniz. İstemciyi SOAP için IBM MQ iletimi ile çalışacak şekilde dönüştürmek için kod ve oluşturma işleminde bazı küçük değişiklikler yapmanız gerekir.

Kodlama

JAX-RPC istemcileri Java içinde yazılmalıdır. .NET Framework 1 ve 2 istemcileri, Common Language Runtime 'ı kullanan herhangi bir dilde yazılabilir. Kod örnekleri C# ve Visual Basic içinde sağlanmıştır.

İşlemsel destek düzeyi, istemci ortamına ve SOAP etkileşiminin örüntülerine bağlıdır. SOAP isteği ve SOAP yanıtı, aynı atomik hareketin bir parçası olamaz.

You must call `IBM.WMQSOAP.Register.Extension()` in a .NET Framework 1, .NET Framework 2 client. In a JAX-RPC Java Web service client call `com.ibm.mq.soap.Register.extension` to register the IBM MQ SOAP sender. The method registers the IBM MQ transport for SOAP sender as the handler for SOAP messages using the `jms:` protocol.

Bir .NET Framework 3 istemcisi oluşturmak için, **svcutil** aracını kullanarak bir Windows Communication Foundation istemcisi yetkili sunucusu oluşturun; bkz. [Bir çalışan hizmetten meta veri içeren svcutil aracını kullanarak bir WCF istemcisi yetkili sunucusu ve uygulama yapılandırma dosyaları oluşturma.](#)

.NET Framework 1 ve 2 istemcilerini oluşturmak ve çalıştırmak için gereken kitaplıklar

- amqsoap
- Sistem
- System.Web.Services
- System.Xml

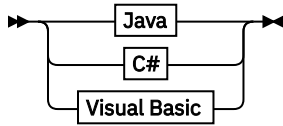
Axis 1.4 istemcilerini oluşturmak ve çalıştırmak için gereken kitaplıklar

- `MQ_Install\java\lib\com.ibm.mq.soap.jar;`
- `MQ_Install\java\lib\com.ibm.mq.commonservices.jar;`
- `MQ_Install\java\lib\soap\axis.jar;`
- `MQ_Install\java\lib\soap\jaxrpc.jar`
- `MQ_Install\java\lib\soap\saa.jar;`
- `MQ_Install\java\lib\soap\commons-logging-1.0.4.jar;`
- `MQ_Install\java\lib\soap\commons-discovery-0.2.jar;`
- `MQ_Install\java\lib\soap\wsdl4j-1.5.1.jar;`
- `MQ_Install\java\jre\lib\xml.jar;`

- `MQ_Install\java\lib\soap\servlet.jar;`
- `MQ_Install\java\lib\com.ibm.mq.jar;`
- `MQ_Install\java\lib\com.ibm.mq.headers.jar;`
- `MQ_Install\java\lib\com.ibm.mq.pcf.jar;`
- `MQ_Install\java\lib\com.ibm.mq.jmqi.jar;`

Not: IBM MQ 8.0, `ldap.jar`, `jndi.jar` ve `jta.jar` 'den şu anda JDK' nin bir parçası olduğu için bu listeden kaldırıldı.

SOAP uzantısını kaydet



Java

➔ `com.ibm.mq.soap.Register.extension()` ➔

C#

➔ `IBM.WMQSOAP.Register.Extension();` ➔

Visual Basic

➔ `IBM.WMQSOAP.Register.Extension` ➔

Client örnekleri

Şekil 34 sayfa 1472 , yerleşik programlama modelini kullanan bir .NET Framework 1 ya da .NET Framework 2 C# istemcisi örneğidir. **IBM.WMQSOAP.Register.Extension()** yöntemi, IBM MQ SOAP göndericisini .NET ile `jms:` iletişim kuralı işleyicisi olarak kaydeder.

```
using System;
namespace QuoteClientProgram {
    class QuoteMain {
        static void Main(string[] args) {
            try {
                IBM.WMQSOAP.Register.Extension();
                Quote q = new Quote();
                Console.WriteLine("Response is: " + q.getQuote("ibm"));
            } catch (Exception e) {
                Console.WriteLine("Exception is: " + e);
            }
        }
    }
}
```

Şekil 34. C# Web hizmeti istemcisi örneği

Şekil 35 sayfa 1473 , JAX-RPC statik yetkili istemci arabirimini kullanan bir Java istemcisine örnek olarak verilebilir. The **`com.ibm.mq.soap.Register.extension();`** method registers the IBM MQ SOAP sender with the service proxy to handle the `jms:` protocol.


```

package org.example.www;
import com.ibm.mq.soap.Register;
public class QuoteClient {
    public static void main(String[] args) {
        try {
            Register.extension();
            QuoteSOAPImplServiceLocator locator = new QuoteSOAPImplServiceLocator();
            System.out.println("Response = "
                + locator.getOrgExampleWwwQuoteSOAPImpl_Wmq().getQuote("IBM"));
        } catch (Exception e) {
            System.out.println("Exception = " + e.getMessage());
        }
    }
}

```

Şekil 35. Java Web hizmeti istemcisi örneği

Kullanıcı çıkışları, API çıkışları ve kurulabilir hizmetler başvurusu

Kullanıcı çıkışlarınızı, API çıkışlarınızı ve kurulabilir hizmet uygulamalarınızı geliştirmenize yardımcı olmak için bu bölümde sağlanan bağlantıları kullanın.

- [“MQIEP yapısı” sayfa 1473](#)
- [“Veri dönüştürme çıkış başvurusu” sayfa 1477](#)
- [“MQ_PUBLISH_EXIT-Yayınlama çıkışı” sayfa 1481](#)
- [“Kanal-çıkış çağrıları ve veri yapıları” sayfa 1489](#)
- [“API çıkış başvurusu” sayfa 1574](#)
- [“Kurulabilir hizmetler arabirimi başvuru bilgileri” sayfa 1635](#)
-  [“Kurulabilir hizmetler arabirimi başvuru bilgileri” sayfa 1697](#)

İlgili kavramlar

[“MQI uygulamaları başvurusu” sayfa 7](#)

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

İlgili başvurular

[“SOAP başvurusu” sayfa 1417](#)

Alfabetik olarak düzenlenmiş SOAP başvuru bilgileri için IBM MQ iletimi.

[“IBM MQ bridge for HTTP için başvuru malzemesi” sayfa 1736](#)

Alfabetik olarak düzenlenmiş, IBM MQ bridge for HTTP için başvuru konuları

[“IBM MQ .NET sınıfları ve arabirimleri” sayfa 1771](#)

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

[“IBM MQ C++ sınıfları” sayfa 1832](#)

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

İlgili bilgiler

[Uygulamaların geliştirilmesi](#)

[Java kitaplıkları için IBM MQ sınıfları](#)

[JMS için IBM MQ Sınıfları](#)

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_KİTAP

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_CMIC_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_SUBRQ_CALL     MQSUBRQ_Call;    /* Address of MQSUBRQ */
PMQ_XCLWLN_CALL    MQXCLWLN_Call;   /* Address of MQXCLWLN */
PMQ_XCNVC_CALL     MQXCNVC_Call;    /* Address of MQXCNVC */
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 çıktıları yazmanız gerekir. Diğer platformlarda, C programlama dilini kullanmanız önerilir.

Bir veri dönüştürme çıkış programı yaratmanıza yardımcı olması için, aşağıdaki bilgiler 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](#)
- [UNIX ve Linux sistemlerinde IBM MQ için veri dönüştürme çıkışı yazılıyor](#)
- [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 188 sayfa 1477 içinde listelenir.

Çizelge 188. Çatı kaynak dosyaları	
Altyapı	Dosya
AIX	amqsvfc0.c
IBM i	QMQMSAMP/QCSRC (AMQSVFC4)

Çizelge 188. Çatı kaynak dosyaları (devamı var)

Altyapı	Dosya
HP-UX	amqsvfc0.c
Linux	amqsvfc0.c
z/OS	CSQ4BAX8 (“1” sayfa 1478) CSQ4BAX9 (“2” sayfa 1478) CSQ4CAX9 (“3” sayfa 1478)
Solaris	amqsvfc0.c
Windows sistemleri	amqsvfc0.c
Notlar: 1. MQXCVNC çağrısını gösterir. 2. Yardımcı program tarafından CICS dışındaki tüm ortamlarda kullanılmak üzere oluşturulan kod parçalarına ilişkin bir sarıcı. 3. Yardımcı program tarafından CICS ortamında kullanılmak üzere oluşturulan kod parçaları için bir sarıcı.	

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, UCS-2 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 891 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)

Windows, UNIX ve Linux sistemleri

crtmqcvx (Create IBM MQ conversion-exit)

➤ **z/OS** 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/OSüzerinde CSQUCVX yardımcı programı çağrılıyor



Şekil 36 sayfa 1479 , CSQUCVX yardımcı programını çağırmak 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.EXITS(MSG1)
```

Şekil 36. CSQUCVX yardımcı programını çağırmak için kullanılan JCL örneği

z/OS veri tanımları



CSQUCVX yardımcı programı, aşağıdaki DDads ile DD deyimleri gerektirir:

DD bildirimi	açıklama
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.

Windows, UNIX ve Linux sistemlerinde hata iletileri

The crrmqcvx command returns messages in the range AMQ7953 through AMQ7970.

Bu iletiler, [İletiler ve neden kodları 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 çağrısı, 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 değiştirme
- İletin konacağı kuyruğu değiştir
- 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, yayının yayınlayıcısına ilişkin bağlamsal bilgiler içerir.

SubContext (MQSBC) - Input/Output

SubContext, yayını alan aboneye ilişkin bağlamsal bilgiler içerir.

MQPSXP-Çıkış veri yapısını yayınlama

MQPSXP yapısı, yayınlama çıkışından geçirilen ve yayınlama çıkışından döndürülen bilgileri açıklar.

[Çizelge 189 sayfa 1482](#), yapıdaki alanları özetler:

Çizelge 189. MQPSXP 'deki alanlar

Alan	Tanım
<u>StrucID</u>	Yapı tanıtıcısı
<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 tanıtıcı
<u>MsgDescPtr</u>	İleti tanımlayıcısının adresi (MQMD)
<u>MsgHandle</u>	İleti özellikleriyle tanıtıcı 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 tanıtıcısıdır. Değer aşağıdaki gibidir:

MQPSXP_STRUCID

MQPSXP_STRUCID , yayınlama çıkış parametresi yapısına ilişkin tanıtıcıdır. 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 çağırılmasına 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_FUNCTION değerini döndürürse, kuyruk yöneticisi iletiyi ölüme mektup kuyruğuna yerleştirdiğinde, ileti için kullanılacak değeri *Feedback* olarak ayarlayın. Çıkıştan dönüşte, *Feedback* özgün değeri MQFB_NONE ise, kuyruk yöneticisi *Feedback* değerini MQFB_STOPPED_BY_PUBSUB_EXIT olarak 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_LENGTH tarafından verilir.

QMGrName , çıkışa ilişkin bir giriş alanıdır.

Hconn (MQHCONN)

Hconn , kuyruk yöneticiyle 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 , işlemekte 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_PROXY değerine 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, MQINQMMP ya da MQDLTMP ileti özelliği işlev çağrıları 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 yoksayılr. 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      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;        /* 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 yayınlayıcıyla ilgili bağlamsal bilgileri içerir; yayınlama çıkışa iletilir.

Çizelge 190 sayfa 1485 , yapıdaki alanları özetler:

Çizelge 190. MQPBC 'deki alanlar	
Alan	Tanım
<u>StrucID</u>	Yapı tanıtıcısı
<u>Version</u>	Yapı sürüm numarası

Çizelge 190. MQPBC 'deki alanlar (devamı var)	
Alan	Tanım
<u>PubTopicString</u>	Konu dizgisini yayınla
<u>MsgDescPtr</u>	İleti tanımlayıcısının adresi (MQMD)

Alanlar

StrucID (MQCHAR4)

StrucID yapı tanıttıcısıdır. Değer aşağıdaki gibidir:

MQPBC_STRUCID

MQPBC_STRUCID , yayın bağlamı yapısına ilişkin tanıttıcıdır. 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 dizesidir.

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 bildirimi-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 geçilir.

Çizelge 191 sayfa 1486 , yapıdaki alanları özetler:

Çizelge 191. MQSBCiçindeki alanlar	
Alan	Tanım
<u>StrucID</u>	Yapı tanıttıcısı
<u>Version</u>	Yapı sürüm numarası
<u>DestinationQMgrName</u>	Hedef kuyruk yöneticisinin adı

Çizelge 191. MQSBCiçindeki alanlar (devamı var)	
Alan	Tanım
<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 548 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 ilişkilendirdiği 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 548, 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 559 ' 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-MQSBC

```
typedef struct tagMQSBC {
    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       PSPProperties;      /* Publish/subscribe properties */
} MQSBC;
```

Kanal-çıkış çağrıları ve veri yapıları

Bu konu derlemi, kanal çıkış programları yazdığınızda kullanabileceğiniz özel IBM MQ çağrıları ve veri yapılarıyla ilgili başvuru bilgileri sağlar.

Bu bilgiler, ürüne duyarlı programlama arabirimi bilgileridir. IBM MQ kullanıcı çıkışlarını aşağıdaki programlama dillerinde yazabilirsiniz:

Altyapı	Programlama dilleri
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 kurallarla ilgili ek bilgi için [“Temel veri tipleri” sayfa 236](#) ' 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 letisi-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 değiştirgeler 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, kanalın davranışını denetleyebilmek için yönetici tarafından ayarlanan parametreleri içerir.

DataLength (MQULENth)-giriş/çıkış

Veri uzunluğu.

Veriler çıkışa göre değişir:

- Bir kanal güvenliği çıkışı için, çıkış çağrıldığında bu değıştirge *AgentBuffer* alanında herhangi bir güvenlik iletisinin uzunluğunu içerir (*ExitReason* MQXR_SEC_MSG ise). Herhangi bir ileti yoksa, bu değır sıfır olur. Çıkış, bu alanı, *ExitResponse* ' u MQXCC_SEND_SEC_MSG ya da MQXCC_SEND_AND_REQUEST_SEC_MSG olarak ayarlarsa, iş ortağına gönderilecek herhangi bir güvenlik iletisinin uzunluğuna ayarlamalıdır. The message data is in either *AgentBuffer* or *ExitBufferAddr*.

güvenlik mesajlarının içeriğı, güvenlik çıkışlarının tek sorumluluğunda.

- Bir kanal letisi çıkışı için, çıkışa çağrıldığında bu parametre iletinin uzunluğunu (iletim kuyruğı üstbilgisi de içinde olmak üzere) içerir. Çıkış, bu alanı, devam etmek için *AgentBuffer* ya da *ExitBufferAddr* ' da iletinin uzunluğuna ayarlamalıdır. Bu değır, iletim kuyruğı üstbilgisinin (MQXQH) uzunluğundan büyük ya da bu uzunluğa eşit olmalıdır.
- Kanal gönderme ya da kanal alma çıkışıysa, çıkış çağrıldığında bu parametre iletim uzunluğunu içerir. Çıkış, bu alanı, devam etmek için *AgentBuffer* ya da *ExitBufferAddr* ' deki iletiminin uzunluğuna 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).

AgentBufferUzunluğu (MQUZE)-giriş

Aracı arabelleğindeki uzunluk.

Bu parametre, başlatma sırasında *DataLength* değırinden 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 AgentBufferUzunluğu)-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 değiştirgenin içerdiği çıkışa ilişkin olarak aşağıdaki komutu içerir:
 - İleti tanımlayıcısını (ileti için bağlam bilgilerini içeren) içeren iletim kuyruğu 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 değişiklikleri yapmak için içeriği *ExitBufferAddr*' e kopyalayın.

Çıkışta iletim kuyruğu üstbilgisinde yapılan değişiklikler denetlenmez; ancak, hatalı değişiklikler, 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 değişiklikleri yapmak için içeriği *ExitBufferAddr*' e kopyalayın.

Verilerin ilk 8 baytı çıkış tarafından değiştirilmemelidir.

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 değiştirge, çıkışta yönetilen depolama arabelleğindeki adresin adresidir; aracının arabelleği varsa ya da yeterince büyük değilse ya da çıkış için daha uygun olduğunda, aracıya ileti ya da iletim verilerini (çıkış tipine bağlı olarak) döndürmeyi seçebileceği bir bellek arabelleğidir.

Çıkışa ilişkin ilk çağırma sırasında, çıkışa geçirilen adres boş değerli. Bundan sonra, her bir çağrıdan çıkışta hangi adres aktarılırsa, çıkışa bir sonraki çağrıldığında bu çıkışa sunulur.

ExitBufferAddr boş değer ise, kullanılan veriler AgentBuffer değiştirgesinden alınır.

ExitBufferAddr boş değerli değilse, kullanılan veriler ExitBufferAddr parametresine işaret edilen arabellekten alınır.

Not: Bu parametre, işaretçi veri tipini desteklemeyen programlama dillerinde yazılan çıkışlar tarafından kullanılmamalıdır.

C çağırma

```
exitname (&ChannelExitParms, &ChannelDefinition,  
&DataLength, &AgentBufferLength, AgentBuffer,  
&ExitBufferLength, &ExitBufferAddr);
```

Çıkışa geçirilen parametreler aşağıdaki 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 çağırma

```
CALL 'exitname' USING CHANNELEXITPARMS, CHANNELDEFINITION,  
                      DATALENGTH, AGENTBUFFERLENGTH, AGENTBUFFER,  
                      EXITBUFFERLENGTH, EXITBUFFERADDR.
```

Çıkışa geçirilen parametreler aşağıdaki 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 çağırısı (ILE)

```
C*..1.....2.....3.....4.....5.....6.....7..  
C                                CALLP    exitname(MQCXP : MQCD : DATLEN :  
C                                ABUFL : ABUF : EBUFL :  
C                                EBUF)
```

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  
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	CMQCPA	,	Channel exit parameter block
CHANNELDEFINITION	CMQCD	,	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 çağırısının 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 yaratılan otomatik tanımlı kanallar, kanal yaratıldığı zaman SSL ya da TLS el sıkışması ortaya çıktığından 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 CMQXPV.
** Channel definition
01 CHANNELDEFINITION.
   COPY CMQCDV.
```

RPG çağırısı (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 1535](#) 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 1496](#) içindeki *Version* alanının tanımına bakın.

MQXWIN-Çıkışta bekle

MQXWINE çağırısı bir olayın oluşmasını bekler. Yalnızca z/OSüzerindeki 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 çağırısının 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 değiştirge, beklenecek olayı açıklar. Bu yapıdaki alanların ayrıntıları için bkz. [“MQXWD-Çıkış bekleme tanımlayıcısı” sayfa 1549](#).

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);
```

Değiştirelileri 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)
```

Değiştirelileri 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 değiştirelileri içerir. İleti kanalı aracısından (MCA) çağrılan her kanal çıkışa geçilir.

Kanal çıkışlarına ilişkin daha fazla bilgi için bkz. “MQ_CHANNEL_EXIT-Kanal çıkışı” sayfa 1489. Bu konudaki açıklama hem ileti kanallarıyla, hem de MQI kanallarıyla ilişkilendirir.

Çıkış adı alanları

Bir çıkış çağrıldığında, *SecurityExit*, *MsgExit*, *SendExit*, *ReceiveExit* ve *MsgRetryExit* 'daki ilgili alan şu anda çağrılmakta olan çıkışa ilişkin adı içerir. Bu alanlardaki adın anlamı, MCA 'nın çalışmakta

olduğu ortama bağlıdır. Burada, dikkat edilenler dışında, ad alanda sola hizalanmış ve boşluk bırakılmaz; ad, alanın uzunluğuna kadar boşluklarla doldurulur. Aşağıdaki açıklamalarda, köşeli ayraçlar ([]) isteğe bağlı bilgileri gösterir:

UNIX sistemleri

Çıkış adı, dinamik olarak yüklenebilir bir modülün ya da kitaplığın adı, o kitaplıkta bulunan bir işlevin adını taşıyan 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 )
```

Ad en çok 128 karakterle sınırlanmıştır.

z/OS

Çıkış adı, LINK ya da LOAD makrosu için EP parametresindeki belirtim için geçerli olan bir yükleme modülünün adıdır. Ad en çok sekiz karakterden oluşan bir değer ile sınırlanmıştır.

Windows

Çıkış adı, o kitaplıkta bulunan bir işlevin adını taşıyan, devingen bağlı bir kitaplığın 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 )
```

Ad en çok 128 karakterle sınırlanmıştır.

IBM i

Çıkış adı, 10 baytlık bir program adı ve ardından 10 baytlık bir kitaplık adıdır. Adlar 10 bayttan daha uzunsa, her ad 10 bayt yapmak için boşluklarla 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 çıkışındaki MQCD alanlarının değiştirilmesi

Bir kanal çıkışı, MQCD 'deki alanları değiştirebilir. Değiştirilen değer MQCD 'de kalır ve bir çıkış zincirindeki geri kalan çıkışlara ve kanal yönetim ortamını paylaşan herhangi bir konuşmaya geçirilir. Değiştirilen MQCD, kanalda devam eden yaşam süresi boyunca olağan işlemesi için MCA tarafından da kullanılır.

Aşağıdaki MQCD alanlarının çıkış tarafından değiştirilmemesi gerekir:

- ChannelName
- ChannelType
- StrucLength
- S\u00f0fcr\u00f0fcm

İlgili başvurular

[“Alanlar” sayfa 1498](#)

Bu konu, MQCD yapısındaki tüm alanları listeler ve her bir alanı açıklar.

[“C bildirimi” sayfa 1522](#)

Bu bildirim, MQCD yapısına ilişkin C bildirimidir.

[“COBOL bildirimi” sayfa 1524](#)

Bu bildirim, MQCD yapısıyla ilgili COBOL bildirimidir.

[“RPG bildirimi \(ILE\)” sayfa 1526](#)

Bu bildirim, MQCD yapısına ilişkin RPG bildirimidir.

[“System/390 çevirici bildirimi” sayfa 1529](#)

Bu bildirim, MQCD yapısı için System/390 çevirici bildirimidir.

[“Visual Basic bildirimi” sayfa 1531](#)

Bu bildirim, MQCD yapısındaki Visual Basic bildirimidir.

[“Kanal çıkışındaki MQCD alanlarının değiştirilmesi” sayfa 1532](#)

Bir kanal çıkışı, MQCD 'deki alanları değiştirebilir. Ancak, listelenen koşullar dışında, bu değişiklikler genellikle üzerinde işlem yapmamış olur.

Alanlar

Bu konu, MQCD 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 özneliğe 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 bildirimi 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_CLUUSSDR ya da MQCHT_CLAUSRCVR ile olan kanallar için geçerlidir.

Bu, çıkışa giriş alanıdır. *Version* değeri MQCD_VERSION_4değerinden 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 değeri 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_ORTAMı
- MQMON_YükSEK

Bu, çıkışa giriş alanıdır. *Version* , MQCD_VERSION_8değerinden 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_ORTAMı
- 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 özniteliği, 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" özniteliğini 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 özniteliği 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 'yi 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 öznitelik, yalnızca istemci-bağlantı kanalı tipi için geçerlidir.

Bu, çıkışa giriş alanıdır. *Sürüm* , MQCD_VERSION_9değerinden 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. Belirttiğiniz 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 addır.
- 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_2değerinden küçükse bu alan yoktur.

DataConversion (MQUZE)

Bu alan, alıcı ileti kanalı aracısının bu dönüşümü gerçekleştiremediğinde, 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* MQCHT_ENDER, MQCHT_SERVER, MQCHT_CLUUSDR ya da MQCHT_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 kanalı için varsayılan yeniden bağlantı özneliği değerini ayarlar.

Varsayılan otomatik istemci yeniden bağlantı seçeneği. Bir istemci uygulamasını otomatik olarak yeniden bağlamak için IBM MQ MQI client konfigürasyonunu tanımlayabilirsiniz. IBM MQ MQI client , bir bağlantı hatasından sonra kuyruk yöneticisine yeniden bağlanmayı dener. Bir MQCONN ya da MQCONN MQI çağrısı veren uygulama istemcisi olmadan yeniden bağlanmaya çalışır.

Yeniden bağlanma bir MQCONNx seçeneğidir. DefReconnect kanal özniteliğini kullanarak, MQCONNseçeneğini kullanan var olan uygulamalara yeniden bağlantı davranışı ekleyebilirsiniz. Ayrıca, MQCONNx' u kullanan uygulamaların yeniden bağlantı davranışını da değiştirebilirsiniz.

Ayrıca, yeniden bağlanma davranışını ayarlamak ya da değiştirmek için mqclient.ini dosyasından DefRecon değerini de ayarlayabilirsiniz. mqclient.ini dosyasındaki DefRecon değeri, DefReconnect kanal özniteliğinden önceliklidir.

Sözdizimi

DefReconnect (MQRCN_NO | MQRCN_YES | MQRCN_Q_MGR | MQRCN_DISABLED)

Parametreler

MQRCN_NO

Varsayılan değer, 'MQRCN_NO ' değeridir.

MQCONNx tarafından geçersiz kılınmadıkça, istemci otomatik olarak yeniden bağlanmaz.

MQRCN_YES

MQCONNx tarafından geçersiz kılınmadıkça, istemci otomatik olarak yeniden bağlanır.

MQRCN_Q_MGR

MQCONNx tarafından geçersiz kılınmadıkça, istemci otomatik olarak yeniden bağlanır, ancak yalnızca aynı kuyruk yöneticisi tarafından değiştirilir. MMGR seçeneği, MQCNO_RECONNECT_Q_MGR ile aynı etkiye sahiptir.

MQRCN_DISABLED

Reconnection is disabled, even if requested by the client program using the MQCONNx MQI call.

Otomatik istemci yeniden bağlantısı, Java için IBM MQ sınıfları tarafından desteklenmez.

Çizelge 192. Otomatik yeniden bağlantı, uygulamadaki ve kanal tanımlamasında belirlenen değerlere bağlıdır.				
DefReconnect	Yeniden bağlanma seçenekleri uygulama içinde ayarlandı			
	MQCNO_RECONNE CT	MQCNO_RECONNE CT_Q_MGR	MQCNO_RECONNE CT_AS_DEF	MQCNO_RECONNE CT_DISABLED
MQRCN_NO	EVET	MMGR	HAYIR	HAYIR
MQRCN_YES	EVET	MMGR	EVET	HAYIR
MQRCN_Q_MGR	EVET	MMGR	MMGR	HAYIR
MQRCN_DISABLED	HAYIR	HAYIR	HAYIR	HAYIR

İlgili başvurular

Bağlanma seçenekleri

MQCONNx 'in işlemini denetleyen seçenekler.

İlgili bilgiler

[Otomatik istemci yeniden bağlantısı](#)

[Kanal ve istemci yeniden bağlantı](#)

[İstemci yapılandırma dosyasının STANA kısmı](#)

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 özneliğiyle 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 boşta durma bağlantı kesme değerini gösterir. Bir sunucu bağlantısı, bu süre için iş ortağı istemcisinden iletişim alamazsa, bağlantıyı sona erdirir. Sunucu bağlantısı boşta 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 öznelik, 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_4 değerinden 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_4 değerinden 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_8 değerinden küçükse bu alan yoktur.

HeartbeatInterval (MQUZE)

Bu alan, sağlıklı işletim bildirimi akışları arasındaki süreyi 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 olmadığında, gönderen MCA ' dan geçirilen sağlıklı işletim bildirimi 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 bildirimi 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 olmadığında, sağlıklı işletim bildirimi akışı arasındaki saniye cinsinden süredir. Bu, kanalın verimli bir şekilde sürdürülmesini 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 yığınının 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 de geçerlidir:

MQKI_AUTO

Otomatik.

Bu değer, canlı tutma aralığının, kararlaştırılan sağlıklı işletim bildirimi aralığından aşağıdaki gibi hesaplandığı anlamına gelir:

- Kararlaştırılan sağlıklı işletim bildirimi aralığı sıfırdan büyükse, kullanılan canlı tutma aralığı, sağlıklı işletim bildirimi aralığı artı 60 saniyedir.
- Kararlaştırılan sağlıklı işletim bildirimi 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 değiştirilmesi 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 yığınının 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.

Bu alan yalnızca, bir MQXPT_TCP *TransportType* olan kanallar ve *ChannelType* MQCHT_SENDER, MQCHT_SERVER, MQCHT_REQUESTER, MQCHT_CLNTCONN, MQCHT_CLUSSDR ya da MQCHT_CLAUSRCVRchannelsalanı için geçerlidir.

Bu alanın uzunluğu MQ_LOCAL_ADDRESS_LENGTH tarafından verilir. *Version* , MQCD_VERSION_7değerinden 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_6değerinden 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 alanıdır. *Version* değeri MQCD_VERSION_6değerinden 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_CLUUSDR 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, çıkış girişi/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 (IBM i, UNIXve Windows).

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 artırıyorsa, 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_2değerinden küçükse bu alan yoktur.

ModeName (MQCHAR8)

Bu alan, LU 6.2 kip adını belirtir.

Bu alan, yalnızca iletim protokolü (*TransportType*) MQXPT_LU62ise 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_8değerinden 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.

- Kanalin kullanıma hazırlanması ve sona erdirilmesi.

This field is not relevant for channels with a *ChannelType* of MQCHT_SVRCONN or MQCHT_CLNTCONN; a message exit is never invoked for such channels.

Çeşitli ortamlardaki bu alanın içeriğine ilişkin açıklamalar için [“MQCD-Kanal tanımlaması” sayfa 1496](#) ' e bakın.

Bu alanın uzunluğu MQ_EXIT_NAME_LENGTH değeri tarafından verilir.

Not: Bu değışmezin değeri ortamdan özgüdür.

MsgExitPtr (MQPTR)

Bu alan, ilk *MsgExit* alanının adresini belirtir.

MsgExitsDefined sıfırdan büyükse, bu adres zincirdeki her bir kanal iletisi çıkışısının adları listesinin adresidir.

Her ad, *ExitNameLength* uzunluktaki bir alanda, boşluklarla doldurulmuğa doldurulur. Her bir çıkış için birden bire bir *MsgExitsDefined* alanı bitişik olanfieldsalanları 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_4değerinden 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_4değerinden 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 iletidir; *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 ileti yeniden deneme çıkış adını belirtir.

İleti yeniden deneme çıkışı, MCA 'nın MQOPED ya da MQPUT çağrısından MQCC_FAILED tamamlanma kodu aldığı anda, 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 denemesini 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 ileti (gönderen tarafından belirtilmişse) oluşturulmasını ve özgün iletiyi 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 iletiyi yeniden teslim etme girişiminde bulunmadan önce beklemeyi gerçekleştirmeden hemen önce
- Kanalin kullanıma hazırlanması ve sona erdirilmesi

Çeşitli ortamlardaki bu alanın içeriğine ilişkin açıklamalar için [“MQCD-Kanal tanımlaması” sayfa 1496](#) ' 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 değişmezin 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 iletiyi 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 değer döndürürse bekleme süresi olarak da kullanılır.

- *MsgRetryExit* boş değilse, *MsgRetryInterval* özniteliği MCA 'yı etkilemez; bunun yerine, MCA' nın ne kadar bekleyeceğini belirleyen, iletiyle 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 ileti 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 alanları 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.

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.

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, çıkışa giriş alanıdır. *Version* değeri MQCD_VERSION_5 değerinden küçükse bu alan yoktur.

Version değeri MQCD_VERSION_6 değerinden 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 sağlanır. Ancak, kalıcı olmayan iletilerin her birine göre sırası korunur. Benzer şekilde, her biri ile görelî kalıcı iletilerin sırası korunur.

Parola (MQCHAR12)

Bu alan, uzak ileti kanalı aracılığıyla güvenli bir SNA oturumu başlatma girişimi sırasında ileti kanalı aracısının kullandığı parolayı belirtir.

Bu alan yalnızca UNIX sistemler ve Windows üzerinde 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_2 değerinden 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 **mqext.**, all message properties are delivered to the application in an MQRFH2 header. Tersî 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

İletin tüm özellikleri, ileti uzak kuyruk yöneticisine gönderildiğinde iletiyle birlikte verilir. İleti tanımlayıcısındaki (ya da uzantısındaki) 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 128

“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/OSüzerinde desteklenir.

MQPA_ONLY_MCA

Varsayılan kullanıcı kimliği kullanılır. Ağdan alınan herhangi bir kullanıcı kimliği kullanılmaz. Bu değer yalnızca z/OSüzerinde desteklenir.

QMGrName (MQCHAR48)

Bu alan, bir çıkışta bağlantı kurabileceği kuyruk yöneticisinin adını belirtir.

MQCHT_CLNTCONN dışında bir *ChannelType* ile kanallar için bu alan, bir çıkışa bağlanabileceği kuyruk yöneticisinin adıdır; UNIX, Linux ve Windows sistemlerinde her zaman boş değildir.

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.
- Kanalin kullanıma hazırlanması ve sona erdirilmesi.

Çeşitli ortamlardaki bu alanın içeriğine ilişkin açıklamalar için [“MQCD-Kanal tanımlaması” sayfa 1496](#) ' 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 doldurulmuğa doldurulur. Her bir çıkış için birden bire bir *ReceiveExitsDefined* alanı bitişik olan fieldsalanları vardır.

Bir çıkış tarafından bu adlarda yapılan değişiklikler korunur, ancak ileti kanalı çıkışı belirttik bir işlem yapmasa da, hangi çıkışlar çağrıldığı değişmez.

ReceiveExitsDefined 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.

ReceiveExitsTanımlı (MQUZE)

Bu alan, zincirde tanımlanan kanal alma çıkışlarının sayısını belirtir.

Sıfır değerinden büyük ya da sıfıra eşit.

Bu, çıkışa giriş 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

görülebilen 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.

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 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.

ReceiveExitsDefined 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.

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, ortamdaki 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_2değerinden küçükse bu alan yoktur.

Version değeri MQCD_VERSION_3değerinden küçükse, bu yapıdaki şu alanlar 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_6değerinden küçükse bu alan yoktur.

Version değeri MQCD_VERSION_7değerinden 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_6değerinden 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, istemci ortamdan elde edilen bir kullanıcı kimliği bulunabilir. Çıkış, bu kullanıcı kimliğinin geçerliliğini denetleyebilir (olasılıkla *RemotePassword* parolasıyla birlikte) ve değeri *MCAUserIdentifier* (sayfa.) olarak güncelleyin.

İstemci bir güvenlik çıkışı varsa, bu bilgiler istemci ortamdan bir güvenlik akışında elde edilebilir.

Bu alanın uzunluğu MQ_USER_ID_LENGTH tarafından verilir. *Version* , MQCD_VERSION_2değerinden 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.
- Kanalin kullanıma hazırlanması ve sona erdirilmesi.

Çeşitli ortamlardaki bu alanın içeriğine ilişkin açıklamalar için “MQCD-Kanal tanımlaması” sayfa 1496 ' 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.

- Kanalin kullanıma hazırlanması ve sona erdirilmesi.

Çeşitli ortamlardaki bu alanın içeriğine ilişkin açıklamalar için [“MQCD-Kanal tanımlaması” sayfa 1496](#) ' e bakın.

Bu alanın uzunluğu MQ_EXIT_NAME_LENGTH değeri tarafından verilir.

Not: Bu değişimin 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 doldurulmuş 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 göstergeci.

Programlama dilinin işaretçi veri tipini desteklemediği altyapılarda, bu alan uygun uzunluğun byte dizisi olarak bildirilir.

Bu, çıkış girişi 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, çıkış girişi alanıdır. *Version* değeri MQCD_VERSION_4 değerinden 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 i ile ilgili değildir.

SendUserDataPtr (MQPTR)

Bu alan, *SendUserData* alanının adresini belirtir.

SendExitsDefined 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 fields alanları 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.

SendExitsDefined sıfırsa, bu alan boş değerli göstergeci.

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.

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 MQ V7.0 davranışı için değer alt sınırı değeridir. Kanal yönetim ortamında yalnızca bir sohbete izin verilse de, ileriye doğru okuma, zamanuyumsuz tüketim ve CLNTCONN-SVRCONN sağlıklı işletim sistemi durdurma ve susturma kanal durdurma davranışlarının Sürüm 7 davranışı kullanılabilir.

Bu, çıkışa giriş alanıdır. *Version* değeri MQCD_VERSION_9 değerinden küçükse, bu değer yoktur.

Bu alanın varsayılan değeri 10 'tır.

Not: *MaxInstances* ve *MaxInstancesPerClient* bir kanala uygulanan sınırlar, bu eşgörünümleri paylaşabilecek etkileşim sayısını değil, kanal eşgörünümlerinin sayısını sınırlıyor.

ShortConnectionAdı (MQCHAR20)

Bu alan, bir bağlantı adının ilk 20 baytını belirtir.

Version alanı MQCD_VERSION_1 ise, *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.

SSLCipherSpec (MQCHAR32)

Bu alan, SSL kullanılırken kullanımda olan şifreleme belirtiminini belirtir.

SSLCipherSpec boşsa, kanal SSL kullanmıyor demektir. Bu alan boş değilse, bu alan, kullanımda CipherSpec belirtimini belirten bir dizgi içerir.

Bu parametre tüm kanal tipleri için geçerlidir. Bu, AIX, HP-UX, Linux, IBM i, Solaris, Windows ve z/OS üzerinde desteklenir. 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_7 değerinden küçükse bu alan yoktur.

SSLClientAuth (MQUZE)

Bu alan, SSL istemci 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_7 değerinden küçükse bu alan yoktur.

SSLPeerNameUzunluğu (MQUZE)

Bu alan, *SSLPeerNamePtr* tarafından işaret edilen SSL eş adının bayt cinsinden uzunluğunu belirtir.

Bu, çıkışa giriş alanıdır. *Version* değeri MQCD_VERSION_7 değerinden küçükse bu alan yoktur.

SSLPeerNamePtr (MQPTR)

Bu alan, SSL eş düzey adının adresini belirtir.

Başarılı bir SSL 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 MQ v7.1 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.

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 UNIX sistemleri ve Windows üzerinde boş bırakılabilir ve yalnızca *ChannelType* MQCHT_SENDER, MQCHT_SERVER, MQCHT_REQUESTER ya da MQCHT_CLNTCONN 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.

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 içermek 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.

```
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];	/* SSL CipherSpec */
MQPTR	SSLPeerNamePtr;	/* Address of SSL peer name */

```

MQLONG    SSLPeerNameLength;      /* Length of SSL peer name */
MQLONG    SSLClientAuth;          /* Whether SSL 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 */
};

```

COBOL bildirimi

Bu bildirim, MQCD yapısıyla ilgili COBOL bildirimidir.

```

** 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-MCNAME 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-MAXMSGLENGTH 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-REMPASSWORD 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-NONPERSISTENTMSGSPD PIC S9(9) BINARY.
** Length of MQCD structure
15 MQCD-STRUCLENGTH 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 **
** SSL CipherSpec
15 MQCD-SSLCIPHERSPEC PIC X(32).
** Address of SSL peer name
15 MQCD-SSLPEERNAMEPTR POINTER.
** Length of SSL peer name
15 MQCD-SSLPEERNAMELENGTH PIC S9(9) BINARY.
** Whether SSL 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-BATCHDATA LIMIT 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 **

```

RPG bildirimi (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 CDCHT 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 CDRCd 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 CDML 1097 1100I 0
D* Length of long remote user identifier
D CDRL 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* SSL CipherSpec
D CDSCS 1217 1248
D* Address of SSL peer name
D CDSPN 1249 1264*
D* Length of SSL peer name
D CDSP 1265 1268I 0
D* Whether SSL 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.

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
*			
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
*			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
*			data
MQCD_MSGRETRYCOUNT	DS	F	Number of times MCA will try to put the message, after the first attempt has failed
*			Minimum interval in milliseconds after which the open or put operation will be retried
MQCD_MSGRETRYINTERVAL	DS	F	
*			Time in seconds between heartbeat flows
MQCD_BATCHINTERVAL	DS	F	Batch duration
MQCD_NONPERSISTENTMSGSPED	DS	F	Speed at which nonpersistent messages are sent
*			Length of MQCD structure
MQCD_STRUCLNGTH	DS	F	
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
*			Address of first SENDEXIT field
MQCD_SENDEXITPTR	DS	F	
MQCD_SENDUSERDATAPTR	DS	F	Address of first SENDUSERDATA field
*			Address of first RECEIVEEXIT field
MQCD_RECEIVEEXITPTR	DS	F	
*			Address of first RECEIVEUSERDATA field
MQCD_RECEIVEUSERDATAPTR	DS	F	
*			Address of a list of cluster names
MQCD_CLUSTERPTR	DS	F	
*			Number of clusters to which the channel belongs
MQCD_CLUSTERSDEFINED	DS	F	
*			Network priority
MQCD_NETWORKPRIORITY	DS	F	
MQCD_LONGMCAUSERIDLENGTH	DS	F	Length of long MCA user identifier
*			Length of long remote user identifier
MQCD_LONGREMOTEUSERIDLENGTH	DS	F	
*			Address of long MCA user identifier
MQCD_LONGMCAUSERIDPTR	DS	F	
*			Address of long remote user identifier
MQCD_LONGREMOTEUSERIDPTR	DS	F	
*			MCA security identifier
MQCD_MCASESECURITYID	DS	XL40	
MQCD_REMOTESECURITYID	DS	XL40	Remote security identifier
MQCD_SSLCIPHERSPEC	DS	CL32	SSL CipherSpec
MQCD_SSLPEERNAMEPTR	DS	F	Address of SSL peer name
MQCD_SSLPEERNAMELENGTH	DS	F	Length of SSL peer name
MQCD_SSLCLIENTAUTH	DS	F	Whether SSL client authentication is required
*			Keepalive interval
MQCD_KEEPAIVEINTERVAL	DS	F	
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
*			Message property control
MQCD_PROPERTYCONTROL	DS	F	
*			Limit on sharing conversations
MQCD_SHARINGCONVERSATIONS	DS	F	
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_USEDLO	DS	F	Use dead-letter queue
MQCD_DEFRECONNECT	DS	F	Default client reconnect option
MQCD_LENGTH	EQU	*-MQCD	

Visual Basic bildirimi

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'
ClusterPtr	As MQPTR	'ReceiveUserData field'
ClustersDefined	As Long	'Address of a list of cluster'
NetworkPriority	As Long	'names'
LongMCAUserIdLength	As Long	'Number of clusters to which the'
LongRemoteUserIdLength	As Long	'channel belongs'
LongMCAUserIdPtr	As MQPTR	'Network priority'
LongRemoteUserIdPtr	As MQPTR	'Length of long MCA user'
MCASecurityId	As MQBYTE40	'identifier'
RemoteSecurityId	As MQBYTE40	'Length of long remote user'
SSLCipherSpec	As String*32	'identifier'
SSLPeerNamePtr	As MQPTR	'Address of long MCA user'
SSLPeerNameLength	As Long	'identifier'
SSLClientAuth	As Long	'Address of long remote user'
KeepAliveInterval	As Long	'identifier'
LocalAddress	As String*48	'MCA security identifier'
BatchHeartbeat	As Long	'Remote security identifier'
HdrCompList(0 to 1)	As Long2	'SSL CipherSpec'
MsgCompList(0 To 15)	As Long16	'Address of SSL peer name'
CLWLChannelRank	As Long	'Length of SSL peer name'
CLWLChannelPriority	As Long	'Whether SSL client'
CLWLChannelWeight	As Long	'authentication is required'
ChannelMonitoring	As Long	'Keepalive interval'
ChannelStatistics	As Long	'Local communications address'
End Type		'Batch heartbeat interval'
		'Header data compression list'
		'Message data compression list'
		'Channel Rank'
		'Channel priority'
		'Channel Weight'
		'Channel Monitoring control'
		'Channel Statistics'

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 alan 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 SharingConversations FALSE olarak ayarlandığı sürece herhangi bir kanaldaki ChannelName alanını değiştirebilir. MQCXP SharingConversations değeri ne olursa olsun, yalnızca bir güvenlik çıkışı MCAUserIdentifier alanını değiştirebilir.

Alan	Çıkış Neden Kodu	Çıkış Tipi	Kanal Tipi
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

Alan	Çıkış Neden Kodu	Çıkış Tipi	Kanal Tipi
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
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

Alan	Çıkış Neden Kodu	Çıkış Tipi	Kanal Tipi
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
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

Alan	Çıkış Neden Kodu	Çıkış Tipi	Kanal Tipi
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 1535](#)

Bu konu, MQCXP yapısındaki tüm alanları listeler ve her bir alanı açıklar.

[“C bildirimi” sayfa 1546](#)

Bu bildirim, MQCXP yapısına ilişkin C bildirimidir.

[“COBOL bildirimi” sayfa 1546](#)

Bu bildirim, MQCXP yapısına ilişkin COBOL bildirimidir.

[“RPG bildirimi \(ILE\)” sayfa 1547](#)

Bu bildirim, MQCXP yapısına ilişkin RPG bildirimidir.

[“System/390 çevirici bildirimi” sayfa 1548](#)

Bu bildirim, MQCXP yapısına ilişkin System/390 çevirici bildirimidir.

Alanlar

Bu konu, MQCXP yapısındaki tüm alanları listeler ve her bir alanı açıklar.

StrucId (MQCHAR4)

Bu alan, yapı tanıtıcısını belirtir.

Değer şu olmalıdır:

MQCXP_STRUC_ID

Kanal çıkış parametresi yapısıyla ilgili tanıtıcı.

C programlama dili için, sabit MQCXP_STRUC_ID_ARRAY değişmezi de tanımlanır; bu değişmez, MQCXP_STRUC_ID ile aynı değere sahiptir, ancak dizgi yerine bir dizi karakter dizisidir.

Bu, çıkışa giriş alanıdır.

Sürüm (MQUZE)

Bu alan, yapı sürüm numarasını belirtir.

Değer, ortama bağlıdır:

MQCXP_VERSION_1

Version-1 kanal çıkışı parametre yapısı.

MQCXP_VERSION_2

Version-2 kanal çıkışı parametre yapısı.

Alan, HP Integrity NonStop Server ortamında bu değere sahiptir.

MQCXP_VERSION_3

Version-3 kanal çıkışı parametre yapısı.

Alan şu ortamlarda bu değere sahiptir: UNIX sistemleri başka bir yerde listelenmez.

MQCXP_VERSION_4

Version-4 kanal çıkış parametresi yapısı.

MQCXP_VERSION_5

Version-5 kanal çıkışı parametre yapısı.

MQCXP_VERSION_6

Version-6 kanal çıkışı parametre yapısı.

MQCXP_VERSION_8

Version-8 kanal çıkışı parametre yapısı.

Alan şu ortamlarda bu değere sahiptir: z/OS, AIX, HP-UX, Linux, IBM i, Solaris, Windows.

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 değiştirge yapısının yürürlükteki sürümü.

Değer, ortama bağlıdır.

Not: MQCXP yapısının yeni bir sürümü kullanıma sunulduğunda, var olan parçanın yerleşim 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 alan çağrılmakta olan çıkış tipini belirtir ve çıkış yordamlarına giriş için ayarlanır.

Aşağıdaki değerler olanaklı:

MQXT_CHANNEL_SEC_EXIT

Kanal güvenlik çıkışı.

MQXT_CHANNEL_MSG_EXIT

Kanal ileti çıkışı.

MQXT_CHANNEL_SEND_EXIT

Kanal çıkışı çıkışı.

MQXT_CHANNEL_RCV_EXIT

Kanal çıkış çıkışı.

MQXT_CHANNEL_MSG_RETRY_EXIT

Kanal iletisi-yeniden deneme çıkışı.

MQXT_CHANNEL_AUTO_DEF_EXIT

Kanal otomatik tanımlama çıkışı.

z/OS üzerinde, bu çıkış tipi yalnızca MQCMT_CLUSSDR ve MQCMT_CLAUSRCVR tipi kanallar için desteklenir.

Bu, çıkışa giriş alanıdır.

ExitReason (MQUZE)

Bu alan, çıkışa neden çağrılmakta olduğunu ve çıkış yordamlarına giriş için ayarlanıp belirlenmemesine neden olur.

Otomatik tanımlama çıkışı tarafından kullanılmaz. Aşağıdaki değerler olanaklı:

MQXR_INIT

Başlatma işleminden çıkılıyor.

Bu değer, çıkışa ilk kez çağrılmakta olduğunu gösterir. Çıkışa, gereksinim duyduğu kaynakları (örneğin: bellek) edinip kullanıma hazırlamaya olanak tanır.

MQXR_TERM

Sondan çıkın.

Bu değer, çıkışa son verilmek üzere olduğunu gösterir. Çıkış, kullanıma hazırlandığı için (örneğin: bellek) kullanıma hazırlandığı tüm kaynakları serbest kılmalıdır.

MQXR_MSG

Bir iletiyi işle.

Bu değer, bir iletiyi işlemek için çıkışa çağrıldığını gösterir. Bu değer yalnızca kanal iletileri çıkışları için gerçekleşir.

MQXR_XMIT

İletişimi işleme.

Bu değer, kanal gönderme ve alma çıkışları için gerçekleşir.

MQXR_SEC_MSG

Güvenlik iletileri 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 iletileri olup olmadığı ya da işlememesinin bir güvenlik iletileri 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 iletileri vardır. İşleyecek bir güvenlik iletileri 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 değiştiricileri

Bu değer yalnızca güvenlik çıkışları için geçerlidir ve çıkışa MQCSP yapısının geçirildiğini gösterir. Daha fazla bilgi için bkz. [“MQCSP-Güvenlik değiştiricileri” sayfa 328](#)

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 iletisi oluşturulmaz. Rapor iletisi başarılı bir şekilde oluşturulamazsa, özgün ileti iletim kuyruğunda da bırakılır.

İletin 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 iletisi yeniden deneme çıkışı için, bu değer MCA 'nın beklediği ve iletiyi yeniden denediğini gösterir; bunun yerine, MCA olağan hata işlemleriyle hemen devam eder (ilet, 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 hiçbirini 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 iletisi gönder.

Bu değer yalnızca kanal güvenliği çıkışı ile ayarlanabilir. Çıkışta, iş ortağına iletilmesi gereken bir güvenlik iletisi 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 1496](#)). Bekleyen yeniden deneme sayısı, *MsgRetryCount* özneliğinin 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

İletin 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, *ExitBufferAddr* içinde olduğunu belirtir.

MQXR2_USE_AGENT_BUFFER ve MQXR2_USE_EXIT_BUFFER ' ın yalnızca biri ayarlanmalıdır.

MQXR2_DEFAULT_CONTINUATION

Varsayılan devamı.

Zincirin sonraki çıkışıyla devam et, çağrılan son çıkıştan gelen yanıtı bağlar:

- 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, çıkış girişi/çıkış alanıdır.

Geribildirim (MQUZE)

Bu alan, geribildirim kodunu belirtir.

Bu alan, çıkış yordamlarına girişte MQFB_NONE olarak ayarlanır.

Bir kanal ileti 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 iletilişi 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 iletilisi 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 sıkıştırılamamalıdır.

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, istemciye oturma 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)

Bu alan, yetenek işaretlerini belirtir.

Şu öge tanımlanır:

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/OSüzerinde 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 yoksayılr. 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 iletiyi görüntüler 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 iletiyi görüntüler 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 SSL kanalı değilse, değer sıfırdır.

SSLRemCertIssNamePtr (PMQVOID)

Bu alan, uzak sertifikanın yayıncısının tam Ayırt Edici Adı 'nın adresini belirtir.

Bu değer, SSL kanalı değilse, boş değerli göstergedir.

Bu, çıkışa giriş alanıdır. *Version* değeri MQCXP_VERSION_6değerinden küçükse bu alan yoktur.

Not: Özel Ayırt Edici Adı ve Sertifika Veren Ayırt Edici Adı 'nın belirlenmesinde kanal güvenliği çıkışlarının davranışı, IBM MQ v7.1yayınında değiştirilmiş olabilir. 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 göstergedir.

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 değer, MQCD ' nin HdrCompList (Liste) alanından erişilen, anlaşmalı desteklenen değerlerden birine kanala gönderilen bir kanaldan çıkış yoluyla değiştirilebilir. Bu, iletinin içeriğine dayalı olarak her ileti için seçilebilecek üstbilgi verilerini sıkıştırmak için kullanılan tekniğin geçerli olmasını sağlar. Değiştirilen değer, yalnızca geçerli ileti için kullanılır. Öznitelik desteklenmeyen bir değere değiştirilirse kanal sona erer. Bir gönderme kanalının ileti çıkışı dışında değiştirilirse, değer yoksayılır.

Bu, çıkışa giriş/çıkış alanıdır. *Version* değeri MQCXP_VERSION_6değerinden küçükse bu alan yoktur.

CurMsgSıkıştırma (MQUZE)

Bu alan, ileti verilerini sıkıştırmak için kullanılmakta olan tekniğin kullanılacağını belirtir.

Aşağıdakilerden birine ayarlanır:

MQCOMPRESS_NONE

Üstbilgi 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.

Bu değer, MQCD ' nin MsgCompList (MsgComp) Listesi alanından erişilen, anlaşmalı desteklenen değerlerden birine kanala gönderilen bir kanaldan çıkış yoluyla değiştirilebilir. Bu, iletinin içeriğine dayalı olarak her ileti için karar verilmek üzere ileti verilerini sıkıştırmak için kullanılan tekniğin kullanılmasını

sağlar. Değiştirilen değer, yalnızca geçerli ileti için kullanılır. Öznitelik desteklenmeyen bir değere değiştirilirse kanal sona erer. Bir gönderme kanalının ileti çıkışı dışında değiştirilirse, değer yoksayılır.

Bu, çıkışa giriş/çıkış alanıdır. *Version* değeri MQCXP_VERSION_6değerinden küçükse bu alan yoktur.

Hconn (MQHCONN)

Bu alan, çıkışta herhangi bir MQI çağrısı yapması gerekiyorsa, çıkışa ilişkin bağlantı tanıtıcısını belirler.

Bu alan, istemci-bağlantı kanallarında çalışan MQHC_UNUABLE_HCONN (-1) değerini içerdiği için çıkışa uygun değildir.

Bu, çıkışa giriş alanıdır. *Version* , MQCXP_VERSION_7' den küçükse bu alan yoktur.

SharingConversations (MQBOOL)

Bu alan, bu kanal yönetim ortamında çalışmakta olan tek bir etkileşim olup olmadığını ya da bu kanal somut örneğinde birden çok konuşmanın çalıştırılıp çalıştırılmayacağını belirtir.

Ayrıca, çıkış programının, aynı anda çalışan başka bir çıkış programı tarafından değiştirilmekte olan MQCD 'nin riskine tabi olup olmadığını da gösterir.

Bu alan yalnızca istemci-bağlantı ya da sunucu bağlantısı kanallarında çalışan çıkış programları için anlamlıdır.

Aşağıdakilerden birine ayarlanır:

YANLIŞ

Çıkış eşgörünümü, bu kanal yönetim ortamında çalışmakta olan tek çıkış eşgörünümdür. Bu, diğer kanal örneklerinde çalışan diğer çıkışlardan çekişme olmadan, MQCD alanlarını güvenli bir şekilde güncellemek için bu çıkışa olanak sağlar. MQCD alanlarına ilişkin değişikliklerin kanalın üzerinde işlem yapılıp yapılamayacağı, [“Kanal çıkışındaki MQCD alanlarının değiştirilmesi” sayfa 1532](#) içindeki MQCD alanları çizelgesiyle tanımlanır.

DOĞRU

Çıkış eşgörünümü, bu kanal yönetim ortamında çalıştırılmakta olan tek çıkış eşgörünümdür değil. MQCD 'de yapılan değişiklikler, MQXR_INIT dışındaki Reasons 'tan çıkmak için [“Kanal çıkışındaki MQCD alanlarının değiştirilmesi” sayfa 1532](#) içindeki MQCD alanları çizelgesinde listelenen değişiklikler 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 bildirimi

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
    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 SSL certificate */
    MQLONG     SSLRemCertIssNameLength; /* Length of
                                distinguished name of issuer
                                of remote SSL certificate */
    MQPTR      SSLRemCertIssNamePtr; /* Address of
                                distinguished name of issuer
                                of remote SSL 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 bildirimi

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-EXITSPACE PIC S9(9) BINARY.
** User Id associated with remote certificate
15 MQCXP-SSLCERTUSERID PIC X(12).
** Length of distinguished name of issuer of remote SSL
** certificate
15 MQCXP-SSLREMCERTISSNAMELENGTH PIC S9(9) BINARY.
** Address of distinguished name of issuer of remote SSL
** 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.

```

RPG bildirimi (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 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 SSL certificate		
D	CXSSLCU	161	172
D*	Length of distinguished name of issuer of remote SSL certificate		
D	CXSRCINL	173	176I 0
D*	Address of distinguished name of issuer of remote SSL 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

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_EXITSPEACE	DS	F	Number of bytes in transmission buffer reserved for exit to use
*			
MQCXP_SSLCERTUSERID	DS	CL12	User identifier associated with remote SSL certificate
*			

MQCXP_SSLREMCERTISSNAMELENGTH	DS	F	Length of distinguished name of issuer of remote SSL certificate
* MQCXP_SSLREMCERTISSNAMEPTR	DS	F	Address of distinguished name of issuer of remote SSL 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_LENGTH	EQU		*-MQCXP
	ORG		MQCXP
MQCXP_AREA	DS		CL(MQCXP_LENGTH)

MQXWD-Çıkış bekleme tanımlayıcısı

MQXWD yapısı, MQXMK çağrısındaki bir giriş/çıkış değiştirir.

Bu yapı yalnızca z/OSüzerinde desteklenir.

İlgili başvurular

[“Alanlar” sayfa 1549](#)

Bu konu, MQXWD yapısındaki tüm alanları listeler ve her bir alanı açıklar.

[“C bildirimi” sayfa 1550](#)

Bu bildirim, MQXWD yapısına ilişkin C bildirimidir.

[“System/390 çevirici bildirimi” sayfa 1550](#)

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 tanıtıcısını belirtir.

Değer şu olmalıdır:

MQXWD_STRUC_ID

Çıkış bekleme tanımlayıcı yapısıyla ilgili tanıtıcı.

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, beklenecek olay denetim bloğunu belirtir.

Bu alan, beklenecek 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 1551](#)

Çıkışta kullanılan yapı veri tipleri aşağıda açıklanmıştır:

- [“MQXCLWLN -Küme iş yükü kayıtlarında dolaşma” sayfa 1553](#)
- [“MQWXP -Küme iş yükü çıkış değiştirgesi yapısı” sayfa 1556](#)
- [“MQWDR-Küme iş yükü hedef kaydı yapısı” sayfa 1564](#)
- [“MQWQR -Küme iş yükü kuyruğu kayıt yapısı” sayfa 1568](#)
- [“MQWCR -Küme iş yükü kümesi kayıt yapısı” sayfa 1573](#)
-  [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 193. Kuyruk yöneticisi öznitelikleri	
Tam ad	MQSC ' de kullanılan ad
<i>ClusterWorkloadData</i>	CLWLDATA
<i>ClusterWorkloadExit</i>	CLAXLEXIT
<i>ClusterWorkloadLength</i>	CLWLEN

Çizelge 194. Kuyruk öznitelikleri	
Tam ad	MQSC ' de kullanılan ad
<i>DefBind</i>	DEFBIND
<i>DefPersistence</i>	DEĞERLERİ
<i>DefPriority</i>	DEĞİŞMEZ
<i>InhibitPut</i>	PUT
<i>QDesc</i>	TASARIMLA

İlgili bilgiler

[Küme iş yükü çıkışlarının yazılması ve derlenmesi](#)

MQ_CLUSTER_WORKLOAD_EXIT -Arama açıklaması

Küme iş yükü çıkışı, bir iletiyi kullanılabilir bir kuyruk yöneticisine yönleltmek 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 öznitelemesiyle 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 sağlanıyor. 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 çıkış işlevinizin adını koyun.

MQ_CLUSTER_WORKLOAD_EXIT parametrelerini aşağıdaki gibi bildirin:

```
MQWXP ExitParms; /* Exit parameter block */
```

High Level Assembler çağırısı

```
CALL EXITNAME, (EXITPARMS)
```

Değiştirgeleri aşağıdaki gibi bildirin:

EXITPARMS	CMQWXA	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 önbellege, kümeyle ilgili bilgileri saklamak için kullanılan ana saklama alanı alanıdır.

Küme önbellege durağansa, sabit bir boyutu vardır. Bu değeri dinamik olarak ayarlarsanız, küme önbellege gerektiği gibi genişletebilir.

Bir sistem parametresi ya da makrosu kullanarak küme önbellege tipini STATIC ya da DYNAMIC olarak ayarlayın.

- The system parameter `CLUSTERCacheTürü` on platforms other than z/OS
- z/OS üzerindeki CSQ6SYSP makrosu içindeki CLCACHE parametresi.

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 çağırılır.

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:

- MQWDR içindeki `ChannelDefGörelî Konumu` alanı
- MQWDR içindeki `ClusterRecGörelî 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 (MQLONG) -çıkış

Tamamlanma kodu.

Tamamlanma kodu aşağıdaki değerlerden birine sahiptir:

MQCC_OK

İşlem başarıyla tamamlandı.

MQCC_WARNING

Uyarı (kısmi tamamlama).

MQCC_FAILED

Arama başarısız oldu.

Reason (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 çıkışının, 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

Düzeltilici 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.
- StructId alanı MQWXP_STRUC_IDdeğil.
- Sürüm alanı MQWXP_VERSION_2değil.
- 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 Assemblerürünlerini 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 Assemblerürünlerini destekler.

MQXCLWLN' in dil çağrıları

MQXCLWLN , iki dil, C ve High Level Assemblerürünlerini 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 çağırısındaki 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 195. MQWXP' deki alanlar		
Alan	Tanım	Sayfa
<i>StrucId</i>	Yapı tanıtıcısı	StrucId
<i>Version</i>	Yapı sürüm numarası	S\u00f0fcr\u00f0cm
<i>ExitId</i>	Çıkış tipi	ExitId
<i>ExitReason</i>	Çıkışa çağırılması nedeni	ExitReason
<i>ExitResponse</i>	Çıkıştan yanıt	ExitResponse
<i>ExitResponse2</i>	Çıkıştan ikincil yanıt	ExitResponse2
<i>Feedback</i>	Feedback kodu	Geribildirim
<i>Flags</i>	İşaretler değerleri. Bu bit işaretleri, konulmakta olan iletiye ilişkin bilgileri belirtmek için kullanılır	İşaretler
<i>ExitUserArea</i>	Kullanıcı alanından çık	ExitUserAlanı
<i>ExitData</i>	Verilerden çık	ExitData
<i>MsgDescPtr</i>	İleti tanımlayıcısının adresi (MQMD)	MsgDescPtr 'si
<i>MsgBufferPtr</i>	İleti verilerinin bazılarını ya da tümünü içeren arabelleğin adresi	MsgBufferPtr
<i>MsgBufferLength</i>	İleti verilerini içeren arabelleğin uzunluğu	MsgBufferUzunluğu
<i>MsgLength</i>	İletinin tam uzunluğu	MsgLength
<i>QName</i>	Kuyruğun adı	QName

Çizelge 195. MQWXP' deki alanlar (devamı var)		
Alan	Tanım	Sayfa
<i>QMgrName</i>	Yerel kuyruk yöneticisinin adı	QMgrName
<i>DestinationCount</i>	Olası varış noktalarının sayısı	DestinationCount
<i>DestinationChosen</i>	Hedef seçildi	DestinationChosen
<i>DestinationArrayPtr</i>	Hedef kayıtlarla ilgili bir dizi işaretçi adresi (MQWDR)	DestinationArrayPtr 'si
<i>QArrayPtr</i>	Kuyruk kayıtlarıyla ilgili bir dizi işaretinin adresi (MQWQR)	QArrayPtr
Not: Sürüm MQWXP_VERSION_2değerinden küçükse, geri kalan alanlar yoksayılr.		
<i>CacheContext</i>	Bağlam Bilgileri	CacheContext
<i>CacheType</i>	Küme önbelleğinin tipi	CacheType
Not: Sürüm MQWXP_VERSION_3değerinden küçükse, geri kalan alanlar yoksayılr.		
<i>CLWLMRUChannels</i>	İzin verilen etkin giden küme kanalı sayısı üst sınırı	CLWLMRUChannels
Not: Sürüm MQWXP_VERSION_4değerinden küçükse, geri kalan alanlar yoksayılr.		
<i>pEntryPoints</i>	MQI ve DCI çağrılarının yapılmasına izin vermek için MQIEP yapısının adresi	pEntryNoktaları

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 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.

[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_IDdölur.
- For the C programming language, the constant MQWXP_STRUC_ID_ARRAY is also defined. Bu değeri, MQWXP_STRUC_IDile aynı değere sahiptir. Bu, bir dizgi yerine bir karakter dizisidir.

Sürüm (MQLONG) -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 şu ortamlarda desteklenmektedir: AIX, HP-UX, Linux, IBM i, Solaris ve Windows.

MQWXP_VERSION_3

Version-3 küme iş yükü çıkış parametre yapısı.

MQWXP_VERSION_3 şu ortamlarda desteklenmektedir: AIX, HP-UX, Linux, IBM i, Solaris ve Windows.

MQWXP_VERSION_4

Version-4 kümesi iş yükü çıkış parametre yapısı.

MQWXP_VERSION_4 şu ortamlarda desteklenmektedir: AIX, HP-UX, Linux, IBM i, Solaris ve 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 (MQLONG) -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 (MQLONG) -çıkış

İletin 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 196. Kuyruk yöneticisi tarafından yapılan işlemler.

Bu tabloda iki sütun vardır. İlk sütunda, çıkış nedenleri ve ikinci sütunda her bir çıkış nedeni için yapılan işlem açıklanmaktadır.

ExitReason	Uygulanan İşlem
- MQXR_CLWL_OPEN - MQXR_CLWL_REPOS - MQXR_CLWL_PUT	MQOPEN, MQPUTya da MQPUT1 çağrısı, tamamlanma kodu MQCC_FAILED ve neden kodu MQRC_STOPPED_BY_CLUSTER_EXITile başarısız olur.
- MQXR_CLWL_MOVE - MQXR_CLWL_REPOS_MOVE	İleti, ölü-harf kuyruğunda yer alıyor.

MQXCC_SUPPRESS_EXIT

Geçerli iletiyi olağan şekilde işlemeye devam edin. Kuyruk yöneticisi sona erdirilinceye kadar çıkışı yeniden çağırmayın.

Kuyruk yöneticisi, ClusterWorkloadExit kuyruk yöneticisi özniteliği boş olduğunda sonraki iletileri işler. 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 (MQLONG) -giriş

Ayrılmış bir alan. Değer sıfır.

İşaretler (MQLONG) -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.

Ayrılmış (MQLONG) -giriş

Ayrılmış bir alan. Değer sıfır.

ExitUserAlanı (MQBYTE16) -giriş/çıkış

Çıkış çağrılar arasında iletişim kurmak için ExitUserArea seçeneğini belirleyin.

- ExitUserAlan , 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 queue-manager özniteliğinin değeri. Bu öznitelik için herhangi bir değer tanımlanmadıysa, bu alan tamamen boşluktur.

- ExitData uzunluğu MQ_EXIT_DATA_LENGTHtarafından verilir.

MsgDescPtr (PMQMD) -giriş

İleti tanımlayıcısının kopyasının adresi (MQMD) İşlenmekte olan ileti için.

- 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 verilerde yapılan değişiklikler kuyruk yöneticisi tarafından yoksayılır.
- Şu durumda çıkışa ileti verisi iletilmedi:
 - MsgDescPtr boş değerli işaretçi.
 - İletinin veri yok.
 - ClusterWorkloadLength queue-manager özniteliği sıfır.

Bu durumda, MsgBufferPtr boş değerli göstergedir.

MsgBufferUzunluğu (MQLONG) -giriş

Çıkışa aktarılan ileti verilerini içeren arabelleğin uzunluğu.

- Uzunluk, ClusterWorkloadLength kuyruk yöneticisi öznitelice denetlenir.
- Uzunluk, iletinin tam uzunluğundan az olabilir, bkz. MsgLength.

MsgLength (MQLONG) -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.

- MQgrName , MQ_Q_MGR_NAME_LENGTH uzunluğudur.

DestinationCount (MQLONG) -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 (MQUSEL) -giriş/çıkış

Seçilen istikamet.

- İletin 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 (PPMQWQR) -giriş

Kuyruk kayıtlarına ilişkin bir dizi işaretçi dizisi (MQWQR).

- 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 büyümmez.
- Bu önbellekteki kayıtlarda dolaşmak için MQXCLWLN çağrısını kullanmanız gerekmez.

MQCLCT_DYNAMIC

Önbellek dinamik.

- Değişen küme bilgilerini barındırmak için önbelleğin boyutu artırabilir.
- Bu önbellekteki kayıtlarda dolaşmak için MQXCLWLN çağrısını kullanmanız gerekir.

CLWLMRUChannels (MQLONG) : Sürüm 3-giriş

Küme iş yükü seçimi algoritması tarafından kullanılmak üzere değerlendirilmek üzere etkin giden küme kanalı sayısı üst sınırını belirtir.

- CLWLMRUChannels , 1-999 999 999 numaradır.

pEntryNoktaları (PMQIEP) : Sürüm 4

MQI ve DCI çağrılarının yapılabileceği MQIEP yapısının adresi.

İlgili başvurular

MQWXP için 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 197. 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:		
1. ↵ simgesi tek bir boş karakteri temsil eder.		
2. C programlama dilinde, MQWXP_DEFAULT makro değişkeni varsayılan değerleri içerir. Yapıdaki alanlara ilişkin ilk değerleri sağlamak için bunu aşağıdaki şekilde kullanın:		
<pre>MQWDR MyWXP = {MQWXP_DEFAULT};</pre>		

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
ORG MQWXP
MQWXP_AREA DS CL(MQWXP_LENGTH)

```

İlgili başvurular

[MQWXP içindeki alanlar-Küme iş yükü çıkış değiştirgesi yapısı](#)

MQWDR-Küme iş yükü hedef kaydı yapısı

Aşağıdaki çizelge, MQWDR -Küme iş yükü hedefi kayıt yapısındaki alanları özetler.

Çizelge 198. MQWDR' daki alanlar		
Alan	Tanım	Sayfa
<i>StrucId</i>	Yapı tanıtıcısı	StrucId
<i>Version</i>	Yapı sürüm numarası	S\u00fcr\u00fcm
<i>StrucLength</i>	MQWDR yapısının uzunluğu	StrucLength
<i>QMgrFlags</i>	Kuyruk yöneticisi işaretleri	QMgrFlags
<i>QMgrIdentifier</i>	Kuyruk yöneticisi tanıtıcısı	QMgrIdentifier
<i>QMgrName</i>	Kuyruk yöneticisi adı	QMgrName
<i>ClusterRecOffset</i>	İlk küme kaydının mantıksal görel konumu (MQWCR)	ClusterRecKayması
<i>ChannelState</i>	Kanal durumu	ChannelState
<i>ChannelDefOffset</i>	Kanal tanımlama yapısının mantıksal görel konumu (MQCD)	ChannelDefGörel Konumu
Not: Sürüm MQWDR_VERSION_2değerinden küçükse, geri kalan alanlar yoksayılr.		
<i>DestSeqNumber</i>	Kanal hedef sıra numarası	DestSeqNumarası
<i>DestSeqFactor</i>	Ağırlıklandırma için kanal hedef sıra katsayısı	DestSeqKatsayısı

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 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.

[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_IDdur.

- 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.

- ChannelDefGörelİ Konumu , ClusterRecKaymasıgibidir
- İşaretçi aritmetikte mantıksal görelİ konum kullanılamaz. Sonraki kaydın adresini elde etmek için, MQXCLWLN çağırısı kullanılmalıdır.

DestSeqFactor (MQLONG) -giriş

Kanal seçimini ağırlığa dayalı olarak seçmenize olanak tanıyan hedef sıra katsayısı.

- Kuyruk yöneticisi bunu değiştirmeden önceDestSeqFactor 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

MQWDRiçin ilk değerler ve dil bildirimleri

İlk değerler ve C ve High Level Assembler Dil bildirimleri MQWDR -Küme iş yükü hedef kaydı için.

MQWDRiçin ilk değerler ve dil bildirimleri

İlk değerler ve C ve High Level Assembler Dil bildirimleri MQWDR -Küme iş yükü hedef kaydı için.

Çizelge 199. MQWDR' da alanların ilk değerleri

Alan adı	Değişmezin adı	Değişmezin değeri
StrucId	MQWDR_STRUC_ID	'WDR↵'
Version	MQWDR_VERSION_1	1
StrucLength	MQWDR_CURRENT_LENGTH ³	136
QMgrFlags	MQWDR_NONE	0
QMgrIdentifier	Yok	" "
QMgrName	Yok	" "
ClusterRecOffset	Yok	0
ChannelState	Yok	0
ChannelDefOffset	Yok	0
DestSeqNumber	Yok	0
DestSeqFactor	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
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 200. MQWQRiçindeki alanlar		
Alan	Tanım	Sayfa
<i>StrucId</i>	Yapı tanıtıcısı	StrucId
<i>Version</i>	Yapı sürüm numarası	S\u00fcr\u00fcm
<i>StrucLength</i>	MQWQR yapısının uzunluğu	StrucLength
<i>QFlags</i>	Kuyruk işaretleri	QFlags
<i>QName</i>	Kuyruk adı	QName
<i>QMgrIdentifier</i>	Kuyruk yöneticisi tanıtıcısı	QMgrIdentifier
<i>ClusterRecOffset</i>	İlk küme kaydının kayması (MQWCR)	ClusterRecKayması
<i>QType</i>	Kuyruk tipi	QType
<i>QDesc</i>	Kuyruk tanımlaması	QDesc
<i>DefBind</i>	Varsayılan bağ tanımı	DefBind
<i>DefPersistence</i>	Varsayılan ileti kalıcılığı	DefPersistence
<i>DefPriority</i>	Varsayılan ileti önceliği	DefPriority
<i>InhibitPut</i>	Kuyruğa ilişkin işlemlere izin verilip verilmeyeceğini	InhibitPut
Not: Sürüm MQWQR_VERSION_2değerinden küçükse, geri kalan alanlar yoksayılr.		
<i>CWLQueuePriority</i>	Kuyruğun önceliğini gösteren 0-9 arasında bir değer	CWLQueuePriority
<i>CLWLQueueRank</i>	Kuyruğun sırasını simgeleyen 0-9 arasında bir değer	CLWLQueueRank
Not: Sürüm MQWQR_VERSION_3değerinden küçükse, geri kalan alanlar yoksayılr.		
<i>DefPutResponse</i>	Varsayılan yanıt koyma yanıtı	DefPutYanıtı

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 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.

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 tanıttıcısı.

- 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_LENGTHHuzunluğlıdır.

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_LENGTHHuzunluğlıdır.

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 çağırısı 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_LENGTHHuzunluğlıdır.

DefBind (MQLONG) -giriş

Kuyruk yöneticinde tanımlı olan varsayılan bağ tanımlama kuyruğu özniteliği, MQWQR yapısı tarafından tanımlanan hedef kuyruğunun eşgörünümünü barındırır. Gruplarla gruplar kullanılırken MQBND_BIND_ON_OPEN ya da MQBND_BIND_ON_GROUP belirtilmeli. 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 201. MQWQRiçindeki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
StrucId	MQWQR_STRUC_ID_ARRAY	'WQR→'
Version	MQWQR_VERSION_1	1
StrucLength	MQWQR_CURRENT_LENGTH ³	212
QFlags	Yok	0
QName	Yok	" "
QMgrIdentifier	Yok	" "
ClusterRecOffset	Yok	0
QType	Yok	0
QDesc	Yok	" "
DefBind	Yok	0

Çizelge 201. 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 bildirimi

```
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

* MQWQR_DEFPUTRESPONSE MQWQR_LENGTH MQWQR_AREA	DS EQU ORG DS	F *-MQWQR MQWQR CL (MQWQR_LENGTH)	the queue are allowed Default put response Length of structure

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 202. MQWCR' deki alanlar.		
Bu tabloda üç sütun vardır. İlk sütunda alan adları listelenir, ikinci sütun her alanın bir açıklamasını sağlar ve üçüncü sütunda her bir alana ilişkin daha fazla bilgi için bir bağlantı bulunur.		
Alan	Tanım	Sayfa
<i>ClusterName</i>	Kümenin adı	ClusterName
<i>ClusterRecOffset</i>	Sonraki küme kaydının kayması (MQWCR)	ClusterRecKayması
<i>ClusterFlags</i>	Küme işaretleri	ClusterFlags

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 tanıtilan 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 203. MQWCR' deki alanların ilk değerleri		
Alan adı	Değişmezin adı	Değişmezin değeri
ClusterName	Yok	" "
ClusterRecOffset	Yok	0
ClusterFlags	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
MQWCR          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 çıkış 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 karşılaşrsa (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 Websphere MQ Sürüm 7.0 and Sürüm 7.1 is to write the exit in a way which links at Sürüm 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 Sürüm 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 yoksa; 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 1580.](#)

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 1580.](#)

Feedback (MQUZE)-giriş/çıkış

Çıkış işlevi çağrıları arasında geribildirim kodlarını iletin. Bu, şu şekilde kullanıma hazırlandı:

```
MQFB_NONE (0)
```

bir zincirdeki ilk çıkışa ait ilk işlevi çağırmadan 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 çağırısının 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 ExitInfonesnesiyle 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 çıkışa 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 işleyici, 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 iletme 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 1583](#) 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 1585’inde açıklanan MQXF_ * değışmezleri olan geçerli değler.

Çıkış işleyicisi bu alanı, çağrılmakta olan çıkışa neden olan IBM MQ API çağrısına bağı olarak, her bir çıkış işlevine giriş sırasında doğru değere ayarlar.

ExitMsgHandle (MQHMSG)-giriş/çıkış

İşlev MQXF_GET ve ExitReason ise, MQXR_AFTER olduğıunda, bu alanda geçerli bir ileti tanıtıcısı döndürölür. Bu alanda, API çıkışı kaydedilirken, MQXEPO yapısında belirtilen ExitProperties dizisiyle eşleşen diğler tüm özellikler, ileti tanımlayıcı alanlarına ve diğler tüm özelliklere ilişkin tüm özellikler için döndürölür.

ExitMsgHandle içinde döndürölün ileti dışı tanımlayıcı özellikleri, belirtilmişse, MQGMO yapısındaki MsgHandle ' den ya da ileti verilerinde kullanılmaz.

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 değışir.

Çizelge 204 sayfa 1580 , 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) değıştirilebilir.

Çizelge 204. MQXR_BEFORE çıkış işlemi			
ExitResponsesdeğeri	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

Çizelge 204. MQXR_BEFORE çıkış işlemi (devamı var)

ExitResponsedeğeri	CompCode (Derleyici Kodu) ve Neden	ExitResponse2 değeri (varsayılan devamı) Zinciri	ExitResponse2 değeri (varsayılan devam) API
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 değıştirgeler şunlardır:

StrucId (MQCHAR4)-giriş

MQAXC_STRUC_IDdeğerine sahip çıkış bağlamı yapısı tanıttıcısı. C programları için, MQAXC_STRUC_ID_ARRAY değışmezi, MQAXC_STRUCT_ID değeriyle aynı değeri, ancak dizgi yerine bir karakter dizisi olarak tanımlanır.

Çıkış işleyici, bu alanı her çıkış işlevine girişte ayarlar.

Sürüm (MQUZE)-giriş

Yapı sürüm numarası, şu değeri kullanarak:

MQAXC_VERSION_2

Çıkış bağlamı yapısına ilişkin sürüm numarası.

MQAXC_CURRENT_VERSION

Çıkış bağlamı yapısına ilişkin yürürlükteki sürüm numarası.

Çıkış işleyici, bu alanı her çıkış işlevine girişte ayarlar.

Ortam (MQUZE)-giriş

Bir çıkış işleviyle sonuçlanan bir IBM MQ API çağrısının yayınlandığı ortam. Bu alan için geçerli değerler şunlardır:

MQXE_DIĞER

Bu değ er, bir sunucu uygulamasından çıkışa çağ ırılıp çağ ırılmı yorsa, bir API çıkış ı çağ ırılarla tutarlı olur. Yani, bir istemcide bir API çıkış ı değ işmeden çalış ır ve farklı bir şey görmeyen bir çıkış olur.

Çıkışta istemcinin çalış ıp çalışmadığını belirlemesi gereki yorsa, çıkışta *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 değ işiklik 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 değ iştiren 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 değ iştiren 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 değ eri 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 çağrısını 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 çağrısını 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 çağrısı 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      StrucId;           /* Structure identifier */
    MQLONG       Version;          /* Structure version number */
    MQLONG       StrucLength;       /* 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_STRUCT_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 ExitInfoadı ile kullanıma hazırlamalıdır; böylece, daha sonra bu MQACH yapısı bu çıkışa ilişkin başka bir yönetim ortamı tarafından ya da bir işbirliği çıkışa göre bulunabilir.

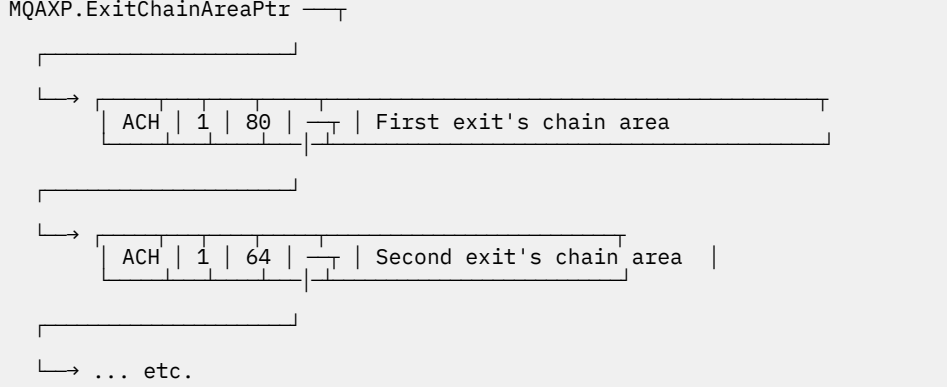
MQACH_DEFAULT ile tanımlanan bu alanın başlangıç değeri sıfır uzunluklu bir dizgidir ({}).

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:



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 işlevine 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 tipdefler hakkında bilgi sağlanır.

Aşağıda, API çıkışlarıyla ilişkili C dili yazı tipi tipleri yer alıyor:

```
typedef PMQLONG MQPOINTER PPMQLONG;
```

```

typedef PMQBYTE MQPOINTER PPMQBYTE;
typedef PMQHOBJ MQPOINTER PPMQHOBJ;
typedef PMQOD MQPOINTER PPMQOD;
typedef PMQMD MQPOINTER PPMQMD;
typedef PMQPMO MQPOINTER PPMQPMO;
typedef PMQGMQ MQPOINTER PPMQGMQ;
typedef PMQCNO MQPOINTER PPMQCNO;
typedef PMQBO MQPOINTER PPMQBO;

typedef MQAXP MQPOINTER PPMQAXP;
typedef MQACH MQPOINTER PPMQACH;
typedef MQAXC MQPOINTER PPMQAXC;

typedef MQCHAR MQCHAR16[16];
typedef MQCHAR16 MQPOINTER PPMQCHAR16;

typedef MQLONG MQPID;
typedef MQLONG MQTID;

```

Çıkış giriř noktası kayıt çağırısı (MQXEP)

MQXEP, MQXEP C dili çağırısı ve MQXEP C iřlev prototipi hakkında bilgi edinmek için bu bilgileri kullanın.

MQXEP çağırısını řu řekilde kullanın:

1. Register the before and after IBM MQ API exit invocation points at which to invoke exit functions
2. Çıkış iřlevi giriř noktalarını belirle
3. Çıkış iřlevi giriř noktalarının kaydını sil

MQXEP çağırılarını genellikle MQ_INIT_EXIT çıkış iřlevinde kodlayabilirsiniz, ancak sonraki çıkış iřlevlerinde bunları belirtebilirsiniz.

Önceden kayıtlı bir çıkış iřlevini kaydettirmek için bir MQXEP çağırısı kullanırsanız, kayıtlı çıkış iřlevini yerine ikinci MQXEP çağırısı başarıyla tamamlanır.

Bir NULL çıkış iřlevini kaydetmek için bir MQXEP çağırısı kullanırsanız, MQXEP çağırısı başarıyla tamamlanır ve çıkış iřlevi kayıt dıřı olur.

MQXEP çağırıları, bir bağlantı isteęinin ömrü sırasında belirli bir çıkış iřlevini kaydetmek, kayıttan kaldırmak ve yeniden kaydettirmek için kullanılırsa, önceden kayıtlı çıkış iřlevi yeniden etkinleřtirilir. Hala ayrılmıř ve bu çıkış iřlevi eřgörünümüyle iliřkilendirilmiř herhangi bir saklama alanı, çıkışa iliřkin iřlevler tarafından kullanılabilir. (Bu saklama alanı genellikle sonlandırma çıkış iřlevinin çağırılması sırasında serbest bırakılır.)

MQXEP arabirimi ařaęıdaki gibi olur:

```
MQXEP (Hconfig, ExitReason, Function, EntryPoint, &ExitOpts, &CompCode, &Reason)
```

Burada:

Hconfig (MQHCONFIG)-giriř

Bařlatılmakta olan iřlevlerin kümesini içeren API çıkışını temsil eden yapılandırma tanıtıcısı. Bu deęer, MQ_INIT_EXIT iřlevini çağırmadan hemen önce kuyruk yöneticisi tarafından oluřturulur ve MQAXP¹ de her bir API çıkış iřlevine geçirilir.

ExitReason (MQHOT)-giriř

Giriř noktasının kaydettirilmesinin nedeni, ařaęıdaki nedenlerden.

- Bağlantı düzeyinde kullanıma hazırlama ya da sonlandırma (MQXR_CONNECTION)
- Bir IBM MQ API çağırısından önce (MQXR_BEFORE)
- IBM MQ API çağırısından sonra (MQXR_AFTER)

İřlev (MQUZE)-giriř

İřlev tanıtıcısı, MQXF_ * deęiřmezleri olan geçerli deęerler (bkz. [“Dıř sabitler” sayfa 1585](#)).

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 205 sayfa 1588.](#))

MQRC_FUNCTION_HATASI

(2281, X'8E9') Belirtilen işlev tanıtıcısı, API çıkış nedeni için geçerli değil. Aşağıdaki çizelge, işlev tanıtıcılarının ve ExitReasons' ın geçerli birleşimlerini göstermektedir.

Çizelge 205. İşlev tanıtıcılarının ve ExitReasons birleşimlerinin geçerli birleşimleri	
İşlev	ExitReason
MQXF_INIT MQXF_TERM	MQXR_BAĞLANTI

Çizelge 205. İşlev tanıtıcılarının ve ExitReasons birleşimlerinin geçerli birleşimleri (devamı var)

İşlev	ExitReason
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 */
MQLONG       EntryPoint;     /* Function entry point */
MQXEP        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 */
PMQXEP0	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 değiştireliler 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 edemediğinizi unutmayın. An API exit BEFORE call might not be followed immediately by an AFTER call.

BEFORE çağrısı, başka bir BEFORE çağrısı tarafından izlenebilir. Örneğin:

```
MQCTL ' DEN
Geri Çağırmadan önce
MQPUT ' DEN
MQPUT ' DEN
Geri Çağırma SONR
MQCTL ' DEN
```

ya da

```
XAXX_ENCODE_CASE_ONE open ' DAN
```

MQCONN ' DEN
MQCONN ' 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 değiştirgelerden herhangi birini değiştirebilir. İstemci önce bu çıkışı çağırır, sonra MQCONN ya da MQCONNX çağrısını çağırır. Yalnızca ilk MQCONN ya da MQCONNX çağrısının API çıkışını çağırdığını 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_ * değerlerine dönüştürülür ve CompCode ve Reason alanlarında uygulamaya geri iletilir. Ancak, çıkış işleyici mantığında saptanan hatalar, CompCode ve neden alanlarında MQCC_ * ve MQRC_ * değerleri 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)
- Çıkış yürütme ortamı temizlemesi *değil* gerçekleştirildi

Hala kullanımda olabileceği için çıkış boşaltılamıyor. Ayrıca, *önce* çıkışı başarılı olan çıkış zincirinde diğer 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) çağrılmadan önce, çıkış yürütme ortamını ayarlar. Çıkış yürütme ortamı uyarı, 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) çağırdıktan 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 çağrılarından 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 çağrılması gerekir.

Yalnızca başlangıçtaki MQCONN ya da MQCONNX çağrısının çıkışa çağrıldığını 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 iletişi 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_EXPORTED 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

Kismi 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 çağırısı 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 */
PMQGM       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 */
PPMQGM       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 */
PMQHOBJ     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 işlev 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 (PMQHOBj)-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 */
PMQHOBj	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 */
    PPMQHOBj    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_CMIT_EXIT

MQ_CMIT_EXIT, *önce* ve *bundan sonra* kesinleştirme işlemlerini gerçekleştirmek için bir kesinleştirme çıkış işlevi sağlar. *önce* ve *bundan sonra* kesinleştirme çağrısı çıkış işlevlerini kaydettirmek için MQXR_BEFORE ve MQXR_AFTER çıkış nedenleriyle MQXF_CMIT işlev tanıtıcısını kullanın.

Kesinleştirme işlemi başarısız olursa ve hareket geriletirse, MQCMIT çağrısı MQCC_UYARI ve MQRC_BACKED_OUT ile başarısız olur. Bu dönüş ve neden kodları, çıkışı vermek için herhangi bir *bundan sonra* MQCMIT çıkış işlevine geçirilir ve iş biriminin yedeklendiğine ilişkin bir gösterge sağlar.

Bu işleve ilişkin arabirim şöyledir:

```
MQ_CMIT_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_CMIT_EXIT (&ExitParms, &ExitContext,&Hconn, &CompCode, &Reason);
```

Çıkışınız, aşağıdaki C işlevi prototipini eşleştirmelidir:

```
void MQENTRY MQ_CMIT_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 1605 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.

Bağlan ve uzantı bağlan-MQ_CONNX_EXIT

MQ_CONNX_EXIT şunları sağlar:

- *önce ve bundan sonra* MQCONN işlemlerini gerçekleştirmek için bağlantı çıkış işlevi
- *önce ve bundan sonra* MQCONNX işlemlerini gerçekleştirmek için bağlantı genişletme çıkış işlevi

Burada tanımlanan 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 çağrısını 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 çağrılarına 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_CONNX_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_CONNX işlev tanıtıcılarını kullanın.

An MQ_CONNX_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_CONNX_EXIT, çağrılmakta olduğu bağlantı için bir API çıkış çağrısından MQDISC ' yi çağırabilir. Bu kısıtlama, hem istemci hem de sunucu API çıkışları için geçerlidir.

MQCONN ve MQCONNX için arabirim aynı:

```
MQ_CONNX_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

MQCONN 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_CONN_EXIT (&ExitParms, &ExitContext, &pQMgrName, &pConnectOps,
               &pHconn, &CompCode, &Reason);
```

Çıkışınız, aşağıdaki C işlevi prototipini eşleştirmelidir:

```
void MQENTRY MQ_CONN_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_CONN_EXIT işlev arabirimi, hem MQCONN çağrısı, hem de MQCONNX çağrısı için kullanılır. Ancak, bu iki arama için ayrı giriş noktaları tanımlanır. *her ikisi* çağrılarını engellemek için,

MQXEP çağrısının, 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 çağrısı 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 çağrısını yayınlayabilir. 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 değıştirtgeler geçersiz olabilir (parametrelerin adresleri için geçersiz işaretçiler de içinde olmak üzere).

MQ_CONNX_EXIT işlevi, kuyruk yöneticisi tarafından herhangi bir yetki denetimi gerçekleştirilmeden önce çağrılır.

4. Çıkış işlevi, MQCONN ya da MQCONNX çağrısında belirlenen kuyruk yöneticisinin adını değıştirmemelidir. Çıkış, çıkış işleviyle değıştirilirse, sonuçlar tanımsız olur.

5. MQ_CONNX_EXIT için bir MQXR_BEFORE çıkış işlevi, MQXEP dışında MQ çağrıları yayınlayamaz.

Geri çağırma denetimi-MQ_CTL_EXIT

MQ_CTL_EXIT, *önce* ve *bundan sonra* denetim geri bildirme işlemlerini gerçekleştirmek için bir abonelik isteğı çıkış işlevi sağlar. *önce* ve *bundan sonra* denetim geri bildirme çağrısı 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 */
    PMQHCONN    pHconn,         /* Address of ptr to connection handle */
    MQLONG      pCompCode,       /* Address of completion code */
    MQLONG      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 1605](#) 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     pBuffer,        /* 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 1605 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

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_SUPPRESS_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 karşılaşır, 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* değıştirtgelerinin 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 */
PMQHOBj	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 */
PMQOD       ppObjDesc,       /* Address of ptr to object descriptor */
PMQLONG     pOptions,        /* Address of open options */
PMQHOBJS    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 (MQHOBJS)-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 (
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 */
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 */
```

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 aracısı 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 (
  MQAXP      pExitParms,      /* Address of exit parameter structure */
  MQAXC      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.

pIntAttrs (PMQLong)-giriş/çıkış

Tamsayı öznitelik değerleri dizisine işaretçi.

CharAttrUzunluk (MQUZE)-giriş/çıkış

Karakter öznitelikleri dizisi uzunluğu.

pCharAttrs (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  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_SET_EXIT (&ExitParms, &ExitContext, &Hconn, &Hobj, &SelectorCount,
             &pSelectors, &IntAttrCount, &pIntAttrs, &CharAttrLength,
             &pCharAttrs, &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 */
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 */
```

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 erdirmiş 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 karşılaşrsa, 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 */
MQLONG     CompCode;       /* Completion code */
MQLONG     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 */
PMQLONG     pCompCode,     /* Address of completion code */
PMQLONG     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 çağırısı yayınlandığında bir MQ_TERM_EXIT işlevi kaydedildiyse, tüm MQDISC çıkış işlevleri çağırıldıktan 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* değıştirtgelerinin uygun değerlere ayarlanmasına neden olan MQ_TERM_EXIT' in çağrılmasına neden olan MQDISC çağırısı 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 */
PPMQHOBJ  ppHobj;        /* Object Handle */
PPMQHOBJ  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 çağrısı 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 */
PMQSRO    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 işleminden 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 işleminden 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 işleminden 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

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_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 */
    MQPTR 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 işleminden önce ve sonra gerçekleştirilmek üzere bir xa_rollback çıkış işlevi sağlar. Önceki ve xa_geridönüş çağrısı 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 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_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 yapar.

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 işlemiden 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 işleminden 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_CONNX_EXIT	1
e3	before MQ_CONNX_EXIT	2
e4		--> MQCONN
e5	after MQ_CONNX_EXIT	2
e6	after MQ_CONNX_EXIT	1
e7		

```

MQOPEN <--
-->
    before MQ_OPEN_EXIT 1
e8
    before MQ_OPEN_EXIT 2
e9
--> MQOPEN
e10
    after MQ_OPEN_EXIT 2
e11
    after MQ_OPEN_EXIT 1
e12

MQPUT <--
-->
    before MQ_PUT_EXIT 1
e13
    before MQ_PUT_EXIT 2
e14
--> MQPUT
e15
    after MQ_PUT_EXIT 2
e16
    after MQ_PUT_EXIT 1
e17

MQCLOSE <--
-->
    before MQ_CLOSE_EXIT 1
e18
    before MQ_CLOSE_EXIT 2
e19
--> MQCLOSE
e20
    after MQ_CLOSE_EXIT 2
e21
    after MQ_CLOSE_EXIT 1
e22

MQDISC <--
-->
    before MQ_DISC_EXIT 1
e23
    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.

Çizelge 206. API çıkış hataları ve yapılması gereken uygun işlemler		
Err Pt	Tanım	İşlemler
e1	Ortam ayarları ayarlanırken hata oluştu.	1. Ortam kurulumunu gereken şekilde geri al 2. Çıkış işlevsiz sürücü 3. MQCC_FAILED ile MQCONN başarısız oldu, MQRC_API_EXIT_LOAD_ERROR

Çizelge 206. API çıkış hataları ve yapılması gereken uygun işlemler (devamı var)

Err Pt	Tanım	İşlemler
e2	MQ_INIT_EXIT işlevi şu şekilde tamamlanır: • MQXCC_FAILED • MQXCC_*	• MQXCC_FAILED için: 1. Ortamı temizle 2. MQCC_FAILED ile MQCONN başarısız oldu, MQRC_API_EXIT_INIT_ERROR • MQXCC_* için * 1. MQXCC_* ve MQXR2_*¹değerleri için hareket edin. 2. Ortamı temizle
e3	Önce MQ_CONNX_EXIT 1 işlevi aşağıdaki gibi tamamlanır: • MQXCC_FAILED • MQXCC_*	• MQXCC_FAILED için: 1. Sürücü MQ_TERM_EXIT işlevi 2. Ortamı temizle 3. MQCC_FAILED ile MQCONN çağrısı başarısız oldu, MQRC_API_EXIT_ERROR • MQXCC_* için * 1. MQXCC_* ve MQXR2_*¹değerleri için hareket edin. 2. Gerekliyse MQ_TERM_EXIT işlevini kullan 3. Gerekirse ortamı temizle
e4	Önce MQ_CONNX_EXIT 2 işlevi aşağıdaki gibi tamamlanır: • MQXCC_FAILED • MQXCC_*	• MQXCC_FAILED için: 1. Sürücü <i>bundan sonra</i> MQ_CONNX_EXIT 1 işlevi 2. Sürücü MQ_TERM_EXIT işlevi 3. Ortamı temizle 4. MQCC_FAILED ile MQCONN çağrısı başarısız oldu, MQRC_API_EXIT_ERROR • MQXCC_* için * 1. MQXCC_* ve MQXR2_*¹değerleri için hareket edin. 2. Drive <i>bundan sonra</i> MQ_CONNX_EXIT 1 function if exit not suppressed 3. Gerekliyse MQ_TERM_EXIT işlevini kullan 4. Gerekirse ortamı temizle
e5	MQCONN çağrısı başarısız oldu.	1. Geçiş MQCONN CompCode ve Reason 2. <i>önce</i> MQ_CONNX_EXIT 2 başarılı olursa ve çıkış engellenmezse, <i>bundan sonra</i> MQ_CONNX_EXIT 2 işlevini kullanın. 3. <i>önce</i> MQ_CONNX_EXIT 1 başarılı olursa ve çıkış engellenmezse, <i>bundan sonra</i> MQ_CONNX_EXIT 1 işlevini kullanın. 4. Sürücü MQ_TERM_EXIT işlevi 5. Ortamı temizle

Çizelge 206. API çıkış hataları ve yapılması gereken uygun işlemler (devamı var)

Err Pt	Tanım	İşlemler
e6	Sonra MQ_CONNX_EXIT 2 işlevi aşağıdaki gibi tamamlanır: • MQXCC_FAILED • MQXCC_*	• MQXCC_FAILED için: 1. Sürücü <i>bundan sonra</i> MQ_CONNX_EXIT 1 işlevi 2. MQCC_FAILED ile MQCONN çağrısını tamamla, MQRC_API_EXIT_ERROR • MQXCC_* için * 1. MQXCC_* ve MQXR2_*¹değerleri için hareket edin. 2. Gerekliyse <i>bundan sonra</i> MQ_CONNX_EXIT 1 işlevini kullanın
e7	Sonra MQ_CONNX_EXIT 1 işlevi aşağıdaki gibi tamamlanır: • MQXCC_FAILED • MQXCC_*	• MQXCC_FAILED için, MQCONN çağrısını MQCC_FAILED ile tamamla, MQRC_API_EXIT_ERROR • MQXCC_* için, MQXCC_* ve MQXR2_*¹değerlerine göre hareket edin.
e8	Önce MQ_OPEN_EXIT 1 işlevi şu şekilde tamamlanır: • MQXCC_FAILED • MQXCC_*	• MQXCC_FAILED için, MQCC_FAILED ile MQXX_ENCODE_CASE_ONE open çağrısını tamamla, MQRC_API_EXIT_ERROR • MQXCC_* için, MQXCC_* ve MQXR2_*¹değerlerine göre hareket edin.
e9	Önce MQ_OPEN_EXIT 2 işlevi şu şekilde tamamlanır: • MQXCC_FAILED • MQXCC_*	• MQXCC_FAILED için: 1. Sürücü <i>bundan sonra</i> MQ_OPEN_EXIT 1 işlevi 2. MQCC_FAILED ile MQXX_ENCODE_CASE_ONE open çağrısını tamamla, MQRC_API_EXIT_ERROR • MQXCC_* için, MQXCC_* ve MQXR2_*¹değerlerine göre hareket edin.
e10	MQOPEN çağrısı başarısız oldu	1. MQOPEN CompCode ögesini geçti ve neden 2. Çıkış engellenmezse <i>bundan sonra</i> MQ_OPEN_EXIT 2 işlevini kullan 3. Çıkış engellenmezse ve zincirleme çıkışlar engellenmezse, <i>bundan sonra</i> MQ_OPEN_EXIT 1 işlevini kullanın
e11	Sonra MQ_OPEN_EXIT 2 işlevi şu şekilde tamamlanır: • MQXCC_FAILED • MQXCC_*	• MQXCC_FAILED için: 1. Sürücü <i>bundan sonra</i> MQ_OPEN_EXIT 1 işlevi 2. MQCC_FAILED ile MQXX_ENCODE_CASE_ONE open çağrısını tamamla, MQRC_API_EXIT_ERROR • MQXCC_* için * 1. MQXCC_* ve MQXR2_*¹değerleri için hareket edin. 2. Çıkış engellenmezse <i>bundan sonra</i> MQ_OPEN_EXIT 1 işlevini kullan

Çizelge 206. API çıkış hataları ve yapılması gereken uygun işlemler (devamı var)

Err Pt	Tanım	İşlemler
e25	Sonra MQ_DISC_EXIT 2 işlevi aşağıdaki gibi tamamlanır: • MQXCC_FAILED • MQXCC_*	• MQXCC_FAILED için: 1. Sürücü <i>bundan sonra</i> MQ_DISC_EXIT 1 işlevi 2. Sürücü MQ_TERM_EXIT işlevi 3. Çıkış yürütme ortamını temizle 4. MQCC_FAILED ile MQDISC çağrısını tamamla, MQRC_API_EXIT_ERROR • MQXCC_* için * 1. MQXCC_* ve MQXR2_*¹değerleri için hareket edin. 2. Sürücü MQ_TERM_EXIT işlevi 3. Çıkış yürütme ortamını temizle

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 döndürülebiliyor.

- 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.

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 236’ 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 çağrıları 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 çağıran 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 arabirimi, bileşeni tanımlı bir şekilde kullanmasını gerektirmez.

SecurityParms

Tip: MQCSP-input

Güvenlik parametreleri. Kullanıcı kimliği, parola ve kimlik doğrulama türleriyle ilgili veriler. MQCSP yapısının AuthenticationType özneliği 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 [“MQCSP-Güvenlik değıştirgeleri” sayfa 328](#) başlıklı konuya bakın.

Bir MQCONN MQI çağırısı sırasında bu değıştirge boş değeri (null) ya da varsayılan değeri içeriyor.

ApplicationContext

Tip: MQZAC-input

Uygulama bağlamı. Çağrılan uygulamayla ilgili veriler. Ayrıntılar için bkz. [MQZAC-Application context](#).

Tüm MQCONN ya da MQCONNX MQI çağırısı sırasında, MQZAC yapısındaki kullanıcı bağlamı bilgileri yeniden alınır.

IdentityContext

Tip: MQZIC-giriş/çıkış

Kimlik bağlamı. Kullanıcı kimliği doğrulamasına giriş sırasında, bu değeri yürürlükteki kimlik bağlamını tanımlar. Kullanıcı kimliği doğrulama işlevi, kuyruk yöneticisinin yeni kimlik bağlamını benimsediği noktada bu değışikliğı değıştirebilir. 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 diğeri 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 değışiklikler korunur ve bu bileşenin sonraki işlevlerinden biri çağrıldığında sunulur.

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 değeri belirtebilirsiniz:

MQZCI_VARSAYILAN

Diğeri bileşenlere bağlı olarak devam edin.

MQZCI_STOP

Sonraki bileşenle devam ettirmeyin.

CompCode

Tip: MQXX_ENCODE_CASE_ONE long-output

Tamamlanma kodu. Bu değeri aşağıdaki değeri den 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 değiştirgeleri şu şekilde 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 işlev bir MQZAS_VERSION_1 yetki hizmeti bileşeni tarafından sağlanır ve kuyruk yöneticisi tarafından, belirli bir nesne üzerinde belirli bir eylemi ya da eylemleri gerçekleştirme yetkisinin olup olmadığını denetlemek için kuyruk yöneticisi tarafından başlatılır.

Bu işleve ilişkin işlev tanıtcısı (MQZEP için) MQZID_CHECK_AUTHORITY olur.

Sözdizimi

MQZ_CHECK_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, bileşeni tanımlı bir şekilde kullanmasını gerektirmez.

EntityName

Tip: MQCHAR12 -giriş

Varlık adı. Nesneye ilişkin yetkisi denetlenmek üzere olan varlığın adı. Dizilimin uzunluk üst sınırı 12 karakterdir; boşlukla doldurulsa boşluk karakterinden daha kısadır. Ad boş değerli bir karakter tarafından sonlandırılmadı.

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. EntityName ile 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.

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 nesneler 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* değiştirgesi MQZCI_default ya da MQZCI_CONTINUE ise, kuyruk yöneticisi varsa diğer bileşenleri çağırmaya 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 çağrılmaz 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 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.

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 olmadığını 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 , 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, 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 1692.](#)

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_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.

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 nesneler 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_YETKILIVER

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.

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') Uundering hizmeti kullanılamıyor.

Bu neden kodlarına ilişkin ek bilgi için [API tamamlama ve neden kodları](#) başlıklı konuya bakın.

C çağırma

```
MQZ_CHECK_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 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 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, 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 1692.

EntityType

Tip: MQHOME-girişi

Varlık tipi. EntityDatatarafından belirtilen varlık tipi. Bu değer aşağıdaki değerlerden biri olmalıdır:

MQZAET_PRINCIPAL

Müdür.

MQZAET_GRP

Grup.

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.

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 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_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') Underling hizmeti kullanılamıyor.

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 işlem, 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 işlem tanıttıcısı (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, değiştirgenin tam uzunluğuna kadar boşluklarla dolduruldu; ad boş değerli bir karakterle sonlandırılmaz.

Kuyruk yöneticisi adı, bileşene bilgi için iletilir; yetki hizmeti arabirimi, bileşeni tanımlı bir şekilde kullanmasını gerektirmez.

RefObjectAd

Tip: 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 sonlandırılmaz.

ObjectName

Tip: 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 sonlandırılmaz.

ObjectType

Tip: MQHOME-giriş

Nesne tipi. *RefObjectName* ve *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 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_SERVICE_ERROR

(2289, X'8F1') Hizmet erişimi beklenmeyen bir hata oluştu.

MQRC_SERVICE_NOT_VAR

(2285, X'8ED') Underling hizmeti kullanılamıyor.

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 (QMgriName, 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 işlem, bir yetki hizmeti bileşeni tarafından sağlanır ve kuyruk yöneticisi tarafından, belirtilen nesneyle ilişkili tüm yetkileri silmek için başlatılır.

Bu işleme ilişkin işlem tanıttıcısı (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, 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') Underling hizmeti kullanılamıyor.

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 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, bileşeni tanımlı bir şekilde kullanmasını gerektirmez.

StartEnumeration

Tip: MQHOME-girişi

Çağrılarının 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 çağrılarının 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 yoksayılır ve boş (null) gösterge olarak belirtilebilir (seçim ölçütleri, *StartEnumeration* tarafından MQZSE_START değerine ayarlanmış, çağrıyla belirtilen *Filter* parametresi tarafından belirlenir).

Filter

Tip: MQZAD-input

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. “MQZAD-Yetki verileri” sayfa 1689 .

StartEnumeration MQZSE_CONTINUE ise, *Filter* yoksayılr ve boş değerli gösterge olarak belirtilebilir.

AuthorityBufferUzunluğu

Tip: MQHOME-girişi

*AuthorityBuffer*uzunluğu. Bu, *AuthorityBuffer* parametresindeki bayt cinsinden uzunluğdır. Yetki arabelleği, döndürülebilmek için gereken verileri sığdırabilmek için yeterince büyük olmalıdır.

AuthorityBuffer

Tip: MQZAD-output

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: Not: Bu değıştirge, arabellek başlangıcındaki MQZAD ' ın 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.

AuthorityDataUzunluk

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 işlevlerden herhangi biri tarafından yapılan değışiklikler 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_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 değër aşağıdaki değërlerden 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 tanıtcısı (MQZEP için) MQZID_FREE_USER.

Sözdizimi

MQZ_FREE_USER(QMgrName , FreeParms , 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, 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 değıştirmeler” sayfa 1695.](#)

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 değışiklikler 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

Devam işareti. Aşağıdaki değler belirlenebilir:

MQZCI_VARSAYILAN

Diğer bileşenlere bağılı olarak devam edin.

MQZCI_STOP

Sonraki bileşenle devam ettirmeyin.

CompCode

Tip: MQXX_ENCODE_CASE_ONE long-output

Tamamlanma kodu. Bu değer aşağıdaki değlerden 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, değiştirgenin tam uzunluğuna kadar boşluklarla dolduruldu; ad boş değerli bir karakterle sonlandırılmaz.

Kuyruk yöneticisi adı, bileşene bilgi için iletilir; yetki hizmeti arabirimi, 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 doldurulsa boşluk karakterinden daha kısadır. Ad boş değerli bir karakter tarafından sonlandırılmadı.

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 alınmak üzere 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

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_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 (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_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 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, 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 1692.](#)

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') Underlying 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.

Sözdizimi

MQZ_GET_AUTHORITY_2 (*QMgrName*, *EntityData*, *EntityType*, *ObjectName*, *ObjectType*, *Authority*, *ComponentData*, *Continuation*, *CompCode*, *Reason*)

C çağırma

```
MQZ_GET_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_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 UNIX platformları, 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 , 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, 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-giriş

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. 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 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 bağlı.

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') Underling 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, değıştirgenin tam uzunluğuna kadar boşluklarla dolduruldu; ad boş değeri bir karakterle sonlandırılmaz.

Kuyruk yöneticisi adı, bileşene bilgi için iletilir; yetki hizmeti arabirimi, 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 1692.](#)

EntityType

Tip: MQHOME-giriş

Varlık tipi. *EntityData* tarafından belirtilen varlık tipi. Bu değeri aşağıdaki değeriilerden 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 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 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.

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 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_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 işlev, 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 tanıttıcısı (MQZEP için) MQZID_INIT_AUTHORITY 'dir.

Sözdizimi

*MQZ_INIT_AUTHORITY(Hconfig , Options , QMgrName , ComponentDataLength ,
ComponentData , Version , CompCode , Reason)*

Parametreler

Hconfig

Tip: MQHCONFIG-input

Yapılandırma tanıttıcısı. Bu tanıttıcı, ilk kullanıma hazırlanmakta olan bileşeni gösterir. Bu bileşen, kuyruk yöneticisini MQZEP işleviyle çağırırken, bu bileşen tarafından kullanılır.

Seçenekler

Tip: MQHOME-giriş

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, değıştirmenin tam uzunluğuna kadar boşluklarla dolduruldu; ad boş değeri bir karakterle sonlandırılmaz.

Kuyruk yöneticisi adı, bileşene bilgi için iletilir; yetki hizmeti arabirimi, 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 çağırmadan 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 çağrılır.

Bu veri alanının uzunluğu, MQZ_INIT_AUTHORITY çağrısının *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, 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

Sürüm 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 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, 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 özneliği 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 göründükleri sırayla geri döndürülür.

Character attributes identified by selectors are returned in the *CharAttrs* parameter in the same order as they appear *Selectors*.

IntAttrSayı

Tip: MQHOME-giriş

IntAttrs değiştirilmesinde 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-giriş

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 dizgisi 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 tanıttıcısı (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, değiştirmenin 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

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 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

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_UYARI ise:

MQRC_SERVICE_ERROR

(2289, X'8F1') Hizmet erişimi beklenmeyen bir hata oluştu.

C çağırma

```
MQZ_REFRESH_CACHE (QMgrName, ComponentData,  
                   &Continuation, &CompCode, &Reason);
```

Değiştiricileri 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 tanıttıcısı (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, değıştirgenin tam uzunluğuna kadar boşluklarla dolduruldu; ad boş değeri bir karakterle sonlandırılmaz.

Kuyruk yöneticisi adı, bileşene bilgi için iletilir; yetki hizmeti arabirimi, 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 değeriilerden 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 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 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.

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_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') Underlying 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 (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 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 tanıtcısı (MQZEP için) MQZID_SET_AUTHORITY olur.

Not: Bu işlev, var olan yetkileri geçersiz kılar. Varolan yetkileri korumak için, bu işlevi bu işleyle yeniden ayarlamanız 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* ,
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, 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 1692.](#)

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 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 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.

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') Underling 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,  
                    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 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 tanıtcısı (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-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 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, 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 değiştirgesindeki 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') Underlying hizmeti kullanılamıyor.

MQRC_TERMINATION_FAILED

(2287, X'8FF') Sona erdirmeye 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 */  
MQQLONG    Options;          /* Termination options */  
MQCHAR48   QMgrName;         /* Queue manager name */  
MQBYTE     ComponentData[n]; /* Component data */  
MQQLONG    CompCode;         /* Completion code */  
MQQLONG    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 tanıtcısı (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 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, 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** değıştirgesinde ne döndürüldüğü önemli deęil, başka bir bileşeni başlatmaya çalışmamaktadır.

CompCode

Tip: MQXX_ENCODE_CASE_ONE long-output

Tamamlanma kodu. Bu deęer aşığıdaki deęerlerden 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 kullanılamıyor.

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 aşığıdaki 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 çağırması 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, 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, 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') Udering 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 tanıttıcısı (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, 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') Underlying 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 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, 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: 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: MQXX_ENCODE_CASE_ONE long-output

Bileşene göre devam göstergesi kümesi. MQZ_LOOKUP_NAME için, kuyruk yöneticisi aşağıdaki gibi başka bir ad hizmeti bileşeninin başlatılıp başlatılmayacağını belirtir:

- *CompCode* MQCC_OK ise, başka bileşen başlatılmıyorsa, *Devamı*' da 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 bağlıdır.

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') Underlying 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 gerçekleştirmelidir.

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-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, değıştirgenin tam uzunluğuna kadar boşluklarla dolduruldu; ad boş değeri bir karakterle sonlandırılmaz.

Kuyruk yöneticisi adı, bileşene bilgi için iletilir; yetki hizmeti arabirimi, 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 çağırılır.

Bileşen verileri, paylaşılan belleğde tüm süreçler tarafından erişilebilir.

Bu veri alanının uzunluğu, MQZ_INIT_NAME çağırısının *ComponentDataLength* parametresindeki kuyruk yöneticisi tarafından geçirilir.

MQZ_TERM_NAME çağırısı tamamlanınca, 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_TERMINATION_FAILED

(2287, X'8FF') Sona erdirmeye tanımlanmamış bir nedenden dolayı başarısız oldu.

MQRC_SERVICE_NOT_VAR

(2285, X'8ED') Underlying hizmeti kullanılamıyor.

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 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 */
```

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 207. MQZAC 'deki alanlar	
Alan	Tanım
<u>StrucId</u>	Yapı tanıtcısı
<u>S\u00f0fcr\u000fcm</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>	Çağırıcı tipi
<u>AuthenticationType</u>	Kimlik denetimi tipi

Çizelge 207. MQZAC ' deki alanlar (devamı var)	
Alan	Tanım
<u>BindType</u>	Bağ Tipi

Alanlar

StrucId

Tip: MQCHAR4 -giriş

Yapı tanıtıcısı. Değer aşağıdaki gibidir:

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.

S\|u00fcr\|u00fcm

Tip: MQHOME-girişi

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 tanıtıcısı.

ThreadId

Tip: MQTID-input

Uygulamanın iş parçacığı tanıtıcısı.

ApplName

Tip: MQCHAR28 -giriş

Uygulama adı.

UserID

Tip: MQCHAR12 -giriş

Kullanıcı kimliği. UNIX sistemleri üzerinde bu alan, uygulamanın gerçek kullanıcı kimliğini belirtir. Windows tarihinde bu alan, uygulamanın kullanıcı kimliğini belirtir.

EffectiveUserTanıtıcısı

Tip: MQCHAR12 -giriş

Etkin kullanıcı kimliği. UNIX sistemleri tarihinde bu alan, uygulamanın etkin kullanıcı kimliğini belirtir. Windows üzerinde bu alan boştur.

Ortam

Tip: MQHOME-girişi

Ortam. Bu alan, çağrılar yapıldığı ortamı belirtir. Alan, aşağıdaki değerlerden biridir:

MQXE_COMMAND_SERVER

Komut sunucusu

MQXE_MQSC

runmqsc komut yorumlayıcısı

MQXE_MCA

İleti kanalı aracı MQXE_OTHER

MQXE_DIĞER

Tanımsız ortam

CallerType

Tip: MQHOME-giriş

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 arasındır.

AuthenticationType

Tip: MQHOME-giriş

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-giriş

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 bildirimi

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.

- 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 208. MQZAD 'daki alanlar	
Alan	Tanım
<u>StrucId</u>	Yapı tanıttıcısı
<u>S\u00f0fcr\u00f0fcm</u>	Yapı sürüm numarası
<u>ProfileName</u>	Tanıttım adı
<u>ObjectType</u>	Nesne tipi
<u>Yetki</u>	Yetki
<u>EntityDataPtr</u>	Varlık verilerine işarettçi
<u>EntityType</u>	Varlık tipi
<u>Seçenekler</u>	Seçenekler

Alanlar

StrucId

Tip: MQCHAR4 -giriş

Yapı tanıttıcısı. Değer aşğıdaki gibidir:

MQZAD_STRUC_ID

Yetki verileri yapısına ilişkin tanıttıcı.

C programlama dili için, MQZAD_STRUC_ID_ARRAY değışmezi de tanımlanır; bu, MQZAD_STRUC_ID ile aynı değere sahiptir, ancak dizgi yerine bir karakter dizisidir.

S\u00f0fcr\u00f0fcm

Tip: MQHOME-giriş

Yapı sürüm numarası. Değer aşğıdaki gibidir:

MQZAD_VERSION_1

Version-1 uygulama bağılamı yapısı. Değışmez MQZAD_CURRENT_VERSION, yürürlükteki sürümün sürüm numarasını belirtir.

Aşğıdaki değışmez, 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 olduğı profil adıdır. Ad alanın sonuna kadar boş bırakılırsa ya da ilk boş karakter, tüm tanıttım adları için yetki verileri döndürölür.

AuthorityBuffer parametresi için bu alan, belirtilen seçim ölçütleriyle eşleşen bir profilin adıdır.

ObjectType

Tip: MQHOME-giriş

Nesne tipi.

Filter parametresi için bu alan, yetki verilerinin gerekli olduğı nesne tipidir. Bu değeri MQOT_ALL ise, tüm nesne tiplerine ilişkin yetki verileri döndürölür.

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

Seenekler

Tip: MQAUTHOPT-input

Seenekler. Bu alan, grntlenen tanıtımlar zerinde denetim veren seenekleri belirtir. Ařağıdaki deęerlerden biri belirtilmeli:

MQAUTHOPT_NAME_ALL_MATCHING

Tm profilleri grntler

MQAUTHOPT_NAME_EXPLICIT

ProfileName alanında belirtildięi gibi, tam olarak aynı ada sahip profilleri grntler.

Ayrıca, ařağıdakilerden birinin de belirtilmesi gerekir:

MQAUTHOPT_ENTITY_SET

Varlığın, **ProfileName** parametresi tarafından belirtilen nesneye sahip olduęu birikmeli yetkiyi hesaplamak iin kullanılan tm profilleri grntler. **ProfileName** parametresi genel arama karakteri iermemelidir.

- 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 iin geerli olan gruptan en geerli profil grntlenir.
- 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 belirlediyseniz, ařağıdaki deęeri de belirtebilirsiniz:

MQAUTHOPT_ENTITY_EXPLICIT

EntityDataPtr MQZED yapısında belirtilen varlık adıyla tam olarak aynı varlık adına sahip profilleri grntler.

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 denetleneceęi varlığı belirtmek iin bir dizi yetki hizmeti aęırısında kullanılır.

Tablo 1. Yapıdaki alanları zetler.

izelge 209. MQZED 'deki alanlar	
Alan	Tanım
<u>StrucId</u>	Yapı tanıtıcısı
<u>S\u00f0fcr\u00f0fcm</u>	S\u00f0fcr\u00f0fcm
<u>EntityName Ptr</u>	Varlık Adı
<u>EntityDomainPtr</u>	Varlık etki alanı gstergesi

Çizelge 209. MQZED ' deki alanlar (devamı var)	
Alan	Tanım
<u>SecurityId</u>	Güvenlik tanıtıcısı
<u>CorrelationPtr</u>	İlinti göstergesi

Alanlar

StrucId

Tip: MQCHAR4 -giriş

Yapı tanıtıcısı. Değer aşağıdaki gibidir:

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.

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 işarettir.

SecurityId

Tip: MQBYTE40 -giriş

Otorite.

Güvenlik tanıtıcısı. Bu, yetkisini denetlenecek güvenlik tanımlayıcısıdır.

CorrelationPtr

Tip: MQPTR-input

İlinti göstergesi. Bu, kimlik doğrulama kullanıcı işlevi ile diğer uygun OAM işlevleri arasında ilişkisel 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 */
```

Alanlar

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 tanıtıcısı. Bu tanıtıcı, bu kurulabilir hizmet için yapılandırılmakta olan bileşeni gösterir. Bileşen başlatma çağrısındaki kuyruk yöneticisi tarafından bileşen yapısını işlevine geçirilen bileşenle aynı olmalıdır.

İşlev

Tip: MQHOME-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

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 210. MQZFP içindeki alanlar	
Alan	Tanım
<u>StrucId</u>	Yapı tanıtıcısı
<u>S\u00fcr\u00fcr</u>	S\u00fcr\u00fcr
<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\u00fcr

Tip: MQHOME-girişi

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 */
};
```

Alanlar

MQZIC-Kimlik bağlamı

MQZIC yapısı, *IdentityContext* parametresi için MQZ_AUTHENTICATE_USER çağrısında kullanılıyor.

MQZIC yapısı, iletiyi ilk olarak bir kuyruğa koyan uygulamanın kullanıcılarını tanıtan 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 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 211. MQZIC 'deki alanlar	
Alan	Tanım
<u>StrucId</u>	Yapı tanıtıcısı
<u>S\u00f0fcr\u00f0fcm</u>	S\u00f0fcr\u00f0fcm
<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\u00f0fcr\u00f0fcm

Tip: MQHOME-girişi

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 447.](#)

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 dizisi olarak değerlendirir ve içeriğini denetmez. *AccountingToken* alanıyla ilgili daha fazla bilgi için bkz. [“AccountingToken \(MQBYTE32\)” sayfa 407.](#)

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 409.](#)

C bildirimi

```
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 */
};
```

Alanlar

Kurulabilir hizmetler arabirimi başvuru bilgileri

Kurulabilir hizmetlere ilişkin başvuru bilgilerini anlamak için bu bilgileri kullanın.

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 967](#)’inde açıklanan temel veri tipleridir.

Parametrelerin açıklamasından sonra, C dili çağrısına da verilir.

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 işlevle 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 konuyu bakın.

C çağırma

```
MQZEP (Hconfig, Function, EntryPoint, &CompCode, &Reason);
```

Değiştirelileri aşağıdaki gibi bildirin:

```
MQHCONFIG  Hconfig;      /* Configuration handle */
MQQLONG    Function;     /* Function identifier */
PMQFUNC    EntryPoint;    /* Function entry point */
```

```
MQLONG    CompCode;    /* Completion code */
MQLONG    Reason;      /* Reason code qualifying CompCode */
```

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.

Uygulamalar, yalnızca eşitlik için bu tipteki değişkenleri test etmelidir.

C bildirimi

```
typedef void MQPOINTER MQHCONFIG;
```

PMQFUNC-İşaretçi işlevi

Bir işleve işaretçi.

C bildirimi

```
typedef void MQPOINTER PMQFUNC;
```

MQZ_AUTHENTICATE_USER-Kullanıcı kimliği doğrulamalı kullanıcı

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 çağrıları 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 çağırان 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, 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. “MQZAC-Uygulama bağlamı” sayfa 1729. 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 “MQZIC-Kimlik bağlamı” sayfa 1735 konusuna bakın.

CorrelationPtr (MQPTR)-çıkış

İlinti göstergesi.

Herhangi bir ilinti verisinin adresini belirtir. Bu işaretçi daha sonra diğer OAM çağrılarına 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 */
```

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 tanıttıcısı (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 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 ilişkin yetkisi denetlenmek üzere olan varlığın adı. Dizilimin uzunluk üst sınırı 12 karakterdir; boşlukla doldurulsa boşluk karakterinden daha kısadır. Ad boş değerli bir karakter tarafından sonlandırılmaz.

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_GRP

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 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 (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 nesneler 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_YETKILIVER

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 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 geçilir.

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') Underlying 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,  
ObjectTypes, 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 ObjectTypes;         /* 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 tanıtcısı (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 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, 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 1692.

EntityType

Tip: MQHOME-girişi

Varlık tipi. EntityDatatarafından belirtilen varlık tipi. Bu değer aşağıdaki değerlerden biri olmalıdır:

MQZAET_PRINCIPAL

Müdür.

MQZAET_GRP

Grup.

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 geçilir.

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 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 değeri aşağıdaki 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 değeri.

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') Underlying hizmeti kullanılamıyor.

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 çağrılır.

Bu işleve ilişkin işlev tanıtcısı (MQZEP için) MQZID_COPY_ALL_AUTHORITY olur.

Sözdizimi

MQZ_COPY_ALL_AUTHORITY (*QMgrName, RefObjectName, ObjectName, ObjectType, 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 (MQZE)-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') Underlying 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 */
```

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 tanıttıcısı (MQZEP için) MQZID_DELETE_AUTHORITY 'dir.

Sözdizimi

**MQZ_DELETE_AUTHORITY (QMgrName, ObjectName, ObjectType,
ComponentData, Continuation, CompCode, Reason)**

Parametreler

MQZ_DELETE_AUTHORITY çağrısının değıştirtgeleri şunlardır:

QMgrName (MQCHAR48)-giriş

Kuyruk yöneticisi adı.

Bileşeni çağıran 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 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 değeri 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 bağlı.

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 */
```

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 sürekli olarak çağrılır.

Bu işleve ilişkin işlev tanıtcısı (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 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.

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 çağrılar 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 1731](#).

StartEnumeration MQZSE_CONTINUE 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 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: Bu değiştirge, arabellek başlangıcındaki MQZAD ' ın her zaman oluştuğu 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.

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 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.

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.

EntityName tarafından belirtilen varlık tipi. Aşağıdaki değer belirlenebilir:

MQZAET_PRINCIPAL

Müdür.

MQZAET_GRP

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') Underling 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_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_REZILIT_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 değişirgeler var.

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.

EntityName (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 (MQUBE)-giriş

Varlık tipi.

EntityName tarafından belirtilen varlık tipi. Aşağıdaki değer belirlenebilir:

MQZAET_PRINCIPAL

Müdür.

MQZAET_GRP

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_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') Undering 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 */
```

MQZ_INIT_AUTHORITY-Yetkilendirme hizmetini kullanıma hazırla

Bu işlev bir yetkilendirme hizmeti bileşeni tarafından sağlanır ve bileşen yapılandırması sırasında kuyruk yöneticisi tarafından çağırılır. 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 çağrısının değıştirmeleri şunlardır:

Hconfig (MQHCONFIG)-giriş

Yapılandırma tanıtıcısı.

Bu tanıtıcı, ilk kullanıma hazırlanmakta olan bileşeni gösterir. Bu bileşen, kuyruk yöneticisini MQZEP işleviyle çağırırken, bu bileşen tarafından kullanılır.

Seçenekler (MQUZE)-giriş

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, değıştirgenin tam uzunluğuna kadar boşluklarla dolduruldu; ad boş değeri 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 çağırmadan 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 değışiklikler korunur ve bu bileşenin sonraki işlevlerinden biri çağrıldığında 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 değeri 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

Sürüm 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 kullanılamıyor.

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 */
```

MQZ_SORMA-YETKILENDİRME HİZMETİ

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 tanıtcısı (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 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.

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 değiştirgesinde 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; *MQZSL_NOT_RETURNED*. Bir yetki hizmeti bileşeni bir öznitelik döndürdüğünde, dizideki uygun değeri *MQZSL_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.

MQRC_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 */
MLONG	SelectorCount;	/* Selector count */
MLONG	Selectors[n];	/* Selectors */
MLONG	IntAttrCount;	/* IntAttrs count */
MLONG	IntAttrs[n];	/* Integer attributes */
MLONG	CharAttrCount;	/* CharAttrs count */
MLONG	CharAttrs[n];	/* Chatacter attributes */
MLONG	SelectorReturned[n];	/* Selector returned */
MQBYTE	ComponentData[n];	/* Component data */
MLONG	Continuation;	/* Continuation indicator set by component */
MLONG	CompCode;	/* Completion code */
MLONG	Reason;	/* Reason code qualifying CompCode */

MQZ_REFRESH_CACHE-Tüm yetkileri yenile

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, 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* değiştirgesinde 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);
```

Değiştirgeleri 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 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 ayarlamanız gerekir.

Sözdizimi

MQZ_SET_AUTHORITY (QMgrName, EntityName, EntityType, ObjectName, ObjectType, Authority, ComponentData, Continuation, CompCode, Reason)

Parametreler

MQZ_SET_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.

EntityName (MQCHAR12)-giriş

Varlık adı.

Nesneye erişimin ayarlanacağı varlığın adı. Dizilimin uzunluk üst sınırı 12 karakterdir; boşlukla doldurulsa boşluk karakterinden daha kısadır. Ad boş değerli bir karakter tarafından sonlandırılmaz.

EntityType (MQUBE)-giriş

Varlık tipi.

EntityName tarafından belirtilen varlık tipi. Aşağıdaki değer belirlenebilir:

MQZAET_PRINCIPAL

Müdür.

MQZAET_GRP

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 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.

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 bağımlı.

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 tanıttıcısı (MQZEP için) MQZID_TERM_AUTHORITY olur.

Sözdizimi

MQZ_TERM_AUTHORITY (Hconfig, Options, QMgrName, ComponentData, CompCode, Reason)

Parametreler

MQZ_TERM_AUTHORITY çağrısının değişitirgeleri şunlardır:

Hconfig (MQHCONFIG)-giriş

Yapılandırma tanıttıcısı.

Bu tanıttıcı, 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 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.

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') Underlying hizmeti kullanılamıyor.

MQRC_TERMINATION_FAILED

(2287, X'8FF') Sona erdirmeye 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 */
```


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 (MQUZN)

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.

For IBM i systems the user profile that the application job was created under. (IBM i: when a profile swap is done with the QWTSETP API in the application job the current user profile is returned).

EffectiveUserkimliği (MQCHAR12)

Etkin kullanıcı kimliği.

IBM i sistemleri için, uygulama işinin yürürlükteki kullanıcı tanıtımı.

Ortam (MQLONG)

Ortam.

Bu alan, çağrıların 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 bildirimi

```
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.

İki parametre şunlardır:

- 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 tanımlı 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 tanıtlar 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_MATCHINGbelirttiyse, 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 */  
};
```

MQZED-Varlık tanımlayıcısı

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 işarettir.

EntityDomainPtr (PMQCHAR)

Varlık etki alanı adının adresi.

Bu, yetkisini denetlenecek olan varlığın tanımını içeren etki alanının adına işarettir.

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 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 */
}
```

MQZFP-Serbest değiştirgeler

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 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* 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 Windowsaltında 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 447](#).

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 dizisi olarak değerlendirir ve içeriğini denetmez. *AccountingToken* alanıyla ilgili daha fazla bilgi için bkz. [“AccountingToken \(MQBYTE32\)” sayfa 407](#).

ApplIdentityVerileri (MQCHAR32)

Kimlikle ilgili uygulama verileri.

Bu, iletinin **kimlik bağlamının** bir parçasıdır.

ApplIdentityData, iletinin kökeniyle ilgili ek bilgi sağlamak için kullanılabilecek 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 409](#).

C bildirimi

```
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 bridge for HTTP için başvuru malzemesi

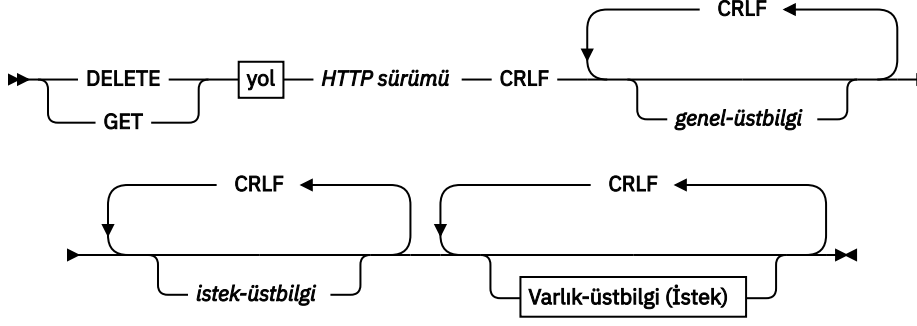
Alfabetik olarak düzenlenmiş, IBM MQ bridge for HTTP için başvuru konuları

HTTP DELETE: IBM MQ bridge for HTTP komutu

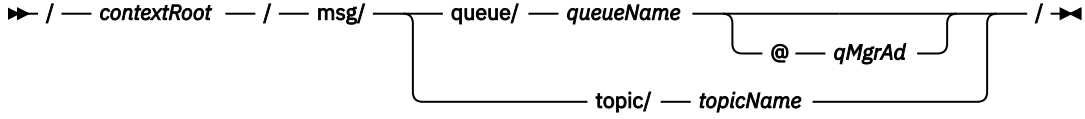
HTTP **DELETE** işlemi, IBM MQ kuyruğundan bir ileti alır ya da bir konudan bir yayını alır. İleti kuyruktan kaldırılır. Yayın alıkonursa, kaldırılmaz. İletiyeye ilişkin bilgiler de içinde olmak üzere, istemciye bir yanıt iletisi gönderilir.

Sözdizimi

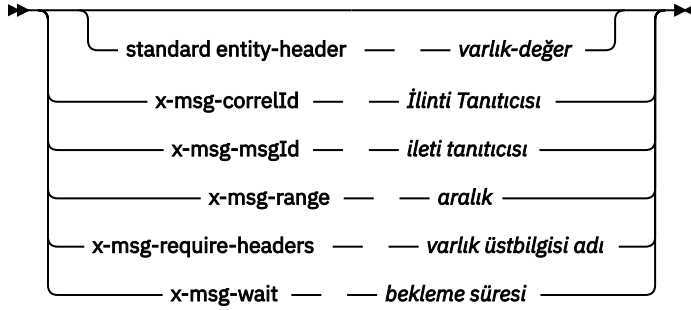
İstek



Yol



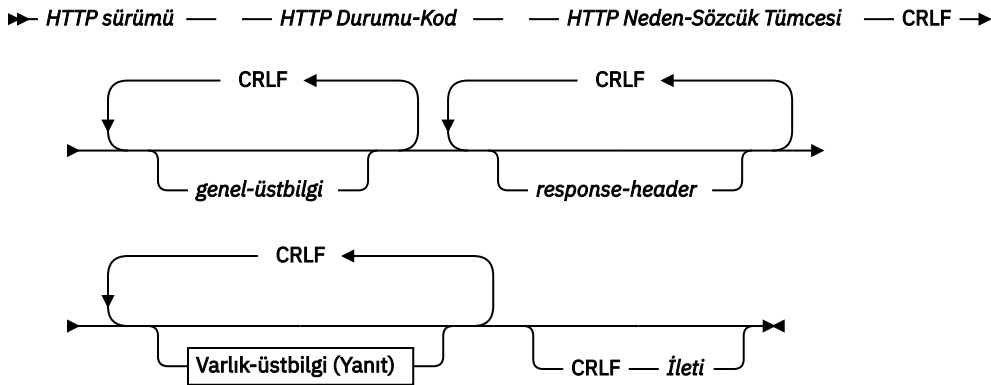
entity-header (İstek)



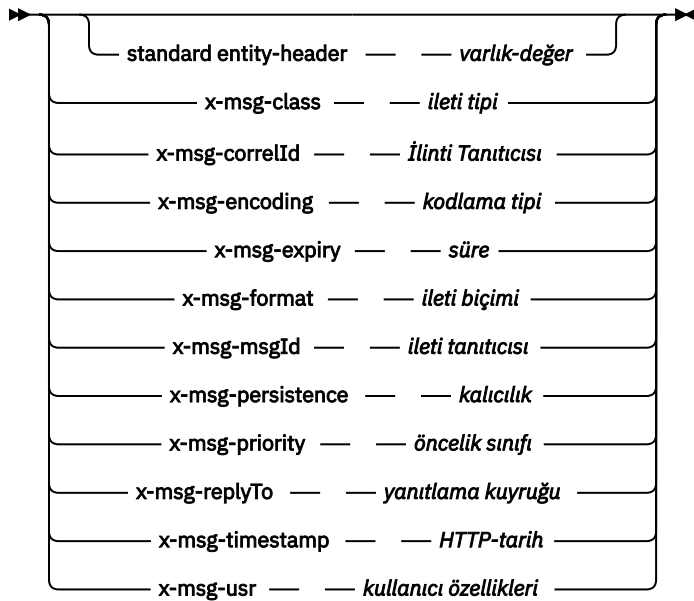
Not:

1. If a question mark (?) is used it must be substituted with %3f. Örneğin, orange?topic , orange%3ftopicolarak belirtilmeli.
2. @ qMgrAd yalnızca HTTP **POST**' de geçerlidir

Yanıt



entity-header (Yanıt)



İstek parametreleri

Yol

Bkz. [“URI Biçimi” sayfa 1771.](#)

HTTP sürümü

HTTP sürümü; örneğin, HTTP/1.1

genel-üstbilgi

Bkz. [HTTP/1.1 - 4.5 Genel Üstbilgi Alanları.](#)

istek-üstbilgi

Bkz. [HTTP/1.1 - 5.3 İstek Üstbilgisi Alanları.](#) Host (Anasistem) alanı HTTP/1.1 isteğinde zorunludur. Genellikle, istemci isteği yaratmak için kullandığınız araç tarafından otomatik olarak araya eklenir.

entity-header (İstek)

Bkz. [HTTP/1.1 - 7.1 Varlık Üstbilgisi Alanları.](#) İstek sözdizimi şemasında listelenen varlık üstbilgilerinden biri.

Yanıt değiştirgeleri

Yol

Bkz. [“URI Biçimi” sayfa 1771.](#)

HTTP sürümü

HTTP sürümü; örneğin, HTTP/1.1

genel-üstbilgi

Bkz. [HTTP/1.1 - 4.5 Genel Üstbilgi Alanları.](#)

yanıt-üstbilgi

Bkz. [HTTP/1.1 - 6.2 Yanıt Üstbilgisi Alanları.](#)

varlık-üstbilgi (Yanıt)

Bkz. [HTTP/1.1 - 7.1 Varlık Üstbilgisi Alanları.](#) Yanıt sözdizimi şemasında listelenen varlık ya da yanıt üstbilgilerinden biri. İçerik-Uzunluğu her zaman bir yanıtta bulunur. İleti gövdesi yoksa sıfır olarak ayarlanır.

İleti

İleti gövdesi.

Tanım

HTTP **DELETE** isteği başarılı olursa, yanıt iletisi IBM MQ kuyruğundan alınan verileri içerir. HTTP İçerik-Uzunluğu üstbilgisinde, iletinin gövdesindeki byte sayısı döndürülür. HTTP yanıtına ilişkin durum kodu 200 OK olarak ayarlandı. x-msg-range , 0 ya da 0-0 olarak belirtilirse, HTTP yanıtının durum kodu 204 No Content olur.

HTTP **DELETE** isteği başarısız olursa, yanıt bir IBM MQ bridge for HTTP hata iletisi ve bir HTTP durum kodu içerir.

HTTP DELETE örneği

HTTP **DELETE** , kuyruktan bir ileti alır ve iletiyi siler ya da bir yayını alır ve siler. **HTTPDELETE** Java örneği, bir kuyruktan ileti okuma isteğinde bulunan bir HTTP **DELETE** örneğidir. Java kullanmak yerine, bir tarayıcı formu ya da AJAX araç takımı kullanarak bir HTTP **DELETE** isteği yaratabilirsiniz.

Aşağıdaki şekil, myQueueadlı kuyruğdaki bir sonraki iletiyi silmek için bir HTTP isteğini göstermektedir. Yanıtta, ileti gövdesi istemciye döndürülür. IBM MQ terimlerinde, HTTP **DELETE** yıkıcı bir alma koşuldur.

The request contains the HTTP request header x-msg-wait, which instructs IBM MQ bridge for HTTP how long to wait for a message to arrive on the queue. İstek, istemcinin yanıtta ileti ilintilendirme tanıtıcısını almak olduğunu belirten x-msg-require-headers istek üstbilgisini de içerir.

```
DELETE /msg/queue/myQueue/ HTTP/1.1
Host: www.example.org
x-msg-wait: 10
x-msg-require-headers: correlID
```

Şekil 37. HTTP **DELETE** isteği örneği

Aşağıdaki şekil, istemciye döndürülen yanıtı gösterir. İlinti tanıtıcısı, isteğin x-msg-require-üstbilgileri içinde istendiği şekilde istemciye döndürülür.

```
HTTP/1.1 200 OK
Date: Wed, 2 Jan 2007 22:38:34 GMT
Server: Apache-Coyote/1.1 WMQ-HTTP/1.1 JEE-Bridge/1.1
Content-Length: 50
Content-Type: text/plain; charset=utf-8
x-msg-correlId: 1234567890

Here is my message body that is retrieved from the queue.
```

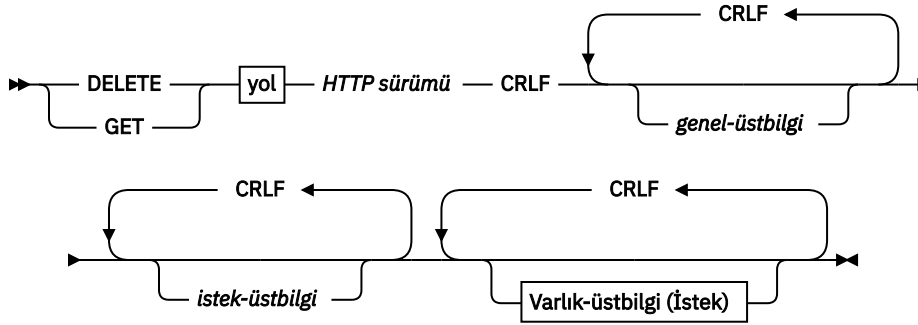
Şekil 38. HTTP **DELETE** yanıtı örneği

HTTP GET: IBM MQ bridge for HTTP komutu

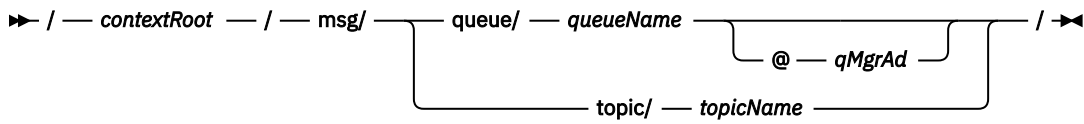
HTTP **GET** işlemi bir IBM MQ kuyruğundan ileti alır. İleti kuyruğun üzerinde kalır. HTTP **GET** işlemi, bir IBM MQ kuyruğuna göz atılabilmek için eşdeğerdir.

Sözdizimi

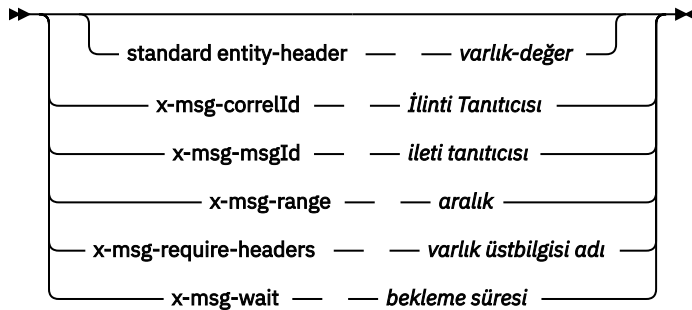
İstek



Yol



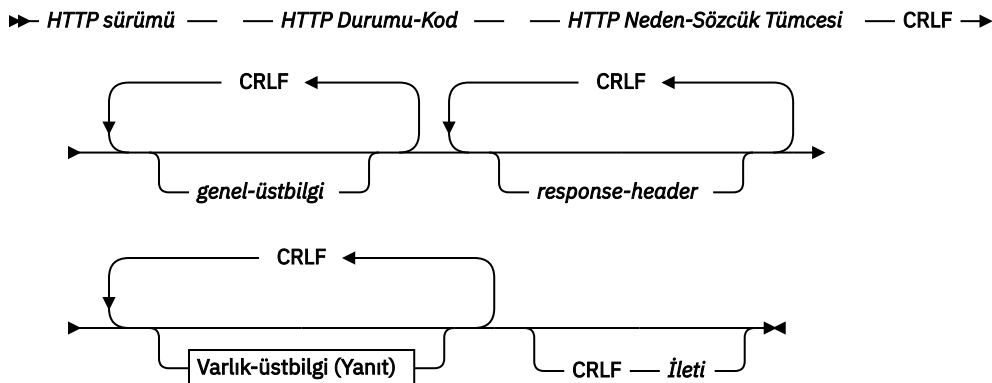
entity-header (İstek)



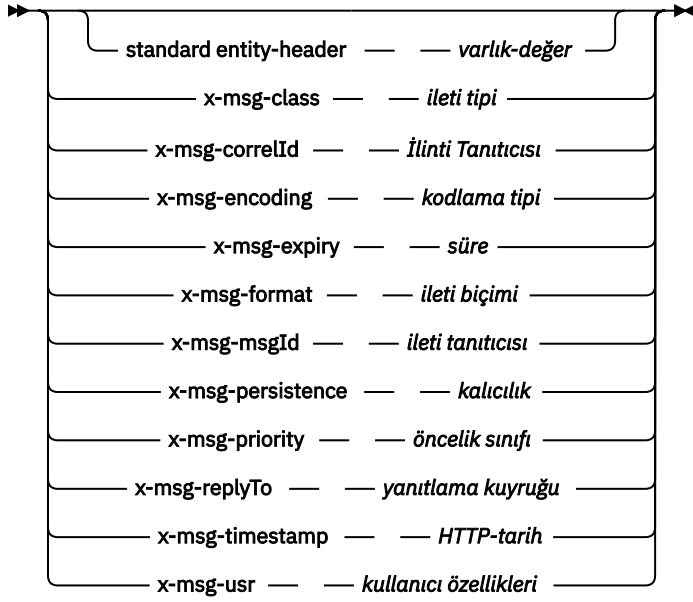
Not:

1. If a question mark (?) is used it must be substituted with %3f. Örneğin, orange?topic , orange%3ftopicolarak belirtilmeli.
2. @ qMgrAd yalnızca HTTP **POST**' de geçerlidir

Yanıt



entity-header (Yanıt)



İstek parametreleri

Yol

Bkz. [“URI Biçimi” sayfa 1771.](#)

HTTP sürümü

HTTP sürümü; örneğin, HTTP/1.1

genel-üstbilgi

Bkz. [HTTP/1.1 - 4.5 Genel Üstbilgi Alanları.](#)

istek-üstbilgi

Bkz. [HTTP/1.1 - 5.3 İstek Üstbilgisi Alanları.](#) Host (Anasistem) alanı HTTP/1.1 isteğinde zorunludur. Genellikle, istemci isteği yaratmak için kullandığınız araç tarafından otomatik olarak araya eklenir.

entity-header (İstek)

Bkz. [HTTP/1.1 - 7.1 Varlık Üstbilgisi Alanları.](#) İstek sözdizimi şemasında listelenen varlık üstbilgilerinden biri.

Yanıt değıştirgeleri

Yol

Bkz. [“URI Biçimi” sayfa 1771.](#)

HTTP sürümü

HTTP sürümü; örneğin, HTTP/1.1

genel-üstbilgi

Bkz. [HTTP/1.1 - 4.5 Genel Üstbilgi Alanları.](#)

yanıt-üstbilgi

Bkz. [HTTP/1.1 - 6.2 Yanıt Üstbilgisi Alanları.](#)

varlık-üstbilgi (Yanıt)

Bkz. [HTTP/1.1 - 7.1 Varlık Üstbilgisi Alanları.](#) Yanıt sözdizimi şemasında listelenen varlık ya da yanıt üstbilgilerinden biri. İçerik-Uzunluğu her zaman bir yanıtta bulunur. İleti gövdesi yoksa sıfır olarak ayarlanır.

İleti

İleti gövdesi.

Tanım

HTTP **GET** isteği başarılı olursa, yanıt iletisi IBM MQ kuyruğundan alınan verileri içerir. HTTP İçerik-Uzunluğu üstbilgisinde, iletinin gövdesindeki byte sayısı döndürülür. HTTP yanıtına ilişkin durum kodu 200 OK olarak ayarlandı. x-msg-range , 0 ya da 0-0 olarak belirtilirse, HTTP yanıtının durum kodu 204 No Content olur.

HTTP **GET** isteği başarısız olursa, yanıt bir IBM MQ bridge for HTTP hata iletisi ve bir HTTP durum kodu içerir.

HTTP GET örneği

HTTP **GET** , bir kuyruktan ileti alır. İleti kuyruğun üzerinde kalır. IBM MQ terimlerinde, HTTP **GET** bir göz atma isteğidir. Bir Java istemcisi, bir tarayıcı formu ya da AJAX araç takımı kullanarak bir HTTP **GET** isteği yaratabilirsiniz.

Aşağıdaki şekil, myQueueadlı kuyruğun sonraki iletisine göz atmak için bir HTTP isteğini göstermektedir.

The request contains the HTTP request header x-msg-wait, which instructs IBM MQ bridge for HTTP how long to wait for a message to arrive on the queue. İstek, istemcinin yanıtta iletinin ilintilendirme tanıtıcısını almak olduğunu belirten x-msg-require-headers istek üstbilgisini de içerir.

```
GET /msg/queue/myQueue/ HTTP/1.1
Host: www.example.org
x-msg-wait: 10
x-msg-require-headers: correlID
```

Şekil 39. HTTP **GET** isteği örneği

Aşağıdaki şekil, istemciye döndürülen yanıtı gösterir. İlinti tanıtıcısı, isteğin x-msg-require-üstbilgileri içinde istendiği şekilde istemciye döndürülür.

```
HTTP/1.1 200 OK
Date: Wed, 2 Jan 2007 22:38:34 GMT
Server: Apache-Coyote/1.1 WMQ-HTTP/1.1 JEE-Bridge/1.1
Content-Length: 50
Content-Type: text/plain; charset=utf-8
x-msg-correlId: 1234567890
```

Here is my message body that appears on the queue.

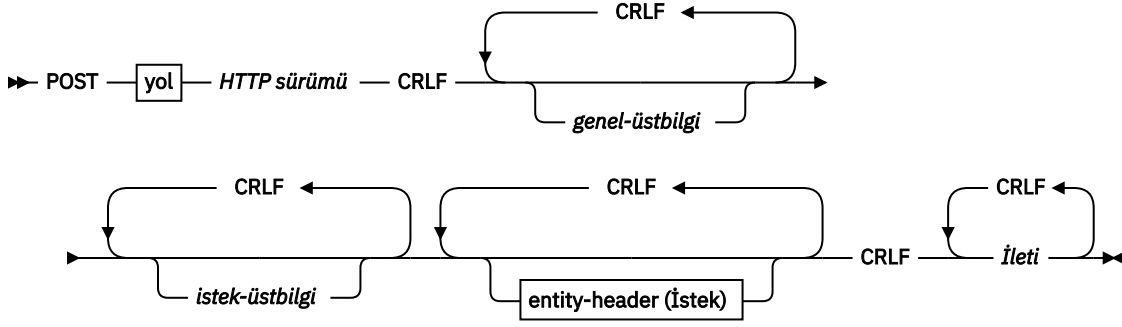
Şekil 40. HTTP **GET** yanıtı örneği

HTTP POST: IBM MQ bridge for HTTP komutu

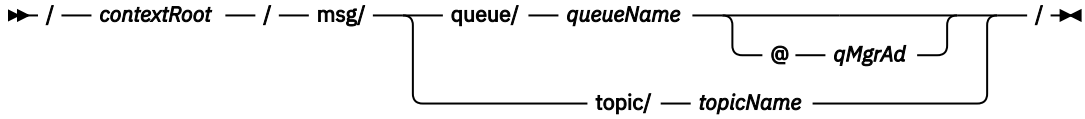
HTTP **POST** işlemi, bir iletiyi IBM MQ kuyruğuna koyar ya da bir konuya ileti yayınlar.

Sözdizimi

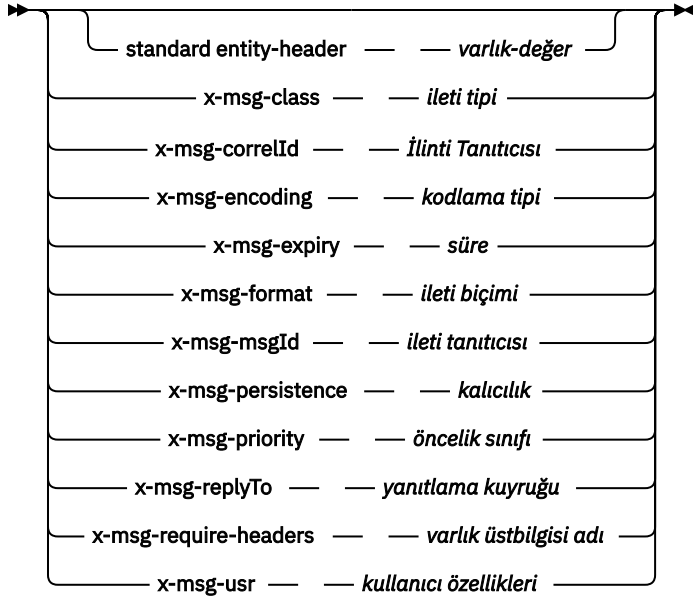
İstek



Yol



entity-header (İstek)

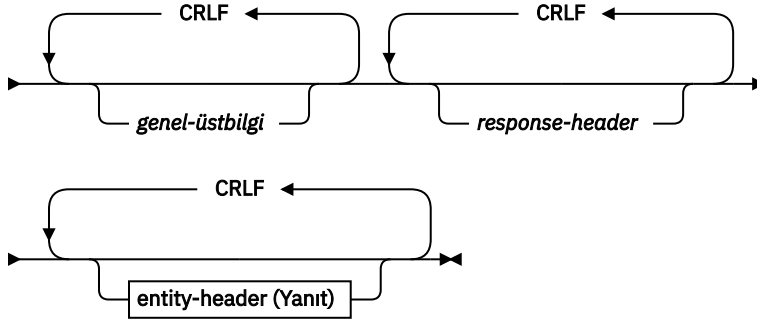


Not:

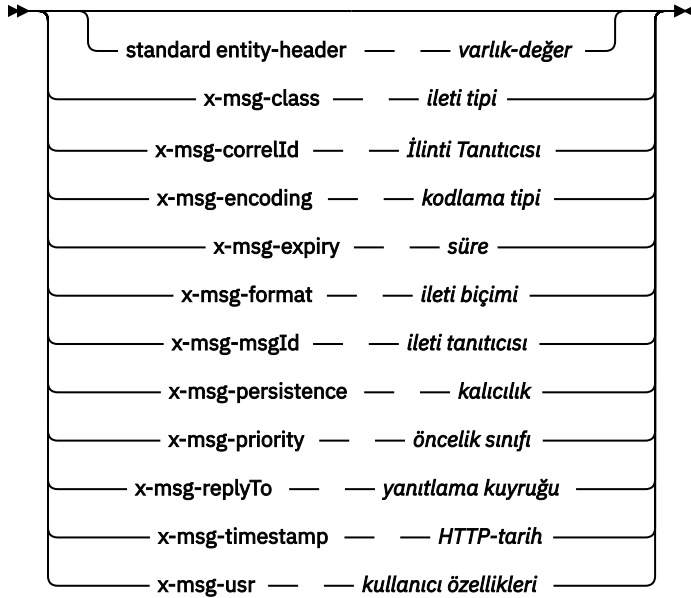
1. If a question mark (?) is used it must be substituted with %3f. Örneğin, orange?topic , orange%3ftopicolarak belirtilmeli.
2. @ qMgrAd yalnızca HTTP **POST**' de geçerlidir

Yanıt

➡ HTTP sürümü — — HTTP Durumu-Kod — — HTTP Neden-Sözcük Tümcesi — CRLF ➡



entity-header (Yanıt)



İstek parametreleri

Yol

Bkz. [“URI Biçimi” sayfa 1771.](#)

HTTP sürümü

HTTP sürümü; örneğin, HTTP/1.1

genel-üstbilgi

Bkz. [HTTP/1.1 - 4.5 Genel Üstbilgi Alanları.](#)

istek-üstbilgi

Bkz. [HTTP/1.1 - 5.3 İstek Üstbilgisi Alanları.](#) Host (Anasistem) alanı HTTP/1.1 isteğinde zorunludur. Genellikle, istemci isteği yaratmak için kullandığınız araç tarafından otomatik olarak araya eklenir.

entity-header (İstek)

Bkz. [HTTP/1.1 - 7.1 Varlık Üstbilgisi Alanları.](#) İstek sözdizimi şemasında listelenen varlık üstbilgilerinden biri. İçerik-Uzunluğu ve İçerik Tipi bir isteğe eklenmelidir ve genellikle istemci isteği yaratmak için kullandığınız araç tarafından otomatik olarak eklenir. The İçerik Tipi must match the type defined in the x-msg-sınıf custom entity-header, if it is specified.

İleti

Kuyruğa konmak için ileti ya da bir konuya göndermek için yayın.

Yanıt deęiřtirgeleri

Yol

Bkz. "URI Biçimi" sayfa 1771.

HTTP sürümü

HTTP sürümü; örneęin, HTTP/1.1

genel-üstbilgi

Bkz. HTTP/1.1 - 4.5 Genel Üstbilgi Alanları.

yanıt-üstbilgi

Bkz. HTTP/1.1 - 6.2 Yanıt Üstbilgisi Alanları.

varlık-üstbilgi (Yanıt)

Bkz. HTTP/1.1 - 7.1 Varlık Üstbilgisi Alanları. Yanıt sözdizimi řemasında listelenen varlık ya da yanıt üstbilgilerinden biri. İçerik-Uzunluęu her zaman bir yanıtta bulunur. İleti gövdesi yoksa sıfır olarak ayarlanır.

Tanım

x-msg-usr üstbilgisi içerilmezse ve ileti sınıfı BYTE ya da TEXTise, kuyruęun içerdii ileti MQRFH2' a sahip deęildir.

Use HTTP entity and request headers in the HTTP **POST** request to set the properties of the message that is put onto the queue. Yanıt iletisinde hangi üstbilgilerin döndürülebileceęini istemek için x-msg-require-üstbilgileri seęeneęini de kullanabilirsiniz.

HTTP **POST** isteęi başarılı olursa, yanıt iletisinin varlıęı boş olur ve İçerięi Uzunluęu sıfır olur. HTTP durum kodu: 200 OK.

HTTP **POST** isteęi başarısız olursa, yanıt bir IBM MQ bridge for HTTP hata iletisi ve bir HTTP durum kodu içerir. The IBM MQ message is not put on the queue or topic.

HTTP POST örneęi

HTTP **POST** , bir iletiyi kuyruęa koyar ya da bir konuya yayın yapar. **HTTPPOST** Java örneęi, bir iletinin kuyruęa iliřkin HTTP **POST** isteęinin bir örneęidir. Java kullanmak yerine, bir tarayıcı formu ya da AJAX araç takımı kullanarak bir HTTP **POST** isteęi yaratabilirsiniz.

Ařağıdaki řekil, myQueueadlı kuyruęa ileti koymak için bir HTTP isteęini göstermektedir. Bu istek, IBM MQ iletisinin ilinti tanıtıcısını ayarlamak için x-msg-correlId HTTP üstbilgisini içerir.

```
POST /msg/queue/myQueue/ HTTP/1.1
Host: www.example.org
Content-Type: text/plain
x-msg-correlID: 1234567890
Content-Length: 50
```

Here is my message body that is posted on the queue.

Şekil 41. Bir kuyruęa iliřkin HTTP **POST** isteęi örneęi

Ařağıdaki řekil, istemciye geri gönderilen yanıtı gösterir. Yanıt içerięi yok.

```
HTTP/1.1 200 OK
Date: Wed, 2 Jan 2007 22:38:34 GMT
Server: Apache-Coyote/1.1 WMQ-HTTP/1.1 JEE-Bridge/1.1
Content-Length: 0
```

Şekil 42. HTTP POST yanıtına ilişkin örnek

HTTP üstbilgileri

IBM MQ bridge for HTTP , özel istek HTTP üstbilgilerini, özel varlık HTTP üstbilgilerini ve standart HTTP üstbilgilerinin bir alt kümesini destekler.

HTTP uygulaması, tüm özel üstbilgilerin x- ile önekleneceğini, bu nedenle HTTP üstbilgilerine ilişkin IBM MQ Bridge 'in başına x-msg- öneki ekleneceğini ifade eder. Örneğin, öncelik üstbilgisini ayarlamak için x-msg-priority seçeneğini kullanın.

Not:

- Çoğu üstbilgi değeri büyük/küçük harfe duyarlıdır. Örneğin, msgId üstbilgisi kullanıldığında, NONE bir anahtar sözcüktür, none ise bir msgID.
- Yanlış yazılan üstbilgiler yoksayılr.

Özel varlık HTTP üstbilgileri

Özel varlık HTTP üstbilgileri, IBM MQ iletilerine ilişkin bilgi içerir. Varlık üstbilgilerini kullanarak, değerleri ileti tanımlayıcısında (MQMD) ya da MQMD' daki sorgu değerlerinde ayarlayabilirsiniz. Ek bir varlık üstbilgisi (x-msg-usr), bir istekle ilişkilendirmek istediğiniz kullanıcı özellik bilgilerini ayarlar ve döndürür.

Farklı HTTP isteği bağlamlarında varlık üstbilgilerini kullanabilirsiniz:

DELETE

x-msg-correlId ya da x-msg-msgId ya da her ikisi de varlık üstbilgilerini bir **DELETE** HTTP isteğiyle kullanabilirsiniz. The effect of the headers is to select a particular message by MsgId and CorrelId in an MQGET, and to delete the message from its queue.

GET

x-msg-correlId ya da x-msg-msgId ya da her ikisi de varlık üstbilgilerini bir **GET** HTTP isteğiyle kullanabilirsiniz. The effect of the headers is to select a particular message by MsgId and CorrelId in an MQGET for browse.

POST

Herhangi bir varlık üstbilgisini, x-msg-timestamp dışında bir **POST** HTTP isteğinde kullanabilirsiniz.

x-msg-require-üstbilgiler

Herhangi bir **GET**, **POST** ya da **DELETE** HTTP isteğinde, virgülle ayrılmış olarak, x-msg-require-headers istek üstbilgisine birden çok varlık üstbilgisi ekleyebilirsiniz. Bu etki, belirtilen varlık üstbilgilerini, ilişkili ileti özelliğinin değerini içeren HTTP yanıt iletisinde geri döndürecek şekilde geri dönmektedir.

The description of each header lists in which contexts the header is processed by IBM MQ bridge for HTTP. Örneğin, **POST**, x-msg-require-üstbilgiler üstbilgisinde, üstbilgi IBM MQ bridge for HTTP tarafından bir HTTP **POST** isteğinde ya da bir HTTP **POST**, **GET** ya da **DELETE** isteğindeki x-msg-require-üstbilgiler istek üstbilgisinde işlenir. Üstbilgi, izin verilmeyen bir bağlamda yer alırsa, üstbilgi yoksayılr. Hata bildirilmedi.

Herhangi bir standart HTTP üstbilgisini Web sunucusu tarafından işlenecek isteklere ya da diğer istek işleyicilerine yerleştirebilirsiniz. Benzer şekilde yanıt, Web sunucusu ya da diğer yanıt işleyiciler tarafından eklenen diğer standart HTTP üstbilgilerini de içerebilir.

Özel istek HTTP üstbilgileri

Üç özel istek HTTP üstbilgileri, x-msg-aralık, x-msg-requires-headersve x-msg-wait, HTTP isteğiyle ilgili ek bilgileri sunucuya iletir. Bunlar istek değiştiricileri olarak işlev görmektedir. x-msg-aralıkile, bir yanıtta döndürülen ileti verilerinin miktarını kısıtlayabilirsiniz. x-msg-require-headersile, isteğin sonucuna ilişkin bilgileri içermek için yanıtı isteyebilirsiniz. x-msg-waitile, istemcinin bir HTTP yanıtı için bekleyeceği süreyi değiştirebilirsiniz.

Standart HTTP üstbilgileri

Anasistem standart HTTP isteği üstbilgisi, bir HTTP/1.1 isteğinde belirtilmelidir.

İçerik Uzunluğu ve İçerik Tipi standart HTTP varlığı üstbilgileri bir istekte belirtilebilir.

Bir isteğe yanıt olarak İçerik-Uzunluğu, İçerik-Konum, İçerik-Aralık, İçerik Tipive Sunucu standart HTTP varlığı üstbilgileri döndürülebilir. İstek iletilisinde x-msg-request-header üstbilgisinde standart HTTP üstbilgilerinden birini ya da birkaçını belirtin.

HTTP üstbilgilerinin alfabetik listesi

class: HTTP x-msg-class entity-header

İleti tipini ayarlayın ya da döndürün.

Tip	Açıklama
HTTP üstbilgi adı	x-msg-sınıfı
HTTP üstbilgi tipi	Varlık-üstbilgi
HTTP istek iletilisinde geçerli	POST, x-msg-require-üstbilgiler
İzin verilen değerler	BYTES MAP STREAM TEXT
Varsayılan değer	BYTES

Tanım

- Bir HTTP **POST** isteğinde, yaratılan iletinin tipini ayarlar.
- Specifying the class header on a **GET** or **DELETE** returns a 400 Bad Request with entity body of MQHTTP40007.
- x-msg-require-headersiçinde belirtilen, HTTP yanıt iletilisinde x-msg-class ögesini bir iletinin tipine ayarlar.
- Bu üstbilgi için geçersiz bir değer belirtilirse, bir MQHTTP40005 iletisi döndürülür.
- x-msg-class üstbilgisi belirtilmediyse ve iletinin içerik tipi application/x-www-form-urlencodedise, verilerin bir JMS eşlem nesnesi olduğu varsayılır.

Content-Length: HTTP entite-header

İletinin gövdesinin uzunluğunu bayt cinsinden ayarlayın ya da döndürün.

Tip	Açıklama
HTTP üstbilgi adı	İçerik Uzunluğu
HTTP üstbilgi tipi	Varlık-üstbilgi

Tip	Açıklama
HTTP istek iletisinde geçerli	x-msg-require-üstbilgiler
İzin verilen ve döndürülen değer	Integer value İleti gövdesinin bayt cinsinden uzunluğu.

Tanım

- İçerik Uzunluğu, HTTP isteğinde isteğe bağlıdır. Bir **GET** ya da **DELETE** için uzunluk sıfır olmalıdır. **POST** için, İçerik-Uzunluk belirtilirse ve ileti satırının uzunluğu ile eşleşmiyorsa, ileti kesilir ya da boş değerler ile belirtilen uzunluğa kadar doldurulur.
- İçerik olmadığında da İçerik Uzunluğu her zaman HTTP yanıtında döndürülür; bu durumda değer sıfır olur.

İçerik-Konum: HTTP entite-header

İstekte başvuru kuyruğu ya da konuyu, HTTP yanıt iletisinde standart İçerik-Konum üstbilgisinde döndürür.

Tip	Açıklama
HTTP üstbilgi adı	İçerik-Konum
HTTP üstbilgi tipi	Varlık-üstbilgi
HTTP istek iletisinde geçerli	x-msg-require-üstbilgiler
Döndürülen değer	Biçimdeki URI, /msg/queue/queueName ya da /msg/topic/topicName

Tanım

- x-msg-require-headers içinde istendiğinde, Content-Location entity-header, HTTP isteğinde başvuru kuyruğu ya da konuyu döndürür.

İçerik-Aralık: HTTP entite-header

Bir HTTP yanıtındaki İçerik Aralığı üstbilgisindeki bir IBM MQ iletisinden seçilen byte aralığını döndürür.

Tip	Açıklama
HTTP üstbilgi adı	İçerik Aralığı
HTTP üstbilgi tipi	Varlık-üstbilgi
HTTP istek iletisinde geçerli	x-msg-require-üstbilgiler

Tip	Açıklama
Döndürülen değer	String Döndürülen alt dizginin m ve üst sınırı olan n alt sınırını ve tüm iletinin $length$ değerini döndürür. Örneğin, $m - n/length$

Tanım

-
- The İçerik Aralığı is only returned in the HTTP response when İçerik Aralığı is specified in a **GET** or **DELETE** request that contains an x-msg-aralık request header.
- Bir **GET** ya da **DELETE** isteğinde x-msg-aralık değeri belirlenirse, yanıtta Content-Range üstbilgisinde belirtilen bayt aralığı döndürülür. Örneğin, 100 bayt içeren bir ileti isteği için x-msg-range: 0-60 kullanılırsa, içerik aralığı üstbilgisi 0-60/100dizgisini içerir.
- Bir x-msg-aralığı isteği, HTTP yanıtındaki x-msg-aralık üstbilgisindeki içerik aralığını da döndürür.

Content-Type: HTTP entite-header

Bir IBM MQ iletisinde JMS iletisinin sınıfını HTTP içerik tipine göre ayarlayın ya da döndürün.

Tip	Açıklama
HTTP üstbilgi adı	İçerik Tipi
HTTP üstbilgi tipi	Varlık-üstbilgi
HTTP istek iletisinde geçerli	POST , x-msg-require-üstbilgiler
İzin verilen ya da döndürülen değer	media-type Desteklenen ortam tipleri için bkz. Çizelge 212 sayfa 1749.

Çizelge 212. Mapping between x-msg-sınıf and HTTP İçerik Tipi	
x-msg-sınıfı	HTTP İçerik Tipi
BYTES	application/octet-stream application/xml
TEXT	text/*
MAP	application/x-www-form-urlencoded application/xml (isteğe bağlı)
STREAM	application/xml (isteğe bağlı)

Tanım

- Bir HTTP **POST** isteğinde, İçerik-Tipi ya da x-msg-sınıfıdeğerlerini belirtin. Her ikisini de belirtirseniz, bunların tutarlı olması ya da HTTP Bad Request kural dışı durumu olması gerekir, Status code 400 değeri döndürülür. If you omit both, the İçerik Tipi and the x-msg-sınıf, a İçerik Tipi of text/* is assumed.

- İçerik Tipi , ileti gövdesi olan bir HTTP **GET** ya da **DELETE** yanıtında her zaman ayarlanır. İçerik-Tipi , Çizelge 213 sayfa 1750 içindeki kurallara göre ayarlanır.

Çizelge 213. İleti tiplerinin x-msg-class ve Content-Type olarak eşlenmesi			
İleti biçimi	JMS İleti tipi	x-msg-sınıfı	İçerik Tipi
MQFMT_STRING dışında herhangi bir şey	Yok	BYTES	application/octet-stream
MQFMT_STRING	Yok	TEXT	text/plain
MQFMT_NONE	jms_bytes	BYTES	application/octet-stream
MQFMT_NONE	jms_text	TEXT	text/plain
MQFMT_NONE	jms_map	MAP	application/xml
MQFMT_NONE	jms_stream	STREAM	application/xml

correlId: HTTP x-msg-correlId entity-üstbilgisi

İlanti tanıtıcısını ayarlayın ya da döndürün.

Tip	Açıklama
HTTP üstbilgi adı	x-msg-correlId
HTTP üstbilgi tipi	Varlık-üstbilgi
HTTP istek iletisinde geçerli	DELETE, GET, POST , x-msg-require-üstbilgiler
İzin verilen değerler	String value Örneğin: <pre>x-msg-correlId: mycorrelationid</pre> Tırnak işareti içine alınmış dizgillere izin verilir; örneğin: <pre>x-msg-correlId: "my id"</pre> Hex value Başında 0x : öneki olan bir onaltılı değer; örneğin: <pre>x-msg-correlId: 0x:43c1d23a</pre> The hex value following 0x: is limited to 48 characters representing 24 bytes. Ek veriler yok sayılır.
Varsayılan değer	Burada geçerli değil

Tanım

- Bir HTTP **POST** isteği üzerine, oluşturulan iletinin ilanti tanıtıcısını ayarlar.
- Bir HTTP **GET** ya da **DELETE** isteği üzerine, kuyruktan ya da konudan iletiyi seçer. Belirtilen ilanti tanıtıcısına sahip bir ileti yoksa, bir HTTP 504 Gateway Timeout yanıtı döndürülür. x-msg-correlId , hem seçicilerle eşleşen bir kuyruktan ya da konudan bir ileti seçmek için x-msg-msgID ile kullanılabilir.

- `x-msg-require-headers` içinde belirtilen, HTTP yanıt iletilisinde `x-msg-coreId` ögesini, bir iletinin ilinti tanıtıcısına ayarlar.
- `0x`: önekinden sonra yatay beyaz alana izin verilir.

Not:

- HTTP **GET** ya da **DELETE** isteğinde bir değer olmadan `x-msg-correlId` belirtilirse; örneğin, "`x-msg-correlId:` ", kuyruk ya da konu ilinti tanıtıcısından bağımsız olarak bir sonraki iletiyi döndürür.
- 24 ya da daha az karakterden oluşan bir seçici belirtirseniz ya da `0x`: en çok 48 karakter ya da daha az karakter izlediye, IBM MQ bridge for HTTP gelişmiş performans için eniyilenmiş bir seçici kullanır.
- Kuyruktan ileti seçerken, JMSCorrelationID değerini içeren bir JMS ileti seçicisi kullanılır. Bu seçici, Seçim davranışı içinde açıklandığı gibi hareket eder.

kodlama: HTTP `x-msg-encoding` entity-header

İleti kodlamasını ayarlayın ya da döndürün.

Tip	Açıklama
HTTP üstbilgi adı	<code>x-msg-encoding</code>
HTTP üstbilgi tipi	Varlık-üstbilgi
HTTP istek iletilisinde geçerli	POST , <code>x-msg-require-üstbilgiler</code>
İzin verilen değerler	Aşağıdaki değerlerin virgülle ayrılmış listesi: DECIMAL_NORMAL DECIMAL_REVERSED FLOAT_IEEE_NORMAL FLOAT_IEEE_REVERSED FLOAT_S390 INTEGER_NORMAL INTEGER_REVERSED Örneğin, <pre>x-msg-encoding: INTEGER_NORMAL,DECIMAL_NORMAL,FLOAT_IEEE_NORMAL</pre> Not: Değer büyük ve küçük harfe duyarlıdır
Varsayılan değer	<code>DECIMAL_NORMAL, FLOAT_IEEE_NORMAL, INTEGER_NORMAL</code>

Tanım

- Bir HTTP **POST** isteği üzerine, oluşturulan iletinin kodlamasını belirtir.
- Bir HTTP **GET** ya da **DELETE** isteğinde, `x-msg-encoding` üstbilgisi yoksayılır.
- `x-msg-require-headers` içinde belirtilen, HTTP yanıt iletilisinde `x-msg-encoding` ögesini, bir iletinin kodlama özelliğine ayarlar.

süre bitimi: HTTP `x-msg-süre bitimi` entite-üstbilgi

İleti süre bitimi süresini ayarlayın ya da döndürün.

Tip	Açıklama
HTTP üstbilgi adı	<code>x-msg-süre Bitimi</code>

Tip	Açıklama
HTTP üstbilgi tipi	Varlık-üstbilgi
HTTP istek iletisinde geçerli	POST , x-msg-require-üstbilgiler
İzin verilen değerler	UNLIMITED Örneğin; <pre>x-msg-expiry: UNLIMITED</pre> Integer value Süre bitiminden önce salise. Örneğin; <pre>x-msg-expiry: 10000</pre>
Varsayılan değer	UNLIMITED

Tanım

- HTTP **POST** isteği üzerine ayarlandığında, istek iletisinin süresi belirtilen süre içinde sona erer.
- Bir HTTP **GET** ya da **DELETE** isteğinde, x-msg-süre bitimi üstbilgisi dikkate alınmaz.
- x-msg-require-headersiçinde belirtilen, HTTP yanıt iletisinde x-msg-expiry ögesini bir iletinin süre bitimi süreye ayarlar.
- UNLIMITED , iletinin hiçbir zaman süre bitimine geçmeyeceğini belirtir.
- Bir iletinin süre bitimi, bir sonuç ağı gecikme süresi dikkate alınmadıkça, ileti kuyruğa geldiği zamandan başlar.
- Değer üst sınırı IBM MQ ile 214748364700 milisaniyeye sınırlanır. Bundan büyük bir değer belirtilirse, olası süre bitimi üst sınırı varsayılır.

format: HTTP x-msg-format entity-header

IBM MQ ileti biçimini ayarlayın ya da döndürün.

Tip	Açıklama
HTTP üstbilgi adı	x-msg-biçimi
HTTP üstbilgi tipi	Varlık-üstbilgi
HTTP istek iletisinde geçerli	POST , x-msg-require-üstbilgiler
İzin verilen değerler	NONE Örneğin, <pre>x-msg-format: NONE</pre> String value En çok sekiz karakterden oluşan kullanıcı tanımlı değerler. Örneğin, <pre>x-msg-format: myformat</pre>
Varsayılan değer	None

Tanım

- HTTP **POST** isteği üzerine ayarlandığında, istek iletisi biçimini ayarlayın.
- Bir HTTP **GET** ya da **DELETE** isteğinde, x-msg-format üstbilgisi yoksayılır.
- x-msg-require-headersiçinde belirtilen, HTTP yanıt iletisinde x-msg-format ögesini iletinin biçimine ayarlar.
- NONE , büyük/küçük harfe duyarlıdır ve ileti biçiminin boş olduğunu gösterir.
- HTTP isteğinin ortam tipiyle çelişse bile, x-msg-format değeri kullanılır. Bkz. Çizelge 214 sayfa 1753.

Çizelge 214. İçerik tipi ve x-msg-class eşleme içerik tipi ileti biçimine eşleniyor		
x-msg-sınıfı	İçerik Tipi	Kuyruk/konu üzerinde ileti biçimi
BYTES	• application/octet-stream • application/xml	IBM MQ iletisi: MQFMT , MQC.MQFMT_NONEolarak ayarlanır
TEXT	• text/*	IBM MQ iletisi: MQFMT , MQC.MQFMT_STRINGolarak ayarlanır
MAP	• application/x-www-form-urlencoded • application/xml (isteğe bağlı)	JMSMap
STREAM	• application/xml (isteğe bağlı)	JMSStream

msgId: HTTP x-msg-msgId entity-header

İleti tanıtıcısını ayarlayın ya da döndürün.

Tip	Açıklama
HTTP üstbilgi adı	x-msg-msgId
HTTP üstbilgi tipi	Varlık-üstbilgi
HTTP istek iletisinde geçerli	DELETE, GET, POST , x-msg-require-üstbilgiler
İzin verilen değerler	String value Örneğin, <pre>x-msg-msgId: mymsgid</pre> Örneğin tırnak işareti içine alınmış dizgiler, x-msg-msgId: "my id" Hex value Başında 0x : öneki olan bir onaltılı değer; Örneğin, <pre>x-msg-msgId: 0x:43c1d23a</pre>
Varsayılan değer	Burada geçerli değil

Tanım

- Bir HTTP **POST** isteği üzerine, oluşturulan iletinin ileti tanıtıcısını ayarlar.

- Bir HTTP **GET** ya da **DELETE** isteği üzerine, kuyruktan ya da konudan iletiyi seçer. Belirtilen ileti tanıtıcısı ile herhangi bir ileti yoksa, bir HTTP 504 Gateway Timeout yanıtı döndürülür. x-msg-msgId , hem seçicilerle eşleşen bir kuyruktan ya da konudan bir ileti seçmek için x-msg-correlID ile kullanılabilir.
- x-msg-require-headersiçinde belirtilen, bir iletinin ileti tanıtıcısının HTTP yanıtında x-msg-msgId değerini döndürür.
- 0x: önekinden sonra yatay beyaz alana izin verilir.

Not: HTTP **GET** ya da **DELETE** isteğinde bir değer olmadan x-msg-msgId belirtilirse; örneğin, "x-msg-msgId: ", kuyruk ya da konu üzerindeki bir sonraki iletiyi, ileti tanıtıcısı ne olursa olsun, döndürür.

Kuyruktan ileti seçerken, JMSMessageID değerini içeren bir JMS ileti seçicisi kullanılır. Bu seçici, [Seçim davranışii](#) içinde açıklandığı gibi hareket eder.

persistence: HTTP x-msg-persistence entity-header

İleti kalıcısını ayarlayın ya da döndürün.

Tip	Açıklama
HTTP üstbilgi adı	x-msg-persistence
HTTP üstbilgi tipi	Varlık-üstbilgi
HTTP istek iletisinde geçerli	POST , x-msg-require-üstbilgiler
İzin verilen değerler	NON_PERSISTENT İleti, sistem arızalarını ya da kuyruk yöneticisini yeniden başlatma işlemini sürdürmez. Örneğin, <pre>x-msg-persistence: NON_PERSISTENT</pre> PERSISTENT İleti, kuyruk yöneticisinin sistem hatalarından ve yeniden başlatılabilmesinin ardından devam eder. Örneğin, <pre>x-msg-persistence: PERSISTENT</pre> AS_DESTINATION Yalnızca POST için geçerlidir. İleti sağlayıcının belirlediği şekilde, hedefin varsayılan kalıcılarını kullanın. Not: Büyük/küçük harfe duyarlı
Varsayılan değer	NON_PERSISTENT

Tanım

- Bir HTTP **POST** isteği üzerine ayarlandığında, istek iletisi kalıcısını ayarlayın.
- Bir HTTP **GET** ya da **DELETE** isteğinde, x-msg-persistence üstbilgisi yoksayılır.
- x-msg-require-headersiçinde belirtilen, HTTP yanıt iletisinde x-msg-persistence ögesini bir iletinin kalıcılığına ayarlar.

priority: HTTP x-msg-priority entity-header

İleti önceliğini ayarlayın ya da döndürün.

Tip	Açıklama
HTTP üstbilgi adı	x-msg-priority
HTTP üstbilgi tipi	Varlık-üstbilgi
HTTP istek iletisinde geçerli	POST , x-msg-require-üstbilgiler
İzin verilen değerler	LOW Örneğin, <pre>x-msg-priority: LOW</pre> MEDIUM Bu öncelik, 4 olan bir IBM MQ öncelik düzeyine eşittir. Örneğin, <pre>x-msg-priority: MEDIUM</pre> HIGH Örneğin, <pre>x-msg-priority: HIGH</pre> Integer value 0 ile 9 arasındaki aralıktaki bir tamsayının dizgi gösterimi; örneğin, <pre>x-msg-priority: 3</pre> AS_DESTINATION Yalnızca POST için geçerlidir. İleti sağlayıcının belirlediği şekilde, hedefin varsayılan önceliğini kullanın. Not: Büyük/küçük harfe duyarlı
Varsayılan değer	MEDIUM

Tanım

- HTTP **POST** isteği üzerine ayarlandığında, istek iletisi önceliğini ayarlayın.
- Bir HTTP **GET** ya da **DELETE** isteğinde, x-msg-priority üstbilgisi yoksayılır.
- x-msg-require-headersiçinde belirtilen, HTTP yanıt iletisinde x-msg-priority (x-msg-öncelik) iletisini bir iletinin önceliğine ayarlar.

priority: HTTP x-msg-priority entity-header

İleti önceliğini ayarlayın ya da döndürün.

Tip	Açıklama
HTTP üstbilgi adı	x-msg-priority
HTTP üstbilgi tipi	Varlık-üstbilgi
HTTP istek iletisinde geçerli	POST , x-msg-require-üstbilgiler

Tip	Açıklama
İzin verilen değerler	LOW Örneğin, <pre>x-msg-priority: LOW</pre> MEDIUM Bu öncelik, 4 olan bir IBM MQ öncelik düzeyine eşittir. Örneğin, <pre>x-msg-priority: MEDIUM</pre> HIGH Örneğin, <pre>x-msg-priority: HIGH</pre> Integer value 0 ile 9 arasındaki aralıktaki bir tamsayının dizgi gösterimi; örneğin, <pre>x-msg-priority: 3</pre> AS_DESTINATION Yalnızca POST için geçerlidir. İleti sağlayıcının belirlediği şekilde, hedefin varsayılan önceliğini kullanın. Not: Büyük/küçük harfe duyarlı
Varsayılan değer	MEDIUM

Tanım

- HTTP **POST** isteği üzerine ayarlandığında, istek iletisi önceliğini ayarlayın.
- Bir HTTP **GET** ya da **DELETE** isteğinde, x-msg-priority üstbilgisi yoksayılır.
- x-msg-require-headersinde belirtilen, HTTP yanıt iletisinde x-msg-priority (x-msg-öncelik) iletisini bir iletinin önceliğine ayarlar.

replyTo: HTTP x-msg-replyTo entity-header

İleti yanıtı kuyruğu ve kuyruk yöneticisi adını ayarlayın ya da döndürün.

Tip	Açıklama
HTTP üstbilgi adı	x-msg-replyTo
HTTP üstbilgi tipi	Varlık-üstbilgi
HTTP istek iletisinde geçerli	POST , x-msg-require-üstbilgiler
İzin verilen değerler	URI Bir noktadan noktaya iletişim URI 'si; örneğin, <pre>x-msg-replyTo: /msg/queue/myReplyQueue x-msg-replyTo: /msg/queue/myReplyQueue@myReplyQueueManager</pre> Not: Büyük/küçük harfe duyarlı
Varsayılan değer	MEDIUM

Tanım

- Bir HTTP **POST** isteği üzerine ayarlandığında, istek iletisi replyTo hedefini ayarlayın.
- Bir HTTP **GET** ya da **DELETE** isteğinde, x-msg-replyTo üstbilgisi yoksayılır.
- Specified in x-msg-require-üstbilgiler, sets x-msg-replyTo in the HTTP response message to the reply-to queue and queue manager name of a message.

Not: HTTP yanıtındaki URI, IBM MQ bridge for HTTP ' in bağlı olduğu kuyruk yöneticisinin adını içerebilir.

Sunucu: HTTP yanıt-üstbilgisi

İstemcinin bağlandığı sunucu ve protokolle ilgili bilgileri döndürür.

Tip	Açıklama
HTTP üstbilgi adı	Sunucu
HTTP üstbilgi tipi	Yanıt-üstbilgi
HTTP istek iletisinde geçerli	x-msg-require-üstbilgiler
Döndürülen değer	WMQ-HTTP/1.1 JEE-Bridge/1.1 ya da Server: Product-token WMQ-HTTP/1.1 JEE-Bridge/1.1

Tanım

- HTTP için IBM MQ Bridge bir uygulama sunucusuna konuşlandırıldıysa, IBM MQ bridge for HTTP ayrıntıları sunucu yanıt üstbilgisinin sonuna eklenir. Örneğin, Apache-Coyote adı verilen WebSphere Application Server Community Edition' a konuşlandırılan IBM MQ bridge for HTTP , yanıtı verir:

Server: Apache-Coyote/1.1 WMQ-HTTP/1.1 JEE-Bridge/1.1

require-headers: HTTP x-msg-require-headers request-header

HTTP yanıt iletisinde hangi üstbilgilerin döndürüleceğini belirleyin.

Tip	Açıklama
HTTP üstbilgi adı	x-msg-require-üstbilgiler
HTTP üstbilgi tipi	İstek-üstbilgi
HTTP istek iletisinde geçerli	POST, GET, DELETE
İzin verilen değerler	Varlık üstbilgisi adlarının virgülle ayrılmış listesi: ALL ALL-USR class content-location correlId encoding expiry

Tip	Açıklama
	format msgId NO_require-headers persistence priority replyTo server timestamp usr- <i>property name</i> Örneğin, <pre>x-msg-require-headers: msgId</pre> ya da <pre>x-msg-require-headers: expiry,correlId,timestamp</pre> Belirli bir özelliği istemek için: <pre>x-msg-require-headers: usr-myCustomProperty</pre> Tüm özellikleri istemek için: <pre>x-msg-require-headers: ALL-USR, ALL</pre>
Varsayılan değer	NO_require-headers

Tanım

- ALL, NO_require-headers ve ALL-USR değişmezleri ve *property-name* değişkeni dışında, x-msg-require-headers değeri büyük/küçük harfe duyarlı değildir.

timestamp: HTTP x-msg-timestamp entity-header

İleti zaman damgasını döndürür.

Tip	Açıklama
HTTP üstbilgi adı	x-msg-zaman damgası
HTTP üstbilgi tipi	Varlık-üstbilgi
HTTP istek iletilisinde geçerli	x-msg-require-üstbilgiler
Döndürülen değer	HTTP-date Biçimdeki bir tarih; gün, tarih ay yıl saat-bölge; örneğin, <pre>Sun, 06 Nov 1994 08:49:37 GMT</pre> RFC 822 tarafından tanımlanır ve RFC 1123 içinde güncellenir.
Varsayılan değer	Burada geçerli değil

Tanım

- Bir HTTP **POST**, **GET** ya da **DELETE** isteğinde, x-msg-timestamp üstbilgisi yoksayılır.
- x-msg-require-headersiçinde belirtilen, HTTP yanıt iletisinde x-msg-timestamp iletisini bir iletinin zaman damgasına ayarlar.

usr: HTTP x-msg-usr entity-header

Kullanıcı özelliklerini ayarlayın ya da döndürün.

Tip	Açıklama
HTTP üstbilgi adı	x-msg-usr
HTTP üstbilgi tipi	Varlık-üstbilgi
HTTP istek iletisinde geçerli	POST , x-msg-require-üstbilgiler
İzin verilen değerler	Bkz. "Sözdizimi" sayfa 1759 ; Örneğin, <pre>x-msg-usr: myProp1;5;i1, x-msg-usr: myProp2;"My String";string</pre>
Varsayılan değer	Burada geçerli değil

Tanım

- Bir HTTP **POST** isteği üzerine ayarlandığında, istek iletisi kullanıcı özelliklerini ayarlayın.
- Bir HTTP **GET** ya da **DELETE** isteğinde, x-msg-usr üstbilgisi yoksayılır.
- x-msg-require-headersiçinde belirtilen, HTTP yanıt iletisinde x-msg-usr ögesini bir iletinin kullanıcı özelliklerine ayarlar.
- İletide birden çok özellik ayarlanabilir. Tek bir x-msg-usr üstbilgisinde birden çok virgülle ayrılmış özellik belirtin ya da x-msg-usr üstbilgisinin iki ya da daha fazla örneğini kullanın.
- Bir **GET** ya da **DELETE** isteğine yanıt olarak belirli bir özellik döndürülebileceğini isteyebilirsiniz. Specify the name of the property in the x-msg-require-üstbilgiler header of the request, using the prefix usr-. Örneğin,

```
x-msg-require-headers: usr-myProp1
```

- Bir yanıtta tüm kullanıcı özelliklerinin döndürülmesini istemek için ALL-USR sabitini kullanın. Örneğin,

```
x-msg-require-headers: ALL-USR
```

Sözdizimi



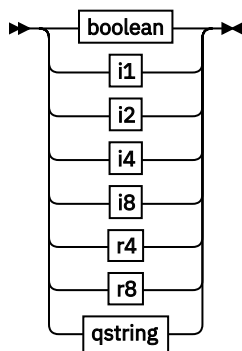
usr-property-value



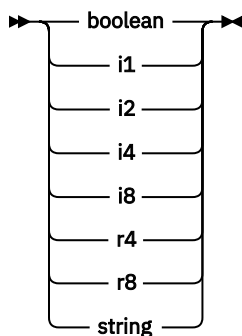
property-name

➤ *dizgi* ➤

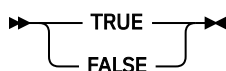
usr-value



usr-type



boolean



i1

➤ -128 — ≤n≤ — +127 ➤

i2

➤ -32768 — ≤n≤ — +32767 ➤

i4

➤ -2147483648 — ≤n≤ — +2147483647 ➤

i8

➤ -9223372036854775808 — ≤n≤ — +9223372036854775807 ➤

r4

➤ -1.4E-45 — ≤n≤ — +3.4028235E38 ➤

r8

➤ -4.9E-324 — ≤n≤ — +1.7976931348623157E308 ➤

qstring

➤ " — dize — " ➤

wait: HTTP x-msg-wait isteği-üstbilgi

Bir HTTP 504 Gateway Timeout yanıt iletisi döndürülmeden önce bir iletinin gelmesi için beklenecek süreyi ayarlayın.

Tip	Açıklama
HTTP üstbilgi adı	x-msg-wait
HTTP üstbilgi tipi	İstek-üstbilgi
HTTP istek iletilisinde geçerli	GET, DELETE
İzin verilen değer	NO_WAIT Örneğin, <pre>x-msg-wait: NO_WAIT</pre> Integer value IBM MQ bridge for HTTP 'ın bir iletinin gelmesi için bekleyeceği süre (milisaniye); örneğin, <pre>x-msg-wait: 8</pre>
Varsayılan değer	NO_WAIT

Tanım

- Bir HTTP **POST** isteğinde, x-msg-wait üstbilgisi yoksayılır.
- On an HTTP **GET** or **DELETE** request, x-msg-wait specifies time to wait for a message to arrive before returning an HTTP 504 Gateway Timeout response.
- NO_WAIT , büyük/küçük harfe duyarlıdır.
- Varsayılan bekleme süresi üst sınırı 35000 'dir. Sunucu uygulamacığının maximum_wait_time parametresini ayarlayarak, varsayılan değeri değiştirebilirsiniz. Daha fazla bilgi için [IBM MQ bridge for HTTP](#) ürününün kurulması, yapılandırılması ve doğrulanması bölümüne bakın.
- maximum_wait_time değeri büyük bir değer ayarlarsa, bunun yerine maximum_wait_time kullanılır.

HTTP dönüş kodları

IBM MQ bridge for HTTP' den gelen dönüş kodlarının listesi

IBM MQ bridge for HTTP , dört hata tipini döndürür:

Sunucu uygulamacığı hataları

MQHTTP0001 ve MQHTTP0002 sunucu uygulamacığı hatalarıdır. Bunlar günlüğe kaydedilir, ancak HTTP istemcisine döndürülmez.

Başarılı işlemler

200-299 aralığındaki bir HTTP durum kodu başarılı bir işlemi gösterir.

İstemci hataları

400-499 aralığındaki bir HTTP durum kodu, bir istemci hatasının olduğunu gösterir. IBM MQ Bridge for HTTP return codes in the range MQHTTP40001 - MQHTTP49999 , istemci hatalarına karşılık gelir.

Sunucu hataları

500-599 aralığındaki bir HTTP durum kodu, bir istemci hatasının olduğunu gösterir. IBM MQ Bridge for HTTP dönüş kodları- MQHTTP50001 - MQHTTP59999 , sunucu hatalarına karşılık gelir.

Bir sunucu hatası ortaya çıkarsa, tam yığın izlemesi, uygulama sunucusu hata günlüğünün çıkışına çıkar. Yığın izlemesi, HTTP yanıtında HTTP istemcisine de döndürülür. İstemci uygulamasındaki yığın izlemesini işleyin ya da sorunu çözmek için uygulama sunucusu denetimcisine başvurun.

Yığın izinde kaynak bağdaştırıcısı hataları varsa, kaynak bağdaştırıcınızla ilgili belgelere bakın.

Dönüş kodlarının alfabetik listesi

HTTP 200: Tamam

Bu sınıf durumu kodu, isteğin başarıyla alındığını, anlaşıldığını ve kabul edildiğini gösterir.

HTTP durum kodu

200 OK

HTTP 204: İçerik yok

Başarılı bir HTTP **GET** ya da **DELETE** ve **x-msg-range: 0** isteği gönderildiğini takiben gönderildi.

HTTP durum kodu

204 No Content

MQHTTP0001: Sunucu uygulamacığı bağlamında bağlantı üreticisi belirtilmedi.

Sunucu uygulamacığı hatası

Açıklama

Sunucu uygulamacığı hatası

HTTP durum kodu

Yok

Programcı yanıtı

Bu hataların günlüğe kaydedildiği yer, uygulama sunucunuza özeldir. Uygulama sunucunuz için belgelere bakın.

MQHTTP0002: *jndiNameTried*JNDI adını kullanarak *queueOrTopic* için bağlantı yöneticisi alınamadı

Sunucu uygulamacığı hatası

Açıklama

Sunucu uygulamacığı hatası

HTTP durum kodu

Yok

Programcı yanıtı

Bu hataların günlüğe kaydedildiği yer, uygulama sunucunuza özeldir. Uygulama sunucunuz için belgelere bakın.

MQHTTP40001: Ayrılmış

Ayrıldı

HTTP durum kodu

400 Bad Request

MQHTTP40002: URI, HTTP için IBM MQ iletimi için geçerli değil

HTTP isteğinde belirtilen URI geçerli değil.

Açıklama

HTTP isteğinde belirtilen URI geçerli değil.

HTTP durum kodu

400 Bad Request

Programcı yanıtı

Belirtilen URI ' nin biçiminin ve sözdiziminin doğru olduğunu onaylayın.

MQHTTP40003: URI geçerli değil. @qmgr yalnızca POST üzerinde geçerlidir

POST isteği olmayan bir HTTP isteği için bir URI ' de @qmgr URI seçeneği belirtildi.

Açıklama

POST isteği olmayan bir HTTP isteği için bir URI ' de @qmgr URI seçeneği belirtildi.

HTTP durum kodu

400 Bad Request

Programcı yanıtı

POST komutunu kullanarak bir ileti yerleştirmeye çalışıyorsanız, HTTP isteğini **POST** isteğine çevirin. If you are attempting to get a message by using the **DELETE** or **GET** verbs, remove @qmgr from the URI.

MQHTTP40004: Belirtilen İçerik-tipi geçersiz.

Bir **POST** isteğinde belirtilen İçerik-Tipi üstbilgi alanı, x-msg-class üstbilgi değeriyle uyumlu değil.

Açıklama

Bir **POST** isteğinde belirtilen İçerik-Tipi üstbilgi alanı, x-msg-class üstbilgi değeriyle uyumlu değil.

HTTP durum kodu

400 Bad Request

Programcı yanıtı

İçerik-Tür üstbilgi alanını desteklenen bir alana çevirin. İçerik Tipi üstbilgisi, belirtilen x-msg-sınıfı üstbilgi alanıyla uyumlu olmalıdır.

MQHTTP40005: İleti üstbilgisi değeri hatalı

Desteklenen bir üstbilgi alanı, belirtilen istek için geçerli olmayan bir değerle belirtildi.

Açıklama

Desteklenen bir üstbilgi alanı, belirtilen istek için geçerli olmayan bir değerle belirtildi.

HTTP durum kodu

400 Bad Request

Programcı yanıtı

Verili üstbilgi alanı için belirtilen değeri geçerli bir değer olarak değiştirin. Bazı üstbilgi alanlarının büyük/küçük harf duyarlı değerleri olduğu için, belirtilen değerın büyük/küçük harf durumu olup olmadığını denetleyin.

MQHTTP40006: *Header_name* geçerli bir istek üstbilgisi değil

Bir HTTP isteği iletisinde yalnızca bir HTTP yanıt iletisinde geçerli olan bir üstbilgi belirtildi.

Açıklama

Bir HTTP isteği iletisinde yalnızca bir HTTP yanıt iletisinde geçerli olan bir üstbilgi belirtildi.

HTTP durum kodu

400 Bad Request

Programcı yanıtı

Yalnızca HTTP yanıtında geçerli olan HTTP isteğindeki üstbilgileri kaldırın; örneğin, x-msg-timestampgibi.

MQHTTP40007: *Header_name* yalnızca şu yerde geçerli ...

Bir HTTP isteğinde üstbilgi belirtildi, ancak verilen istek komutu için üstbilgi alanı geçerli değil.

Açıklama

Bir HTTP isteğinde üstbilgi belirtildi, ancak verilen istek komutu için üstbilgi alanı geçerli değil.

HTTP durum kodu

400 Bad Request

Programcı yanıtı

Belirtilen istek komutu için geçerli olmayan tüm üstbilgileri HTTP isteğinden kaldırın. Örneğin, x-msg-encoding , HTTP **POST** istekleri için geçerlidir, ancak HTTP **GET** ya da HTTP **DELETE** istekleri için geçerli değildir.

MQHTTP40008: *Header_name* uzunluk üst sınırı ...

Belirtilen üstbilgi alanına ilişkin uzunluk üst sınırı aşıldı.

Açıklama

Belirtilen üstbilgi alanına ilişkin uzunluk üst sınırı aşıldı.

HTTP durum kodu

400 Bad Request

Programcı yanıtı

Üstbilgi alanının değerini, üstbilgi alanı için izin verilen aralığın içinde olan bir değerle değiştirin.

MQHTTP40009: Üstbilgi alanı *header_field* için geçerli değil ...

HTTP isteğinde belirtilen bir üstbilgi alanı, IBM MQ bridge for HTTP ' nin bağlı olduğu ileti alışverişi sağlayıcısı tarafından desteklenmiyor.

Açıklama

HTTP isteğinde belirtilen bir üstbilgi alanı, IBM MQ bridge for HTTP 'nin bağlı olduğu ileti alışverişi sağlayıcısı tarafından desteklenmiyor. Hata, IBM MQ bridge for HTTP' in tüm özelliklerini destekleyemeyen bir ileti alışverişi sağlayıcısı kullanılıyorsa ortaya çıkar.

HTTP durum kodu

400 Bad Request

Programcı yanıtı

Desteklenmeyen üstbilgiyi HTTP isteğinden kaldırın.

MQHTTP40010: Content-Type *content_type* ile ileti ayrıştırılamadı

HTTP isteğinin içeriği, isteğin İçerik-tipi ile uyumlu değil.

Açıklama

HTTP isteğinin içeriği, isteğin İçerik-tipi ile uyumlu değil. A common cause is badly formed application/x-www-form-urlencoded or application/xml data.

HTTP durum kodu

400 Bad Request

Programcı yanıtı

HTTP isteğinin içeriğini, isteğin İçerik-Tipi için doğru biçimde olacak şekilde düzeltin.

MQHTTP40301: Erişiminiz yasaklandı ...

IBM MQ bridge for HTTP , belirtilen hedef için kendisini doğrulayamadı.

Açıklama

IBM MQ bridge for HTTP , belirtilen hedef için kendisini doğrulayamadı.

HTTP durum kodu

403 Forbidden

Programcı yanıtı

Change the authentication properties of the destination so that the IBM MQ Bridge for HTTP is authorized to connect to it. Alternatively, specify a destination to which the IBM MQ bridge for HTTP is authorized to connect.

MQHTTP40302: Bu durumda yasaklandınız ...

IBM MQ bridge for HTTP kuyruk yöneticisine bağlanamadı.

Açıklama

IBM MQ bridge for HTTP kuyruk yöneticisine bağlanamadı. IBM MQ bridge for HTTP güvenlik yapılandırması yanlış.

HTTP durum kodu

403 Forbidden

Programcı yanıtı

Kuyruk yöneticisinin kimlik doğrulama yapılandırmasını değiştirerek, HTTP için IBM MQ Köprüsünün bağlantı kurma yetkisi olacak şekilde değiştirin. Diğer bir seçenek olarak, IBM MQ bridge for HTTP konfigürasyonunu, bağlanma yetkisi olan bir kuyruk yöneticisine bağlanmasını da sağlar.

MQHTTP40401: Hedef *destination_name* bulunamadı.

HTTP isteği URI 'sında belirtilen hedef, IBM MQ bridge for HTTP tarafından bulunamıyor.

Açıklama

HTTP isteği URI 'sında belirtilen hedef, IBM MQ bridge for HTTP tarafından bulunamıyor.

HTTP durum kodu

404 Not found

Programcı yanıtı

HTTP isteği URI 'sında belirtilen hedefin var olup olmadığını denetleyin ya da alternatif bir hedef belirtin.

MQHTTP40501: Method *method_name* not allowed

HTTP isteğinde belirtilen yöntem, IBM MQ bridge for HTTP tarafından desteklenmiyor.

Açıklama

HTTP isteğinde belirtilen yöntem, IBM MQ bridge for HTTP tarafından desteklenmiyor.

HTTP durum kodu

405 Method not allowed

Programcı yanıtı

HTTP isteğinde belirtilen yöntemi, IBM MQ bridge for HTTP tarafından desteklenen bir yönteme çevirin.

MQHTTP41301: İletilen ileti hedef için çok büyük.

HTTP POST isteği URI 'sında belirtilen hedef, HTTP isteğinde belirtilen ileti kadar uzun iletileri kabul edemiyor.

Açıklama

HTTP POST isteği URI 'sında belirtilen hedef, HTTP isteğinde belirtilen ileti kadar uzun iletileri kabul edemiyor.

HTTP durum kodu

413 Request entity too large

Programcı yanıtı

HTTP isteğinde belirtilen iletinin boyutunu küçültür. Diğer bir seçenek olarak, aranan uzunluktaki iletileri destekleyebilecek bir hedef belirtin.

MQHTTP41501: Ortam tipi karakter kümesi desteklenmiyor

İçerik-Türü üstbilgi alanında belirtilen karakter kümesi IBM MQ bridge for HTTP tarafından desteklenmiyor.

Açıklama

İçerik-Türü üstbilgi alanında belirtilen karakter kümesi IBM MQ bridge for HTTP tarafından desteklenmiyor.

HTTP durum kodu

415 Unsupported media type

Programcı yanıtı

Content-Type üstbilgi alanının karakter kümesini IBM MQ bridge for HTTP tarafından desteklenen bir karakter olarak değiştirin.

MQHTTP41502: Ortam tipi *media-type* desteklenmiyor ...

HTTP isteğinde belirtilen ortam tipi, belirtilen HTTP yüklemi için IBM MQ bridge for HTTP tarafından desteklenmiyor.

Açıklama

HTTP isteğinde belirtilen ortam tipi, belirtilen HTTP yüklemi için IBM MQ bridge for HTTP tarafından desteklenmiyor.

HTTP durum kodu

415 Unsupported media type

Programcı yanıtı

HTTP isteğinde belirtilen ortam tipini, belirtilen HTTP komutu için HTTP için IBM MQ Bridge tarafından desteklenen bir istek üzerine değiştirin.

MQHTTP41503: Ortam tipi *media-type* desteklenmiyor ...

HTTP isteğinde belirtilen ortam tipi, belirtilen x-msg-sınıfı üstbilgi alanı için IBM MQ bridge for HTTP tarafından desteklenmiyor.

Açıklama

HTTP isteğinde belirtilen ortam tipi, belirtilen x-msg-sınıfı üstbilgi alanı için IBM MQ bridge for HTTP tarafından desteklenmiyor.

HTTP durum kodu

415 Unsupported media type

Programcı yanıtı

HTTP isteğinde belirtilen ortam tipini, belirtilen x-msg-sınıfı üstbilgi alanı için HTTP için IBM MQ Bridge tarafından desteklenen bir istek üzerine değiştirin.

MQHTTP41701: HTTP üstbilgisi (Expect) desteklenmiyor

IBM MQ bridge for HTTP , Expect üstbilgi alanını desteklemez.

Açıklama

The Expect header has been specified in an HTTP request. IBM MQ bridge for HTTP , Expect üstbilgi alanını desteklemez.

HTTP durum kodu

417 Expectation failed

Programcı yanıtı

HTTP isteğinden Expect üstbilgisini kaldırın.

MQHTTP50001: Beklenmeyen bir sorun var ...

IBM MQ bridge for HTTP içinde bir hata oluştu.

Açıklama

IBM MQ bridge for HTTP içinde bir hata oluştu.

HTTP durum kodu

500 Internal server error

Programcı yanıtı

HTTP için IBM MQ Bridge sistem yöneticisiyle iletişim kurun.

MQHTTP50201: IBM MQ bridge for HTTP ile kuyruk yöneticisi arasında bir hata oluştu.

IBM MQ bridge for HTTP ile kuyruk yöneticisi arasında bir hata oluştu

Açıklama

IBM MQ bridge for HTTP ile kuyruk yöneticisi arasında bir hata oluştu

HTTP durum kodu

502 Bad Gateway

Programcı yanıtı

HTTP için IBM MQ Bridge sistem yöneticisiyle iletişim kurun.

MQHTTP50401: İleti alma zamanaşımına uğradı

Bir HTTP **GET** ya da HTTP **DELETE** içinde belirtilen istek değiştiricileriyle eşleşen hiçbir ileti zamanaşımı süresinde döndürülmedi.

Açıklama

Bir HTTP **GET** ya da HTTP **DELETE** içinde belirtilen istek değiştiricileriyle eşleşen hiçbir ileti zamanaşımı süresinde döndürülmedi. Dönüş kodu, HTTP isteğinin ömrü boyunca herhangi bir zaman uygun bir ileti olmadığını belirtir.

HTTP durum kodu

504 Gateway timeout

Programcı yanıtı

Bir ileti bekleniyorsa, HTTP isteğinin x-msg-correlId ve x-msg-msgid gibi üstbilgi alanlarını denetleyin. HTTP isteği URI 'sında belirtilen hedefin doğru olup olmadığını denetleyin. HTTP isteğinin bekleme süresini x-msg-wait üstbilgi alanını kullanarak genişletmeyi deneyin.

MQHTTP50501: HTTP 1.1 ve yukarı ...

HTTP isteğinde kullanılan HTTP protokolü, IBM MQ bridge for HTTP tarafından desteklenmiyor.

Açıklama

HTTP isteğinde kullanılan HTTP protokolü, IBM MQ bridge for HTTP tarafından desteklenmiyor.

HTTP durum kodu

505 HTTP version not supported

Programcı yanıtı

HTTP isteğini HTTP iletişim kuralını V1.1 ya da daha sonraki bir sürümü kullanacak şekilde değiştirin.

HTTP için WebSphere Köprüsü için ileti tipleri ve ileti eşlemeleri

IBM MQ bridge for HTTP , dört ileti sınıfını destekler: TEXT, BYTES, STREAM ve MAP. İleti sınıfları JMS ileti tipleriyle ve HTTP Content-Type ile eşlenir.

HTTP POST

Hedefe ulaşan ileti tipi, x-msg-class üstbilgisinin değerine ya da HTTP isteğinin Content-Type değerine bağlıdır. [Çizelge 215 sayfa 1769](#) , her x-msg-class değerine karşılık gelen HTTP Content-Type tipini gösterir. Her iki alan da ileti tipini ve ileti biçimini ayarlamak için kullanılabilir. Her iki alan da ayarlanmış ve sürekli olarak ayarlanmışsa, bir Bad Request exception döndürülür (HTTP 400, MQHTTP20004).

<i>Çizelge 215. Mapping between x-msg-sınıf and HTTP İçerik Tipi</i>	
x-msg-sınıfı	HTTP İçerik Tipi
BYTES	application/octet-stream application/xml
TEXT	text/*
MAP	application/x-www-form-urlencoded application/xml (isteğe bağlı)
STREAM	application/xml (isteğe bağlı)

If the JMS message type is set in the MQRFH2 header, it is mapped according to [Çizelge 216 sayfa 1769](#).

<i>Çizelge 216. x-msg-class ve JMS ileti tipleri arasında eşleme.</i>	
x-msg-sınıfı	JMS ileti tipi
BYTES	jms_bytes
TEXT	jms_text
MAP	jms_map
STREAM	jms_stream

JMS ileti tipi her zaman MAP ya da STREAM ileti sınıfı için ayarlıdır. Her zaman BYTES ya da TEXT ileti sınıfı için ayarlanmaz. İstek için bir MQRFH2 yaratılacaksa, JMS ileti tipi her zaman ayarlıdır. Ters durumda, MQRFH2 yaratılmıyorsa, hiçbir JMS ileti tipi ayarlanmaz. İstekte kullanıcı özellikleri ayarlandıysa, x-msg-usr üstbilgisi kullanılarak bir MQRFH2 yaratılır.

JMS ileti tipi ayarlandıysa, ileti biçimi MQFMT_NONE olarak ayarlandıysa, bkz. [Çizelge 218 sayfa 1770](#):

Çizelge 217. *x-msg-sınıf* ile IBM MQ ileti biçimi arasında eşleme

x-msg-sınıfı	İletide MQRFH2 ile ileti biçimi var	İletide yok MQRFH2 olan ileti biçimi
BYTES	MQFMT_NONE	MQFMT_NONE
TEXT	MQFMT_NONE	MQFMT_STRING
MAP	MQFMT_NONE	Mümkün değil
STREAM	MQFMT_NONE	Mümkün değil

HTTP GET ya da DELETE

Alınan ileti tipi ya da biçim, `x-msg-class` üstbilgisinin değerini ve HTTP yanıtının İçerik-Türü ' ne saptar. `x-msg-sınıf` üstbilgisi, `x-msg-headers` isteğinde istenirse döndürülür.

Çizelge 218 sayfa 1770 , `x-msg-class` ve Content-Type arasındaki eşlemeleri ve kuyruktan ya da konudan alınan ileti tipini açıklar.

Çizelge 218. İleti tiplerinin *x-msg-class* ve Content-Type olarak eşlenmesi

İleti biçimi	JMS İleti tipi	x-msg-sınıfı	İçerik Tipi
MQFMT_STRING dışında herhangi bir şey	Yok	BYTES	application/octet-stream
MQFMT_STRING	Yok	TEXT	text/plain
MQFMT_NONE	jms_bytes	BYTES	application/octet-stream
MQFMT_NONE	jms_text	TEXT	text/plain
MQFMT_NONE	jms_map	MAP	application/xml
MQFMT_NONE	jms_stream	STREAM	application/xml

MAP ve STREAM ileti sınıfı serileştirme

MAP ve STREAM ileti sınıfları, bir ileti kuyrukta diziselleştirildiği şekilde, aynı şekilde HTTP yanıtında istemciye yeniden diziselleştirilir.

MAP için, XML adı, tipi ve değer üçüzleri şu şekilde kodlanır:

```
<map>
  <elt name="elementname1" dt="datatype1">value1</elt>
  <elt name="elementname2" dt="datatype2">value2</elt>
  ...
</map>
```

STREAM , MAP gibi, ancak öge adları yok:

```
<stream>
  <elt dt="datatype1">value1</elt>
  <elt dt="datatype2">value2</elt>
  ...
</stream>
```

Not: datatype , kullanıcı tanımlı özellikleri tanımlamak için tanımlanan ve “[usr: HTTP x-msg-usr entity-header](#)” sayfa 1759 içinde listelenen veri tiplerinden biridir. Varsayılan veri tipi string olduğu için, dizgi ögeleri için dt="string" özniteliği atlandı.

- [“Oluşturucular” sayfa 1772](#)

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;}

Çözümlemiş nesne adı.

public static string ResolvedObjectQMgrName {get;}

Çözümlemiş 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 SSL istemcisi 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 1773
- “Oluşturucular” sayfa 1773

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.

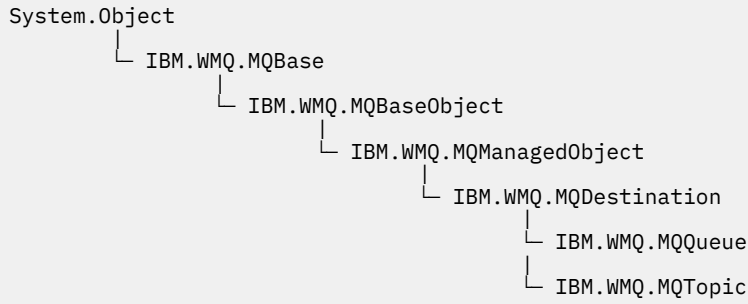
Oluşturucular

`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 1774](#)
- [“Yöntemler” sayfa 1774](#)
- [“Oluşturucular” sayfa 1776](#)

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.

message

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, MQQM_SYNCPOINT altında alınan iletiler için, başarılı yeniden bağlantı sonrasında MQRC_BACKED_OUT neden kodunu döndürür.

getMessageOptions

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.

MaxMsgSize

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.

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);
```

message

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.

putMessageOptions

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 1776](#)
- [“Özellikler” sayfa 1777](#)
- [“Oluşturucular” sayfa 1778](#)

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 kullanılan SSL/TLS sertifika geçerlilik denetimi ilkesini ayarlayın. 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 ortamı somutlaştırılmadan öncesetayarlamamız 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 SSL ' yi etkinleştirmek üzere SVRCONN kanalındaki CipherSpec kümesinin değerine SSLCipherSpec değerini ayarlayın. Varsayılan değer Boş Değerdir ve bağlantı için SSL 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 yoksayılr.

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 SSL etkileşimi içinde gönderilen ve alınan şifrelenmemiş baytların sayısını belirtir.

public static ArrayList MQAIRArray {get; set;}

MQAuthenticationInformationRecord nesneleri 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 paylaşılabilecek etkileşim sayısı üst sınırını belirler.

0 değeri, kanalın etkileşim paylaşımı, okuma ve sağlıklı işletim bildirimi 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 süreç, 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 1779](#)
- [“Oluşturucular” sayfa 1779](#)

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.

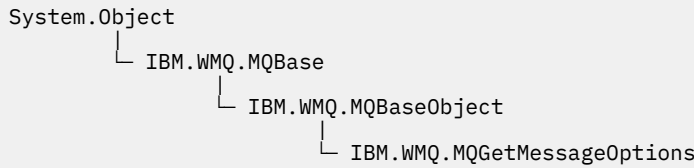
reasonCode

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 1779](#)
- [“Oluşturucular” sayfa 1782](#)

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 gerekiyorsa, 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.MQGMNO_NO_SYNCPOINT

Eşitleme noktası denetimi olmadan ileti al.

MQC.MQGMNO_NO_WAIT

Uygun bir ileti yoksa hemen geri dönün.

MQC.MQGMNO_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.MQGMNO_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.MQGMNO_PROPERTIES_FORCE_MQRFH2

MQRFH2 üstbilgilerinde, ileti tanımlayıcısında ya da uzantıda bulunan özellikler dışında ileti özelliklerini alır. Özellikleri almayı bekleyen, ancak ileti tanıtıcılarını kullanmak üzere değiştirilemeyecek uygulamalarda MQC.MQGMNO_PROPERTIES_FORCE_MQRFH2 kullanın.

MQC.MQGMNO_PROPERTIES_IN_HANDLE

Retrieve message properties using a MsgHandle.

MQC.MQGMNO_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.MQGMNO_SYNCPOINT_IF_PERSISTENT*

İleti kalıcıysa, eşitleme noktası denetimi ile ileti alın.

MQC.MQGMNO_UNLOCK*

Önceden kilitlemiş bir iletinin kilidini açın.

MQC.MQGMNO_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.MQGMNO_WAIT ile WaitInterval seçeneğini kullanın. Bir ileti için sınırsız süre beklemek üzere MQC.MQWI_UNLIMITED değerini ayarlayın.

Oluşturucular

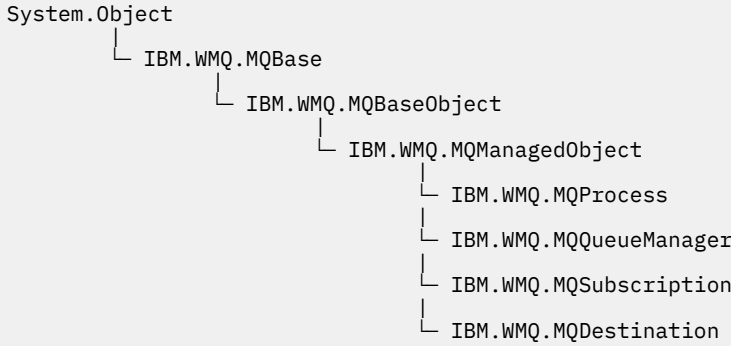
public MQGetMessageOptions()

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 1782](#)
- [“Yöntemler” sayfa 1783](#)
- [“Oluşturucular” sayfa 1784](#)

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'dir. Kalıcı dinamik kuyruklar, geçici dinamik kuyruklar, abonelikler ve bunları yaratan nesneler 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.

selector

Hangi özniteliğin sorgulanmakta olduğunu gösteren tamsayı.

length

Gereken 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.

selectors

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 dizisinin uzunluğu değişmez.

public void Set(int[] selectors, int[] intAttrs, byte[] charAttrs);

MQException yayınlıyor.

Seçiciler vektöründe tanımlanan öznitelikleri ayarlar. Ayarlanacak öznitelikler, seçiciler dizisinde belirtilir.

selectors

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 bitleştirileceği 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 dizgisi belirler.

selector

Hangi özniteliğin ayarlanmakta olduğunu gösteren tamsayı.

value

Öznitelik değeri olarak ayarlanacak dizgi.

length

Gereken dizilimin uzunluğunu gösteren tamsayı.

Oluşturucular

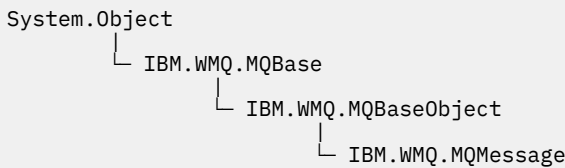
protected MQManagedObject()

Oluşturucu yöntemi. Bu nesne, kendisi tarafından somutlaştırılamayan 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 nesneler arasında veri aktarmak için Read ve Write yöntemlerini kullanın. MQDestination, MQQueue ve MQTopic sınıflarının Put ve Get yöntemlerini kullanarak MQMessage nesnelerini 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 1785](#)
- [“Read ve Write ileti yöntemleri” sayfa 1790](#)
- [“Arabellek yöntemleri” sayfa 1793](#)
- [“Özellik yöntemleri” sayfa 1794](#)

- “Oluşturucular” sayfa 1796

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'dir.

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 ""'dır.

public string ApplicationOriginData {get; set;}

İletinin kaynağı hakkında ek bilgi sağlamak için kullanılacak uygulama tarafından tanımlanan bilgiler. Varsayılan değer ""'dır.

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.

İletideki 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_MGR'deki 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 örnek olarak UTF-8 dizesini dönüştürür.
- Byte yöntemleri, uygulama ve ileti arabelleğiyle değiştirilmeden byte 'dır

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 ileti 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 duyarlılık 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'dir.

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_IIH_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 yanıtlama 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 bölümlenmesinin 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;}

İletiyi koyan uygulamanın adı. Varsayılan değer ""değeridir.

public int PutApplicationType {get; set;}

İletiyi 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 ""'dır 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 seçenekten 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

-999999999999-999999999999999

- 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 219. Okuma ve Yazma ileti yöntemleri

Hedef türü	Yöntem imzaları
Boolean	<pre>public bool ReadBoolean(); public void WriteBoolean(bool value);</pre>
Byte	<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>
Bytes	<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>
Decimal12	<pre>public void WriteDecimal2(short value)</pre>

Çizelge 219. Okuma ve Yazma ileti yöntemleri (devamı var)

Hedef türü	Yöntem imzaları
Decimal4	<pre>public void WriteDecimal4(short value)</pre>
Decimal8	<pre>public void WriteDecimal8(short value)</pre>
Double	<pre>public double ReadDouble()</pre> <pre>public void WriteDouble(double value)</pre>
Float	<pre>public float ReadFloat()</pre> <pre>public void WriteFloat(float value)</pre>
Int2	<pre>public void WriteInt2(int value)</pre>
Int4	<pre>public int readDecimal4() public int ReadInt() public int ReadInt4()</pre> <pre>public void WriteInt(int value) public void WriteInt4(int value)</pre>
Int8	<pre>public void WriteInt8(long value)</pre>
Long	<pre>public long ReadDecimal8() public long ReadLong() public long ReadInt8()</pre> <pre>public void WriteLong(long value)</pre>
Object	<pre>public Object ReadObject()</pre> <pre>public void WriteObject(Object object)</pre>
Short	<pre>public short ReadShort() public short ReadDecimal2() public short ReadInt2()</pre> <pre>public void WriteShort(int value)</pre>

Çizelge 219. Okuma ve Yazma ileti yöntemleri (devamı var)

Hedef türü	Yöntem imzaları
string	<pre>public string ReadString(int length) public void WriteString(string string)</pre>
Unsigned Short	<pre>public ushort ReadUnsignedShort() public ushort ReadUInt2()</pre>
Unicode	<pre>public string ReadLine() public char ReadChar() public void WriteChar(int value) public void WriteChars(string string)</pre>
UTF	<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 ileri 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 yayınlıyor.

İletiden belirtilen adı taşıyan bir özelliği siler.

name

Silinecek özelliğin adı.

public System.Collections.IEnumerator GetPropertyNames(string name)

MQException yayınlıyor.

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.

name

Eşleştirmek için özelliğin adı.

SetProperty ve GetProperty yöntemleri

Tüm SetProperty ve GetProperty yöntemleri MQException yayınlıdır.

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.

Çizelge 220. SetProperty ve GetProperty yöntemleri	
Tip	Yöntem imzaları
Boolean	<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>
Byte	<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>
Bytes	<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 220. SetProperty ve GetProperty yöntemleri (devamı var)

Tip	Yöntem imzaları
Double	<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>
Float	<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>
Int2	<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>
Int4	<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>
Int8	<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>
Long	<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>
Object	<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 220. SetProperty ve GetProperty yöntemleri (devamı var)

Tip	Yöntem imzaları
Short	<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>
string	<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

```
System.Object
├── IBM.WMQ.MQBase
│   ├── IBM.WMQ.MQBaseObject
│   │   ├── IBM.WMQ.MQManagedObject
│   │   │   └── IBM.WMQ.MQProcess
```

```
public class IBM.WMQ.MQProcess extends IBM.WMQ.MQManagedObject;
```

- [“Özellikler” sayfa 1796](#)
- [“Oluşturucular” sayfa 1797](#)

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.MQOO_FAIL_IF QUIESCING
- MQC.MQOO_INQUIRE
- MQC.MQOO_SET
- MQC.MQOO_ALTERNATE_USER_AUTHORITY

queueManagerName

İşlemin tanımlandığı kuyruk yöneticisinin adı. Kuyruk yöneticisi, sürecin erişmekte olduğu kuyruk yöneticisi aynıysa, boş değerli ya da boş bir kuyruk yöneticisi adı bırakabilirsiniz.

alternateUserId

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 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.MQ00_FAIL_IF QUIESCING
- MQC.MQ00_INQUIRE
- MQC.MQ00_SET
- MQC.MQ00_ALTERNATE_USER_AUTHORITY

queueManagerName

İşlemin tanımlandığı kuyruk yöneticisinin adı. Kuyruk yöneticisi, sürecin erişmekte olduğu kuyruk yöneticisi aynıysa, boş değerli ya da boş bir kuyruk yöneticisi adı bırakabilirsiniz.

alternateUserId

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 SetProperty methods. MQPropertyDescriptor, bir MQMessage özelliğini açıklar.

Sınıf

```
System.Object  
└─ IBM.WMQ.MQPropertyDescriptor
```

```
public class IBM.WMQ.MQPropertyDescriptor extends System.Object;
```

- [“Özellikler” sayfa 1799](#)
- [“Oluşturucular” sayfa 1800](#)

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 hiçbirisi ya da hiçbirisi 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_PROPERTY değerini 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.Put davranışını değiştirir.

Sınıf

```
System.Object
├── IBM.WMQ.MQBase
│   └── IBM.WMQ.MQBaseObject
│       └── IBM.WMQ.MQPutMessageOptions
```

```
public class IBM.WMQ.MQPutMessageOptions extends IBM.WMQ.MQBaseObject;
```

- [“Özellikler” sayfa 1800](#) [“Oluşturucular” sayfa 1802](#)

Özellikler

Özellikler alınırken MQException için test ediliyor.

Not: Bu sınıftaki bazı seçeneklerin davranışı, kullanılırların 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_CONTEXT içeriyorsa, 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 çözümleyen 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.puteylemine denetleyen seçenekler. Aşağıdaki değerlerin hiçbiri ya da hiçbiri belirlenebilir. Birden fazla seçenek gerekliyse, 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 özniteliğinden 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 eşanlamlıdır.

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 dizgisinde bir genel arama karakteri içerebilir. Konu ağacında, abonelikte 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 çözümleyen kuyruklara başarılı bir şekilde gönderdiği ileti sayısını bildirir.

.NET dağıtım listelerini desteklemez, ancak tek bir kuyruk açılırken InvalidDestCount (InvalidDust Sayısı) ayarı ayarlanır.

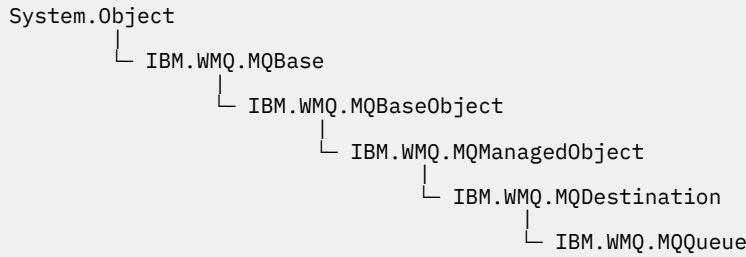
Oluşturucular**public MQPutMessageOptions();**

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 özniteliklerini 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 1803](#)
- [“Yöntemler” sayfa 1805](#)
- [“Oluşturucular” sayfa 1807](#)

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 değıştirme geçerli değildir. Bu değıştirme yalnızca yerel kuyruklar için geçerlidir.

public DateTime CreationDateTime {get;}

Bu kuyruğun yaratıldığı tarih ve saat.

public int CurrentDepth {get;}

Şu anda kuyruktaki ileti sayısını alır. Bu değır, bir put çağırısı sırasında ve bir alma çağırısının 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 değırler 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 değırler şunlardır:

- MQC.MQQA_GET_INHIBITED
- MQC.MQQA_GET_ALLOWED

public int InhibitPut {get; set;}

Bu kuyruğa ya da bu konuya ilişkin iletiler yerleştirtip koyamayacağınızı denetler. Olası değırler 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 toplamının etkinleştirilip etkinleştirilmeyeceğini belirler.

public int QueueMonitoring {get;}

Kuyruğun izlenmesini etkinleştirip etkinleştirmeyip etkinleştirmeyeceğini belirler.

public int QueueStatistics {get;}

Kuyruğa ilişkin istatistik toplamının 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 iletisine eklediği serbest biçimli veriler. Bu kuyruğa gelen bir ileti, bir tetikleme iletisinin 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 iletisi yazılmadan önce kuyruğun üzerinde olması gereken ileti sayısı.

public int TriggerMessagePriority {get; set;}

İletilerin hangi iletilerin oluşturulacağı, tetikleme iletilerinin oluşturulmasına katkıda bulunmamasını sağlar. Yani, kuyruk yöneticisi bir tetikleme oluşturulup oluşturulmayacağına karar verirken bu iletileri yoksayar. Sıfır değeri, tüm iletilerin, 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.

message

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.

getMessageOptions

Alma işleminin denetlenmesini denetleyen seçenekler.

MQC.MQGM_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 MQGM_NOWAIT olur.

Yeniden bağlanabilir bir istemcide MQGM_LOGICAL_ORDER seçeneğini kullanırsanız, MQRC_RECONNECT_INCOMPATIBLE neden kodu döndürülür.

MaxMsgSize

Bu ileti nesnesinin alınabileceği en büyük ileti. Kuyruktaki ileti bu boyuttan büyükse, iki şeyden biri oluşur:

- If the MQGM_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.
- MQGM_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.

```
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);
```

message

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 .

putMessageOptions

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.

message

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 .

putMessageOptions

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.

message

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.

putMessageOptions

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.

message

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.

putMessageOptions

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 yöneticisindeki bir kuyruğa erişir.

İletiler alabilir ya da bunlara göz atabilir, iletileri yerleştirebilir, kuyruğun özniteliklerini sorgulayabilir ya da kuyruğun özniteliklerini ayarlayabilirsiniz. Kuyruk adı bir model kuyruğa, dinamik bir yerel kuyruk yaratılır. Dinamik kuyruğun adını bulmak için, sonuçtaki MQQueue nesnesinin name özniteliğini 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.

queueManagerName

Kuyruğun tanımlı olduğu kuyruk yöneticisinin adı. Tamamen boş ya da boş değer olan bir ad, MQQueueManager nesnesinin bağlı olduğu kuyruk yöneticisini belirtir.

dynamicQueueName

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.

alternateUserId

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 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.

queueManagerName

Kuyruğun tanımlı olduğu kuyruk yöneticisinin adı. Tamamen boş ya da boş değer olan bir ad, MQQueueManager nesnesinin bağlı olduğu kuyruk yöneticisini belirtir.

dynamicQueueName

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.

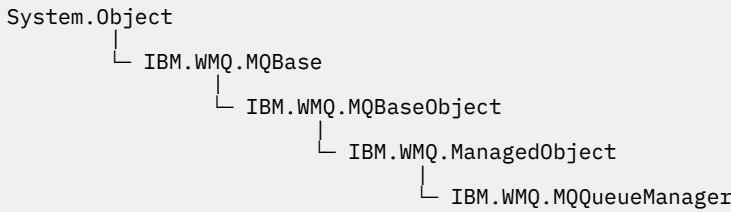
alternateUserId

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 1810](#)
- [“Yöntemler” sayfa 1813](#)
- [“Oluşturucular” sayfa 1819](#)

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 hala 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 'nün 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 MaximumMessageLength değerinden 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.MQSQM_USE

İletiler, paylaşılan kuyruğa konmadan önce nesne kuyruk yöneticisine teslim edilir.

MQCMQSQM_IGNORE

İletiler doğrudan paylaşılan kuyruğa konadır.

public int SSLEvent {get;}

SSL 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 SSL etkileşimi içinde gönderilen ve alınan şifrelenmemiş baytların sayısını belirtir.

public int ClusterSenderStatistics {get;}

İstatistiklerin ardışık toplantıları 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;}

TCPStackTypedegerine 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 özneliklerine ilişkin bilgi edinmek için bu kuyruk yöneticisine ilişkin bir IBM MQ işleme 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

queueManagerName

İşlemin tanımlandığı kuyruk yöneticisinin adı. Kuyruk yöneticisi, sürecin erişmekte olduğu kuyruk yöneticisi aynıysa, boş değerli ya da boş bir kuyruk yöneticisi adı bırakabilirsiniz.

alternateUserId

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 özneliklerini sorgulayabilir ya da kuyruğun özneliklerini ayarlayabilirsiniz. Kuyruk adı bir model kuyruğsa, 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.

queueManagerName

Kuyruğun tanımlı olduğu kuyruk yöneticisinin adı. Tamamen boş ya da boş değer olan bir ad, MQQueueManager nesnesinin bağlı olduğu kuyruk yöneticisini belirtir.

dynamicQueueName

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.

alternateUserId

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 nesneleri, bazen konu nesneleri olarak adlandırılan denetim konusu nesnelerle 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 wild cards.

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 abonelikte tutarlı olduğundan emin olmanız gerekir.

destination

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 bitştirilir. *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

options

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.

alternateUserId

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.

properties

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

İletin yerleştirilecek kuyruğun adı.

destinationName

Hedef nesnenin adı. Bu, *typedeğerine* bağlı olarak bir kuyruk ya da bir konudur.

type

Hedef nesnenin tipi. Seçenekleri birleştirmemelisiniz.

MQC.MQOT_Q

Kuyruk

MQC.MQOT_TOPIC

Konu

queueManagerName

Kuyruğun tanımlandığı kuyruk yöneticisi ya da kuyruk yöneticisi diğer adının adı.

MQC.MQOT_TOPIC tipi belirtilirse, bu parametre yoksayılır.

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* yoksayılır.

message

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](#)).

putMessageOptions

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.

alternateUserId

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 yöneticiyle 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*, *Port* ya da *Channel*, *properties* içinde belirtilir.
3. *MQEnvironment.HostName*, *MQEnvironment.Port* ya 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.

queueManagerName

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.

options

MQCNO bağlantı seçeneklerini belirtin. Değerlerin, yapılmakta olan bağlantı tipine uygun olması gerekir. Örneğin, bir client istemci bağlantısı için aşağıdaki sunucu bağlantısı özelliklerini belirtirseniz, bir MQException dosyası atılır.

- MQC.MQCNO_FASTPATH_BINDING
- MQC.MQCNO_STANDARD_BINDING

properties

Özellikler parametresi, MQEnvironment tarafından ayarlanan özellikleri geçersiz kılan bir dizi anahtar/değer çiftini alır; Örneğin, “MQEnvironment özelliklerini geçersiz kıl” sayfa 1822 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 WebSphere 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 43. 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 44. 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.MQCNORECONNECT); // 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);

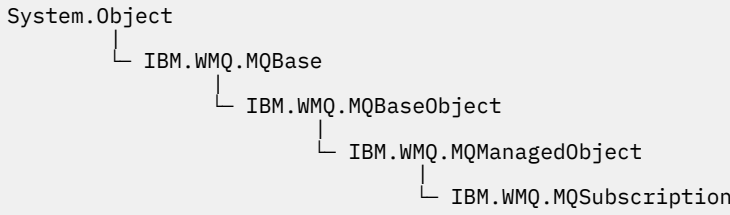
```

Şekil 45. 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 1823
- “Yöntemler” sayfa 1823
- “Oluşturucular” sayfa 1823

Özellikler

MQManagedObject sınıfını kullanarak abonelik özelliklerine erişin; bkz. “Özellikler” sayfa 1782.

Yöntemler

Access subscription Inquire, Set and Get methods using the MQManagedObject class; see “Yöntemler” sayfa 1783.

```
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.

options

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.

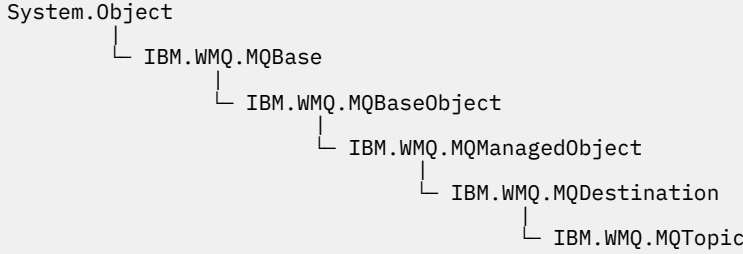
RequestPublicationUpdate yöntemini çağırın. MQSubscription , MQManagedObject sınıfının alt sınıfıdır. MQManagedObject' un özelliklerine ve yöntemlerine erişmek için başvuruyu kullanın.

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 1824](#)
- [“Yöntemler” sayfa 1824](#)
- [“Oluşturucular” sayfa 1826](#)

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 abonelik ile 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 abonelik ile 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);
```

message

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 .

putMessageOptions

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 zamanuymulu 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.

message

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.

getMessageOptions

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.

MaxMsgSize

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` nesneleri, bazen konu nesneleri olarak adlandırılan denetim konusu nesnelerle 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 wild cards.

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 abonelikte tutarlı olduğundan emin olmanız gerekir.

queueManager

Bir konuya erişmek için kuyruk yöneticisi.

destination

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 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

options

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.

alternateUserId

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.

properties

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 yöneticisininde bir konuya erişin.

MQTopic nesneleri, bazen konu nesneleri olarak adlandırılan denetim konusu nesnelerle 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 dizesini birleştirmenin sonudur. Birinci konu dizgisi, *topicObject* ile tanımlanan yönetimle ilgili konu nesnesi tarafından tanımlanır. İkinci konu dizesi: *topicString*. The resulting topic string associated with the MQTopic object can identify multiple topics by including wild cards.

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.

destination

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 dizgisi. *topicName* , *topicObject* yönetim konusu nesnesinde tanımlanan konu dizgisiyle bitştirilir. *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 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

options

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.

alternateUserId

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.

properties

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

IMQObjectTrigger.NET arabirimi

runmqdmn.NET izleme programı tarafından geçirilen 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.

queue

İzlenmekte olan kuyruk.

message

Kuyruktan ileti okundu.

param

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 tanıttıcıları

.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

Karakter kümesi	Tanım
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 1834). Ayrıntılı açıklamalar için bkz. “Nesnelerin öznitelikleri” sayfa 778.

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 kullanılabilecek 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) bu da 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 nesneleri 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 parçacılarındaki nesnelerin birbirlerinden yalıtıldığından emin olmak için, kendi bağımsız derlemine sahip olmalıdır.

Sınıflar şunlardır:

- “ImqAuthenticationKayıt C++ sınıfı” sayfa 1848
- “ImqBinary C++ sınıfı” sayfa 1851
- “ImqCache C++ sınıfı” sayfa 1852
- “ImqChannel C++ sınıfı” sayfa 1855
- “ImqCICSBridgeHeader C++ sınıfı” sayfa 1861
- “ImqDeadLetterHeader C++ sınıfı” sayfa 1867
- “ImqDistributionListe C++ sınıfı” sayfa 1869
- “ImqError C++ sınıfı” sayfa 1871

- [“ImqGetMessageOptions C++ sınıfı” sayfa 1872](#)
- [“ImqHeader C++ sınıfı” sayfa 1875](#)
- [“ImqIMSBridgeHeader C++ sınıfı” sayfa 1877](#)
- [“ImqItem C++ sınıfı” sayfa 1880](#)
- [“ImqMessage C++ sınıfı” sayfa 1881](#)
- [“ImqMessageTracker C++ sınıfı” sayfa 1888](#)
- [“ImqNamelist C++ sınıfı” sayfa 1890](#)
- [“ImqObject C++ sınıfı” sayfa 1892](#)
- [“ImqProcess C++ sınıfı” sayfa 1897](#)
- [“ImqPutMessageOptions C++ sınıfı” sayfa 1899](#)
- [“ImqQueue C++ sınıfı” sayfa 1901](#)
- [“ImqQueueManager C++ sınıfı” sayfa 1911](#)
- [“ImqReferenceHeader C++ sınıfı” sayfa 1928](#)
- [“ImqString C++ sınıfı” sayfa 1930](#)
- [“ImqTrigger C++ sınıfı” sayfa 1935](#)
- [“ImqWorkÜstbilgi C++ sınıfı” sayfa 1938](#)

C++ ve MQI çapraz başvurusu

Bu konu derlemi, C++ ile MQI 'a ilişkin bilgileri içerir.

Bu bilgileri [“MQI ' da kullanılan veri tipleri” sayfa 236](#) 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.

Çizelge 221. Veri yapısı, sınıf ve include-file çapraz başvurusu		
Veri yapısı	Sınıf	İçerme dosyası
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
MQIHL	ImqIMSBridgeHeader	imqiil.hpp
	ImqItem	imqitm.hpp
Mqmd	ImqMessage	imqmsg.hpp
	ImqMessageİzleyicisi	imqmtr.hpp

Çizelge 221. Veri yapısı, sınıf ve include-file çapraz başvurusu (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	imqqueue.hpp
MQBO, MQCNO, MQCSP	ImqQueueYöneticisi	imqmgr.hpp
MQRMH	ImqReferenceÜstbilgisi	imqrfh.hpp
	ImqString	imqstr.hpp
MQTM	ImqTrigger	imqtrg.hpp
MQTMC		
MQTMC2	ImqTrigger	imqtrg.hpp
MQXQH		
MQWIH	ImqWorkÜstbilgisi	imqwih.hpp

ImqAuthenticationKayıt çapraz başvurusu

Öznitelikler, veri yapıları, alanlar ve ImqAuthenticationKayıt C++ sınıfı çağrıları çapraz başvuru.

Öznitelik	Veri yapısı	Alan	Ara
Bağlantı adı	MQAIR	AuthInfoConnName	MQCONN
parola	MQAIR	LDAPPassword	MQCONN
tip	MQAIR	AuthInfoTipi	MQCONN
kullanıcı adı	MQAIR	LDAPUserNamePtr	MQCONN
	MQAIR	LDAPUserNameGörelili Konumu	MQCONN
	MQAIR	LDAPUserNameUzunluğu	MQCONN

ImqCache çapraz başvuru

Cross-reference of attributes and calls for the ImqCache C++ class.

Öznitelik	Ara
otomatik arabellek	MQGet
arabellek uzunluğu	MQGet
arabellek göstergesi	MQGET, MQPUT
veri uzunluğu	MQGet
veri görelili konumu	MQGet
veri göstergesi	MQGet
ileti uzunluğu	MQGET, MQPUT

ImqChannel çapraz başvuru

Cross-reference of attributes, data structures, fields, and calls for the ImqChannel C++ class.

Öznitelik	Veri yapısı	Alan	Ara
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
SSL eş adı	MQCD	sslPeerAdı	MQCONN
hareket programı adı	MQCD	TpName	MQCONN
taşıma tipi	MQCD	TransportType	MQCONN
kullanıcı kimliği	MQCD	UserIdentifier	MQCONN

ImqCICSBridgeHeader apraz bařvurusu

ImqCICSBridgeHeader C++ sınıfına ilişkin zniteliklere, veri yapılarına ve alanlara apraz bařvuru.

znitelik	Veri yapısı	Alan
kpr olağandıřı bitiř kodu	MQCIH	AbendCode
ADS tanımlayıcı	MQCIH	AdsDescriptor
uyarı tanıtıcısı	MQCIH	AttentionId
kimlik doğrulayıcı	MQCIH	Kimlik doğrulayıcı
kpr tamamlama kodu	MQCIH	BridgeCompletionKodu
kpr hata grelı konumu	MQCIH	ErrorOffset
kpr neden kodu	MQCIH	BridgeReason
kpr iptal kodu	MQCIH	CancelCode
etkileřimli grev	MQCIH	ConversationalTask
imle konumu	MQCIH	CursorPosition
tesis simgesi	MQCIH	Tesis
tesis zaman tutma	MQCIH	FacilityKeepSresi
gibi bir tesis	MQCIH	FacilityLike
iřlev	MQCIH	İřlev
bekleme sresi aralığı	MQCIH	GetWaitAralığı
Bağlantı tipi	MQCIH	LinkType
sonraki iřlem tanıtıcısı	MQCIH	NextTransactionTanıtıcısı
ıkıř veri uzunluğ	MQCIH	OutputDataUzunluğ
yanıtlama biimi	MQCIH	ReplyToBiimi
kpr dnř kodu	MQCIH	ReturnCode
bařlatma kodu	MQCIH	StartCode
grev bitiř durumu	MQCIH	TaskEndDurumu
iřlem tanıtıcısı	MQCIH	TransactionId
uow denetimi	MQCIH	UowControl
srm	MQCIH	S\u00fcr\u00fcm

ImqDeadLetterHeader apraz bařvurusu

ImqDeadLetterHeader C++ sınıfına ilişkin zniteliklerin, veri yapılarının ve alanların apraz bařvurusu.

znitelik	Veri yapısı	Alan
l harf neden kodu	MQDLH	Neden
Hedef kuyruk yneticisi adı	MQDLH	DestQMgrAdı
hedef kuyruk adı	MQDLH	DestQName
Koyma Uygulaması Adı	MQDLH	PutApplAdı
Koyma Uygulaması Tipi	MQDLH	PutApplTipi

Öznitelik	Veri yapısı	Alan
bitiş tarihi	MQDLH	PutDate
sürme zamanı	MQDLH	PutTime

ImqError çapraz başvurusu

Cross-reference of attributes and calls for the ImqError C++ class.

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.

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	Bölömlöme
bölömlö 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.

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.

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 başvurusu

Cross-reference of attributes and calls for the ImqItem C++ class.

Öznitelik	Ara
yapı tanıtıcısı	MQGet

ImqMessage çapraz başvuru

Cross-reference of attributes, data structures, fields, and calls for the ImqMessage C++ class.

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	
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	

Öznitelik	Veri yapısı	Alan	Ara
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.

Öznitelik	Veri yapısı	Alan
muhasabe 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.

Öznitelik	Sorgu	Ara
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 çağrıları çapraz başvuru.

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

Öznitelik	Veri yapısı	Alan	Sorgu	Ara
kuyruk yöneticisi tanıtıcısı	kuyruk yöneticisi tanıtıcısı		MQCA_Q_MGR_IDENTIFIER	MQINQ

ImqProcess çapraz başvurusu

Öznitelikler, sorgular ve ImqAuthenticationKayıt C++ sınıfı çağrıları çapraz başvuru.

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.

Çizelge 222. ImqPutMessageOptions çapraz başvurusu		
Öznitelik	Veri yapısı	Alan
bağlam başvurusu	MQPMO	Bağlam
	MQPMO	InvalidDestSayısı
	MQPMO	KnownDestSayı
seçenekler	MQPMO	Seçenekler
kayıt alanları	MQPMO	PutMsgRecFields
çözümlemiş kuyruk yöneticisi adı	MQPMO	ResolvedQMgrAdı
çözümlemiş 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 223. ImqQueue çapraz başvurusu				
Öznitelik	Veri yapısı	Alan	Sorgu	Ara
backout request adı			MQCA_BACKUP_REQ_Q_NAME	MQINQ
Geriletme eşiği			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

Çizelge 223. ImqQueue çapraz başvurusu (devamı var)

Öznitelik	Veri yapısı	Alan	Sorgu	Ara
küme iş yükü sırası			MQIA_CLWL_Q_RANK	MQINQ
küme iş yükü önceliği			MQIA_CLWL_Q_PRIORITY	MQINQ
küme iş yükü kullanım kuyruğu			MQIA_CLWL_USEQ	MQINQ
yaratma tarihi			MQCA_CREATION_DEĞERİ	MQINQ
Oluşturulma zamanı			MQCA_CREATION_TIME	MQINQ
yürürlükteki derinlik			MQIA_CURRENT_Q_DEPTH	MQINQ
varsayılan bağlama			MQIA_DEF_BIND	MQINQ
varsayılan giriş açma seçeneği			MQIA_DEF_INPUT_OPEN_OPTION	MQINQ
varsayılan kalıcılık			MQIA_DEF_PERSISTENCE	MQINQ
varsayılan öncelik			MQIA_DEF_PRIORITY	MQINQ
Tanımlama Tipi			MQIA_TANIMLI_TIPİ	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
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

Çizelge 223. ImqQueue çapraz başvurusu (devamı var)

Öznitelik	Veri yapısı	Alan	Sorgu	Ara
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
çözümmlenen kuyruk yöneticisi adı	MQOD	ResolvedQMgrAdı		
çözümmlenen kuyruk adı	MQOD	ResolvedQName		
alıkoyma aralığı			MQIA_RETENTION_INTERVAL	MQINQ
kapsam			MQIA_SCOPE	MQINQ
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_ADı	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.

Öznitelik	Veri yapısı	Alan	Sorgu	Ara
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
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

Öznitelik	Veri yapısı	Alan	Sorgu	Ara
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
komut sunucusu denetimi			MQIA_CMD_SERVER_CONTROL	MQINQ
bağlantı seçenekleri	MQCNO	Seçenekler		MQCONN, MQCONNX
Bağlantı Tanıtıcısı	MQCNO	ConnectionId		MQCONN
Bağlantı durumu				MQCONN, MQCONNX, MQDISC
bağlantı etiketi	MQCD	ConnTag		MQCONN
şifreleme donanımı	MQSCO	CryptoHardware		MQCONN
-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		MQCONN
	MQSCO	AuthInfoRecPtr		MQCONN
etkinliği engelle			MQIA_INHIBIT_EVENT	MQINQ
ip adresi sürümü			MQIA_IP_ADDRESS_VERSION	MQINQ
anahtar havuzu	MQSCO	KeyRepository		MQCONN

Öznitelik	Veri yapısı	Alan	Sorgu	Ara
anahtar sıfırlama sayısı	MQSCO	KeyResetSayı		MQCONN
dinleyici zamanlayıcısı			MQIA_LISTENER_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_HANDSHANS	MQINQ
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

Öznitelik	Veri yapısı	Alan	Sorgu	Ara
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
tcp kanalları			MQIA_TCP_KANALS	MQINQ
tcp canlı tutma			MQIA_TCP_KEEP_ALIVE	MQINQ
tcp adı			MQCA_TCP_ADI	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	CSPUserIdGörelili Konumu		MQCONN
	MQCSP	CSPUserIduzunluğu		MQCONN

ImqReferenceÜstbilgi çapraz başvurusu

ImqAuthenticationRecord C++ sınıfına ilişkin özniteliklere, veri yapılarına ve alanlara çapraz başvuru.

Öznitelik	Veri yapısı	Alan
hedef ortam	MQRMH	DestEnvUzunluğu, DestEnvGörelili Konumu
Hedef Adı	MQRMH	DestNameUzunluğu, DestNameGörelili Konumu
Eşgörünüm Tanıtıcısı	MQRMH	ObjectInstanceTanıtıcısı
mantıksal uzunluk	MQRMH	DataLogicalLength
mantıksal görelili konum	MQRMH	DataLogicalGörelili Konumu
mantıksal görelili konum 2	MQRMH	DataLogicalOffset2
Başvuru tipi	MQRMH	ObjectType
Kaynak Ortamı	MQRMH	SrcEnvUzunluğu, SrcEnvGörelili Konumu

Öznitelik	Veri yapısı	Alan
KAYNAK ADI	MQRMH	SrcNameUzunluğu, SrcNameGörelİ Konumu

ImqTrigger çapraz başvurusu

ImqAuthenticationRecord C++ sınıfına ilişkin özniteliklere, veri yapılarına ve alanlara çapraz başvuru.

Çizelge 224. ImqTrigger çapraz başvurusu		
Öznitelik	Veri yapısı	Alan
Uygulama tanıtıcısı	MQTM	ApplId
Uygulama Tipi	MQTM	ApplType
ortam verileri	MQTM	EnvData
İşlem adı	MQTM	ProcessName
Kuyruk adı	MQTM	QName
tetikleyici verileri	MQTM	TriggerData
Kullanıcı verileri	MQTM	UserData

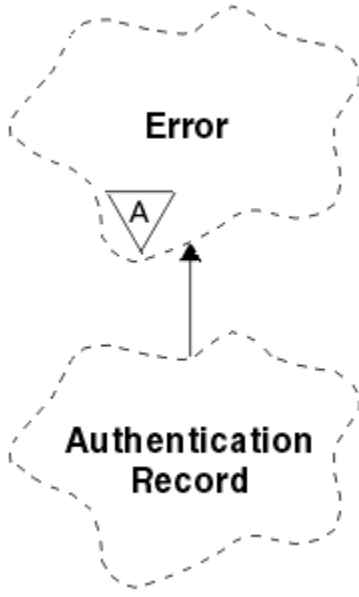
ImqWorkÜstbilgi çapraz başvurusu

ImqAuthenticationRecord C++ sınıfına ilişkin özniteliklere, veri yapılarına ve alanlara çapraz başvuru.

Öznitelik	Veri yapısı	Alan
ileti simgesi	MQWIH	MessageToken
Hizmet Adı	MQWIH	ServiceName
hizmet adımı	MQWIH	ServiceStep

ImqAuthenticationKayıt C++ sınıfı

Bu sınıf, özel SSL istemci bağlantıları için, ImqQueueManager: :connect yönteminin yürütülmesi sırasında kullanılmak üzere bir kimlik doğrulama bilgisi kaydını (MQAIR) sarmalıyor.



Şekil 46. *ImqAuthenticationKayıt* sınıfı

Daha fazla ayrıntı için *ImqQueueManager*: :connect yönteminin açıklamalarına bakın. Bu sınıf z/OS platformunda kullanılamaz.

- “Nesne öznitelikleri” sayfa 1849
- “Oluşturucular” sayfa 1849
- “Nesne yöntemleri (genel)” sayfa 1850
- “Nesne yöntemleri (korunan)” sayfa 1850

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 işleci = (const ImqAuthenticationKayıt ve *hava*);

Eşgörünüm verilerini, var olan eşgörünüm verilerini değiştirerek *hava*' dan kopyalar.

const ImqString & connectionName () const;

Bağlantı adıdeğerini döndürür.

void setConnectionAd (const ImqString & *ad*);

Bağlantı adı' yı ayarlar.

void setConnectionAd (const char * *ad* = 0);

Bağlantı adı' yı ayarlar.

ImqQueueManager * connectionReference () const;

Bağlantı başvurusudeğerini döndürür.

void setConnectionReferansı (ImqQueueManager & *yönetici*);

Bağlantı başvurusunuayarlar.

void setConnectionReferansı (ImqQueueManager * *yönetici* = 0);

Bağlantı başvurusunuayarlar.

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 kaydideğerini döndürür.

const ImqString & password () const;

Paroladeğerini döndürür.

void setPassword (const ImqString & *parola*);

Parolaseçeneğini belirler.

void setPassword (const char * *password* = 0);

Parolaseçeneğini belirler.

ImqAuthenticationKayıt * previousAuthenticationRecord () const;

Önceki kimlik doğrulama kaydınıdöndürür.

MQTHOE tipi () const;

tipdeğerini döndürür.

void setType (const MQHOMET *tip*);

tip' i ayarlar.

const ImqString & userName () const;

Kullanıcı adıdeğerini döndürür.

void setUserAdı (const ImqString & *ad*);

Kullanıcı adı' yı ayarlar.

void setUserAdı (const char * *ad* = 0);

Kullanıcı adı' yı ayarlar.

Nesne yöntemleri (korunan)

void setNextAuthenticationRecord (ImqAuthenticationRecord * *pAir* = 0);

Sonraki kimlik doğrulama kaydınıayarlar.

Dikkat: Bu işlevi yalnızca, kimlik doğrulama kaydı listesini bozmayacağından eminseniz kullanın.

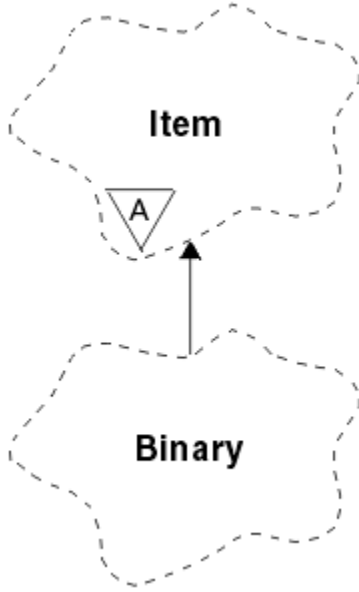
void setPreviousAuthenticationRecord (ImqAuthenticationRecord * *pAir* = 0);

Önceki kimlik doğrulama kaydınıayarlar.

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 kullanılabilecek bir ikili bayt dizisini sarsalıyor. Kolay atama, kopyalama ve karşılaştırma olanağı sağlar.



Şekil 47. ImqBinary sınıfı

- “Nesne öznitelikleri” sayfa 1851
- “Oluşturucular” sayfa 1851
- “Aşırı yüklenmiş ImqItem yöntemleri” sayfa 1851
- “Nesne yöntemleri (genel)” sayfa 1852
- “Nesne yöntemleri (korunan)” sayfa 1852
- “Neden kodları” sayfa 1852

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 & *ikili*);

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 biçim* ' i MQFMT_NONE olarak ayarlar.

Ek ayrıntılar için ImqItem sınıfı yöntem açıklamasına bakın.

sanal ImqBoolean pasteIn (ImqMessage & msg);

Var olan **verileride** değiştirerek, 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);

Copies bytes from *ikili*.

ImqBoolean işleci == (const ImqBinary & binary);

Bu nesneyi *binary* ile karşılaştırır. Eşit değilse FALSE değerini ve tersi durumda TRUE değerini döndürür. Nesneler 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. *uzunluk* aynı zamanda sıfırsa *arabellek* sıfır olabilir. *uzunluk* negatif olmamalıdır. Başarılı olursa TRUE değerini döndürür.

size_t dataLength () const ;

Veri uzunluğu değerini döndürür.

ImqBoolean setDataLength (const size_t length);

Veri uzunluğu nı belirler. 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 uzunluğu sıfır ya da tüm **verilerde** byte 'lar sıfırda TRUE değerini döndürür. Ters durumda FALSE değerini döndürür.

ImqBoolean set (const void * buffer, const size_t length);

Copies *uzunluk* bytes from *arabellek*. Başarılı olursa TRUE değerini döndürür.

Nesne yöntemleri (korunan)

void clear ();

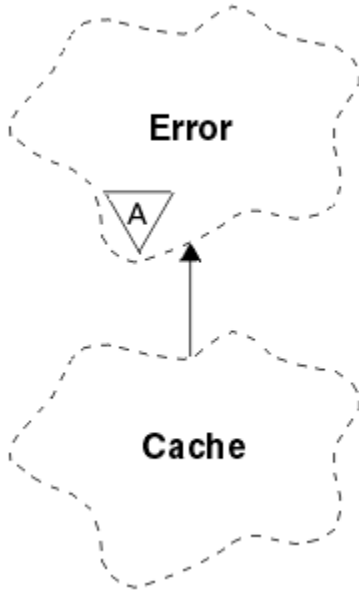
Veri uzunluğu değerini sıfıra indirir.

Neden kodları

- MQRC_NO_BUFFER
- MQRC_STORAGE_NOT_AVAM
- MQRC_INCONSTENT_BIÇIMI

ImqCache C++ sınıfı

Belleğindeki verileri tutmak ya da verileri bellemek için bu sınıfı kullanın.



Şekil 48. ImqCache sınıfı

Belleğindeki verileri tutmak ya da verileri bellemek için bu sınıfı kullanın. Sabit büyüklükte bir bellek arabelleği gösterebilirsiniz ya da sistem, otomatik olarak esnek bir bellek miktarı sağlayabilir. Bu sınıf, “ImqCache çapraz başvuru” sayfa 1835’inde listelenen MQI çağrılarını ilgilidir.

- “Nesne öznitelikleri” sayfa 1853
- “Oluşturucular” sayfa 1854
- “Nesne yöntemleri (genel)” sayfa 1854
- “Neden kodları” sayfa 1855

Nesne öznitelikleri

otomatik arabellek

Arabellek belleğinin 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 büyümeyi 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 görelili 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 bufferLength () 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 uzunluğu değerini döndürür.

ImqBoolean setMessageLength (const size_t length);

İleti uzunluğu nı 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-gerekli);

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 (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* ile artırılır. Bu yöntem başarılı olursa TRUE değerini döndürür.

ImqBoolean resizeBuffer (const size_t length);

Automatic buffer (otomatik arabellek) değeri TRUE (Doğru) olduğunda **arabellek uzunluğunugösterir**. Bu, arabellek belleğindeki yerini yeniden ayırarak elde edilir. Varolan arabellekten **ileti uzunluğu** byte 'a kadar veri, yeni bir veri kaynağı arabelleğinden kopyalanır. Kopyalanan sayı üst sınırı *uzunluk* byte 'tır. **Arabellek göstergesi** değiştirilir. **ileti uzunluğu** ve **veri görelili 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 MQRD_STORAGE_NOT_AVAM ile kullanılabilir durumda olabilir.

ImqBoolean useEmptyBuffer (const char * external-buffer, const size_t length);

Boş bir kullanıcı arabelleğinden birini tanımlar; **arabellek göstergesi** için *dış-arabellek*değerini, **arabellek uzunluğu** değerini *uzunluk*değerine ve **ileti uzunluğu** değerini sıfıra 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 değeri değildir ve *uzunluk* sıfır değildir), bu durumda **otomatik arabellek** değeri FALSE olarak ayarlanır ya da sistem tarafından yönetilen esnek belleğe geri dönmek için kullanılabilir (*dış-arabellek* boş değeri ve *uzunluk* sıfır), bu durumda **otomatik arabellek** TRUE olarak ayarlanır.

ImqBoolean useFullBuffer (const char * externalBuffer, const size_t length);

useEmptyArabellekiçin 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 görelili konumu** is increased by *uzunluk*, and the **ileti uzunluğu** is increased if necessary to ensure that it is no less than the new **veri görelili konumu** value. Bu yöntem başarılı olursa TRUE değerini döndürür.

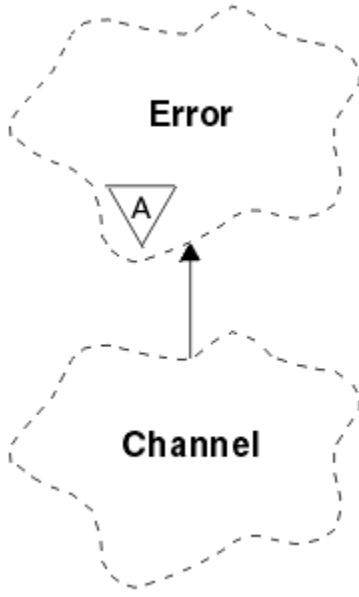
Otomatik arabellek değeri TRUE ise, yeterli miktarda bellek garanti edilir; tersi durumda, en son **veri görelili konumu** , **arabellek uzunluğunuaşmamalıdır**.

Neden kodları

- MQRD_BUFFER_NOT_AUTOMAKRED
- MQRD_DATA_TRUNCATED
- MQRD_INSUFFICIENT_BUFFER
- MQRD_INSUFFICIENT_DATA
- MQRD_NULL_POINTER
- MQRD_STORAGE_NOT_AVAM
- MQRD_ZERO_LENGTH

ImqChannel C++ sınıfı

Bu sınıf, özel istemci bağlantıları için, Yönetici: :connect yöntemi yürütülürken kullanılacak bir kanal tanımlamasını (MQRD) kapsüller.



Şekil 49. ImqChannel sınıfı

Daha fazla ayrıntı için, Yönetici: :connect yöntemi ve [Örnek program HELLO WORLD \(imqwrlld.cpp\)](#) 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/OS üzerinde desteklenmez.

- “Nesne öznitelikleri” sayfa 1856
- “Oluşturucular” sayfa 1857
- “Nesne yöntemleri (genel)” sayfa 1857
- “Neden kodları” sayfa 1861

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 geçerli.

Bağlantı adı

Bağlantının adı. Örneğin, bir anasistem bilgisayarının IP adresi. Başlangıç değeri boş geçerli.

üstbilgi sıkıştırması

Kanal tarafından desteklenen üstbilgi veri sıkıştırma tekniklerinin listesi. İlk değerler, tümü MQCOMPRESS_NOT_AVAM değerine ayarlanır.

heart-beat interval

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 MQKAI_AUTO 'dur.

Yerel adres

Kanala ilişkin yerel iletişim adresi.

ileti uzunluđu üst sınırı

Tek bir iletişimde kanal tarafından desteklenen ileti uzunluđu üst sınırı. Başlangıç değeri 4 194 304 'tür.

ileti sıkıştırma

Kanal tarafından desteklenen ileti veri sıkıştırma tekniklerinin listesi. İlk değeri, tümü MQCOMPRESS_NOT_AVAM değerine ayarlanır.

kip adı

Kipin adı. Başlangıç değeri boş değeri.

Parola

Bağlantı kimlik doğrulaması için sağlanan parola. Başlangıç değeri boş değeri.

çıkış sayısı alma

Giriş çıkış sayısı. Başlangıç değeri sıfır. Bu öznitelik salt okunurdur.

çıkış adlarının alınması

Alma çıkışlarının adları.

kullanıcı verilerini al

Alma çıkışlarıyla ilişkili veriler.

güvenlik çıkış adı

Bağlantının sunucu tarafında çağrılacak güvenlik çıkışa ilişkin ad. Başlangıç değeri boş değeri.

güvenlik kullanıcı verileri

Güvenlik çıkışa geçirilecek veriler. Başlangıç değeri boş değeri.

çıkış sayısı gönder

Çıkış çıkış sayısı. Başlangıç değeri sıfır. Bu öznitelik salt okunurdur.

çıkış adlarının gönderilmesi

Çıkış gönderilerinin adları.

kullanıcı verilerinin gönderilmesi

Gönderme çıkışlarıyla ilişkili veriler.

SSL CipherSpec

SSL ile kullanmak içinCipherSpec .

SSL istemcisi kimlik denetimi tipi

SSL ile kullanmak için istemci kimlik denetimi tipi.

SSL eş adı

SSL ile kullanım için eşdüzey ad.

hareket programı adı

Hareket programının adı. Başlangıç değeri boş değeri.

iletim tipi

Bağlantının iletim tipi. İlk değeri MQXPT_LU62' dir.

kullanıcı kimliği

Yetki için sağlanan bir kullanıcı kimliği. Başlangıç değeri boş değeri.

Oluşturucular**ImqChannel() ;**

Varsayılan oluşturucu.

ImqChannel(const ImqChannel & kanal);

Kopya oluşturucusu.

Nesne yöntemleri (genel)**void işleci = (const ImqChannel & kanal);**

Var olan yönetim ortamı verilerini değiştirerek eşgörünüm verilerini *kanalı*çinden kopyalar.

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 MQORT uzunluk = 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 sayısıdeğerini döndürür.

ImqString receiveExitAd ();
Varsa, **çıkış çıkış** adlarını alışlevinin ilk ögesini döndürür. **Çıkış sayısını al** sıfırsa, boş bir dizgi döndürür.

ImqBoolean receiveExitAdlar (const size_t sayı, ImqString * adlar []);
adlariçinde **çıkış adlarının al** kopyalarının kopyalarını döndürür. Sets any **adlar** in excess of **çıkış sayısını alma** to null strings. Bu yöntem başarılı olursa TRUE değerini döndürür.

ImqBoolean setReceiveExitName(const char * ad = 0);
Çıkış adlarını al seçeneğini tek bir **ad**olarak ayarlar. **ad** boş değerli ya da boş olabilir. **çıkış sayısını alma** 'ı 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 ögesini **adlar**olarak ayarlar. Tek tek **adlar** değerleri boş bırakılmamalı ya da boş bırakılmamalı. **Alma çıkış sayısını 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 ögesini **adlar**olarak ayarlar. Tek tek **adlar** değerleri boş bırakılmamalı ya da boş bırakılmamalı. **Alma çıkış sayısını 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** sıfırsa, 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 sayısını alma** to null strings. Bu yöntem başarılı olursa TRUE değerini döndürür.

ImqBoolean setReceiveUserData(const char * data = 0);
Kullanıcı verilerini al seçeneğini tek bir öğeye (**data**) ayarlar. **data** boş değerli değilse, **alma çıkış sayısını** 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 çıkış adıdeğerini döndürür.

ImqBoolean setSecurityExitName(const char * ad = 0);
Güvenlik çıkış adını ayarlar. Bu yöntem başarılı olursa TRUE değerini döndürür.

ImqString securityUserData () const;
Güvenlik kullanıcı verilerideğerini döndürür.

ImqBoolean setSecurityUserData(const char * data = 0);
Güvenlik kullanıcı verilerini ayarlar. Bu yöntem başarılı olursa TRUE değerini döndürür.

size_t sendExitSayı () const;
çıkış sayısını gönderdeğerini döndürür.

ImqString sendExitAd ();

Varsa, **çıkış adlarını gönder** ' in ilk ögesini döndürür. **Çıkış sayısı gönder** sıfırsa, boş bir dizgi döndürür.

ImqBoolean sendExitAdlar (const size_t count, ImqString * names []);

adları içinde **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 ' i tek bir *ad* olarak 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 ' i *adlar* olarak ayarlar. Tek tek *adlar* değerleri boş bırakılmamalı ya da boş bırakılmamalı. **Çıkış sayısını gönder** ' i *sayı* olarak 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 ' i *adlar* olarak ayarlar. Tek tek *adlar* değerleri boş bırakılmamalı ya da boş bırakılmamalı. **Çıkış sayısını gönder** ' i *sayı* olarak 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** sıfırsa, boş bir dizgi döndürür.

ImqBoolean sendUserData (const size_t count, ImqString * data []);

veri ' de **kullanıcı verilerinin gönderilmesi** öğelerinin kopyalarını döndürür. Sets any *veri* in excess of **çıkış sayısı gönder** to null strings. Bu yöntem başarılı olursa TRUE değerini döndürür.

ImqBoolean setSendUserData(const char * data = 0);

Kullanıcı verilerini gönder seçeneğini tek bir öğeye (*data*) ayarlar. *data* 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;

SSL şifre belirtimini döndürür.

ImqBoolean setSslCipherSpecification(const char * ad = 0);

SSL şifre belirtimini ayarlar. Bu yöntem başarılı olursa TRUE değerini döndürür.

MQHOT sslClientKimlik Doğrulaması () const;

SSL istemcisi kimlik denetimi tipini döndürür.

ImqBoolean setSslClientAuthentication(const MQORA auth = MQSCA_REQUIREMOND);

SSL istemcisi kimlik denetimi tipini ayarlar. Bu yöntem başarılı olursa TRUE değerini döndürür.

ImqString sslPeerAd () const;

SSL eş adını döndürür.

ImqBoolean setSslPeerName(const char * ad = 0);

SSL 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 MQORT tip = MQXPT_LU62);
İletim tipini belirler. Bu yöntem başarılı olursa TRUE değerini döndürür.

ImqString userId() const;
Kullanıcı kimliği değerini döndürür.

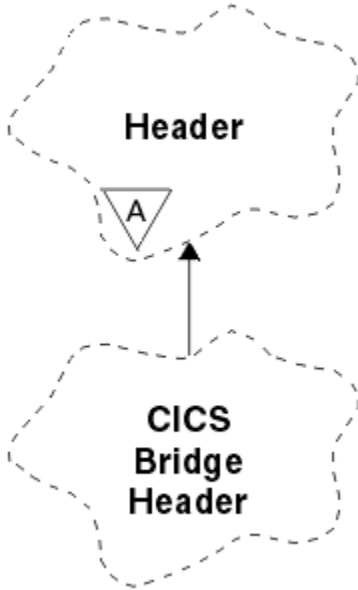
ImqBoolean setUserId (const char * id = 0);
Kullanıcı kimliği' yi ayarlar. Bu yöntem başarılı olursa TRUE değerini döndürür.

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 50. 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 1861](#)
- [“Oluşturucular” sayfa 1864](#)
- [“Aşırı yüklenmiş ImqItem yöntemleri” sayfa 1864](#)
- [“Nesne yöntemleri \(genel\)” sayfa 1864](#)
- [“Nesne verileri \(korunan\)” sayfa 1866](#)
- [“Neden kodları” sayfa 1866](#)
- [“Dönüş kodları” sayfa 1866](#)

Nesne öznitelikleri

ADS tanımlayıcısı

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 225 sayfa 1866](#) .

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 225 sayfa 1866](#) .

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 225 sayfa 1866](#) .

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 süresi

CICS bridge olanağı yayın süresi.

gibi bir tesis

Uçbirim öykünülen özneliliği. Alan, MQ_FACILITY_LIKE_LENGTH uzunluğunda olmalıdır.

tesis simgesi

BVT simge değeri. Alan, MQ_FACILITY_LENGTH uzunluğunda olmalıdır. İlk değer MQCFAC_NONE olur.

function

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 225 sayfa 1866](#) .

Aşağıdaki ek değerler, **işlev** bir IBM MQ arama adı içerdiğinde mümkündür:

- MQCFUNC_MQCONN
- MQCFUNC_MQGET
- MQCFUNC_MQINQ
- MQCFUNC_NONE

- MQCFUNC_MQAC
- MQCFUNC_PUT
- MQCFUNC_MQPUT1

bekleme aralığı al

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 verileri 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

görevin 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 değer MQCUOWC_ONLY değeridir. Aşağıdaki ek değerler kullanılabilir:

- İLK MQCUOWC_FIRST
- MQCUOWC_ORTA
- MQCUOWC_SON
- YALNIZCIK
- MQCUOWC_COMMIT
- MQCUOWC_BACKOUT
- MQCUOWC_CONTINY

sürüm

MQCIH sürüm numarası. İlk değer MQCIH_VERSION_2' dir. Desteklenen diğer değer MQCIH_VERSION_1' dir.

Oluşturucular

ImqCICSBridgeHeader();

Varsayılan oluşturucu.

ImqCICSBridgeHeader(const ImqCICSBridgeHeader & üstbilgi);

Kopya oluşturucusu.

Aşırı yüklenmiş ImqItem yöntemleri

sanal ImqBoolean copyOut(ImqMessage & msg);

Bir MQCIH veri yapısını başlangıçta ileti arabelleğine ekler, var olan ileti verilerini daha ileriye taşıyarak bu ileti biçimini MQFMT_CICSolarak ayarlar.

Daha fazla ayrıntı için üst sınıf yöntemi açıklamasına bakın.

sanal ImqBoolean pasteIn(ImqMessage & msg);

İleti arabelleğinden bir MQCIH 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_CICSolmalıdır.

Daha fazla ayrıntı için üst sınıf yöntemi açıklamasına bakın.

Nesne yöntemleri (genel)

void işleci = (const ImqCICSBridgeHeader & üstbilgi);

Eşgörünüm verilerini, var olan eşgörünüm verilerini değiştirerek, üstbilgisinden kopyalar.

MQUQT ADSDescriptor () const;

ADS tanımlayıcısı' nın bir kopyasını döndürür.

void setADSDescriptor(const MQORA tanımlayıcı = MQCADSD_NONE);

ADS tanımlayıcısı' ı ayarlar.

ImqString attentionIdentifier() const;

MQ_ATTENTION_ID_LENGTH uzunluğuna sahip sondaki boşluklarla doldurulmuş olan **uyarı tanıtıcısı** kopyasını döndürür.

void setAttentionTanıtıcısı (const char * data = 0);

Sondaki boşluklarla doldurulan **dikkat tanıtıcısını** MQ_ATTENTION_ID_LENGTH uzunluğuna ayarlar. veri belirtilmezse, **uyarı tanıtıcısı** ilk değerini ilk değerine döndürür.

ImqString kimlik doğrulayıcı () const;

MQ_AUTHENTICATOR_LENGTH uzunluğuna sahip sondaki boşluklarla doldurulan **kimlik doğrulayıcı** nın bir kopyasını döndürür.

void setAuthenticator(const char * data = 0);

Sondaki boşluklarla doldurulan **kimlik doğrulayıcıyı** MQ_authenticator_length uzunluğuna ayarlar. veri belirtilmezse, **kimlik doğrulayıcı** ilk değerini ilk değerine döndürür.

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** nın bir kopyasını döndürür.

void setBridgeCancelCode(const char * data = 0);

İzleyen boşluklarla doldurulan **köprü iptal kodunu** MQ_CANCEL_CODE_LENGTH uzunluğuna kadar ayarlar. veri belirtilmezse, **köprü iptal kodu** değerini ilk değer olarak ilk durumuna getirir.

MQlong bridgeCompletionKodu () const;
Köprü tamamlanma kodu' nun bir kopyasını döndürür.

MQHOT bridgeErrorGörelİ Konumu () const;
Köprü hata görelİ konumu' nin bir kopyasını döndürür.

MQHOT bridgeReasonKodu () const;
Köprü neden kodu' nun bir kopyasını döndürür.

MQlong bridgeReturnKodu () const;
Köprü dönüş kodudeğerini döndürür.

MQlong conversationalTask() const;
conversasyonel tasköğesinin bir kopyasını döndürür.

void setConversationalGörev (const MQWOLD task = MQCCT_NO);
etkileşimli göreviayarlar.

MQLONG cursorPosition() const;
İmleç konumu' nun bir kopyasını döndürür.

void setCursorKonumu (const MQORT konum = 0);
İmleç konumunuayarlar.

MQUZE facilityKeepTime () const;
Tesise tutma süresi' nin bir kopyasını döndürür.

void setFacilityKeepTime(const MQHOMET 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 simgesiöğesinin bir kopyasını döndürür.

ImqBoolean setFacilitySimgesi (const ImqBinary & simge);
tesis simgesi' ı ayarlar. *belirteç* ' in **veri uzunluğu** sıfır ya da MQ_FACILITY_LENGTH olmalıdır. Başarılı olursa TRUE değerini döndürür.

void setFacilityToken (const MQBYTE8 simge = 0);
tesis simgesi' ı ayarlar. *simge* sıfır olabilir; bu, MQCFAC_NONE belirtilerek aynıdır. *simge* sıfır değilse, 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 intervalöğesinin bir kopyasını döndürür.

void setGetWaitInterval(const MQUZE aralık = MQCGWI_DEFA
Bekleme süresi aralığını al seçeneğini belirler.

MQLONG linkType() const;
link type' in bir kopyasını döndürür.

void setLinkTip (const MQORT tip = MQCLT_PROPROGRA;
Bağlantı tipiniayarlar.

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 MQORT uzunluk = MQCODL_AS_INPUT);

Çıkış veri uzunluğunubelirler.

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* belirtilmezse, **yanıtlama biçimi** ilk değerini ilk değerine döndürür.

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* belirtilmezse, **başlatma kodu** ilk değerini ilk değerine döndürür.

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* belirtilmezse, **işlem tanıtıcısı** ilk değerini ilk değerine döndürür.

MQLONG UOWControl () const;

UOW denetimi' nin bir kopyasını döndürür.

void setUOWControl(const MQlong control = MQCUOWC_ONLY);

UOW denetimi' ni ayarlar.

MQUZE sürümü () const;

version numarasını döndürür.

ImqBoolean setVersion(const MQORT version = MQCIH_VERSION_2);

sürüm numarasını ayarlar. Başarılı olursa TRUE değerini döndürür.

Nesne verileri (korunan)

MQUZE olVersion

opcihi 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

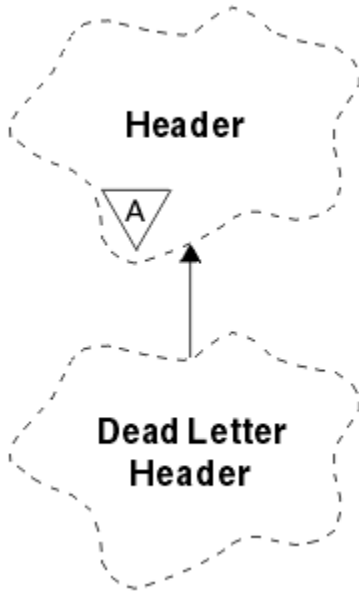
Çizelge 225. ImqCICSBridgeHeader sınıf dönüş kodları				
Dönüş Kodu	İşlev	CompCode	Neden	Olağandışı Son Kod
MQCRC_OK				

Çizelge 225. ImqCICSBridgeHeader sınıf dönüş kodları (devamı var)

Dönüş Kodu	İşlev	CompCode	Neden	Olağandışı Son Kod
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 51. 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 1868
- “Oluşturucular” sayfa 1868
- “Aşırı yüklenmiş ImqItem yöntemleri” sayfa 1868
- “Nesne yöntemleri (genel)” sayfa 1868
- “Nesne verileri (korunan)” sayfa 1869

- [“Neden kodları” sayfa 1869](#)

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.

koyma tarihi

İletinin, ölü-mektup kuyruğuna konduğu tarih. Tarih, MQ_PUT_DATE_LENGTH uzunluğunda bir dizilimdir. Başlangıç değeri, boş değerli bir dizgidir.

koyma süresi

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 & üstbilgi);

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* **biçimi** olarak MQFMT_DEAD_LETTER_HEADER olarak ayarlanır.

See the ImqHeader class method description on page [“ImqHeader C++ sınıfı” sayfa 1875](#) for further details.

sanal ImqBoolean pasteIn (ImqMessage & msg);

İleti arabelleğinden bir MQDLH veri yapısını okur.

Başarılı olmak için, ImqMessage **biçim** MQFMT_DEAD_LETTER_HEADER olmalıdır.

See the ImqHeader class method description on page [“ImqHeader C++ sınıfı” sayfa 1875](#) 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-harflik **neden kodu** değerini döndürür.

void setDeadLetterReasonKodu (const MQlong neden);

Ölü harf **neden kodu**' yı belirler.

ImqString destinationQueueManagerName () const ;

Returns the **hedef kuyruk yöneticisi adı**, stripped of any trailing blanks.

void setDestinationQueueManagerAd (const char * ad);

Hedef kuyruk yöneticisi adını belirler. MQ_Q_MGR_NAME_LENGTH ' den (48 karakter) daha uzun veri keser.

ImqString destinationQueueAd () const ;

Sondaki boşluklardan oluşan, **hedef kuyruk adı** kopyasının bir kopyasını döndürür.

void setDestinationQueueName (const char * ad);

Hedef kuyruk adı ' na ayarlanır. MQ_Q_NAME_LENGTH ' den (48 karakter) daha uzun veri keser.

ImqString putApplicationAd () const ;

Sondaki boşluklardan oluşan, **uygulama adını girin** kopyasının bir kopyasını döndürür.

void setPutApplicationName (const char * ad = 0);

Uygulama adını koy ögesini ayarlar. MQ_PUT_APPL_NAME_LENGTH ' den (28 karakter) daha uzun veri keser.

MQLONG putApplicationTipi () const ;

put application type değerini döndürür.

void setPutApplicationType (const MQlong type = MQAT_NO_CONTEXT);

Uygulama tipini indir seçeneğini belirler.

ImqString putDate () const ;

Sondaki boşluklardan oluşan, **bitiş tarihi** kopyasının bir kopyasını döndürür.

void setPutDate (const char * date = 0);

put date (put tarihi) seçeneğini belirler. MQ_PUT_DATE_LENGTH (8 karakter) değerinden daha uzun veri keser.

ImqString putTime () const ;

Sondaki boşluklardan oluşan, **sürme zamanı** kopyasının bir kopyasını döndürür.

void setPutTime (const char * time = 0);

put time (put time) ögesini ayarlar. MQ_PUT_TIME_LENGTH (8 karakter) uzunluğundan daha uzun verileri keser.

Nesne verileri (korunan)

MQDLH omqdlh

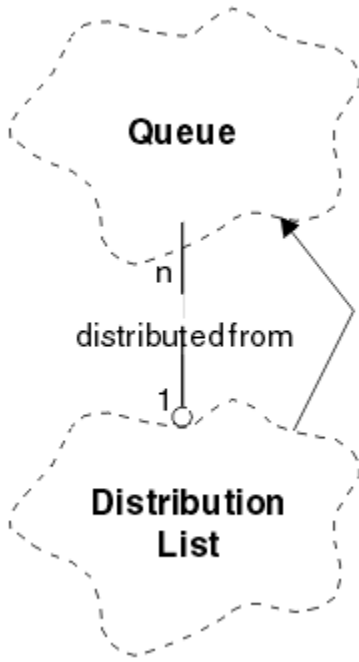
MQDLH veri yapısı.

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 52. *ImqDistributionListe* Sınıfı

- “Nesne öznitelikleri” sayfa 1870
- “Oluşturucular” sayfa 1870
- “Nesne yöntemleri (genel)” sayfa 1870
- “Nesne yöntemleri (korunan)” sayfa 1871

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 nesneler 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 kuyrukdeğerini döndürür.

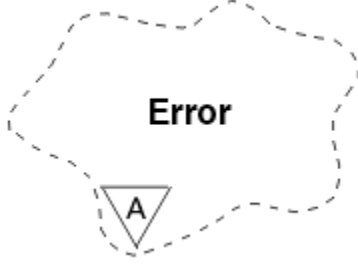
Nesne yöntemleri (korunan)

void setFirstDistributedQueue (* queue = 0);

İlk dağıtılmış kuyruğu ayarlar.

ImqError C++ sınıfı

Bu soyut sınıf, bir nesneyle ilişkili hatalarla ilgili bilgi sağlar.



Şekil 53. ImqError sınıfı

- “Nesne öznitelikleri” sayfa 1871
- “Oluşturucular” sayfa 1871
- “Nesne yöntemleri (genel)” sayfa 1871
- “Nesne yöntemleri (korunan)” sayfa 1872
- “Neden kodları” sayfa 1872

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);

Var olan eşgörünüm verilerini değiştirerek eşgörünüm verilerini *error* adresinden kopyalar.

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 * *işaretçi*, 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);

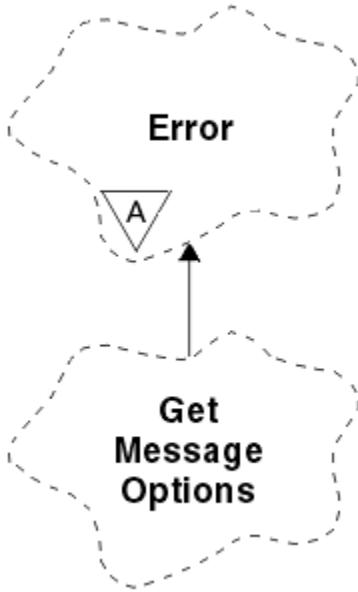
Neden kodu' yı belirler.

Neden kodları

- MQRC_BUFFER_ERROR

ImqGetMessageOptions C++ sınıfı

Bu sınıf MQGMO veri yapısını sarsalıyor



Şekil 54. ImqGetMessageOptions sınıfı

- [“Nesne öznitelikleri” sayfa 1872](#)
- [“Oluşturucular” sayfa 1874](#)
- [“Nesne yöntemleri \(genel\)” sayfa 1874](#)
- [“Nesne yöntemleri \(korunan\)” sayfa 1875](#)
- [“Nesne verileri \(korunan\)” sayfa 1875](#)
- [“Neden kodları” sayfa 1875](#)

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

eşleştirme 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

çözülen 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.

segmentasyon

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

eşitleme noktası katılımı

İletilerin syncpoint denetimi altında alındığında TRUE (Doğru) değeri alınır.

Bekleme Aralığı

Bir iletinin gelmesi için uygun bir ileti beklenirken, sınıfın **alma** yöntemi duraksaması gereken süre (zaten varsa). İlk değer sıfır (sıfır), belirsiz bir bekleme durumu etkiler. MQWI_UNUSISINI ek değeri mümkün. Bu öznitelik, **seçenekler** MQGMO_WEKE içermediği sürece yoksayılır.

Oluşturucular

ImqGetMessageOptions();

Varsayılan oluşturucu.

ImqGetMessageOptions(const ImqGetMessageOptions & gmo);

Kopya oluşturucusu.

Nesne yöntemleri (genel)

void operator = (const ImqGetMessageOptions & gmo);

Eşgörünüm verilerini, var olan eşgörünüm verilerini değiştirerek gmo adresinden kopyalar.

MQCHAR groupStatus () const ;

Grup durumudeğerini döndürür.

void setGroupStatus (const MQCHAR status);

Grup durumunuayarlar.

MQlong matchOptions () const ;

eşleşme seçeneklerideğerini döndürür.

void setMatchSeçenekleri (const MQlong seçenekler);

eşleşme seçenekleriniayarlar.

ImqBinary messageToken() const;

İleti simgesideğerini döndürür.

ImqBoolean setMessageSimgesi (const ImqBinary & simge);

İleti belirtecini belirler. The **veri uzunluğu** of *simge* must be either zero or MQ_MSG_TOKEN_LENGTH. Bu yöntem başarılı olursa TRUE değerini döndürür.

void setMessageSimgesi (const MQBYTE16 simge = 0);

İleti belirtecini belirler. *simge* sıfır olabilir; bu, 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.

MQLONG seçenekler () const ;

seçeneklerdeğerini döndürür.

void setOptions (const MQlong seçenekler);

syncpoint katılımı değeri de dahil olmak üzere **seçenekleri**ayarlar.

ImqString resolvedQueueAd () const ;

Çözümlenen kuyruk adı' nun bir kopyasını döndürür.

MQHOT returnedLength() const;

döndürülen uzunlukdeğerini döndürür.

MQCHAR kesimlere ayırma () const ;

segmentasyon' ı döndürür.

void setSegmentation (const MQCHAR değer);

segmentasyon' ı ayarlar.

MQCHAR segmentStatus () const ;
segment statusdeğerini döndürür.

void setSegmentStatus (const MQCHAR status);
bölüm durumunuayarlar.

ImqBoolean syncPointKatılım () const ;
seçenekler değeri MQGMO_SYNCPOINT ya da MQGMO_SYNCPOINT_IF_PERSISTENT ise TRUE olan **syncpoint katılımı** değerini döndürür.

void setSyncPointParticipation (const ImqBoolean sync);
syncpoint katılım değerini ayarlar. *eşitleme* TRUE ise, **seçenekler** ' i MQGMO_SYNCPOINT ögesini içerecek şekilde ve hem MQGMO_NO_SYNCPOINT hem de MQGMO_SYNCPOINT_IF_PERSISTENT ögesini dışlamak için değiştirir. *eşitleme* FALSE ise, **seçenekler** 'i MQGMO_NO_SYNCPOINT' ı içerecek şekilde ve hem MQGMO_SYNCPOINT hem de MQGMO_SYNCPOINT_IF_PERSISTENT ögesini dışlamak için değiştirir.

MQUZE waitInterval () const ;
Bekleme aralığıdeğerini döndürür.

void setWaitInterval (const MQLong interval);
Bekleme aralığı' yı ayarlar.

Nesne yöntemleri (korunan)

static void setVersionDestekleniyor (const MQHOTR);
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.

MQLONG olVersion

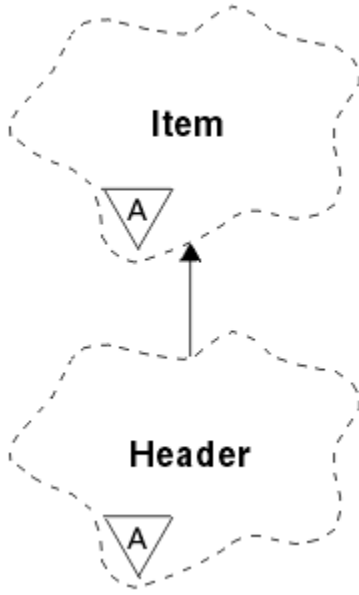
*opgmo*tarafından adreslenen MQGMO veri yapısının sürüm numarası.

Neden kodları

- MQRC_BINARY_DATA_LENGTH_ERROR

ImqHeader C++ sınıfı

Bu soyut sınıf, MQDLH veri yapısının ortak özelliklerini kapsüller.



Şekil 55. *ImqHeader* sınıfı

- “Nesne öznitelikleri” sayfa 1876
- “Oluşturucular” sayfa 1876
- “Nesne yöntemleri (genel)” sayfa 1876

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 nesneleri için
- MQRMHF_SON ImqReferenceüstbilgi sınıfı nesneleri için
- MQCIH_NONE, ImqCICSBridgeHeader sınıfının nesneleri için
- MQWIIH_NONE, ImqWorküstbilgi sınıfı nesneleri için

Oluşturucular

ImqHeader();

Varsayılan oluşturucu.

ImqHeader(const ImqHeader & header);

Kopya oluşturucusu.

Nesne yöntemleri (genel)

void operator = (const ImqHeader & header);

Eşgörünüm verilerini, var olan eşgörünüm verilerini değiştirerek *üstbilgisinden* 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.

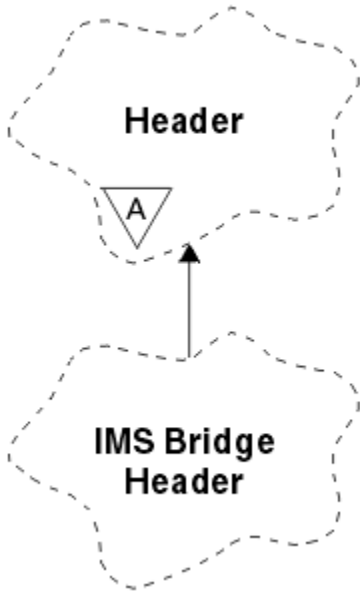
virtual void setFormat (const char * ad = 0);
Sondaki boşluklar içeren 8 karaktere kadar doldurulmuş **biçim'** i ayarlar.

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 56. 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 **encoding** değerleri varsayılan değerlere sahip olmalı ve başka bir değer ayarlanmamalıdır.

- [“Nesne öznitelikleri” sayfa 1878](#)
- [“Oluşturucular” sayfa 1878](#)
- [“Aşırı yüklenmiş ImqItem yöntemleri” sayfa 1878](#)
- [“Nesne yöntemleri \(genel\)” sayfa 1878](#)
- [“Nesne verileri \(korunan\)” sayfa 1879](#)
- [“Neden kodları” sayfa 1879](#)

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 değeri

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 & üstbilgi);

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 biçim* ' i 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 **kodlama** değeri MQENC_NATIVE olmalıdır. İletileri MQGMO_CONVERT ile MQENC_NATIVE ile alın.

Başarılı olmak için, ImqMessage **format** , MQFMT_IMS olmalıdır.

Ek ayrıntılar için üst sınıf yöntemi açıklamasına bakın.

Nesne yöntemleri (genel)

void operator = (const ImqIMSBridgeHeader & header);

Eşgörünüm verilerini, var olan eşgörünüm verilerini değiştirerek *üstbilgisinden* kopyalar.

ImqString kimlik doğrulayıcı () const ;

MQ_AUTHENTICATOR_LENGTH uzunluğuna sahip sondaki boşluklarla doldurulan **kimlik doğrulayıcı** 'nın bir kopyasını döndürür.

void setAuthenticator (const char * *ad*);

Kimlik doğrulayıcıyı ayarlar.

MQCHAR commitMode () const ;

Kesinleştirme kipideğerini döndürür.

void setCommitKipi (const MQCHAR *mode*);

Kesinleştirme kipi' ne ayarlar.

ImqString logicalTerminalGeçersiz Kılma () const ;

Mantıksal uçbirim geçersiz kılma değeri' nin bir kopyasını döndürür.

void setLogicalTerminalOverride (const char * *override*);

mantıksal uçbirim geçersiz kılma' ı ayarlar.

ImqString messageFormatServicesMapAd () 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-biçimi** ögesinin bir kopyasını döndürür.

void setReplyToFormat (const char * *biçim*);

Sondaki boşluklarla doldurulan **yanıt-biçimi**' yi MQ_FORMAT_LENGTH uzunluğuna ayarlar.

MQCHAR securityScope () const ;

Güvenlik kapsamı değerini döndürür.

void setSecurityKapsam (const MQCHAR *scope*);

Güvenlik kapsamını ayarlar.

ImqBinary transactionInstanceTanıtıcısı () const ;

işlem eşgörünümü tanıtıcısı' nın bir kopyasını döndürür.

ImqBoolean setTransactionInstanceId (const ImqBinary & *id*);

işlem eşgörünümü tanıtıcısı' ı ayarlar. The **veri uzunluğu** of *simge* must be either zero or MQ_TRAN_INSTANCE_ID_LENGTH. Bu yöntem başarılı olursa TRUE değerini döndürür.

void setTransactionInstanceId (const MQBYTE16 *id* = 0);

işlem eşgörünümü tanıtıcısı' ı ayarlar. *id* sıfır (sıfır) olabilir; bu, MQITII_NONE belirtilerek aynıdır.

id sıfır değilse, 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 durumudeğerini döndürür.

void setTransactionState (const MQCHAR *state*);

Hareket durumunu ayarlar.

Nesne verileri (korunan)

MQIIH omqiih

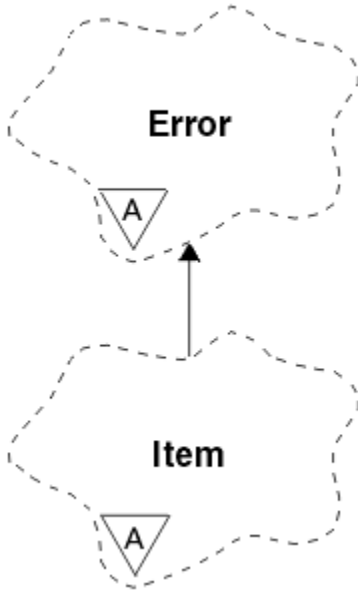
MQIIH veri yapısı.

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 ögeyi (belki de birden çok) temsil eder.



Şekil 57. ImqItem sınıfı

Öğeler, bir ileti arabelleğindeki birleştirilir. Her uzmanlaşma, 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 1880
- “Oluşturucular” sayfa 1880
- “Sınıf yöntemleri (genel)” sayfa 1880
- “Nesne yöntemleri (genel)” sayfa 1881
- “Neden kodları” sayfa 1881

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 ögesinin **yapı tanıtıcısı** , *yapı-tanıtıcısı-test* ile aynıysa TRUE değerini döndürür. Sonraki öge, 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);

Var olan eşgörünüm verilerini değiştirerek eşgörünüm verilerini *item*den kopyalar.

sanal 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 **kodlama** , 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.

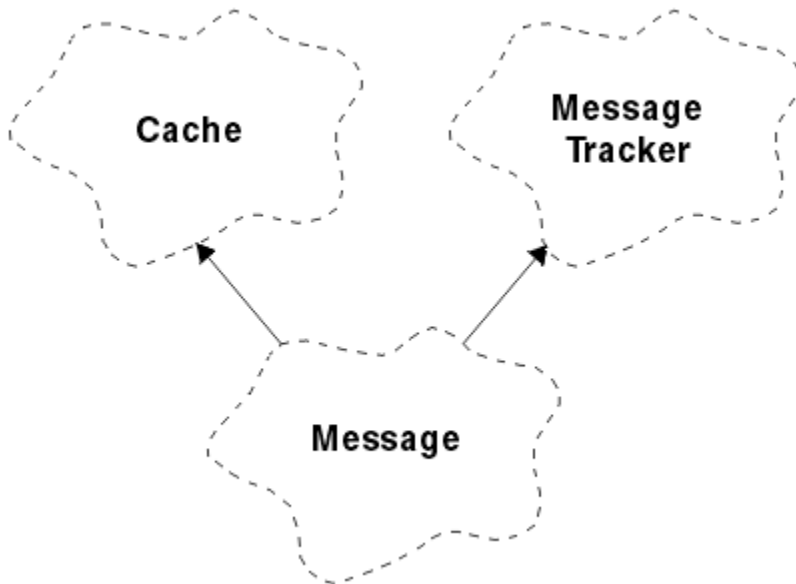
Belirli bir alt sınıfla çalışmak için bu yöntemi geçersiz kılın.

Neden kodları

- MQRCENCODING_ERROR
- MQRC_STRUC_ID_ERROR
- MQRC_INCONSTENT_BIÇIMI
- 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 58. ImqMessage sınıfı

- [“Nesne öznitelikleri” sayfa 1882](#)
- [“Oluşturucular” sayfa 1885](#)
- [“Nesne yöntemleri \(genel\)” sayfa 1885](#)
- [“Nesne yöntemleri \(korunan\)” sayfa 1887](#)
- [“Nesne verileri \(korunan\)” sayfa 1887](#)

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 909](#).

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 419 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 430 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_KöPRÜSü
- 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 \(MQUZE\)” sayfa 435 alanına](#) bakın.

koyma tarihi

Bir iletinin yerleştirdiği tarih. Başlangıç değeri, boş değerli bir dizgidir.

koyma süresi

Bir iletinin yerleştirdiği saat. Başlangıç değeri, boş değerli bir dizgidir.

yanıtlama kuyruğu yöneticisi adı

Herhangi bir yanıtın gönderileceği kuyruk yöneticisinin adı. Başlangıç değeri, boş değerli bir dizgidir.

yanıtlama 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 değerleri gösterir.

Sıra Numarası

Grup içindeki bir iletiyi tanımlayan sıra bilgileri. Başlangıç değeri bir değerdir.

toplam ileti uzunluğu

En son ileti okuma girişimi sırasında kullanılabilir olan bayt sayısı. Son ileti kısaltıldıysa, bu sayı ImqCache **ileti uzunluğu** değerinden büyük ya da kesme gerçekleşmiş olduğu için son ileti okunmuyorsa daha büyük olur. Bu öznitelik salt okunurdur. Başlangıç değeri sıfır.

Bu öznitelik, kesilmiş iletiler içeren herhangi bir durumda yararlı olabilir.

kullanıcı kimliği

Bir iletiyle ilişkili kullanıcı kimliği. Başlangıç değeri, boş değerli bir dizgidir.

Oluşturucular

ImqMessage();

Varsayılan oluşturucu.

ImqMessage(const ImqMessage & msg);

Kopya oluşturucusu. Ayrıntılar için **operator =** yöntemine bakın.

Nesne yöntemleri (genel)

void operator = (const ImqMessage & msg);

MQMD ' yi ve ileti verilerini msgadresinden kopyalar. Kullanıcı tarafından bu nesne için bir arabellek sağlandıysa, kopyalanan veri miktarı, kullanılabilir arabellek boyuyla sınırlandırılır. Ters durumda, sistem, kopyalanan veriler için yeterli büyüklükte bir arabellek sağlanmasını sağlar.

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) ögesinin bir kopyasını döndürür.

void setApplicationOriginData (const char * data = 0);

Uygulama kaynağı verileriniayarlar.

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-ile-test* ile aynıysa TRUE değerini döndürür.

void setFormat (const char * *ad* = 0);
Sondaki boşluklar içeren sekiz karaktere kadar doldurulmuş **biçim'** i ayarlar.

MQUZE messageFlags () const ;
İleti işaretleri döndürür.

void setMessageFlags (const MQlong flags);
İleti işaretleri' i ayarlar.

MQUZE messageType () const ;
İleti tipini döndürür.

void setMessageType (const MQlong tip);
İleti tipini ayarlar.

MQUZE göreli konumu () const ;
göreli konum değerini döndürür.

void setOffset (const MQlong göreli konum);
Göreli Konum' u ayarlar.

MQLONG originalLength () const ;
Özgün uzunluk değerini döndürür.

void setOriginalLength (const MQlong length);
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 & öğe);

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 setSequenceNumara (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 & öğe);

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)

MQMD1 omqmd

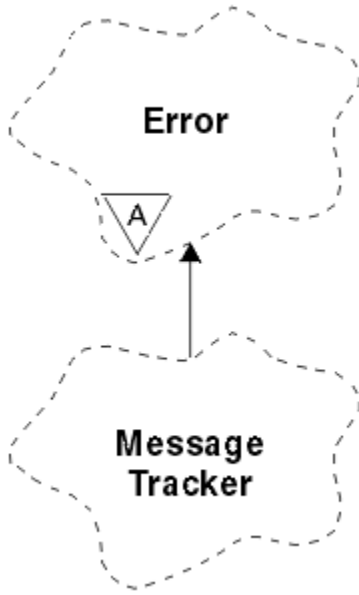
(Yalnızca IBM MQ for z/OS .) MQMD veri yapısı.

MQMD2 omqmd

(z/OS'deki platformlar.) MQMD veri yapısı.

ImqMessageTracker C++ sınıfı

Bu sınıf, herhangi bir nesneyle ilişkilendirilebilecek bir ImqMessage ya da ImqQueue nesnesinin özniteliklerini sarsalır.



Şekil 59. ImqMessageİzleyici sınıfı

Bu sınıf, [“ImqMessageİzleyici çapraz başvurusu”](#) sayfa 1840’inde listelenen MQI çağrıları ile ilgilidir.

- [“Nesne öznitelikleri”](#) sayfa 1888
- [“Oluşturucular”](#) sayfa 1889
- [“Nesne yöntemleri \(genel\)”](#) sayfa 1889
- [“Neden kodları”](#) sayfa 1890

Nesne öznitelikleri

muhasabe 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 416 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 & izleyici);

Kopya oluşturucusu. Ayrıntılar için **operator =** yöntemine bakın.

Nesne yöntemleri (genel)

void operator = (const ImqMessageTracker & tracker);

Eşgörünüm verilerini, var olan eşgörünüm verilerini değiştirerek *izleyici* kopyasından kopyalar.

ImqBinary accountingToken () const ;

muhasabe simgesi ögesinin bir kopyasını döndürür.

ImqBoolean setAccountingSimgesi (const ImqBinary & simge);

muhasabe belirteci'yi ayarlar. *simge* çizelgesinin **veri uzunluğu** değeri 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, MQACT_NONE belirtilerek aynı olur. *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 & *belirteç*);

İlinti tanıtıcısını ayarlar. *simge* çizelgesinin **veri uzunluğu** değeri 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 değeriyle belirtme ile aynıdır. *id* sıfır değilse, bu değer, MQ_COREL_ID_LENGTH of binary verilerini ele almalıdır. 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 & *belirteç*);

Grup tanıtıcısı'yi ayarlar. *simge* çizelgesinin **veri uzunluğu** değeri 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 & *belirteç*);

İleti tanıtıcısını belirler. *simgesi* için **veri uzunluğu** sıfır ya da MQ_MSG_ID_LENGTH olmalıdır. Bu yöntem başarılı olursa TRUE değerini döndürür.

void setMessageId (const MQBYTE24 *id* = 0);

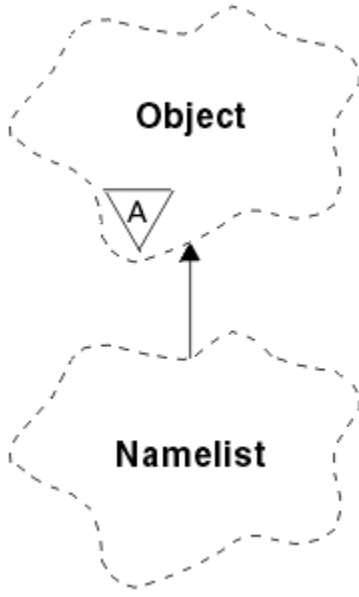
İleti tanıtıcısını belirler. *id* sıfır (sıfır) olabilir; bu, MQMI_NONE belirtme ile aynıdır. *id* sıfır değilse, bu, ikili verilerin MQ_MSG_ID_LENGTH byte 'ları ile adreslenmelidir. MQMI_NONE gibi önceden tanımlanmış değerleri kullanırken, imza eşleştirmesini sağlamak için bir tip dönüştürme yapmanız gerekebilir; örneğin (MQBYTE *) MQMI_NONE.

Neden kodları

- MQRC_BINARY_DATA_LENGTH_ERROR

ImqNamelist C++ sınıfı

Bu sınıf bir ad listesini sarsalıyor.



Şekil 60. *ImqNamelist* sınıfı

Bu sınıf, [“ImqNamelist çapraz başvurusu” sayfa 1840](#) içinde listelenen MQI çağrıları ile ilgilidir.

- [“Nesne öznitelikleri” sayfa 1891](#)
- [“Oluşturucular” sayfa 1891](#)
- [“Nesne yöntemleri \(genel\)” sayfa 1891](#)
- [“Neden kodları” sayfa 1892](#)

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 işleci = (const ImqNamelist & list);

Eşgörünüm verilerini, var olan eşgörünüm verilerini değiştirerek *list*(liste) içinden kopyalar. **ImqObject open status** değeri false (yanlış) değeridir.

ImqBoolean nameCount(MQBOLD & sayı);

Ad sayısı'nın 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 MQORT *dizin*);

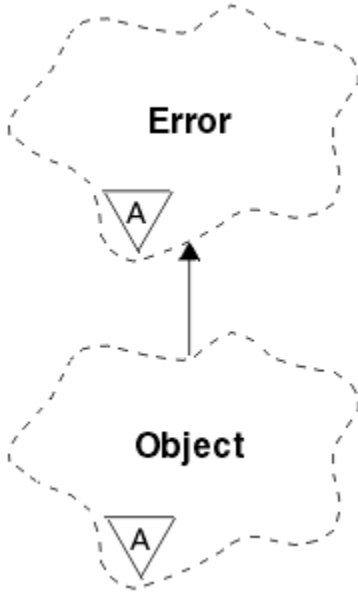
Herhangi bir hata belirtisi göstermeksizin, sıfır tabanlı *dizin*le **ad listesi adlarından** birini döndürür.

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 61. ImqObject sınıfı

Bu sınıf, [“ImqObject çapraz başvurusu”](#) sayfa 1840’inde listelenen MQI çağrılarını ile ilgilidir.

- [“Sınıf öznitelikleri”](#) sayfa 1892
- [“Nesne öznitelikleri”](#) sayfa 1893
- [“Oluşturucular”](#) sayfa 1894
- [“Sınıf yöntemleri \(genel\)”](#) sayfa 1894
- [“Nesne yöntemleri \(genel\)”](#) sayfa 1894
- [“Nesne yöntemleri \(korunan\)”](#) sayfa 1896
- [“Nesne verileri \(korunan\)”](#) sayfa 1897
- [“Neden kodları”](#) sayfa 1897
-

Sınıf öznitelikleri

davranış

Örtülü açıklığın davranışını denetler.

IMQ_IMPL_ASK (8L)

Örtülü açma işlemine izin verilir. Bu varsayılandır.

Nesne öznitelikleri

değiştirme tarihi

Değişim randevusu. Bu öznitelik salt okunurdur.

değiştirme 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 ' dir.

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, adlandırılmış bir kuyruk için kuyruk yöneticisini (büyük olasılıkla uzak) tanıtan **kuyruk yöneticisi adı** ile karıştırmayın.

Açıklama

Kuyruk yöneticisi, kuyruk, ad listesi ya da süreç için açıklayıcı ad (en çok 64 karakter). Bu öznitelik salt okunurdur.

NAME

Kuyruk yöneticisi, kuyruk, ad listesi ya da işlem için ad (en çok 48 karakter). Başlangıç değeri, boş değerli bir dizgidir. Model kuyruğunun adı, sonuçtaki dinamik kuyruk adının **açılmasından** sonra değişir.

Not: ImqQueueYöneticisi, varsayılan kuyruk yöneticisini temsil eden boş değerli bir ada sahip olabilir. Başarılı bir **açık**olduktan sonra ad, 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 nesne ile aynı **bağlantı başvurusu** olabilir. 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 **open** yöntemini kullanmayın. IBM MQ , **açık seçenekleri** otomatik olarak ayarlar ve otomatik olarak açılır, yeniden açar ve nesneleri gerektiği gibi kapatır. Bu, gereksiz yeniden açma işlemleriyle sonuçlanabilir; IBM MQ , **openFor** yöntemini kullanır ve bu da yalnızca artımsal olarak **açık seçenekleri** ekler.
2. Bir MQI çağrısına neden olan yöntemleri kullanmadan önce **açık seçenekleri** ayarlayın (bkz. [“C++ ve MQI çapraz başvurusu” sayfa 1834](#)). Bu, gereksiz yeniden açma işlemlerinin gerçekleşmemesini sağlar. Olası yeniden açma sorunlarından herhangi biri gerçekleşecekse açık seçenekleri açık bir şekilde ayarlayın (bkz. [Yeniden Aç](#)).

open yöntemini kullanırsanız, önce **açma seçenekleri** ' nin uygun olduğundan emin olmak için **gerekir** . Ancak, **open** yönteminin kullanılması zorunlu değildir; IBM MQ yine de 1. vakada olduğu gibi aynı davranışı sergiler, ancak bu durumda davranış verimlidir.

Sıfır geçerli bir değer değil; nesneyi açma girişiminde bulunmadan önce uygun değeri ayarlayın. Bu işlem, **setOpenSeçenekleri** (*lOpenSeçenekleri*) kullanılarak yapılabilir. ardından **open** () ya da **openFor** (*lRequiredOpenOption*).

Not:

1. Bir dağıtım listesi için **aç** yöntemi sırasında MQOO_OUTPUT yerine MQOO_OUTPUT yerine konu; MQOO_OUTPUT, şu anda geçerli olan tek **açma seçeneği**. Ancak, **open** yöntemini kullanan uygulama programlarında MQOO_OUTPUT ögesini belirttik olarak ayarlamak her zaman iyi bir uygulamadır.
2. Sınıfın **çözülen kuyruk yöneticisi adı** ve **çözülen kuyruk adı** 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 düzende), bu nesne ile aynı **bağlantı başvuru**ya sahip olmalıdır. Başlangıç değeri sıfır.

kuyruk yöneticisi tanıtıcısı

Kuyruk yöneticisi tanıtıcısı. Bu öznitelik salt okunurdur.

Oluşturucular

ImqObject();

Varsayılan oluşturucu.

ImqObject(const ImqObject & object);

Kopya oluşturucusu. **Açık durumu** FALSE olur.

Sınıf yöntemleri (genel)

statik MQUZE davranışı ();

Davranışdeğerini döndürür.

void setBehavior(const Mqlong davranış = 0);

Davranış' i ayarlar.

Nesne yöntemleri (genel)

void operator = (const ImqObject & object);

Gerekirse bir kapatma gerçekleştirir ve eşgörünüm verilerini *nesne* içinden kopyalar. **Açık durumu** FALSE olur.

ImqBoolean alterationDate(ImqString & date);

Değişiklik tarihi' nin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

ImqString alterationDate();

Olası hatalara ilişkin herhangi bir belirti olmaksızın **değiştirme tarihini** döndürür.

ImqBoolean alterationTime(ImqString & zaman);

Değişiklik süresi' nin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

ImqString alterationTime();

Olası hatalara ilişkin herhangi bir belirti olmaksızın **değiştirme süresini** döndürür.

ImqString alternateUserId () const ;

Diğer kullanıcı kimliği' nin bir kopyasını döndürür.

ImqBoolean setAlternateUserId (const char * id);

diğer kullanıcı kimliği' i ayarlar. **Diğer kullanıcı kimliği** yalnızca **open status** (açık durumu) FALSE (Yanlış) olduğunda ayarlanabilir. Bu yöntem başarılı olursa TRUE değerini döndürür.

ImqBinary alternateSecurityId () const;

Diğer güvenlik tanıtıcısı' nin bir kopyasını döndürür.

ImqBoolean setAlternateSecurityId(const ImqBinary & simge);

Diğer güvenlik tanıtıcısını ayarlar. **Diğer güvenlik tanıtıcısı** yalnızca **open status** (açık durumu) FALSE (Yanlış) olduğunda ayarlanabilir. *belirteç* veri uzunluğunun sıfır ya da MQ_SECURITY_ID_LENGTH olması gerekir. Başarılı olursa TRUE değerini döndürür.

ImqBoolean setAlternateSecurityId(const MQBYTE* *simge* = 0);

Diğer güvenlik tanıtıcısını ayarlar. *simge* sıfır (sıfır) olabilir; bu, MQSID_NONE belirtilerek aynıdır. *belirteç* 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 kimliği yalnızca, **açık durumu** 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.

MQUZE closeOptions () const ;

Kapat seçenekleri değerini döndürür.

void setCloseSeçenekleri (const MQLong *seçenekler* ' i);

Kapat seçenekleri ' ne ayarlar.

ImqQueueManager * connectionReference () const ;

Bağlantı başvurusu değerini döndürür.

void setConnectionReferansı (ImqQueueManager & *yönetici*);

Bağlantı başvurusunu ayarlar.

void setConnectionReferansı (ImqQueueManager * *yönetici* = 0);

Bağlantı başvurusunu ayarlar.

sanal ImqBoolean description (ImqString & *description*) = 0;

Açıklama ' nın bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

ImqString açıklama ();

Olası hatalara ilişkin herhangi bir belirti olmadan **description** ögesinin bir kopyasını döndürür.

sanal ImqBoolean ad (ImqString & *ad*);

Ad ' ın bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

ImqString ad ();

Olası hatalara ilişkin herhangi bir belirti olmaksızın **ad** ögesinin bir kopyasını döndürür.

ImqBoolean setName (const char * *ad* = 0);

Ad ' ı ayarlar. **ad** yalnızca **açık durum** FALSE olduğunda ve ImqQueueManager için **bağlantı durumu** FALSE olduğunda ayarlanabilir. Başarılı olursa TRUE değerini döndürür.

ImqObject * nextManagedNesnesi () const ;

Sonraki yönetilen nesne değerini döndürür.

ImqBoolean open ();

Changes the **açık durum** to TRUE by opening the object as necessary, using among other attributes the **açma seçenekleri** and the **ad**. This method uses the **bağlantı başvurusu** information and the ImqQueueManager **bağlantı** method if necessary to ensure that the ImqQueueManager **bağlantı durumu** is TRUE. **Açık durum** değerini döndürür.

ImqBoolean openFor (const MQLong *gerekli-seçenekler* = 0);

Nesnenin **açma seçenekleri** ile ya da **zorunlu-seçenekler** parametre değerinin belirttiği davranışı garanti eden **açma seçenekleri** ile açık olduğundan emin olmak için yapılan denemeler.

requires-options (zorunlu seçenekler) sıfırsa, giriş gereklidir ve giriş seçeneği yeterli olur. Bu nedenle, **açık seçenekler** aşağıdakilerden birini içeriyorsa:

- MQOO_INPUT_AS_Q_DEF
- MQOO_INPUT_SHARED
- MQOO_INPUT_EXCLUSIVE

açma seçenekleri zaten tatmin edici ve değiştirilmez; **açma seçenekleri** bu seçeneklerden herhangi birini içermiyorsa, **açma seçenekleri** içinde MQOO_INPUT_AS_Q_DEF ayarlıdır.

zorunlu-seçenekler sıfırda, gerekli seçenekler **açma seçenekleri** ' e eklenir. *zorunlu-seçenekler* bu seçeneklerden herhangi biri ise, diğerleri sıfırlanır.

Açık seçenekler değiştiyse ve nesne önceden açılmışsa, nesne geçici olarak kapatılır ve **açma seçenekleri** ayarlanacak şekilde yeniden açılır.

Başarılı olursa TRUE değerini döndürür. Başarı, nesnenin uygun seçeneklerle açık olduğunu gösterir.

MQUZE openOptions () const ;

Açık seçenekler döndürür.

ImqBoolean setOpenSeçenekleri (const MQlong seçenekler);

Açık seçenekler ayarlar. **Açık seçenekler** yalnızca **açık durum** FALSE olduğunda ayarlanabilir. Başarılı olursa TRUE değerini döndürür.

ImqBoolean openStatus () const ;

Açık durum döndürür.

ImqObject * previousManagedNesnesi () const ;

Önceki yönetilen nesne döndürür.

ImqBoolean queueManagerTanıtıcısı (ImqString & tanıtıcı);

kuyruk yöneticisi tanıtıcısı ' nin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

ImqString queueManagerTanıtıcısı ();

Olası hatalara ilişkin herhangi bir belirti olmadan **kuyruk yöneticisi tanıtıcısı** değerini döndürür.

Nesne yöntemleri (korunan)

sanal ImqBoolean closeTemporarily ();

Yeniden açmadan önce bir nesneyi güvenli bir şekilde kapatır. Başarılı olursa TRUE değerini döndürür. Bu yöntem, **açık durum** ' un TRUE olduğunu varsayar.

MQHCONN connectionHandle () const ;

Bağlantı başvurusu ile ilişkili MQHCONN değerini döndürür. **Bağlantı başvurusu** yoksa ya da Yönetici bağlı değilse, bu değer sıfır olur.

ImqBoolean sorgula (const MQlong int-attr, MQORT & 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 çağrısını uygun parametrelerle kullanmanız gerekir.

virtual void openInformationDasperse ();

MQOPED çağrısından hemen sonra MQOD veri yapısının değişken bölümündeki bilgileri dağıtır.

sanal ImqBoolean openInformationPrepare ();

MQOD veri yapısının değişken kısmına ilişkin bilgileri MQOPED çağrısından hemen önce hazırlar ve başarılı olursa TRUE değerini döndürür.

ImqBoolean set (const MQlong int-attr, const MQlong value);

Bir IBM MQ tamsayı özniteliğini ayarlar.

ImqBoolean set (const MQlong char-attr, const char * buffer, const size_t requires-length);

Bir IBM MQ karakter özniteliğini ayarlar.

void setNextManagedObject (const ImqObject * object = 0);

Sonraki yönetilen nesne ayarlar.

Dikkat: Bu işlevi, yalnızca yönetilen nesne listesinin bozulmayacağından eminseniz kullanın.

void setPreviousManagedObject (const ImqObject * object = 0);

Önceki yönetilen nesne ayarlar.

Dikkat: Bu işlevi, yalnızca yönetilen nesne listesinin bozulmayacağından eminseniz kullanın.

Nesne verileri (korunan)

MQHOBj ohobj

IBM MQ nesne tanıtıcısı (yalnızca **open status** değeri 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* tarafından adreslenen tüm alanlara erişilebilir.

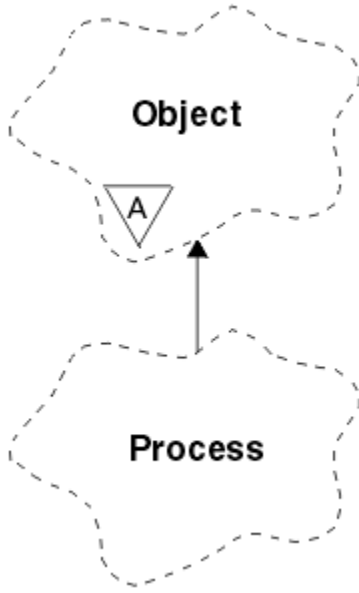
Not: *omqod.pmqod.Version* , *omqod.Version* değeri küçük olabilir; bu, IBM MQ MQI client ' in IBM MQ sunucusundan daha fazla işleve sahip olduğunu gösterir.

Neden kodları

- MQRC_ATTRIBUTE_LOCKED
- MQRC_INCONSISTENT_OBJECT_STATE
- MQRC_NO_CONNECTION_REFERENCE
- MQRC_STORAGE_NOT_AVAIL
- 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 62. *ImqProcess* sınıfı

- “Nesne öznitelikleri” sayfa 1898
- “Oluşturucular” sayfa 1898
- “Nesne yöntemleri (genel)” sayfa 1898

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 *işlemi*çinden 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);

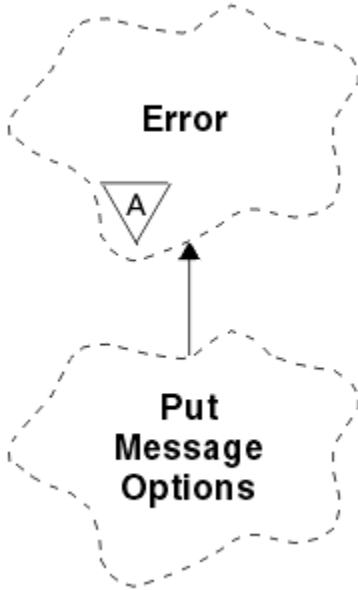
Kullanıcı verileri' nin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

ImqString userData ();

Olası hatalara ilişkin herhangi bir belirti olmaksızın **kullanıcı verilerini** döndürür.

ImqPutMessageOptions C++ sınıfı

Bu sınıf MQPMO veri yapısını sarsalıyor.



Şekil 63. ImqPutMessageOptions sınıfı

- [“Nesne öznitelikleri” sayfa 1899](#)
- [“Oluşturucular” sayfa 1900](#)
- [“Nesne yöntemleri \(genel\)” sayfa 1900](#)
- [“Nesne verileri \(korunan\)” sayfa 1901](#)
- [“Neden kodları” sayfa 1901](#)

Nesne öznitelikleri

bağlam başvurusu

İletiler için bağlam sağlayan bir ImqQueue . Başlangıçta başvuru yok.

seçenekler

Put iletileri 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 öznelikleri, belirtilen herhangi bir alan için nesneden alınır. ImqMessageTracker öznelikleri, *değil* belirtilen herhangi bir alan için ImqMessage nesnesinden alınır.

çözülen kuyruk yöneticisi adı

Put sırasında belirlenen hedef kuyruk yöneticisinin adı. Başlangıç değeri boş değerli. Bu öznelik salt okunurdur.

çözülen kuyruk adı

Put sırasında belirlenen hedef kuyruğun adı. Başlangıç değeri boş değerli. Bu öznelik salt okunurdur.

eşitleme noktası 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);

Eşgörünüm verilerini, var olan eşgörünüm verilerini değiştirerek *pmo*' dan kopyalar.

ImqQueue * contextReference () const ;

Bağlam başvurusunu döndürür.

void setContextBaşvuru (const ImqQueue & queue);

Bağlam başvurusunu ayarlar.

void setContextReference (const ImqQueue * queue = 0);

Bağlam başvurusunu ayarlar.

MQLONG seçenekler () const ;

seçeneklerdeğerini döndürür.

void setOptions (const MQlong seçenekler);

syncpoint katılımı değeri de dahil olmak üzere seçenekleri ayarlar.

MQUZT recordFields () const ;

Kayıt alanlarını döndürür.

void setRecordFields (const MQlong fields);

Kayıt alanlarını belirler.

ImqString resolvedQueueManagerName () const ;

çözülen kuyruk yöneticisi adı' nın bir kopyasını döndürür.

ImqString resolvedQueueAd () const ;

Çözümlenen kuyruk adı' nun bir kopyasını döndürür.

ImqBoolean syncPointKatılım () const ;

seçenekler , MQPMO_SYNCPOINT ögesini içerirse, TRUE olan **syncpoint katılımı** değerini döndürür.

void setSyncPointParticipation (const ImqBoolean sync);

syncpoint katılım değerini ayarlar. *eşitleme* TRUE ise, **seçenekler** , MQPMO_SYNCPOINT ögesini içerecek şekilde değiştirilir ve MQPMO_NO_SYNCPOINT ögesini dışlamak için değiştirilir. *eşitleme* FALSE ise, **seçenekler** , MQPMO_NO_SYNCPOINT ögesini içerecek şekilde değiştirilir ve MQPMO_SYNCPOINT ögesini dışlamak için değiştirilir.

Nesne verileri (korunan)

MQPMO omqpmo

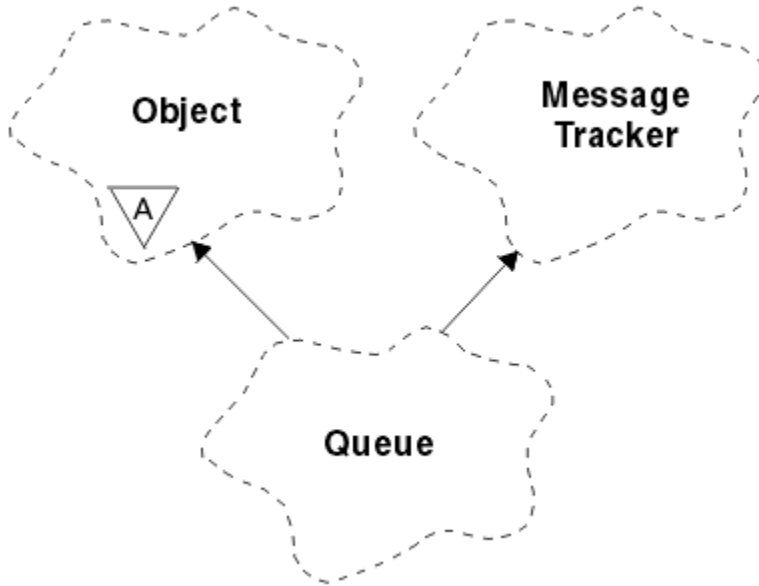
MQPMO veri yapısı.

Neden kodları

- MQRC_STORAGE_NOT_AVAM

ImqQueue C++ sınıfı

Bu sınıf bir ileti kuyruğunu sarsalıyor (MQOT_Q tipinde bir IBM MQ nesnesi).



Şekil 64. ImqQueue sınıfı

Bu sınıf, Çizelge 223 sayfa 1841 içinde listelenen MQI çağrıları ile ilgilidir.

- “Nesne öznitelikleri” sayfa 1902
- “Oluşturucular” sayfa 1904
- “Nesne yöntemleri (genel)” sayfa 1905
- “Nesne yöntemleri (korunan)” sayfa 1911

- [“Neden kodları” sayfa 1911](#)

Nesne öznitelikleri

geri alma isteği adı

Gereğinden fazla geri alma isteği adı. Bu öznitelik salt okunurdur.

Geriletme eşiği

Geriletme eşiği. Bu öznitelik salt okunurdur.

temel kuyruk adı

Diğer adın çözümlediği 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 kullanılabilecek 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. for all Windows, UNIX, and Linux platforms.

harden geri tepme

Geri sayım için harden de olsun. Bu öznitelik salt okunurdur.

dizin tipi

Dizin tipi. Bu öznitelik salt okunurdur.

İnhibit alma

Alma işlemlerine izin verilip verilmediği. Başlangıç değeri, kuyruk tanımlamasına bağlıdır. Bu öznitelik yalnızca bir diğer ad ya da yerel kuyruk için geçerlidir.

İnhibit put

Koyma işlemlerine izin verilip 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 yöneticisi adı

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 toplamanın 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 paylaşılabilip paylaşılamayacağı. Bu öznitelik salt okunurdur.

depolama sınıfı

Depolama sınıfı. Bu öznitelik salt okunurdur.

iletim kuyruğu adı

İletim kuyruğunun adı. Bu öznitelik salt okunurdur.

tetikleme denetimi

Tetik kontrolü. Başlangıçtaki değer kuyruk tanımlamasına bağlıdır. Bu öznitelik yalnızca yerel bir kuyruk için geçerlidir.

tetikleme 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.

tetikleme iletisi 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 (kuyruk kuyruğu));

Kopya oluşturucusu. ImqObject **open status** , FALSE olur.

ImqQueue(const char * ad);
ImqObject **adını**ayarlar.

Nesne yöntemleri (genel)

void operator = (const ImqQueue & queue);
Gerekirse bir kapatma gerçekleştirir ve eşgörünüm verilerini *queue*(kuyruk) içinden kopyalar.
ImqObject **open status** , FALSE olur.

ImqBoolean backoutRequeueName (ImqString & ad);
backout require nameöğesinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

ImqString backoutRequeueAd ();
Olası hatalara ilişkin herhangi bir belirti olmadan **backout requeue name** değerini döndürür.

ImqBoolean backoutThreshold (MQLONG & 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 & seçenek);

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 (MQORT & 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.

MQUQT 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 (MQBOLD & 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.

MQUQT 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 depthMaximumOlay (MQBOLD & olay);

Provides a copy of the enablement state of the **derinlik üst sınırı olayı**. Başarılı olursa TRUE değerini döndürür.

MQLONG depthMaximumOlay ();

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 başvurusunuayarlar.

void setDistributionListReference (ImqDistributionList * list = 0);

Dağıtım listesi başvurusunuayarlar.

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 & seçenekler);

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* nesnesinin ImqCache **otomatik arabelleği** varsa, arabellek alınan tüm iletileri barındıracak şekilde büyür. The **clearMessage** method is invoked against the *İlt* 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 yansır. 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 kılan *arabellek-büyükülüğü* 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 arabellek-büyükülüğü);

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(MQBOLE & 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 & inhibit);

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 & inhibit);

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 (MQlong & 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 (MQLONG & 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 & acctq);

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 adı 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 'nün bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

ImqString remoteQueueManagerName ();

Olası hatalara ilişkin herhangi bir belirti olmadan **uzak kuyruk yöneticisi adı** nı döndürür.

ImqBoolean remoteQueueName (ImqString & ad);

Uzak kuyruk adı 'nün 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 **çözümlemiş kuyruk yöneticisi adı** değerini döndürür.

ImqBoolean resolvedQueueAd(ImqString & ad);

Çözümlemiş kuyruk 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 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 serviceIntervalOlayı (MQBOLD & 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 paylaşılabirlik (MQORT & paylaşılabirlik);

paylaşılabirlik değerinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

MQHOT paylaşılabirlik ();

Olası hatalara ilişkin herhangi bir belirti olmadan **sharebility** değerini döndürür.

ImqBoolean storageClass(ImqString & sınıf);

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 & denetim);

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 (MQLONG & 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 tetikleseçeneğini 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 & kullanım);

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.

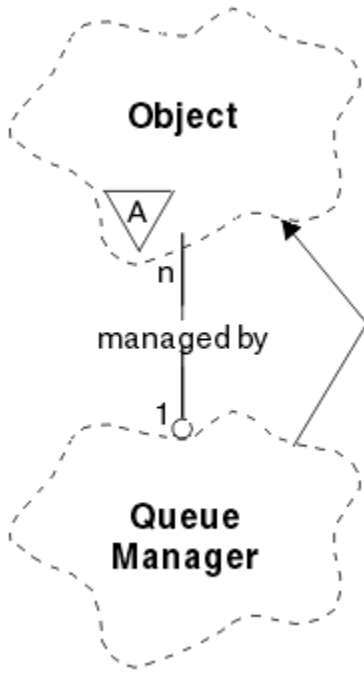
Dikkat: Bu işlevi yalnızca, dağıtılmış kuyruk listesini bozmayacağından eminseniz kullanın.

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 65. *ImqQueueManager* sınıfı

Bu sınıf, “[ImqQueueManager çapraz başvurusu](#)” sayfa 1844’inde listelenen MQI çağrılarını ile ilgilidir. Listelenen tüm yöntemler tüm altyapılar için geçerli değildir; ek ayrıntılar için [ALTER QMGR](#) konusuna bakın.

- “Sınıf öznitelikleri” sayfa 1912
- “Nesne öznitelikleri” sayfa 1913
- “Oluşturucular” sayfa 1918
- “Yok ediciler” sayfa 1918
- “Sınıf yöntemleri (genel)” sayfa 1918
- “Nesne yöntemleri (genel)” sayfa 1918
- “Nesne yöntemleri (korunan)” sayfa 1927
- “Nesne verileri (korunan)” sayfa 1927
- “Neden kodları” sayfa 1927

Sınıf öznitelikleri

davranış

Örtülü bağlantının ve bağlantının kesilmesinin davranışını denetler.

IMQ_EXPL_DISC_BACKOUT (0L)

Bağlantıyı kes yöntemine açık bir çağrı, geri tepme anlamına gelir. Bu öznitelik IMQ_EXPL_DISC_COMMIT ile karşılıklı olarak birbirini dışlar.

IMQ_EXPL_DISC_COMMIT (1L)

Bağlantıyı kes yöntemine açık bir çağrı, kesinleştirmeyi belirtir (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 imhası sırasında oluşabilecek, **bağlantı kesme** yöntemine yönelik örtük bir çağrı geri tepme 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 değer). 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 eşleştiginde otomatik olarak yeniden başlatılıp başlatılmayacağını, yeni mca denetimi değiştirgeleri 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

begin 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 okunurdur.

kanal başlatıcı bağdaştırıcıları

IBM MQ çağrılarını işlemek için kullanılacak bağdaştırıcı alt görevlerinin sayısı. Bu öznitelik salt okunurdur.

kanal başlatıcı denetimi

Kuyruk Yöneticisi başlatıldığında Kanal Başlatıcı 'nın otomatik olarak başlatılıp başlatılmayacağını belirleyin. Bu öznitelik salt okunurdur.

kanal başlatan dağıtıcıları

Kanal başlatıcısı için kullanılacak dağıtıcıların sayısı. Bu öznitelik salt okunurdur.

kanal başlatıcı izleme otomatik başlatma

Kanal başlatıcı izinin otomatik olarak başlamalı ya da başlatılmayacağı. Bu öznitelik salt okunurdur.

kanal başlatıcı izleme çizelgesi büyüklüğü

Kanal başlatıcısının izleme verileri alanının büyüklüğü (MB). Bu öznitelik salt okunurdur.

kanal izleme

Kanallar için çevrimiçi izleme verilerinin toplanmasını denetler. Bu öznitelik salt okunurdur.

kanal başvurusu

İstemci bağlantısı sırasında kullanılacak kanal tanımlamasına yönelik başvuru. Bağlıyken, 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 okunurdur.

karakter kümesi

Kodlanmış karakter takımı tanıtcısı (CCSID). Bu öznitelik salt okunurdur.

küme gönderen izleme

Otomatik olarak tanımlanmış küme gönderen kanallarına ilişkin çevrimiçi izleme verilerinin toplanmasını denetler. Bu öznitelik salt okunurdur.

küme gönderen istatistikleri

Otomatik olarak tanımlanmış küme gönderen kanallarına ilişkin istatistik verileri toplamanın denetlenmesini sağlar. Bu öznitelik salt okunurdur.

küme iş yükü verileri

Küme iş yükü çıkış verileri. Bu öznitelik salt okunurdur.

küme iş yükü çıkışı

Küme iş yükü çıkış adı. Bu öznitelik salt okunurdur.

küme iş yükü uzunluğu

Küme iş yükü uzunluğu. Bu öznitelik salt okunurdur.

küme iş yükü mru

Küme iş yükü en son kullanılan kanallar değeri. Bu öznitelik salt okunurdur.

küme iş yükü kullanım kuyruğu

Küme iş yükü kullanım kuyruğu değeri. Bu öznitelik salt okunurdur.

komut olayı

Komut olaylarının oluşturulup oluşturulmayacağı. Bu öznitelik salt okunurdur.

komut giriş kuyruğu adı

Sistem komutu giriş kuyruğu adı. Bu öznitelik salt okunurdur.

komut düzeyi

Kuyruk yöneticisi tarafından desteklenen komut düzeyi. Bu öznitelik salt okunurdur.

komut sunucusu denetimi

Kuyruk Yöneticisi başlatıldığında Komut Sunucusu 'nın da otomatik olarak başlatılıp başlatılmayacağını belirleyin. Bu öznitelik salt okunurdur.

bağlantı seçenekleri

connect 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 grubuna ilişkin gelen iletimleri işleyen TCP dinleyicisinin, Dinamik Etki Alanı Adı Hizmetleri için İş Yüğü Yöneticisi ile kayıt yaptırmaları 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 **bağlantı başvurusu** 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 SSL etkileşimi içinde gönderilen ve alınan şifrelenmemiş baytların 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 derlemi 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 öznitelik salt okunurdur.

kuyruk istatistikleri

Kuyruklar için istatistik verilerinin toplanmasını denetler. Bu öznitelik salt okunurdur.

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 bekleyeceği (yaklaşık olarak). Bu öznitelik salt okunurdur.

alma zamanaşımı alt sınırı

Bir TCP/IP kanalının, etkin olmayan duruma dönmeden önce iş ortasından kalp atışları da içinde olmak üzere verileri almak için bekleyeceği minimum süre. Bu öznitelik salt okunurdur.

alma zamanaşımı tipi

alma zamanaşımı değerine uygulanan bir niteleyici. Bu öznitelik salt okunurdur.

uzak olay

Uzak olayları denetler. Bu öznitelik salt okunurdur.

Havuz adı

Havuz adı. Bu öznitelik salt okunurdur.

havuz ad listesi

Havuz adı listesi adı. Bu öznitelik salt okunurdur.

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 öznitelik salt okunurdur.

ssl olayı

SSL olaylarının oluşturulup oluşturulmayacağı. Bu öznitelik salt okunurdur.

ssl FIPS

Whether only FIPS-certified algorithms should be used if the cryptography is executed in IBM MQ software. Bu öznitelik salt okunurdur.

ssl anahtarı sıfırlama sayısı

Gizli anahtar yeniden anlaşılmadan önce, bir SSL etkileşimi içinde gönderilen ve alınan şifrelenmemiş baytların sayısı. Bu öznitelik salt okunurdur.

start-stop olayı

Start-stop olaylarını denetler. Bu öznitelik salt okunurdur.

istatistik aralığı

İstatistik izleme verilerinin izleme kuyruğuna ne sıklıkta yazıldığı. Bu öznitelik salt okunurdur.

syncpoint kullanılabilirliği

Eşitleme noktası katılımının kullanılabilirliği. Bu öznitelik salt okunurdur.

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** and **kesinleştir** are supported on the IBM i platform for local units of work coordinated by a queue manager.

tcp kanalları

TCP/IP iletim protokolünü kullanan, bağlanabilecek, yürürlükteki ya da istemci olabilecek kanal sayısı üst sınırı. Bu öznitelik salt okunurdur.

tcp keepalive

Bağlantının diğer ucunun hala kullanılabilir olup olmadığını denetlemek için TCP KEEPSAVE olanağının kullanılıp kullanılmayacağını denetleyin. Bu öznitelik salt okunurdur.

tcp adı

tcp yığını tipideğerine bağlı olarak, kullanılacak tek ya da varsayılan TCP/IP sisteminin adı. Bu öznitelik salt okunurdur.

tcp yığın tipi

Kanal başlatıcının yalnızca **tcp adı** altında belirlenen 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

UNIX ve 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 **ad** ögesini *name* olarak ayarlar.

Yok ediciler

Bir ImqQueueManager nesnesi yok edildiğinde, otomatik olarak kesilir.

Sınıf yöntemleri (genel)

statik MQUZE davranışı ();

Davranış değerini döndürür.

void setBehavior(const MQLong davranış = 0);

Davranış 'ı ayarlar.

Nesne yöntemleri (genel)

void işleci = (const ImqQueueManager & mgr);

Gerekirse bağlantıyı keser ve eşgörünüm verilerini *mgr* ögesinden 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 değerini benimsemenin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

MQlong adoptNewMCACheck ();

Olası hatalara ilişkin herhangi bir belirti olmadan yeni MCA denetim değerini kabul eder.

ImqBoolean adoptNewMCAType (MQBOLD & tip);

Yeni MCA tipini benimsemenin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

MQlong adoptNewMCAType ();

Olası hatalara ilişkin herhangi bir belirti olmadan yeni MCA tipini benimseme olanağı sağlar.

QXX_ENCODE_CASE_ONE long authenticationType () const;

Kimlik doğrulama tipini döndürür.

void setAuthenticationTip (const MQTORT tipi = MQCSP_AUTH_NONE);

Kimlik doğrulama tipini ayarlar.

ImqBoolean authorityEvent(MQBOLD & olay);

Yetki olayı'nın etkinleştirme durumunun bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

MQHOT authorityEvent();

Olası hatalara ilişkin herhangi bir belirti olmaksızın, **yetki olayı** 'in etkinleştirme durumunu döndürür.

ImqBoolean geriletme ();

Kesinleştirilmemiş değişiklikleri yedekler. Başarılı olursa TRUE değerini döndürür.

ImqBoolean begin ();

Bir çalışma birimi başlatır. **başlangıç seçenekleri** , bu yöntemin davranışını etkiler. Başarılı olursa TRUE değerini döndürür; ancak, temeldeki MQBEGIN çağrısı MQRC_NO_EXTERNAL_PARTICILAR ya da MQRC_PARTICIPANT_NOT_AVAM (her ikisi de MQCC_UYARI ile ilişkilidir) döndürse de TRUE değerini döndürür.

MQlong beginOptions() const;

başlangıç seçeneklerideğerini döndürür.

void setBeginSeçenekleri (const MQlong seçenekler = MQBO_NONE);

başlangıç seçenekleri'ne ayarlar.

ImqBoolean bridgeEvent (MQORT & event);

Köprü olay değerinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

MQHOT bridgeEvent ();

Olası hatalara ilişkin herhangi bir belirti olmadan köprü olayı değerini döndürür.

ImqBoolean channelAutoTanımlaması (MQBOLD & değer);

Kanal otomatik tanımlama değerinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

MQOUR channelAutoTanımlaması ();

Olası hatalara ilişkin herhangi bir belirti olmadan **kanal otomatik tanımlaması** değerini döndürür.

ImqBoolean channelAutoDefinitionEvent(MQBOLD & değer);

Kanal otomatik tanımlama olayı değerinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

MQHOT channelAutoDefinitionEvent();

Olası hatalara ilişkin herhangi bir belirti olmadan **kanal otomatik tanımlama olayı** değerini döndürür.

ImqBoolean channelAutoDefinitionExit(ImqString & ad);

Kanal otomatik tanımlama çıkışı adının bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

ImqString channelAutoDefinitionExit();

Olası hatalara ilişkin herhangi bir belirti olmadan **kanal otomatik tanımlama çıkışı** adını döndürür.

ImqBoolean channelEvent (MQORT & event);

Kanal olay değerinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

MQHOMEN channelEvent();

Olası hatalara ilişkin herhangi bir belirti olmadan kanal olay değerini 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 başvurusu'nin bir kopyasını sağlar. **Kanal başvurusu** geçersizse, *pchannel* ögesini boş değer olarak ayarlar. Bu yöntem başarılı olursa TRUE değerini döndürür.

ImqChannel * channelReference();

Olası hatalara ilişkin herhangi bir belirti olmadan **kanal başvurusu** değerini 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(MQBOLD & ccscid);

Karakter kümesi'nin 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 değeri bir kopyasını sağlar. Başarılı olursa TRUE değeri döndürür.

MQHOT clusterSenderMonitoring ();

Olası hatalara ilişkin herhangi bir gösterge olmadan, küme gönderen izleme varsayılan değeri döndürür.

ImqBoolean clusterSenderStatistics (MQUVE & statacls);

Küme gönderen istatistik değeri bir kopyasını sağlar. Başarılı olursa TRUE değeri döndürür.

MQlong clusterSenderStatistics ();

Olası hatalara ilişkin herhangi bir gösterge olmadan, küme gönderen istatistik değeri döndürür.

ImqBoolean clusterWorkloadVerileri (ImqString & data);

Küme iş yükü çıkış verileri' nin bir kopyasını sağlar. Başarılı olursa TRUE değeri döndürür.

ImqString clusterWorkloadData ();

Olası hatalara ilişkin herhangi bir belirti olmadan küme iş yükü çıkış verilerini döndürür.

ImqBoolean clusterWorkloadExit (ImqString & ad);

Küme iş yükü çıkış adı' nun bir kopyasını sağlar. Başarılı olursa TRUE değeri döndürür.

ImqString clusterWorkloadExit ();

Olası hatalara ilişkin herhangi bir belirti olmadan küme iş yükü çıkış adını döndürür.

ImqBoolean clusterWorkloadLength (MQBOLD & uzunluk);

Küme iş yükü uzunluğu' nin bir kopyasını sağlar. Başarılı olursa TRUE değeri döndürür.

MQlong clusterWorkloadLength ();

Olası hatalara ilişkin herhangi bir belirti olmadan küme iş yükü uzunluğunu döndürür.

ImqBoolean clusterWorkLoadMRU (MQUBE & mru);

Küme iş yükünün en son kullanılan kanal değeri bir kopyasını sağlar. Başarılı olursa TRUE değeri döndürür.

MQUZE clusterWorkLoadMRU ();

Olası hatalara ilişkin herhangi bir gösterge olmadan, en son kullanılan kanal iş yükünü en son kullanılan kanallar değeri döndürür.

ImqBoolean clusterWorkLoadUseQ (MQUBY & useq);

Küme iş yükü kullanım kuyruğu değeri içeren bir kopya sağlar. Başarılı olursa TRUE değeri döndürür.

MQlong clusterWorkLoadUseQ ();

Olası hatalara ilişkin herhangi bir belirti olmadan küme iş yükü kullanım kuyruğu değeri döndürür.

ImqBoolean commandEvent (MQOLD & event);

Komut olay değeri bir kopyasını sağlar. Başarılı olursa TRUE değeri döndürür.

MQlong commandEvent ();

Olası hatalara ilişkin herhangi bir belirti olmadan komut olay değeri döndürür.

ImqBoolean commandInputQueueName(ImqString & ad);

komut giriş kuyruğu adının bir kopyasını sağlar. Başarılı olursa TRUE değeri döndürür.

ImqString commandInputQueueName();

Olası hatalara ilişkin herhangi bir belirti olmadan komut giriş kuyruğu adını döndürür.

ImqBoolean commandLevel(MQBOLD & düzey);

Komut düzeyinin bir kopyasını sağlar. Başarılı olursa TRUE değeri 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 değeri döndürür.

ImqBoolean commandServerControl (MQBOLD & server);

Komut sunucusu denetimi başlangıç değeri bir kopyasını sağlar. Başarılı olursa TRUE değeri döndürür.

ImqBoolean commit ();

Kesinleştirilmemiş değişiklikleri kesinleştirin. Başarılı olursa TRUE değerini döndürür.

ImqBoolean connect ();

Belirtilen ImqObject **adile** 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, kanal tanımlamasıyla ilgili bilgileri bir MQCD ' de MQCONNX 'e aktarmak için kullanılır. MQCD ' deki ChannelType , MQCHT_CLNTCONN olarak ayarlıdır. Yalnızca istemci bağlantıları için anlamlı olan **kanal başvurusu** , sunucu bağlantıları için dikkate alınmaz. **Bağlantı seçenekleri** , bu yöntemin davranışını 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. Tümü 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 tag = 0);

Bağlantı etiketini ayarlar. *etiket* sıfırsa, **bağlantı etiketini** temizler. Bu yöntem başarılı olursa TRUE değerini döndürür.

ImqBoolean setConnectionEtiketi (const ImqBinary & tag);

Bağlantı etiketini ayarlar. *etiket* 'in **veri uzunluğu** değeri, sıfır (**bağlantı etiketi** ' ı temizlemek için) olmalıdır ya da MQ_CONN_TAG_LENGTH. Bu yöntem başarılı olursa TRUE değerini döndürür.

MQHOT connectOptions() const;

Bağlantı seçeneklerini döndürür.

void setConnectSeçenekleri (const MQlong seçenekler = MQCNO_NONE);

Bağlantı seçeneklerini ayarlar.

ImqBoolean connectionStatus() const;

Bağlantı durumunu döndürür.

ImqString cryptographicHardware ();

Şifreleme donanımı değerini döndürür.

ImqBoolean setCryptographicDonanımı (const char * hardware = 0);

Şifreleme donanımını ayarlar. Bu yöntem başarılı olursa TRUE değerini döndürür.

ImqBoolean deadLetterQueueName(ImqString & ad);

ölü-harflı kuyruk adı ' nun bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

ImqString deadLetterQueueName ();

Olası hatalara ilişkin herhangi bir belirti olmadan, **dead-letter queue name** (ölü harf kuyruğu adı) ögesinin bir kopyasını döndürür.

ImqBoolean defaultTransmissionQueueName(ImqString & ad);

Varsayılan iletim kuyruğu adı ' nun bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

ImqString defaultTransmissionQueueName ();

Olası hatalara ilişkin herhangi bir belirti olmadan **varsayılan iletim kuyruğu adı** değerini döndürür.

ImqBoolean bağlantı kesme ();

Kuyruk yöneticisinden bağlantıyı keser ve **bağlantı durumu** değerini FALSE olarak ayarlar. Bu nesneyle ilişkili tüm ImqProcess ve ImqQueue nesnelerini kapatır ve bağlantıyı kesmeden önce **bağlantı başvurusunu** getirir. 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(MQBOLE & 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 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 * air = 0);

İlk kimlik doğrulama kaydını ayarlar.

ImqObject * firstManagedNesnesi () const;

ilk yönetilen nesne değerini döndürür.

ImqBoolean inhibitEvent(MQBOLD & olay);

Engelleyici olayı etkinleştirmenin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

MQLONG inhibitEvent();

Olası hatalara ilişkin herhangi bir belirti olmaksızın, **etkinliği engelle** ' in 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 havuzu değerini döndürür.

ImqBoolean setKeyHavuzu (const char * havuz = 0);

Anahtar havuzu ' yı ayarlar. Başarılı olursa TRUE değerini döndürür.

ImqBoolean listenerTimer (MQORT & timer);

Dinleyici süreölçeri değerinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

MQUQT listenerTimer ();

Olası hatalara ilişkin herhangi bir belirti olmadan dinleyici zamanlayıcısı değerini döndürür.

ImqBoolean localEvent(MQBOLD & olay);

Yerel olay ' in etkinleştirme durumunun bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

MQLONG localEvent();

Olası hatalara ilişkin herhangi bir belirti olmadan **local event** (yerel olay) 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 & sonex);

LU62 ARM sonexinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

ImqString lu62ARMSuffix ();

Olası hatalara ilişkin herhangi bir belirti olmadan LU62 ARM sonexini 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(MQBOLD & sayı);

tutamaçların üst sınırı'nin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

MQLONG maximumHandles();

Olası hataların göstergesi olmadan **tutamaçları üst sınırı** değerini 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 sınırı'nin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

Mqlong maximumMessageUzunluk ();

Olası hatalara ilişkin herhangi bir belirti olmadan **ileti uzunluğu üst sınırı** değerini döndürür.

ImqBoolean maximumPriority(MQBOLD & priority);

Öncelik üst sınırı'nin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

MQLONG maximumPriority();

Olası hatalara ilişkin herhangi bir belirti olmadan **en yüksek öncelik** kögesinin 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 üst sınırı'nin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

Mqlong maximumUncommittedİletileri ();

Olası hataların herhangi bir belirtisi olmadan **kesinleştirilmemiş iletilerin üst sınırı** değerini döndürür.

ImqBoolean mqiAccounting (MQUVE & statint);

MQI hesap deęerinin bir kopyasını saęlar. Başarılı olursa TRUE deęerini döndürür.

MQHOT mqiAccounting ();

Olası hatalara ilişkin herhangi bir belirti olmadan MQI hesap deęerini 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 baęlantılarında kullanılan parolayı döndürür.

ImqBoolean setPassword (const ImqString & password);

İstemci baęlantılarında kullanılan parolayı ayarlar.

ImqBoolean setPassword (const char * = 0 password);

İstemci baęlantılarında kullanılan parolayı ayarlar.

ImqBoolean setPassword (const ImqBinary & password);

İstemci baęlantılarında kullanılan parolayı ayarlar.

ImqBoolean performanceEvent(MQBOLD & olay);

Başarım olayı' nın etkinleřtirme durumunun bir kopyasını saęlar. Başarılı olursa TRUE deęerini döndürür.

MQLONG performanceEvent();

Olası hatalara ilişkin herhangi bir belirti olmaksızın, **başarım olayı** ' in etkinleřtirme durumunu döndürür.

ImqBoolean altyapısı (MQUZE & altyapı);

Altyapı' ın bir kopyasını saęlar. Başarılı olursa TRUE deęerini döndürür.

MQLONG platformu ();

Olası hatalara ilişkin herhangi bir belirti olmadan **platform** ' u 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 zamanařımı deęerinin bir kopyasını saęlar. Başarılı olursa TRUE deęerini döndürür.

MQHOMEN receiveTimeout ();

Olası hatalara ilişkin herhangi bir belirti olmadan alma zamanařımı deęerini döndürür.

ImqBoolean receiveTimeoutMin (MQUVE & min);

Alma zamanasını alt sınır değerinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

MQHOMEN receiveTimeoutMin ();

Olası hatalara ilişkin herhangi bir gösterge olmadan minimum alma zamanasını değerini döndürür.

ImqBoolean receiveTimeoutTip (MQBOLD & tip);

Alma zamanasını tipine ilişkin bir kopya sağlar. Başarılı olursa TRUE değerini döndürür.

MQUMET receiveTimeoutTip ();

Olası hatalara ilişkin herhangi bir belirti olmadan alma zamanasını tipini döndürür.

ImqBoolean remoteEvent(MQBOLD & olay);

Uzak olay' in 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 adı' nın 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 adı' nun bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

ImqString repositoryNameListName ();

Olası hataların herhangi bir göstergesi olmaksızın, **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);

start-stop event(başlatma olayı) ögesinin etkinleştirme durumunun bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

MQlong startStopEvent ();

Olası hatalara ilişkin herhangi bir belirti olmaksızın, **start-stop olayı** ' in 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 (MQBOLD & eşitleme);

Syncpoint kullanılabilirlik değerinin bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

MQOUR syncPointUygunluğu ();

Olası hatalara ilişkin herhangi bir belirti olmadan, **syncpoint availability** 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 aralığı' nın bir kopyasını sağlar. Başarılı olursa TRUE değerini döndürür.

MQLONG triggerInterval();

Olası hatalara ilişkin herhangi bir belirti olmadan **tetikleme aralığı** değerini 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 nesneyiayarlar.

Nesne verileri (korunan)

MQHCONN ohconn

IBM MQ bağlantısı tanıtıcısı (yalnızca **bağlantı durumu** TRUE olduğunda anlamlıdır).

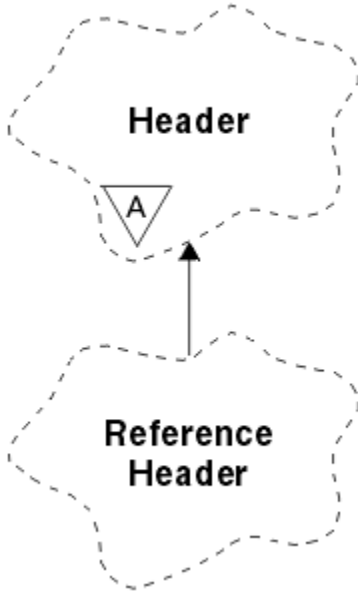
Neden kodları

- MQRC_ATTRIBUTE_LOCKED
- MQRC_ENVIRONMENT_ERROR
- MQRC_FUNCTION_NOT_SUPPORTUL
- MQRC_REFERENCE_ERROR
- (MQBACK için neden kodları)
- (MQBEGIN 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 66. ImqReferenceÜstbilgi sınıfı

Bu sınıf, “ImqReferenceÜstbilgi çapraz başvurusu” sayfa 1847’inde listelenen MQI çağrıları ile ilgilidir.

- “Nesne öznitelikleri” sayfa 1928
- “Oluşturucular” sayfa 1929
- “Aşırı yüklenmiş ImqItem yöntemleri” sayfa 1929
- “Nesne yöntemleri (genel)” sayfa 1929
- “Nesne verileri (korunan)” sayfa 1930
- “Neden kodları” sayfa 1930

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 konum' a 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.

Oluşturucular

ImqReferenceÜstbilgisi ();

Varsayılan oluşturucu.

ImqReferenceÜstbilgisi (const ImqReferenceÜstbilgisi & üstbilgi);

Kopya oluşturucusu.

Aşırı yüklenmiş ImqItem yöntemleri

sanal ImqBoolean copyOut (ImqMessage & msg);

Başlangıçta ileti arabelleğindeki bir MQRMH veri yapısı ekler, var olan ileti verilerini daha da ileriye taşıyıp **msg biçimi** MQFM_REF_MSG_HEDENE değerini ayarlar.

Daha ayrıntılı bilgi için, "[ImqHeader C++ sınıfı](#)" sayfa 1875 ' ta ImqHeader sınıf yöntemi açıklamasına bakın.

sanal ImqBoolean pasteIn (ImqMessage & msg);

İleti arabelleğinden bir MQRMH veri yapısını okur.

Başarılı olmak için, ImqMessage **biçim** MQFMT_REF_MSG_HEADER olmalıdır.

Daha ayrıntılı bilgi için, "[ImqHeader C++ sınıfı](#)" sayfa 1875 ' ta ImqHeader sınıf yöntemi açıklamasına bakın.

Nesne yöntemleri (genel)

void operator = (const ImqReferenceÜstbilgisi & üstbilgi);

Eşgörünüm verilerini, var olan eşgörünüm verilerini değiştirerek *üstbilgisinden* kopyalar.

ImqString destinationEnvironment () const ;

Hedef ortam ' nın bir kopyasını döndürür.

void setDestinationOrtam (const char * environment = 0);

Hedef ortamı ayarlar.

ImqString destinationName () const ;

Hedef adı ' nın bir kopyasını döndürür.

void setDestinationAd (const char * ad = 0);

Hedef adı belirler.

ImqBinary instanceId () const ;

Yönetim ortamı tanıtıcısı ' nın bir kopyasını döndürür.

ImqBoolean setInstanceId (const ImqBinary & id);

eşgörünüm tanıtıcısı ' ı ayarlar. The **veri uzunluğu** of *simge* must be either 0 or MQ_OBJECT_INSTANCE_ID_LENGTH. Bu yöntem başarılı olursa TRUE değerini döndürür.

void setInstanceId (const MQBYTE24 id = 0);

eşgörünüm tanıtıcısı ' ı ayarlar. *id* sıfır (sıfır) olabilir; bu, MQOLI_NONE belirtilerek aynı olur. *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.

MQlong logicalLength () const ;

Mantıksal uzunluk değerini döndürür.

void setLogicalLength (const MQlong length);

Mantıksal uzunluk değerini ayarlar.

MQUZE logicalOffset () const ;
 Mantıksal görelî konumdeğerini döndürür.

void setLogicalOffset (const MQlong görelî konum);
 Mantıksal görelî konumdeğerini ayarlar.

MQUZE logicalOffset2 () const ;
 Mantıksal görelî konum 2değerini döndürür.

void setLogicalOffset2 (const MQlong görelî konum);
 Mantıksal görelî konum 2' yi ayarlar.

ImqString referenceType () const ;
 Başvuru tipi' nin bir kopyasını döndürür.

void setReferenceTip (const char * ad = 0);
 Başvuru tipiniayarlar.

ImqString sourceEnvironment () const ;
 Kaynak ortam' nın bir kopyasını döndürür.

void setSourceOrtam (const char * environment = 0);
 Kaynak ortamıbelirler.

ImqString sourceName () const ;
 Kaynak adı' nın bir kopyasını döndürür.

void setSourceAd (const char * ad = 0);
 Kaynak adı' na ayarlar.

Nesne verileri (korunan)

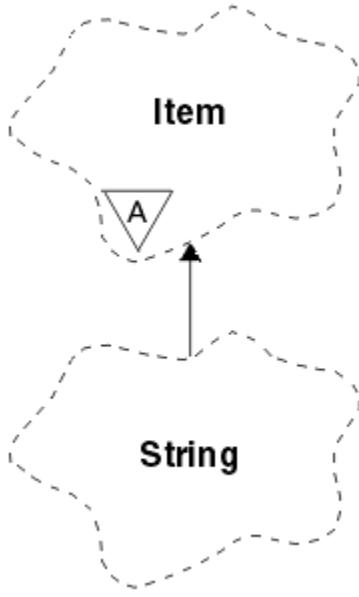
MQRMH omqrmh
 MQRMH veri yapısı.

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 67. ImqString sınıfı

Use an ImqString in place of a **char *** in most situations where a parameter calls for a **char ***.

- “Nesne öznitelikleri” sayfa 1931
- “Oluşturucular” sayfa 1931
- “Sınıf yöntemleri (genel)” sayfa 1932
- “Aşırı yüklenmiş ImqItem yöntemleri” sayfa 1932
- “Nesne yöntemleri (genel)” sayfa 1932
- “Nesne yöntemleri (korunan)” sayfa 1935
- “Neden kodları” sayfa 1935

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** özniteliği yoktur.

Oluşturucular

ImqString();

Varsayılan oluşturucu.

ImqString(const ImqString & dizgi);

Kopya oluşturucusu.

ImqString(const char c);

Karakterler , c' i oluşturur.

ImqString(const char * metin);

Karakterler , metiniçinden 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-arabellek*, const size_t *length*, const char * *kaynak-buffer*, const char *pad* = 0);

Copies up to *uzunluk* bytes from *kaynak arabellek* to *hedef-arabellek*. If the number of characters in *kaynak arabellek* is insufficient, fills the remaining space in *hedef-arabellek* with *doldurma* characters. *kaynak-arabelleği* sıfır olabilir. *uzunluk* aynı zamanda sıfır *hedef-arabellek* sıfır olabilir. Herhangi bir hata kodu kaybedilir. Bu yöntem başarılı olursa TRUE değerini döndürür.

statik ImqBoolean copy (char * *hedef-arabellek*, const size_t *length*, const char * *kaynak-buffer*, ImqError & *hata-nesne*, const char *pad* = 0);

Copies up to *uzunluk* bytes from *kaynak arabellek* to *hedef-arabellek*. If the number of characters in *kaynak arabellek* is insufficient, fills the remaining space in *hedef-arabellek* with *doldurma* characters. *kaynak-arabelleği* sıfır olabilir. *uzunluk* aynı zamanda sıfır *hedef-arabellek* sıfır olabilir. Herhangi bir hata kodu *hata-nesnesi* içinde 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** , 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 *görelî konum*) 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 **karakteri** içinde bir başvuru oluşturmazsa, boş bir ImqString döndürür.

void işleci = (const ImqString & *dizgi*);

Eşgörünüm verilerini, var olan eşgörünüm verilerini değiştirerek *dizgi* ' den kopyalar.

ImqString işleci + (const char *c*) const;

Returns the result of appending *C* to the **karakter**.

ImqString işleci + (const char * *metin*) const;

Returns the result of appending *metin* to the **karakter**. 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 işleci + = (const char * metin);

metin 'ı **karakter**' a ekler.

void işleci + = (const ImqString & dizgi);

dizgi 'ı **karakter**' a ekler.

void operator + = (const double sayı);

Metne dönüştürdükten sonra **karakter** başına *sayı* ekler.

void işleci + = (const long sayı);

Metne dönüştürdükten sonra **karakter** başına *sayı* 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 & dizgi) 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 & dizgi) const;

karakter yöntemini, **karşılaştır** yöntemini kullanarak *dizgi* ile karşılaştırır. TRUE ya da FALSE değerini döndürür.

ImqBoolean işleci != (const ImqString & dizgi) 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 ImqString değerinden az değer olarak kabul edilir.

ImqBoolean copyOut(char * buffer, const size_t length, const char pad = 0);

karakter 'dan *arabellek* ' a kadar olan ve *uzunluk* ' a kadar olan kopya kopyalar. If the number of **karakter** is insufficient, fills the remaining space in *arabellek* with *doldurma* characters. *uzunluk* aynı zamanda sıfırsa *arabellek* sıfır olabilir. Başarılı olursa TRUE değerini döndürür.

size_t copyOut(uzun & sayı) const;

Metinden dönüştürme işleminden sonra **karakter** 'den *sayı* ' ı ayarlar ve dönüştürmede yer alan karakter sayısını döndürür. Bu sıfır ise, dönüştürme yapılmamıştır ve *sayı* ayarlanmaz. Dönüştürülebilir bir karakter sırası aşağıdaki değerlerle başlamalıdır:

<blank(s)>

```
<+|->
digit(s)
```

size_t copyOut(ImqString & belirteç, 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. Bu durumda *simge* , o sıraya ayarlanır ve döndürülen değer, önde gelen karakterlerin (c) sayısının toplamını ve sıranın bayt sayısını içerir. Ters durumda, sıfır değerini döndürür ve *simge* olarak ayarlanmaz.

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 kesilebilir. üç kez:

```
strNumbers = "-1 0    +55 "
while ( strNumbers.cutOut( number ) );
number becomes -1, then 0, then 55
leaving strNumbers == " "
```

size_t cutOut(ImqString & belirteç, 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);

Searches for an exact match for *dizgi* somewhere within the **karakter** from offset *görelî konum* onwards. *dizgi* boş değerliyse, *görelî konum* güncellenmeden TRUE değerini döndürür. Eşleşme bulunamazsa, FALSE değerini döndürür (*görelî konum* değerinin değeri artırılmış olabilir). Bir eşleşme bulunursa, TRUE değerini döndürür ve *görelî konum* değerini **karakteri** içindeki *dizgi* görelî konumuna günceller.

size_t length () const;

uzunluk değerini döndürür.

ImqBoolean pasteIn(const double number, const char * format = "%f");

Metne dönüştürdükten sonra **karakter** başına *sayı* 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 sayı);

Metne dönüştürdükten sonra **karakter** başına *sayı* 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 yazdırılamayan 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 *length*);

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*);

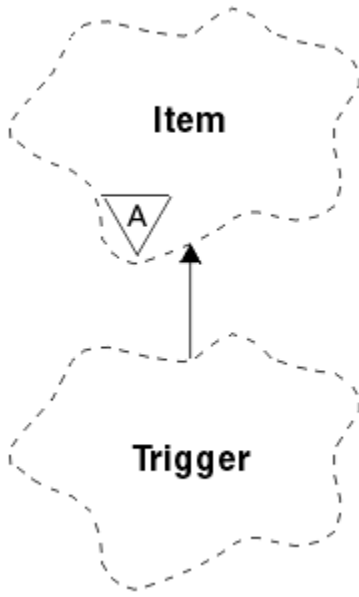
Eşdeğer **işleç** = yöntemine eşdeğer, ancak sanal olmayan bir yöntem. Başarılı olursa TRUE değerini döndürür.

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 68. *ImqTrigger* sınıfı

Bu sınıfın nesneleri 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 1936](#)
- [“Oluşturucular” sayfa 1937](#)
- [“Aşırı yüklenmiş ImqItem yöntemleri” sayfa 1937](#)
- [“Nesne yöntemleri \(genel\)” sayfa 1937](#)
- [“Nesne verileri \(korunan\)” sayfa 1938](#)
- [“Neden kodları” sayfa 1938](#)

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

Süreçe 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.

tetikleme verileri

Süreçe ilişkin verileri tetikler. Başlangıç değeri, boş değerli bir dizgidir.

Kullanıcı verileri

Süreçe ilişkin kullanıcı verileri. Başlangıç değeri, boş değerli bir dizgidir.

Oluşturucular

ImqTrigger();

Varsayılan oluşturucu.

ImqTrigger(const ImqTrigger & tetikleme);

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 1880](#) adresindeki ImqItem sınıf yöntemi açıklamasına bakın.

sanal ImqBoolean pasteIn (ImqMessage & msg);

İleti arabelleğinden bir MQTM veri yapısını okur.

Başarılı olmak için, ImqMessage **biçim** için MQFMT_TRIGGER değeri olmalıdır.

Daha ayrıntılı bilgi için, [“ImqItem C++ sınıfı” sayfa 1880](#) adresindeki ImqItem sınıf yöntemi açıklamasına bakın.

Nesne yöntemleri (genel)

void işleç = (const ImqTrigger & tetikleme);

Eşgörünüm verilerini, var olan eşgörünüm verilerini değiştirerek *tetikleyiciden* kopyalar.

ImqString applicationId () const ;

Uygulama tanıtıcısı' nın bir kopyasını döndürür.

void setApplicationId (const char * id);

Uygulama tanıtıcısı' yı ayarlar.

MQUZE applicationType () const ;

Uygulama tipideğerini döndürür.

void setApplicationType (const MQlong tip);

uygulama tipi' i ayarlar.

ImqBoolean copyOut (MQTMC2 * ptmc2);

Başlatma kuyruklarında alınan MQTM veri yapısını kapsüller. Çağırان 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 verileri' nin bir kopyasını döndürür.

void setEnvironmentData (const char * data);

Ortam verileriniayarlar.

ImqString processName () const ;

Süreç adı' nin bir kopyasını döndürür.

void setProcessAd (const char * ad);

Boşluklarla doldurulan **işlem adını**48 karakter olarak ayarlar.

ImqString queueName () const ;

kuyruk adı' nin bir kopyasını döndürür.

void setQueueAd (const char * ad);

Kuyruk adı' nin, boşluklarla 48 karakter arasında doldurulması sağlar.

ImqString triggerData () const ;

Tetikleme verilerininbir kopyasını döndürür.

void setTriggerData (const char * data);

Tetikleme verileriniayarlar.

ImqString userData () const ;

Kullanıcı verileri' nin bir kopyasını döndürür.

void setUserData (const char * data);

Kullanıcı verileriniayarlar.

Nesne verileri (korunan)

MQTM omqtm

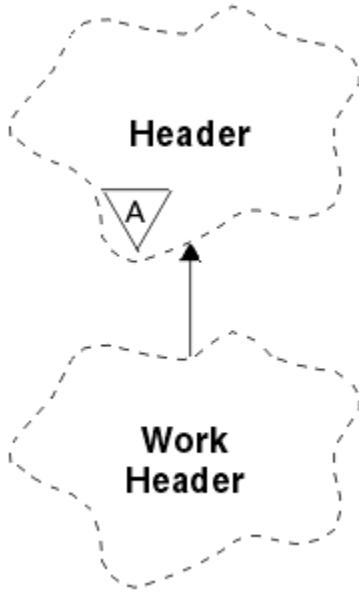
MQTM veri yapısı.

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 69. *ImqWorkÜstbilgi* sınıfı

Bu sınıfın nesneleri, iletileri z/OS Workload Manager tarafından yönetilen kuyruğa ileti koyan uygulamalar tarafından kullanılır.

- “Nesne öznitelikleri” sayfa 1939
- “Oluşturucular” sayfa 1939
- “Aşırı yüklenmiş ImqItem yöntemleri” sayfa 1939
- “Nesne yöntemleri (genel)” sayfa 1940
- “Nesne verileri (korunan)” sayfa 1940
- “Neden kodları” sayfa 1940

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 adımı

İşlem içindeki bir adımın 8 karakterden oluşan adı. Ad, başlangıçta boşluklara neden olur.

Oluşturucular

ImqWorkÜstbilgisi ();

Varsayılan oluşturucu.

ImqWorkÜstbilgisi (const ImqWorkÜstbilgisi & üstbilgi);

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 olarak 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 & üstbilgi);

Eşgörünüm verilerini, var olan eşgörünüm verilerini değiştirerek *üstbilgisinden* kopyalar.

ImqBinary messageToken () const;

İleti simgesideğerini 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, 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 bir tip dönüştürme yapmanız gerekebilir; örneğin, (MQBYTE *) MQMTOK_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 226 sayfa 1941](#) , 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 nesnelerine ilişkin özellikler arasındaki bağımlılıklar” sayfa 1944.

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 sınırlandırılır.

Çizelge 226. Özellik adları ve geçerli nesne tipleri									
Özellik	Kısa Biçim	Nesne tipi							
		CF	QCF	TCF	Q	T	XACF	XAQCF	XATCF
“UYGULAMA ADI” sayfa 1945	appName	Y	Y	Y			Y	Y	Y
“ASYNCEXCEPTION” sayfa 1946	AEX	Y	Y	Y			Y	Y	Y
“BROKERCCDURSUBQ” sayfa 1947 ¹	CCDSUB					Y			
“BROKERCCSUQ” sayfa 1947 ¹	CCSID ' LER	Y		Y			Y		Y
“BROKERCONQ” sayfa 1948 ¹	SIMGE	Y		Y			Y		Y
“BROKERDURSUQ” sayfa 1948 ¹	BDSUB					Y			
“BROKERPUBQ” sayfa 1949 ¹	PUB	Y		Y		Y	Y		Y
“BROKERPUBQMGR” sayfa 1949 ¹	BPQM					Y			
“BROKERQMGR” sayfa 1950 ¹	BQM	Y		Y			Y		Y
“BROKERSUBQ” sayfa 1950 ¹	BSUB	Y		Y			Y		Y
“BROKERVER” sayfa 1950 ¹	BVER	E ²		E ²		Y	Y		Y
“CCDTURL” sayfa 1951 ³	CCDT	Y	Y	Y			Y	Y	Y
“CCSID” sayfa 1951	CCS	Y	Y	Y	Y	Y	Y	Y	Y
“Kanal” sayfa 1952 ³	Chan	Y	Y	Y			Y	Y	Y
“Temizle” sayfa 1952 ¹	CL	Y		Y			Y		Y
“CLEANUPINT” sayfa 1953 ¹	YNT	Y		Y			Y		Y
“CONNECTIONNAMELIST” sayfa 1953	CNIST	Y	Y	Y					
“İSTEMCILERİYÜZLER” sayfa 1954	ÇAPRAZ	Y	Y	Y					
“CLIENTRECONNECTTIMEOUT” sayfa 1955	CRT	Y	Y	Y					
“CLIENTID” sayfa 1955	CID	E ²	Y	E ²			Y	Y	Y
“ÇALIŞTIRMA” sayfa 1955	CLS	Y		Y			Y		Y
“KARMAŞIK” sayfa 1956	HC	Y		Y			Y		Y
“MSG” sayfa 1956	MC	Y	Y	Y			Y	Y	Y

Çizelge 226. Özellik adları ve geçerli nesne tipleri (devamı var)

Özellik	Kısa Biçim	Nesne tipi							
		CF	QCF	TCF	Q	T	XACF	XAQCF	XATCF
“HAYIR” sayfa 1957	CNOPT	Y	Y	Y			Y	Y	Y
“CONNTAG” sayfa 1958	CNTAG	Y	Y	Y			Y	Y	Y
“AÇIKLAMA” sayfa 1958	ALÇ	E ²	Y	E ²	Y	Y	Y	Y	Y
“YÖN YÖNTEMÇİ” sayfa 1959	DAUTH	E ²		E ²					
“Kodlama” sayfa 1959	ENC				Y	Y			
“Son kullanma tarihi” sayfa 1960	ÜS				Y	Y			
“FAILIFQUIESCE” sayfa 1961	FIQ	Y	Y	Y	Y	Y	Y	Y	Y
“Anasistem adı” sayfa 1961	Anasistem	E ²	Y	E ²			Y	Y	Y
“YEREL ADRES” sayfa 1962	LA	E ²	Y	E ²			Y	Y	Y
“MAPNAMESTILI” sayfa 1962	MNST	Y	Y	Y			Y	Y	Y
“MAXBUFFSIZE” sayfa 1963	MBSZ	E ²		E ²					
“MDREAD” sayfa 1963	MDR				Y	Y			
“MDWRITE” sayfa 1964	MDC				Y	Y			
“MDMSGCTX” sayfa 1964	MDCTX				Y	Y			
“MSGBATCHSZ” sayfa 1965 ¹	MBS	Y	Y	Y			Y	Y	Y
“MSGONT” sayfa 1965	GÖVDE				Y	Y			
“MSGRECALOG” sayfa 1966	MRET	Y	Y				Y	Y	
“MSGSEÇEK” sayfa 1967 ¹	MSEL	Y		Y			Y		Y
“ÇOK YA” sayfa 1967	MCET	E ²		E ²		Y			
“EN İYİ BİLDİRİM” sayfa 1968 ¹	OPTPUB	Y		Y					
“KALDIRIM” sayfa 1968 ¹	Bildirim Gönder	Y		Y					
“Kalıcılık” sayfa 1969	HER BİRİ İÇİN				Y	Y			
“GİRİŞ TARİHİ” sayfa 1969 ¹	YAZI	Y	Y	Y			Y	Y	Y
“PORT” sayfa 1970	PORT	E ²	Y	E ²			Y	Y	Y
“Öncelik” sayfa 1970	PRI				Y	Y			
“SÜREÇ” sayfa 1971 ¹	PROC DUR	Y		Y					
“SAĞLAMA SÜ” sayfa 1971	PVER	Y	Y	Y			Y	Y	Y
“PROXYHOSTNAME” sayfa 1974	PHOST	E ²		E ²					
“TANITIM” sayfa 1974	PKAPI	E ²		E ²					
“PUBKACT” sayfa 1975 ¹	PAI	Y		Y			Y		Y
“PUTASINCALLI” sayfa 1975	PAALD				Y	Y			
“YÖNETİM” sayfa 1976	MMGR	Y	Y	Y	Y		Y	Y	Y

Çizelge 226. Özellik adları ve geçerli nesne tipleri (devamı var)

Özellik	Kısa Biçim	Nesne tipi							
		CF	QCF	TCF	Q	T	XACF	XAQCF	XATCF
“kuyruk” sayfa 1976	QU				Y				
“OKUNABİLENİZİN” sayfa 1977	RALAND				Y	Y			
“READAHEADCLOSEPOLICY” sayfa 1977	RACP				Y	Y			
“RECEIVECCSTnt” sayfa 1978	RCCS				Y	Y			
“ALINMASI” sayfa 1978	RCNV				Y	Y			
“RECEIVEI” sayfa 1979 ¹	RCVISOL	Y		Y					
“YENİDEN çıkış” sayfa 1979	RCX	Y	Y	Y			Y	Y	Y
“YENİDEN KALDIRMA” sayfa 1980	RRCXI	Y	Y	Y			Y	Y	Y
“REPLYTOSTYLE” sayfa 1980	RTOST				Y	Y			
“SıFIRA” sayfa 1981 ¹	AYIR	Y	Y				Y	Y	
“SEÇİLEN” sayfa 1981	SCX	Y	Y	Y			Y	Y	Y
“GÜNCELLEMEYİ” sayfa 1982	SCXI	Y	Y	Y			Y	Y	Y
“SENDCHECKCOUNT” sayfa 1982	Gizli	Y	Y	Y			Y	Y	Y
“SENDEXIT” sayfa 1983	SSDX	Y	Y	Y			Y	Y	Y
“SENDEXITINIT” sayfa 1983	SXI	Y	Y	Y			Y	Y	Y
“SENABRALD” sayfa 1984	SALD	Y	Y	Y			Y	Y	Y
“SPARSESUBS” sayfa 1984 ¹	SSUBS	Y		Y					
“SSLCIPHERSUIT” sayfa 1985	SCPHS	Y	Y	Y			Y	Y	Y
“SLSCRL” sayfa 1985	SCRL	Y	Y	Y			Y	Y	Y
“SSLFIPSREQUIRE” sayfa 1986	SFIPS	Y	Y	Y			Y	Y	Y
“SLAYICI ADI” sayfa 1986	SPEER	Y	Y	Y			Y	Y	Y
“SSLRESTCOUNT” sayfa 1987	src	Y	Y	Y			Y	Y	Y
“DEYİM YENİDEN” sayfa 1987 ¹	SRI	Y		Y			Y		Y
“ALT DÜZEN” sayfa 1988 ¹	SS	Y		Y			Y		Y
“SYNCPOINTALLGETS” sayfa 1988	SPAG	Y	Y	Y			Y	Y	Y
“HEDEF İSTEMCİ” sayfa 1989	TC				Y	Y			
“TARGCLIENTEŞLEŞTİRME” sayfa 1989	TCM	Y	Y				Y	Y	
“TEMPMODEL” sayfa 1990	TM	Y	Y				Y	Y	
“TEMPQPREFIX” sayfa 1990	TQP	Y	Y				Y	Y	
“TEMPTOPICFIX” sayfa 1991	TTP	Y		Y			Y		Y
“Konu” sayfa 1991	TOP					Y			
“Aktarım” sayfa 1991	TRAN	E ²	Y	E ²			Y	Y	Y

Çizelge 226. Özellik adları ve geçerli nesne tipleri (devamı var)

Özellik	Kısa Biçim	Nesne tipi							
		CF	QCF	TCF	Q	T	XACF	XAQCF	XATCF
"JOKER BIÇIMI" sayfa 1992	WCFMT	Y		Y			Y		Y

Not:

1. Bu özellik, IBM MQ classes for JMS ' un Sürüm 7.0 ile birlikte kullanılabilir, ancak bağlantı üreticisinin PROVIDERVERSION özelliği 7 'den küçük bir sürüm numarasına ayarlanmadıkça, Sürüm 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 CCDURL ve KANAL özellikleri aynı anda ayarlanmamalıdır.

IBM MQ classes for JMS nesnelere ilişkin özellikler arasındaki bağımlılıklar

Bazı özelliklerin geçerliliği, diğer özelliklerin belirli değerlerine bağımlıdır.

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
- CCDURL
- CCSID
- KARMAŞIK
- MSG
- YENİDEN çıkış
- YENİDEN 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 ' tür. 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 DIRECTTHHTTP 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 DIRECTTHHTTP 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:

- YENİDEN KALDIRMA
- GÜNCELLEMİYİ
- SENDEXITINIT

For example, specifying RECEXITINIT(myString) without specifying RECEXIT(some.exit.classname) causes an error.

İlgili başvurular

“Aktarım” sayfa 1991

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 Nesneler

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 sığalacağı 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`.

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 nesneler

`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

ASYNCEXCEPTIONS_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.

Ortam	Değer
JMS Denetim Aracı	TÜMÜ
Programlı	WMQCONSTANTS.ASYNC_EXCEPTIONS_ALL = -1
IBM MQ Gezgin	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. **V 8.0.0.2** This is the default value from IBM MQ 8.0.0, Düzeltme Paketi 2 (see [JMS: Kural dışı durum dinleyicisi değişiklikleri in Version 8.0](#)).

Ortam	Değer
JMS Denetim Aracı	CONNECTIONBOZUK
Programlı	WMQCONSTANTS.ASYNC_EXCEPTIONS_CONNECTIONBROKEN = 1
MQ Explorer	Bağlantı Kesildi

Aşağıdaki ek sabit değer tanımlıdır:

- **V 8.0.0.2** Sürüm 8.0.0, Düzeltme Paketi 2: WMQCONSTANTS.ASYNC_EXCEPTIONS_DEFAULT = ASYNC_EXCEPTIONS_CONNECTIONBROKEN
- Sürüm 8.0.0, Düzeltme Paketi 2 tarihinden önce: WMQCONSTANTS.ASYNC_EXCEPTIONS_DEFAULT = ASYNC_EXCEPTIONS_ALL

İlgili bilgiler

[IBM MQ classes for JMS](#) içinde kural dışı durumlar

BROKERCCDURSUBQ

ConnectionConsumer(Bağlantı Tüketicisi) için dayanıklı abonelik iletilerinin alındığı kuyruğun adı.

Uygulanabilir nesneler

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 nesneler

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 Nesneler

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 nesneler

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 bilgiler

[IBM MQ ileti alışverişi sağlayıcısı kipini seçme kuralları](#)

BROKERPUBQ

Yayınlanan iletilerin gönderildiği kuyruğun adı (akış kuyruğu).

Uygulanabilir Nesneler

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 bilgiler

[IBM MQ ileti alışverişi sağlayıcısı kipini seçme kuralları](#)

BROKERVER

Kullanılmakta olan aracının sürümü.

Uygulanabilir Nesneler

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 Nesneler

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

Bir bağlantı ya da hedef için kullanılacak kodlanmış karakter takımı tanıtıcısı.

Uygulanabilir Nesneler

ConnectionFactory, QueueConnectionFactory, TopicConnectionFactory, Kuyruk, Konu, XAConnectionFactory, XAQueueConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: CCSID

JMS yönetim aracı kısa adı: CCS

Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setCCSID()
- MQConnectionFactory.getCCSID()

Değerler

819

Bu, bir bağlantı üreticisine ilişkin varsayılan değerdir.

1208

Bu, bir hedef için varsayılan değerdir.

Pozitif bir tamsayı

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 YENİ 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.

İSTEMCİLERİYÜZLER

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, yalnızca başlangıçta bağlı olduğu kuyruk yöneticisine yeniden bağlanabilir.

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, herhangi bir kuyruk yöneticisine 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 özneliğinin değerine bağlı olarak DefReconnect' in değerine bağlıdır.

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 Nesneler

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 Nesneler

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 çalıştırılmayacağı.

Uygulanabilir Nesneler

ConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: CLONESUPP

JMS yönetim aracı kısa adı: CLS

Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setCloneSupport ()
- MQConnectionFactory.getCloneDesteği ()

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 kullanılabilecek tekniklerin listesi.

Uygulanabilir Nesneler

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 kullanılabilecek tekniklerin listesi.

Uygulanabilir Nesneler

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_STANDARD_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 bağlıyken, bir iş birimi içinde uygulama tarafından güncellenen kaynaklarla ilişkilendirdiği bir etiket.

Uygulanabilir Nesneler

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 Nesneler

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 YÖNTEMÇİ

SSL kimlik doğrulamasının bir aracıya gerçek zamanlı bir bağlantıda kullanılıp kullanılmadığını.

Uygulanabilir Nesneler

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 Nesneler

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* ı 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)
```

Son kullanma tarihi

Bir hedefteki iletilerin süresinin dolduğu saat.

Uygulanabilir Nesneler

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 Nesneler

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 Nesneler

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 kapıyı ya da yerel kapı aralığını belirtir.

Uygulanabilir Nesneler

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 stiline kullanıldığını sağlar.

Uygulanabilir Nesneler

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, Sürüm 5.3 öncesi bir IBM MQ classes for JMS sürümünü kullanan bir uygulamaya eşleme iletileri 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 Nesneler

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 Nesneler

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 hiçbirisi 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 belirtik olarak ayarlamadığı özellikler de içinde olmak üzere, MQMD 'deki güncellenmiş alan değerlerini yansıtacak şekilde güncellenir. İleti alınırken, tüm JMS_IBM_MQMD* özellikleri, gönderenin belirtik olarak ayarlamadığı özellikler de içinde olmak üzere, alınan bir iletilerde kullanılabilir.

Programlar için True (Doğru) seçeneğini kullanın.

MDWRITE

Bu özellik, bir JMS uygulamasının MQMD alanlarının değerlerini ayarlayıp ayarlayamayacağını belirler.

Uygulanabilir Nesneler

Kuyruk, Konu

JMS yönetim aracı uzun adı: MDWRITE

JMS yönetim aracı kısa adı: MDR

Programlı erişim

Ayarlayıcılar/alıcılar

- MQDestination.setMQMDWriteEnabled()
- MQDestination.getMQMDWriteEnabled()

Değerler

HAYIR

Tüm JMS_IBM_MQMD* özellikleri yoksayılır ve değerleri, temeldeki MQMD yapısıyla kopyalanmaz. Bu, denetim araçları için varsayılan değerdir.

Programlar için False değerini kullanın.

EVET

JMS_IBM_MQMD* özellikleri işlendi. Değerleri, temeldeki MQMD yapısına kopyalanır.

Programlar için True (Doğru) seçeneğini kullanın.

MDMSGCTX

What level of message context is to be set by the JMS application. Uygulamanın, bu özelliğin geçerli olması için uygun bağlam yetkisi ile çalışıyor olması gerekir.

Uygulanabilir Nesneler

JMS yönetim aracı uzun adı: MDMSGCTX

JMS yönetim aracı kısa adı: MDCTX

Programlı erişim

Ayarlayıcılar/alıcılar

- MQDestination.setMQMDMessageContext()
- MQDestination.getMQMDMessageContext()

Değerler

VARSAYILAN

MQOPER API çağrısı ve MQPMO yapısı belirttik ileti bağlamı seçeneklerini belirtmiyor. Bu, denetim araçları için varsayılan değerdir.

Programlar için WMQ_MDCTX_VARSAYILAN değeri kullanın.

SET_IDENTITY_CONTEXT

MQOPER API çağrısı, 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

MQOPER API çağrısı, 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 Nesneler

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 alınmadır.

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 V1 değerine sahip olduğunda, aracıya göre ileti seçimi desteklenmez.

Uygulanabilir Nesneler

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, aracından 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 Nesneler

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 edilmiyor. ConnectionFactory ve TopicConnectionFactory nesneleri 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 nesneleri için geçerlidir ve Konu nesneleri 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 Nesneler

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.

EVET

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 Nesneler

ConnectionFactory, TopicConnectionFactory

JMS yönetim aracı uzun adı: OUTCOMPENOTIFICATION

JMS yönetim aracı kısa adı: NOTIFY

Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setOutcomeNotification ()
- MQConnectionFactory.getOutcomeNotification ()

Değerler

EVET

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 Nesneler

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 değeriine sahip olduğunda ilişkilidir.

Uygulanabilir Nesneler

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 Nesneler

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 Nesneler

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 yükseğ

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 Nesneler

ConnectionFactory, TopicConnectionFactory

JMS yönetim aracı uzun adı: PROCESSDURATION

JMS yönetim aracı kısa adı: PROC DUR

Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setProcessSüre ()
- MQConnectionFactory.getProcessSüre ()

Değerler

BİLİNİYOR

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 eniyilenir. Kısıtlamalarla IBM MQ ileti alışverişi sağlayıcısı normal kipi, JMS 2.0 API 'sini kullanır, ancak paylaşılan abonelikler, gecikmeli teslim ya da zamanuyumsuz gönderme gibi yeni özellikleri kullanmaz.

Uygulanabilir Nesneler

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 belirtilmemiş (varsayılan değer) herhangi birine ayarlayabilirsiniz. **PROVIDERVERSION** özelliği için belirttiğiniz değer bir dizgi olmalıdır. 8, 7 ya da 6 seçeneklerinden birini belirtiyorsanız, bunu aşağıdaki biçimlerden birinde de yapabilirsiniz:

- V.R.M.F
- V.R.M
- V.R
- V

Burada V, R, M ve F, sıfıra eşit ya da sıfırdan büyük tamsayı değerleridir. Ek R, M ve F değerleri isteğe bağlıdır ve büyük parçalı denetime gerek duyulması durumunda kullanmanız için kullanılabilir. Örneğin, **PROVIDERVERSION** düzeyi 7 kullanmak istediyseniz, **PROVIDERVERSION**=7, 7.0, 7.0.0 ya da 7.0.0.0 ayarlayabilirsiniz.

8 -Normal kip

JMS uygulaması, IBM MQ ileti alışverişi sağlayıcısı olağan kipini kullanır. Normal kip, JMS 'yi uygulamak için bir IBM MQ kuyruk yöneticisinin tüm özelliklerini kullanır. Bu kip, JMS 2.0 API 'sini ve işlevlerini kullanmak için eniyilenir.

Komut düzeyi 800 olan bir kuyruk yöneticisine bağlanıyorsanız, zamanuyumsuz gönderme, gecikmeli teslim alma ya da paylaşılan abonelik gibi tüm JMS 2.0 API 'si ve özellikleri kullanılabilir.

Bağlantı üreticisi ayarlarında belirtilen kuyruk yöneticisi bir Sürüm 8.0.0 kuyruk yöneticisi değilse, createConnection yöntemi bir kural dışı durum JMSFMQ0003 ile başarısız olur.

IBM MQ ileti alışverişi sağlayıcısı normal kipi, paylaşım etkileşimleri özelliğini kullanır ve paylaşılacak etkileşim sayısı, sunucu bağlantı kanalında **SHARECNV()** özelliği tarafından denetlenir. Bu özellik 0 olarak ayarlanırsa, IBM MQ ileti alışverişi sağlayıcısı normal kipini kullanamazsınız ve createConnection yöntemi bir kural dışı durumla başarısız olur: JMSCC5007.

7 -Kısıtlamalarla birlikte normal kip

The JMS application uses the IBM MQ messaging provider normal mode with restrictions. Bu kip JMS 2.0 API 'sini kullanır, ancak paylaşılan abonelikler, gecikmeli teslim ya da zamanuyumsuz gönderme gibi yeni özellikleri kullanmaz.

PROVIDERVERSION ' u 7 olarak ayarlarsanız, kısıtlamalarla ilgili olarak yalnızca IBM MQ ileti alışverişi sağlayıcısı normal bir işlem kipinde kullanılabilir. Bağlantı üreticisi ayarlarında belirtilen kuyruk yöneticisi Sürüm 7.0.1 değil ya da daha sonra kuyruk yöneticisi değilse, createConnection yöntemi JMSFCC5008 kural dışı durumuyla başarısız olur.

Kısıtlamaları olan normal kipi kullanarak bağlantı kullanıyorsanız, 700 ile 800 arasında bir komut düzeyi olan bir kuyruk yöneticisine, daha sonra JMS 2.0 API 'sini kullanabilirsiniz,

ancak zamanuysuz gönderme, gecikmeli teslim alma ya da paylaşılan abonelik özellikleri kullanamazsınız.

Kısıtlamaları olan IBM MQ ileti alışverişi sağlayıcısı normal kipi, paylaşma etkileşimleri özelliğini kullanır ve paylaşılabilir etkileşim sayısı, sunucu bağlantı kanalında **SHARECNV()** özelliği tarafından denetlenir. Bu özellik 0 olarak ayarlanırsa, kısıtlamalarla IBM MQ ileti alışverişi sağlayıcısı normal kipini kullanamazsınız ve createConnection yöntemi bir kural dışı durum ile başarısız olur: JMSCC5007.

6 -Geçiş kipi

JMS uygulaması, IBM MQ ileti alışverişi sağlayıcısı geçiş kipini kullanır.

IBM MQ classes for JMS , IBM WebSphere MQ Sürüm 6.0 ile birlikte verilen özellikleri ve algoritmaları kullanır. IBM WebSphere MQ Enterprise Transport Sürüm 6.0 kullanarak WebSphere Message Broker Sürüm 6.0 ya da 6.1 ile bağlantı kurmak istiyorsanız, bu kipi kullanmanız gerekir. You can connect to a IBM MQ 8.0 queue manager using this mode, but none of the new features of a IBM MQ classes for JMS queue manager are used, for example, read ahead or streaming.

IBM MQ 8.0 kuyruk yöneticisine bağlanan bir IBM MQ 8.0 istemciniz varsa, ileti seçimi istemci sistemi yerine kuyruk yöneticisi tarafından yapılır.

If IBM MQ messaging provider migration mode is specified and you attempt to use any of the JMS 2.0 API, the API method call fails with the exception JMSCC5007.

belirtilmeyen (varsayılan)

PROVIDERVERSION özelliği varsayılan olarak *belirtilmemiş* olarak ayarlanır.

A connection factory that was created with a previous version of IBM MQ classes for JMS in JNDI takes this value when the connection factory is used with the new version of IBM MQ classes for JMS. Hangi işlemin hangi modda kullanıldığını belirlemek için aşağıdaki algoritma kullanılır. Bu algoritma, createConnection yöntemi çağrıldığında ve bağlantı üreticisinin IBM MQ ileti alışverişi sağlayıcısı normal kipinin, kısıtlamaların olduğu normal kipi ya da IBM MQ ileti alışverişi sağlayıcısı geçiş kipinin gerekli olup olmadığını belirlemek için bağlantı üreticisinin diğer özelliklerini kullanıyorsa kullanılır.

1. İlk olarak, IBM MQ ileti alışverişi sağlayıcısı normal kipini kullanma girişiminde bulunuldu.
2. Kuyruk yöneticisi bağlantısı IBM MQ 8.0 değilse ya da sonraki bir sürümü değilse, kısıtlamaları olan IBM MQ ileti alışverişi sağlayıcısı normal kipini kullanma girişiminde bulunmanız gerekir.
3. Kuyruk yöneticisi bağlantısı IBM WebSphere MQ Sürüm 7.0.1 değilse ya da sonraki bir sürümü değilse, bağlantı kapatılır ve bunun yerine IBM MQ ileti alışverişi sağlayıcısı geçiş kipi kullanılır.
4. Sunucu bağlantı kanalındaki **SHARECNV** özelliği 0 olarak ayarlandıysa, bağlantı kapatılır ve bunun yerine IBM MQ ileti alışverişi sağlayıcısı geçiş kipi kullanılır.
5. **BROKERVER** , V1 değerine ya da varsayılan *belirtilmemiş* değerine ayarlıysa, IBM MQ ileti alışverişi sağlayıcısı olağan kipi kullanılmaya devam eder ve bu nedenle, yayınlama/abone olma işlemleri yeni IBM WebSphere MQ Sürüm 7.0.1 ya da sonraki özellikleri kullanır.

Uyumluluğa ilişkin ek bilgi için ALTER QMGR komutunun PSMODE parametresiyle ilgili ek bilgi için [ALTER QMGR](#) başlıklı konuya bakın.

6. **BROKERVER** V2 olarak ayarlanırsa, alınan eylem **BROKERQMGR** değerinin değerine bağlıdır:

• **BROKERQMGR** boş ise:

BROKERCONQ özelliği tarafından belirtilen kuyruk çıkışı için açılabilir (yani, çıkış için MQOPEN başarılı olur) ve kuyruk yöneticisinde **PSMODE** COMPAT ya da DISABLE olarak ayarlandıysa, IBM MQ ileti alışverişi sağlayıcısı geçiş kipi kullanılır.

- **BROKERCONQ** özelliği tarafından belirtilen kuyruk çıkışı için açılmazsa ya da **PSMODE** özniteliği ENABETLE : olarak ayarlandıysa

IBM MQ ileti alışverişi sağlayıcısı olağan kipi kullanılır.

- **BROKERQMGR** boş olmayan ise:

IBM MQ ileti alışverişi sağlayıcısı geçiş kipi kullanılır.

Kullanmakta olduğunuz bağlantı üreticisini değiştiremezseniz, bağlantı üreticisindeki herhangi bir ayarı geçersiz kılmak için `com.ibm.msg.client.wmq.overrideProviderVersion` özelliğini kullanabilirsiniz. Bu geçersiz kılma işlemi, JVM ' deki tüm bağlantı fabrikaları için geçerlidir, ancak gerçek bağlantı üreticisi nesneleri değiştirilmez.

İlgili bilgiler

JMS **PROVIDERVERSION** özelliğini yapılandırma

[IBM MQ ileti alışverişi sağlayıcısı kipini seçme kuralları](#)

PROXYHOSTNAME

Yetkili sunucu aracılığıyla bir aracıya gerçek zamanlı bağlantı kullanılırken yetkili sunucunun çalıştığı sistemin ana makine adı ya da IP adresi.

Uygulanabilir Nesneler

ConnectionFactory, TopicConnectionFactory

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 Nesneler

ConnectionFactory, TopicConnectionFactory

JMS yönetim aracı uzun adı: PROXYPET

JMS yönetim aracı kısa adı: PPORT

Programlı erişim

Ayarlayıcılar/alıcılar

MQConnectionFactory.setProxyPort ()

MQConnectionFactory.getProxyKapası ()

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 alçalttığınızda, 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 Nesneler

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 Nesneler

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](#) istemci kanalı tanımlama çizelgesi kullanılması başlıklı konuya bakın.

Uygulanabilir Nesneler

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 Nesneler

Kuyruk

JMS yönetim aracı uzun adı: QUEUE

JMS yönetim aracı kısa adı: QU

Değerler

Herhangi bir dizgi

Geçerli bir IBM MQ kuyruğu adı.

İlgili bilgiler

[IBM MQ nesnelerini 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 Nesneler

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.

EYET

İlerlemenize izin verilir.

Programlardaki WMQConstants.WMQ_READ_AHEAD_ALLOWED_ENABLED seçeneğini kullanın.

READAHEADCLOSEPOLICY

Zamanuysuz bir ileti dinleyicisine gönderilmekte olan iletiler için, ileti tüketicisi kapatıldığında iç okuma arabelleğindeki iletilere ne olur.

Uygulanabilir Nesneler

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 dönmekten 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_QMGR` olarak ayarlanmadıkça, değer yoksayılır.

Uygulanabilir Nesneler

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 Nesneler

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 Nesneler

ConnectionFactory, TopicConnectionÜreticisi

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.

YENIDEN çıkış

Bir kanal alma çıkışı ya da art arda çalıştırılacak 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 Nesneler

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

boş değerli

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).

Bu varsayılan değerdir.

YENIDEN KALDIRMA

Kanala geçirilen kullanıcı verileri, çağrıldığında çıkış çıkışlarına çıkar.

Uygulanabilir Nesneler

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 öğesini virgüllerle ayrılmış bir dizgi. Bu varsayılan değerdir.

REPLYTOSTYLE

Alınan bir iletteki JMSReplyTo alanının nasıl oluşturulacağını belirler.

Uygulanabilir Nesneler

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 6.0.2.5varsayılan 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 değıştirgenin değeri önem göstermez.

SIFIRA

Noktadan noktaya iletişim etki alanındaki bir ileti tüketicisi, almak istediğı iletileri seçmek için bir ileti seçicisini kullandığında, IBM MQ classes for JMS kuyruğun MsgDeliverySequence özneteliğı tarafından belirlenen sırayla uygun iletiler için IBM MQ kuyruğunu arar.

IBM MQ classes for JMS uygun bir ileti bulup tüketicie teslim ettikten sonra, IBM MQ classes for JMS , kuyrukta 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 kuyrukta arama yapmaya devam eder. Her durumda, IBM MQ classes for JMS aramayı devam ettirmek için kuyruğun başına dönün ve yeni bir zaman aralığı kesinleştirmesi 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 değeri olabilir.

SEÇİLEN

Kanal güvenliğı 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

boş değerli

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 çıkışı 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

boş değerli

Her bir öğenin aşağıdakilerden biri olduğu virgüllerle ayrılmış olarak bir ya da daha çok öğeyi oluşturan herhangi bir dizgi:

- 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).
-

Bu varsayılan değerdir.

SENDEXITINIT

Kanal gönderme çıkışlarına geçirilen kullanıcı verileri çağrıldığında bu verileri gönderir.

Uygulanabilir Nesneler

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 ögesini içeren dizilimlerden biri varsayılan değer olabilir.

SENABRALD

Bu özellik, kanal tanımlarının eşleşmesi durumunda, bir istemci bağlantısının yuvasını aynı kuyruk yöneticisiyle aynı işlem düzeyinden diğer üst düzey JMS bağlantılarıyla paylaşıp paylaşamayacağını belirler.

Uygulanabilir Nesneler

ConnectionFactory, QueueConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XAQueueConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: SHARECVALLED

JMS yönetim aracı kısa adı: SCALD

Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setShareConvAllowed()
- MQConnectionFactory.getShareConvAllowed()

Değerler

EVET

Bu, denetim araçları için varsayılan değerdir.

Programlar için WMQConstants.WMQ_SHARE_CONV_ALLOWED_YES ögesini kullanın.

HAYIR

Bu değer, denetim araçları içindir.

Programlar için WMQConstants.WMQ_SHARE_CONV_ALLOWED_NO kullanın.

SPARSESUBS

Bir TopicSubscriber nesnesine ilişkin ileti alma ilkesini denetler.

Uygulanabilir Nesneler

ConnectionFactory, TopicConnectionFactory

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.

SSLCIPHERSUIT

SSL bağlantısı için kullanılacak CipherSuite .

Uygulanabilir Nesneler

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 değerli

Bu varsayılan değerdir. Daha fazla bilgi için bakınız: [SSL properties of JMS objects](#).

SLSCRL

SSL sertifikasının iptal olup olmadığını denetlemek için kullanılan CRL sunucuları.

Uygulanabilir Nesneler

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 değerli

LDAP URL ' lerinin boşlukla ayrılmış listesi. Bu varsayılan değerdir. Daha fazla bilgi için bakınız: [SSL properties of JMS objects](#).

SSLFIPSREQUIRD

Bu özellik, bir SSL bağlantısının IBM Java JSSE FIPS sağlayıcısı (IBMJSEFIPS) tarafından desteklenen bir CipherSuite ' i kullanıp kullanmayacağını belirler.

Uygulanabilir Nesneler

ConnectionFactory, QueueConnectionFactory, TopicConnectionFactory, XAConnectionFactory, XAQueueConnectionFactory, XATopicConnectionFactory

JMS yönetim aracı uzun adı: SSLFIPSREQUIRD

JMS yönetim aracı kısa adı: SFIPS

Programlı erişim

Ayarlayıcılar/alıcılar

- MQConnectionFactory.setSSLFipsRequired ()
- MQConnectionFactory.getSSLFipsRequired ()

Değerler

HAYIR

Bir SSL bağlantısı, IBM Java JSSE FIPS sağlayıcısı (IBMJSEFIPS) tarafından desteklenmeyen herhangi bir CipherSuite ' i kullanabilir.

Bu varsayılan değerdir. Programlarda false değerini kullanın.

EVET

Bir SSL bağlantısının, IBMJSSEFIPS tarafından desteklenen bir CipherSuite kullanması gerekir. Programlarda true (doğru) seçeneğini kullanın.

SLAYICI ADI

SSL 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: [SSL properties of JMS objects](#).

SSLRESETCOUNT

SSL 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: [SSL properties of JMS objects](#).

DEYİM YENIDEN

Bir abonenin kuyruk yöneticisiyle bağlantısını kaybettiğinde algılayan uzun süre çalışan işlemin yenilenmesi arasındaki aralık (milisaniye).

Bu özellik, yalnızca SUBSTORORE 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: [SSL 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_QUEUEseçeneğini kullanın.

SYNCPOINTALLGETS

Bu özellik, tüm alımların uyumluluk noktası altında gerçekleştirilip gerçekleştirilmeyeceğini belirler.

Uygulanabilir Nesneler

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 Nesneler

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 örnek.

Uygulanabilir Nesneler

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 örnek, diğer tüm altyapılarda z/OS ve AMQ . * üzerinde CSQ . * örnekidir. 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 Nesneler

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 bilgiler

[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_TCIPseçeneğini 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 nesnelerine ilişkin özellikler arasındaki bağımlılıklar” sayfa 1944](#)

Bazı özelliklerin geçerliliği, diğer özelliklerin belirli değerlerine bağlıdır.

JOKER BİÇİMİ

Bu özellik, kullanılacak genel arama karakteri sözdiziminin hangi sürümünü kullanılacağını belirler.

Uygulanabilir Nesneler

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_ONLYöğesini kullanın.

YALNIZCA

Yalnızca, aracı sürüm 1 'de kullanıldığı gibi, karakter genel arama karakterlerini tanır.

Programlar için WMQConstants.WMQ_WILDCARD_CHAR_ONLYöğesini kullanın.

ENCODING özelliği

ENCODING özelliği, on iki olası birleşimde üç alt özelliiden 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*ı 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)
```

JMS nesnelere ilişkin SSL özellikleri

SSLCIPHERSUITE özelliği kullanılarak SSL (Secure Sockets Layer; Güvenli Yuva Katmanı) şifrelemesini etkinleştirin. Bundan sonra, SSL şifrelemesinin özelliklerini birkaç diğer özellik kullanarak değiştirebilirsiniz.

TRANSPORT (CLIENT) belirttiğinizde, SSL (Secure Sockets Layer; Güvenli Yuva Katmanı) şifrelenmiş iletişimi SSLCIPHERSUITE özelliği kullanılarak geçerli kılınabilir. 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 SSL 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. [SSL/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 yoksayılar.

SSLPEERNAME, kanal tanımlarında ayarlanabilen SSLPEER parametresinin biçimiyle eşleşir. Bu, virgülle ya da noktalı virgüllerle ayrılmış öznitelik adı ve değer çiftlerinin listesidir. Örneğin:

```
SSLPEERNAME(CN=QMGR.*, OU=IBM, OU=WEBSPPHERE)
```

Ad ve değeri kümesi, *ayırıcı edici* adlandırır. Ayırıcı edici adlar ve bunların IBM MQ ile kullanımı hakkında daha fazla ayrıntı için bkz. [Güvenlik](#).

Örnek verilen örnek, sunucu tarafından bağlanma sırasında gösterilen sertifikayı denetler. Bağlantının başarılı olması için sertifikada 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 yoksayılır.

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 SSL sertifikası belirtilen CRL sunucularına göre denetlenir. CRL güvenliği hakkında daha fazla bilgi için bkz. [Güvenlik](#).

SSLCRL ayarlanmadıysa, böyle bir denetleme gerçekleştirilmez. SSLCIPHERSUITE ayarlanmadıysa, SSLCRL yoksayılır.

SSLRESETCOUNT özelliği, 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 JMS tarafından gönderilen ve alınan denetim bilgilerini de içerir.

Örneğin, 4 MB 'lık veri akıldıktan sonra yeniden anlaşma sağlanan bir SSL 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 anlaşılabilir. SSLCIPHERSUITE ayarlanmadıysa, SSLRESETCOUNT özelliği yoksayılır.

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. Bulunduğunuz 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 kullanılabileceğini 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 kullanılabilecek 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 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 diğer ülkelerdeki tescilli ticari markasıdır.

Bu ürün, Eclipse Project (<http://www.eclipse.org/>) tarafından geliştirilen yazılımları içerir.

Java ve Java tabanlı tüm markalar ve logolar, Oracle firmasının ve/ya da iştiraklerinin markaları ya da tescilli markalarıdır.



Parça numarası:

(1P) P/N: