

9.4

IBM MQ 構成リファレンス

IBM

注記

本書および本書で紹介する製品をご使用になる前に、[265 ページの『特記事項』](#)に記載されている情報をお読みください。

本書は、IBM® MQ バージョン 9 リリース 4、および新しい版で明記されていない限り、以降のすべてのリリースおよびモディフィケーションに適用されます。

お客様が IBM に情報を送信する場合、お客様は IBM に対し、お客様に対してなんら義務も負うことのない、自ら適切と信ずる方法で情報を使用または配布する非独占的な権利を付与します。

© Copyright International Business Machines Corporation 2007 年, 2024.

目次

構成に関する参照情報	5
例: IBM MQ のクロスプラットフォーム通信のセットアップ.....	5
クロスプラットフォーム通信の使用法の例.....	7
例: AIX での IBM MQ クロスプラットフォーム通信のセットアップ.....	9
例: IBM i での IBM MQ クロスプラットフォーム通信のセットアップ.....	15
例: Linux での IBM MQ クロスプラットフォーム通信のセットアップ.....	32
例: Windows での IBM MQ クロスプラットフォーム通信のセットアップ.....	38
Example: setting up IBM MQ cross-platform communication on z/OS.....	44
Example: setting up IBM MQ cross-platform communication on z/OS using QSGs.....	49
Example: setting up IBM MQ cross-platform communication for intra-group queuing on z/OS.....	57
/var/mqm に適用される IBM MQ ファイル・システム権限.....	64
mqm の setuid が設定された /opt/mqm の IBM MQ のファイル権限.....	69
Windows での IBM MQ ファイル・システム権限.....	70
キューの命名上の制約.....	71
その他のオブジェクトの命名上の制約.....	73
キュー名の解決.....	74
キュー名解決について.....	76
別名キュー、リモート・キュー、クラスター・キューの宛先オブジェクト属性の解決方法.....	77
システムおよびデフォルト・オブジェクト.....	77
SYSTEM.BASE.TOPIC.....	82
構成ファイル・スタンザ情報.....	83
分散キューイング用の構成ファイル・スタンザ.....	86
チャンネルの属性.....	87
MQSC キーワード (A-B) のチャンネル属性.....	92
MQSC キーワードのチャンネル属性 (C).....	96
MQSC キーワード (D-L) のチャンネル属性.....	103
MQSC キーワードのチャンネル属性 (M).....	110
MQSC キーワード (N-R) のチャンネル属性.....	115
MQSC キーワードのチャンネル属性 (S).....	119
MQSC キーワード (T-Z) のチャンネル属性.....	124
IBM MQ クラスターのコマンドおよび属性.....	126
チャンネル定義コマンドで使用可能なクラスター属性.....	127
キュー定義コマンドで使用可能なクラスター属性.....	130
キュー・マネージャー定義コマンドで使用可能なクラスター属性.....	132
DISPLAY CLUSQMGR.....	134
REFRESH CLUSTER.....	136
RESET CLUSTER: クラスターからキュー・マネージャーを強制的に除去する.....	137
SUSPEND QMGR、RESUME QMGR、およびクラスター.....	139
クラスターでのワークロード・バランシング.....	140
Asynchronous behavior of CLUSTER commands on z/OS.....	149
チャンネル・プログラム.....	150
IBM i における相互通信ジョブ.....	150
IBM i におけるチャンネルの状態.....	150
例: AIX, Linux, and Windows でのメッセージ・チャンネルの計画.....	151
AIX, Linux, and Windows でのメッセージ・チャンネルの例のセットアップ.....	152
AIX, Linux, and Windows の例の実行および展開.....	154
例: IBM i でのメッセージ・チャンネルの計画.....	155
IBM i でのメッセージ・チャンネル・エージェントのセットアップ.....	156
IBM i の例の実行および展開.....	159
Example: planning a message channel on z/OS.....	159
Setting up the message channel agent on z/OS.....	161

Running and expanding the example for z/OS.....	163
Example: planning a message channel for z/OS using queue sharing groups.....	163
Setting up the queue sharing group definitions and a queue manager QM3 not in the queue sharing group.....	165
Running the queue sharing group example for z/OS.....	166
別名を使用して MQ ライブラリーを参照する.....	167
Managed File Transfer 構成リファレンス.....	167
MFT プロパティーでの環境変数の使用.....	167
MFT installation.properties ファイル.....	169
MFT agent.properties ファイル.....	173
MFT coordination.properties ファイル.....	196
MFT command.properties ファイル.....	200
MFT logger.properties ファイル.....	204
LogTransfer 関数によって生成される出力.....	214
MFT Java システム・プロパティー.....	217
MFT で使用する SHA-2 の CipherSpec と CipherSuite.....	218
MFT ファイル・ロガーの構成ファイル.....	218
SCSQFCMD ライブラリー.....	226
SYSTEM.FTE トピック.....	227
MFT エージェント・キュー設定.....	228
MFT システム・キューおよびシステム・トピック.....	230
MFT オブジェクト命名規則.....	231
MFT エージェントの状況メッセージ.....	233
IBM MQ Internet Pass-Thru 構成リファレンス.....	233
MQIPT プロパティーの要約.....	235
MQIPT グローバル・プロパティー.....	240
MQIPT 経路のプロパティー.....	243
mqiptAdmin プロパティー.....	263
特記事項.....	265
プログラミング・インターフェース情報.....	266
商標.....	266

構成に関する参照情報

IBM MQ を構成する際には、このセクションの参照情報を使用してください。

構成についてのリファレンス情報は、以下のサブトピックに記載されています。

関連タスク

構成


 z/OS の構成

例: IBM MQ のクロスプラットフォーム通信のセットアップ

この例では、IBM MQ の送信側チャンネルと受信側チャンネルを構成して、サポートされているすべてのプロトコルを介してプラットフォーム間の双方向メッセージ・フローを使用可能にすることにより、作動中の IBM MQ ネットワークを確立する方法を示します。










始める前に

構成の例では、次のように、特定のプラットフォームに特定のネットワーク・インフラストラクチャーが存在すると仮定しています。

-  z/OS は、3745 ネットワーク・コントローラー (または同等のもの) を使用して通信します。

また、SNA については、必要なすべての定義が VTAM およびネットワーク制御プログラム (NCP) に定義済みで、LAN 接続プラットフォームが広域ネットワーク (WAN) を介して通信できるように活動化されていることを前提としています。同様に、TCP の場合は、ドメイン・ネーム・サーバーを使用するか、ローカルに保持されているテーブル (例えば、ホスト・ファイル) を使用することによって、ネーム・サーバー機能が使用可能であると想定されます。

構成例は、以下のネットワーク・ソフトウェア製品を対象としています。

- SNA
 -  IBM Windows の Personal Communications
 -  IBM AIX® 用の Communications Server
 -  IBM i
 - OS/390®
- TCP
 -  Microsoft Windows
 -  AIX
 -  IBM i
 -  z/OS 用の TCP
-  NetBIOS
-  SPX

サポートされる通信プロトコルおよびソフトウェアについて詳しくは、[IBM MQ のシステム要件](#)を参照してください。

このタスクについて

この例では、送信側チャンネルと受信側チャンネルを使用します。送信側と受信側以外のチャンネル・タイプを使用するには、[DEFINE CHANNEL \(define a new channel\)](#)を実行します。

6 ページの図 1 は、単一チャンネルと、それに関連付けられている IBM MQ オブジェクトの概念を表しています。

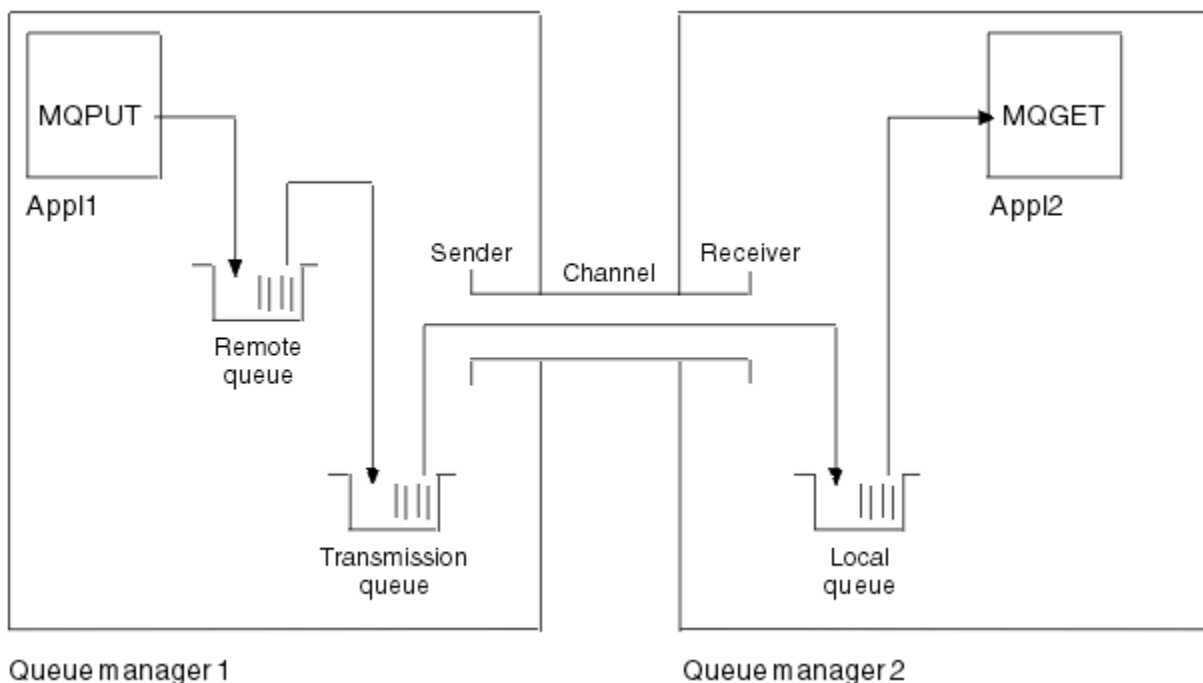


図 1. 構成例でセットアップする IBM MQ チャンネル

これは、IBM MQ ネットワークの基本的なエレメントだけを紹介するための簡単な例です。この例ではトリガー操作の使用法は説明していません。これについては、[チャンネルのトリガー操作](#)で説明しています。

このネットワークのオブジェクトは次のとおりです。

- リモート・キュー
- 伝送キュー
- ローカル・キュー
- 送信側チャンネル
- 受信側チャンネル

Appl1 と Appl2 は、どちらもアプリケーション・プログラムです。Appl1 はメッセージを書き込み、Appl2 はメッセージを受信します。

Appl1 はリモート・キューにメッセージを書き込みます。このリモート・キューの定義は、ターゲット・キュー・マネージャー、そのキュー・マネージャーのローカル・キュー、およびこのローカル・キュー・マネージャーの伝送キューの名前を指定します。

キュー・マネージャーは、リモート・キューへのメッセージ書き込み要求を Appl1 から受け取ったときに、宛先がリモートであることをキュー定義によって判別します。したがって、キュー・マネージャーは、伝送ヘッダーを付けたメッセージを、定義に指定されている伝送キューに直接書き込みます。メッセージは、チャンネルが使用可能になるまで伝送キューに残りますが、即時使用可能になることもあります。

送信側チャンネルには、その定義内に 1 つだけの伝送キューへの参照が備えられています。チャンネルの開始時や、その他の通常操作中のときに、チャンネルはこの伝送キューを参照して、そこにあるすべてのメッセージをターゲット・システムに送信します。メッセージの伝送ヘッダーには、宛先キューとキュー・マネージャーに関する詳細が入っています。

さまざまな組み合わせのプラットフォーム間で相互通信を行う例を使用して、前述の各オブジェクトの作成方法を詳細に説明しています。

ターゲット・キュー・マネージャーでは、ローカル・キューおよびチャンネルの受信側に対する定義が必要です。これらのオブジェクトは、互いに独立して操作し、任意の順序で作成できます。

ローカル・キュー・マネージャーでは、リモート・キュー、伝送キュー、およびチャンネルの送信側に対する定義が必要です。リモート・キュー定義とチャンネル定義の両方が伝送キュー名を参照するので、伝送キューを最初に作成することをお勧めします。

手順

- 7 ページの『[クロスプラットフォーム通信の使用法の例](#)』に記載されている情報を読みます。
- 適切なプラットフォームの指示に従って、ネットワーク接続を確立し、チャンネルを定義します。

a) **AIX**

[9 ページの『例: AIX での IBM MQ クロスプラットフォーム通信のセットアップ』](#)を参照

b) **IBM i**

[15 ページの『例: IBM i での IBM MQ クロスプラットフォーム通信のセットアップ』](#)を参照

c) **Linux**

[32 ページの『例: Linux での IBM MQ クロスプラットフォーム通信のセットアップ』](#)を参照

d) **Windows**

[38 ページの『例: Windows での IBM MQ クロスプラットフォーム通信のセットアップ』](#)を参照

e) **z/OS**

[44 ページの『Example: setting up IBM MQ cross-platform communication on z/OS』](#)を参照

f) **z/OS**

[49 ページの『Example: setting up IBM MQ cross-platform communication on z/OS using QSGs』](#)を参照

g) **z/OS**

[57 ページの『Example: setting up IBM MQ cross-platform communication for intra-group queuing on z/OS』](#)を参照

関連タスク

[分散キューイングの構成](#)

[z/OS での他のキュー・マネージャーとの通信のセットアップ](#)

クロスプラットフォーム通信の使用法の例

IBM MQ のクロスプラットフォーム通信をセットアップするための構成例では、別のプラットフォームとの通信をセットアップするために単一のプラットフォームで実行されるタスクについて説明しています。次に、これらの例では、そのプラットフォームへの作業チャンネルを確立するためのタスクについて説明します。

できるだけ情報を一般的にするようにしています。したがって、それぞれ異なるプラットフォーム上にある 2 つのキュー・マネージャーを接続する場合に参照する必要があるのは、関連する 2 つのセクションだけです。逸脱や特殊な事例は強調表示されています。また、同じプラットフォーム (異なるマシンまたは同一のマシン) で実行される 2 つのキュー・マネージャーも接続することができます。この場合は、すべての情報は 1 つのセクションから得られます。

ALW

AIX, Linux, and Windows では、各プラットフォームに対応する手順を始める前に、各種の環境変数を設定する必要があります。以下のいずれかのコマンドを入力して、これを行います。

- Linux AIX On AIX and Linux:

```
MQ_INSTALLATION_PATH/bin/setmqenv
```

ここで、`MQ_INSTALLATION_PATH` は IBM MQ がインストールされている場所を示しています。このコマンドは、現在作業しているシェルの環境変数を設定します。別のシェルを開いた場合には、コマンドを再入力する必要があります。

- ▶ **Windows** On Windows:

```
MQ_INSTALLATION_PATH/bin/setmqenv
```

ここで、`MQ_INSTALLATION_PATH` は IBM MQ がインストールされている場所を示しています。

サンプル構成で使用されているパラメーターを見つけることができる例があります。各パラメーターには短い説明があり、システムでこれに相当する値を探す場所についての指針が示されています。独自の値のセットがある場合は、このセクションの例を使用する際に、必ずそれらの値を使用してください。

これらの例は、クラスター化を使用している場合の通信のセットアップ方法については取り上げていません。クラスター化を使用しているときの通信のセットアップについては、[キュー・マネージャー・クラスターの構成](#)を参照してください。その場合でも、ここに示した通信の構成値を適用できます。

以下のプラットフォームを対象とした構成例があります。

- ▶ **AIX** 9 ページの『例: AIX での IBM MQ クロスプラットフォーム通信のセットアップ』
- ▶ **IBM i** 15 ページの『例: IBM i での IBM MQ クロスプラットフォーム通信のセットアップ』
- ▶ **Linux** 32 ページの『例: Linux での IBM MQ クロスプラットフォーム通信のセットアップ』
- ▶ **Windows** 38 ページの『例: Windows での IBM MQ クロスプラットフォーム通信のセットアップ』
- ▶ **z/OS** 44 ページの『Example: setting up IBM MQ cross-platform communication on z/OS』
- ▶ **z/OS** 49 ページの『Example: setting up IBM MQ cross-platform communication on z/OS using QSGs』
- ▶ **z/OS** 57 ページの『Example: setting up IBM MQ cross-platform communication for intra-group queuing on z/OS』

IT の責務

例中で使用されている用語を理解するためには、以下の指針を理解することから始めてください。

- システム管理者: 特定のプラットフォームにソフトウェアをインストールし、構成する人(またはグループ)です。
- ネットワーク管理者: LAN 接続、LAN アドレス割り当て、ネットワーク命名規則などのネットワークに関する作業を管理する人です。この管理者は、システム管理グループに属していることも、別のグループに属していることもあります。

ほとんどの z/OS インストール環境には、ネットワーク構成をサポートするために ACF/VTAM、ACF/NCP、および TCP/IP ソフトウェアの更新を担当するグループが存在します。IBM MQ プラットフォームを IBM MQ for z/OS に接続する場合には、主に、このグループのメンバーから必要な情報を入手します。LAN のネットワーク命名規則は、このグループによって指示されたり指定されたりします。独自の定義を作成するときには、このグループの制御スパンを十分確認する必要があります。

- 特定のタイプの管理者、例えば CICS® 管理者は、その人の責務を明確に記述できる場合に指定されています。

構成例のセクションでは、各パラメーター設定の担当者については言及しません。一般的には、数人の担当者で行います。

関連タスク

5 ページの『例: IBM MQ のクロスプラットフォーム通信のセットアップ』

この例では、IBM MQ の送信側チャネルと受信側チャネルを構成して、サポートされているすべてのプロトコルを介してプラットフォーム間の双方向メッセージ・フローを使用可能にすることにより、作動中の IBM MQ ネットワークを確立する方法を示します。

関連資料

[setmqenv](#)

例: AIX での IBM MQ クロスプラットフォーム通信のセットアップ

この例は、AIX 上の IBM MQ から別のプラットフォーム上の IBM MQ への通信リンクをセットアップし、そのプラットフォームへの作業チャンネルを確立する方法を示しています。

始める前に

この例とその使用方法の背景情報については、5 ページの『例: IBM MQ のクロスプラットフォーム通信のセットアップ』および7ページの『クロスプラットフォーム通信の使用法の例』を参照してください。

このタスクについて

この例では、AIX 上の IBM MQ から以下のプラットフォームへのクロスプラットフォーム通信のセットアップについて説明します。

-  Windows
-  Linux
-  IBM i
-  z/OS
- VSE/ESA

手順

1. 以下のいずれかのオプションを使用して、ネットワーク接続を確立します。

- LU 6.2 接続を確立します。SNA over TCP/IP の構成については、「[Communications Server for AIX Library](#)」を参照してください。
- TCP 接続を確立します。

チャンネルを開始するには、必ずその前にリスナーを明示的に開始してください。これにより、インバウンド送信チャンネルからの要求にตอบสนองして受信チャンネルが自動的に開始されるようになります。以下のコマンドを使用して、TCP リスナーの IBM MQ を開始します。

```
runmqclsr -t tcp
```

a. `/etc/services` ファイルを編集します。

注: `/etc/services` ファイルを編集するには、スーパーユーザーまたは root ユーザーとしてログインする必要があります。以下に示す行がこのファイルにない場合には、次のとおりに入力してファイルに追加してください。

```
MQSeries      1414/tcp      # MQSeries channel listener
```

b. `/etc/inetd.conf` ファイルを編集します。以下の行がこのファイルにない場合は、示されているとおりに追加して、`MQ_INSTALLATION_PATH` を IBM MQ がインストールされている上位のディレクトリーに置き換えてください。

```
MQSeries stream tcp nowait root MQ_INSTALLATION_PATH/bin/amqcrsta amqcrsta  
[-m queue.manager.name]
```

c. コマンド `refresh -s inetd` を入力します。

注：mqm グループに **root** を追加しなければなりません。1 次グループを mqm に設定する必要はありません。一連のグループ内に mqm があれば、コマンドを使用することができます。キュー・マネージャーを使用するアプリケーションだけを実行する場合、mqm グループ権限は必要ありません。

2. 接続が確立されたら、[10 ページの『AIX でのチャネルの構成』](#)の説明に従っていくつかのチャネルを定義します。

AIX AIX でのチャネルの構成

AIX での構成例のために IBM MQ を構成するには、キュー・マネージャーの基本構成ステップを実行してから、送信側チャネルと受信側チャネルを構成します。

始める前に

注：

1. インストール・プロセスを開始する前に、まず mqm ユーザーとグループを作成し、パスワードを設定したことを確認してください。
2. ファイル・システム内のスペース不足のためにインストールが失敗した場合は、コマンド `smit C sna` を使用して、以下のようにサイズを増やすことができます。(ファイル・システムの状況を表示するには、`df` を使用します。このコマンドによって、満杯になっている論理ボリュームが分かります。)

```
-- Physical and Logical Storage
-- File Systems
-- Add / Change / Show / Delete File Systems
-- Journaled File Systems
-- Change/Show Characteristics of a Journaled File System
```

3. 次のコマンドを使用して、チャネルを開始します。

```
runmqchl -c channel.name
```

4. サンプル・プログラムは `MQ_INSTALLATION_PATH/samp` にインストールされています。ここで、`MQ_INSTALLATION_PATH` は IBM MQ がインストールされている上位のディレクトリーを表します。
5. エラー・ログは、`/var/mqm/qmgrs/qmgrname/errors` に格納されています。
6. AIX では、標準の IBM MQ トレース・コマンドまたは AIX システム・トレースを使用して、IBM MQ コンポーネントのトレースを開始できます。IBM MQ トレースおよび AIX システム・トレースについて詳しくは、[トレースの使用](#) を参照してください。
7. コマンド・インタープリター `runmqsc` を使用して、管理コマンドを入力している場合、行の終わりにある `+` は、次の行が継続していることを示します。必ず最後のパラメーターと継続文字との間にスペースを入れるようにします。

手順

1. キュー・マネージャーの基本構成を実行するには、以下のステップを実行します。
 - a) 次のコマンドを使用して AIX コマンド・ラインからキュー・マネージャーを作成します。

```
crtmqm -u dlqname -q aix
```

ここで、

aix

キュー・マネージャーの名前

-q

これがデフォルトのキュー・マネージャーになることを示す

-u dlqname

未配布メッセージ・キューの名前を指定する

このコマンドは、キュー・マネージャーおよび一連のデフォルト・オブジェクトを作成します。

b) 次のコマンドを使用して AIX コマンド・ラインからキュー・マネージャーを開始します。

```
stmqm aix
```

ここで、*aix* は、作成時に、キュー・マネージャーに指定された名前です。

c) 以下のコマンドを入力して、AIX コマンド行から **runmqsc** を開始し、それを使用して配信不能メッセージ・キューを作成します。

```
def ql (dlqname)
```

ここで、*dlqname* は、キュー・マネージャーが作成されたとき、未配布メッセージ・キューに指定された名前です。

2. 構成例のチャンネルを構成します。

以下の例で使用されているパラメーターについては、[12 ページの『AIX のチャンネル構成パラメーター』](#)を参照してください。いずれの場合も、この例では MQSC コマンドを示しています。AIX コマンド行から **runmqsc** を開始して各コマンドを順番に入力するか、コマンドをコマンド・ファイルにビルドします。

Windows 以下の例は、AIX 上の IBM MQ を Windows 上の IBM MQ に接続するためのものです。別のプラットフォーム上の IBM MQ に接続するには、Windows の値ではなく、[12 ページの『AIX のチャンネル構成パラメーター』](#)の表の適切な値を使用してください。

a) 以下の例に示すように、送信側チャンネルを定義します。

• SNA の使用:

```
def ql (WINNT) +                               F
  usage(xmitq) +
  replace

def qr (WINNT.REMOTEQ) +                       D
  rname(WINNT.LOCALQ) +                       E
  rqmname(WINNT) +                             C
  xmitq(WINNT) +                               F
  replace

def chl (AIX.WINNT.SNA) chltype(sdr) +        G
  trptype(lu62) +
  conname('WINNTCPIC') +                      17
  xmitq(WINNT) +                               F
  replace
```

• TCP の使用:

```
def ql (WINNT) +                               F
  usage(xmitq) +
  replace

def qr (WINNT.REMOTEQ) +                       D
  rname(WINNT.LOCALQ) +                       E
  rqmname(WINNT) +                             C
  xmitq(WINNT) +                               F
  replace

def chl (AIX.WINNT.TCP) chltype(sdr) +        H
  trptype(tcp) +
  conname(remote_tcpip_hostname) +
  xmitq(WINNT) +                               F
  replace
```

b) 以下の例に示すように、受信側チャンネルを定義します。

• SNA の使用:

```
def ql (AIX.LOCALQ) replace B
def chl (WINNT.AIX.SNA) chltype(rcvr) + I
  trptype(lu62) +
  replace
```

- TCP の使用:

```
def ql (WINNT) + F
  usage(xmitq) +
  replace

def qr (WINNT.REMOTEQ) + D
  rname(WINNT.LOCALQ) + E
  rqmname(WINNT) + C
  xmitq(WINNT) + F
  replace

def chl (AIX.WINNT.TCP) chltype(sdr) + H
  trptype(tcp) +
  conname(remote_tcpip_hostname) +
  xmitq(WINNT) + F
  replace
```

注: 送信側チャンネルが会話を開始するときに SNA 受信側チャンネルが正しく活動化されるようにするための代替方法があります。

AIX Communications Server の構成プロセス中に作成された LU 6.2 TPN プロファイルには、TP 実行可能プログラムへの絶対パスが含まれています。例えば、ファイルは `u/interop/AIX.crs6a` と呼ばれていました。名前を選択することはできますが、その名前に使用するキュー・マネージャーの名前を含めることも検討してください。実行可能ファイルの内容は、次のとおりでなければなりません。

```
#!/bin/sh
MQ_INSTALLATION_PATH/bin/amqcrs6a -m aix
```

ここで、`aix` はキュー・マネージャー名 (A) で、`MQ_INSTALLATION_PATH` は IBM MQ がインストールされている上位のディレクトリーです。このファイルを作成した後、次のコマンドを実行するとその実行が可能になります。

```
chmod 755 /u/interop/AIX.crs6a
```

実行可能ファイルを作成する代わりに、「LU 6.2 TPN プロファイルの追加」パネルでコマンド行パラメーターを使用して上記のパスを指定することもできます。

これら 2 つの方法のどちらかでパスを指定すると、送信側チャンネルが会話を開始した場合、必ず SNA 受信側チャンネルが正しく活動化します。

AIX AIX のチャンネル構成パラメーター

AIX での構成例のチャンネルを構成するために必要なパラメーター。

10 ページの『AIX でのチャンネルの構成』のステップ 11 ページの『2』では、5 ページの『例: IBM MQ のクロスプラットフォーム通信のセットアップ』で説明されているチャンネルを実装するために AIX キュー・マネージャーで実行する構成について説明します。10 ページの『AIX でのチャンネルの構成』の例は、IBM MQ for IBM i と IBM MQ for Windows を接続するためのものです。別のプラットフォーム上の IBM MQ に接続するには、Windows の値の代わりに該当する表の値を使用します。

注: 太字の語は推奨値であり、これらの例で使用されている IBM MQ オブジェクトの名前を反映しています。これらの値は製品インストール済み環境で変更できますが、変更する場合は、このセクションの例を参照する際に必ず独自の値を使用してください。

ローカル・ノードの定義

ID	パラメーター名	参照	使用例
A	キュー・マネージャー名		AIX
B	ローカル・キュー名		AIX.LOCALQ

Windows 上の IBM MQ への接続

Windows

表のこのセクションで指定する値は、42 ページの『Windows のチャンネル構成パラメーター』で指定した値と一致させる必要があります。

ID	パラメーター名	参照	使用例
C	リモート・キュー・マネージャー名	A	WINNT
D	リモート・キュー名		WINNT.REMOTEQ
E	リモート・システムのキュー名	B	WINNT.LOCALQ
F	伝送キュー名		WINNT
G	送信側 (SNA) チャンネル名		AIX.WINNT.SNA
H	送信側 (TCP/IP) チャンネル名		AIX.WINNT.TCP
I	受信側 (SNA) チャンネル名	G	WINNT.AIX.SNA
J	受信側 (TCP) チャンネル名	H	WINNT.AIX.TCP

Linux 上の IBM MQ への接続

Linux

表のこのセクションで指定する値は、36 ページの『Linux のチャンネル構成パラメーター』で指定した値と一致させる必要があります。

ID	パラメーター名	参照	使用例
C	リモート・キュー・マネージャー名	A	LINUX
D	リモート・キュー名		LINUX.REMOTEQ
E	リモート・システムのキュー名	B	LINUX.LOCALQ
F	伝送キュー名		LINUX
G	送信側 (SNA) チャンネル名		AIX.LINUX.SNA
H	送信側 (TCP/IP) チャンネル名		AIX.LINUX.TCP
I	受信側 (SNA) チャンネル名	G	LINUX.AIX.SNA
J	受信側 (TCP/IP) チャンネル名	H	LINUX.AIX.TCP

IBM i 上の IBM MQ への接続

IBM i

表のこのセクションで指定する値は、29 ページの『IBM i のチャンネル構成パラメーター』で指定した値と一致させる必要があります。

表 4. IBM i 上の IBM MQ に接続するための構成例

ID	パラメーター名	参照	使用例
C	リモート・キュー・マネージャー名	A	AS400
D	リモート・キュー名		AS400.REMOTEQ
E	リモート・システムのキュー名	B	AS400.LOCALQ
F	伝送キュー名		AS400
G	送信側 (SNA) チャンネル名		AIX.AS400.SNA
H	送信側 (TCP) チャンネル名		AIX.AS400.TCP
I	受信側 (SNA) チャンネル名	G	AS400.AIX.SNA
J	受信側 (TCP) チャンネル名	H	AS400AIX。TCP (TCP)

IBM MQ for z/OS への接続

z/OS

表のこのセクションで指定する値は、47 ページの『Channel configuration parameters for z/OS』で指定した値と一致させる必要があります。

表 5. IBM MQ for z/OS に接続するための構成例

ID	パラメーター名	参照	使用例
C	リモート・キュー・マネージャー名	A	MVS
D	リモート・キュー名		MVS.REMOTEQ
E	リモート・システムのキュー名	B	MVS.LOCALQ
F	伝送キュー名		MVS
G	送信側 (SNA) チャンネル名		AIX.MVS.SNA
H	送信側 (TCP) チャンネル名		AIX.MVS.TCP
I	受信側 (SNA) チャンネル名	G	MVS.AIX.SNA
J	受信側 (TCP) チャンネル名	H	MVSAIX。TCP (TCP)

キュー共有グループを使用した IBM MQ for z/OS への接続

z/OS

表のこのセクションで指定する値は、55 ページの『Shared channel configuration parameters』で指定した値と一致させる必要があります。

表 6. キュー共有グループを使用して IBM MQ for z/OS に接続するための構成例

ID	パラメーター名	参照	使用例
C	リモート・キュー・マネージャー名	A	QSG
D	リモート・キュー名		QSG.REMOTEQ
E	リモート・システムのキュー名	B	QSG.SHAREDQ
F	伝送キュー名		QSG

ID	パラメーター名	参照	使用例
G	送信側 (SNA) チャンネル名		AIX.QSG.SNA
H	送信側 (TCP) チャンネル名		AIX.QSG.TCP
I	受信側 (SNA) チャンネル名	G	QSG.AIX.SNA
J	受信側 (TCP) チャンネル名	H	QSG.AIX.TCP

IBM i 例: IBM i での IBM MQ クロスプラットフォーム通信のセットアップ

この例は、IBM i 上の IBM MQ から別のプラットフォーム上の IBM MQ への通信リンクをセットアップし、そのプラットフォームへの作業チャンネルを確立する方法を示しています。

始める前に

この例とその使用方法の背景情報については、5 ページの『例: IBM MQ のクロスプラットフォーム通信のセットアップ』および 7 ページの『クロスプラットフォーム通信の使用法の例』を参照してください。

このタスクについて

この例では、IBM i 上の IBM MQ から以下のプラットフォームへのクロスプラットフォーム通信のセットアップについて説明します。

-  Windows
-  AIX
-  Linux
-  z/OS または MVS
- VSE/ESA

手順

1. 以下のいずれかのオプションを使用して、ネットワーク接続を確立します。
 - 15 ページの『IBM i での LU 6.2 接続の確立』の説明に従って、LU 6.2 接続を確立します。
 - 24 ページの『IBM i での TCP 接続の確立』の説明に従って、TCP 接続を確立します。
2. 接続が確立されたら、26 ページの『IBM i でのチャンネルの構成』の説明に従っていくつかのチャンネルを定義します。

IBM i IBM i での LU 6.2 接続の確立

IBM i で LU 6.2 接続を確立するには、ローカル・ノードを構成し、それをパートナー・ノードに接続する必要があります。

このタスクについて

IBM i システムから他のいずれかの IBM MQ プラットフォームへの通信をセットアップするために必要なパラメーターについて詳しくは、20 ページの『IBM i での LU 6.2 接続の構成パラメーター』の表を参照してください。タスク・ステップで大括弧 () で囲まれた番号は、これらの表の「ID」列の値に対応しています。

ローカル・ノードを構成するには、以下を行う必要があります。

- 回線記述の作成
- 経路指定項目を追加してから、サブシステムを開始してください。

パートナー・ノードに接続するには、以下を行う必要があります。

- 制御装置記述の作成
- 装置記述の作成
- CPI-C サイド情報の作成
- APPC の通信項目の追加
- 構成リスト項目の追加

手順

1. 回線記述を作成し、経路指定項目を追加することによって、ローカル・ノードを構成してください。

a) 回線記述を作成します。

回線記述がまだ作成されていない場合は、以下の例に示すように、**CRTLINTRN** コマンドを使用して **回線記述 (6)** および **リソース名 (7)** の値を指定します。

```
Create Line Desc (token-ring) (CRTLINTRN)
Type choices, press Enter.

Line description . . . . . TOKENRINGL Name
Resource name . . . . . LINO41 Name, *NWID
NWI type . . . . . *FR *FR, *ATM
Online at IPL . . . . . *YES *YES, *NO
Vary on wait . . . . . *NOWAIT *NOWAIT, 15-180 (1 second)
Maximum controllers . . . . . 40 1-256
Attached NWI . . . . . *NONE Name, *NONE

Bottom
F3=Exit F4=Prompt F5=Refresh F10=Additional parameters F12=Cancel
F13=How to use this display F24=More keys
Parameter LIND required. +
```

b) 経路指定項目を追加します。

コマンド **ADDRTGE** を入力し、Enter キーを押して、「**Subsystem description (サブシステム記述)**」(5) に独自の値を指定します。また、「**Routing entry sequence number (ルーティング項目シーケンス番号)**」、「**Compare value (値の比較)**」(8)、「**Starting position (開始位置)**」、「**Program to call (呼び出し先プログラム)**」、および呼び出し先プログラムが入っている **Library** の例に示されている値を指定します。


```

Add Routing Entry (ADDRTGE)

Type choices, press Enter.

Subsystem description . . . . . QCMN      Name
Library . . . . . *LIBL      Name, *LIBL, *CURLIB
Routing entry sequence number . 1      1-9999
Comparison data:
Compare value . . . . . 'MQSERIES'

Starting position . . . . . 37      1-80
Program to call . . . . . AMQCRC6B      Name, *RTGDTA
Library . . . . . QMAS400      Name, * LI BL, *CURLIB
Class . . . . . *SBSD      Name, *SBSD
Library . . . . . *LIBL      Name, *LIBL, *CURLIB
Maximum active routing steps . . *NOMAX      0-1000, *NOMAX
Storage pool identifier . . . . . 1      1-10

Bottom
F3=Exit F4=Prompt F5=Refresh F12=Cancel F13=How to use this display
F24=More keys
Parameter SBSDB required.
+

```

コマンド STRSBS *subsystem description* (5) を入力し、Enter キーを押して、サブシステムを開始します。

2. 制御装置記述、装置記述、および CPI-C サイド情報を作成し、APPC の通信項目と構成リスト項目を追加することによって、パートナー・ノードへの接続を作成します。

Windows この例は、Windows システムへの接続についてのものですが、他のノードでも手順は同じです。

a) 制御装置記述の作成

コマンド行で CRTCTLAPPC と入力して Enter キーを押し、「**コントローラー記述**」(12) の値を指定し、「**リンク・タイプ**」を *LAN に設定し、「**IPL 時にオンライン**」を *NO に設定します。

```

Create Ctl Desc (APPC) (CRTCTLAPPC)

Type choices, press Enter.

Controller description . . . . . WINNTCP      Name
Link type . . . . . *LAN      *FAX, *FR, *IDLC,
*LAN...
Online at IPL . . . . . *NO      *YES, *NO

Bottom
F3=Exit F4=Prompt F5=Refresh F10=Additional parameters F12=Cancel
F13=How to use this display F24=More keys
Parameter CTLD required.
+

```

Enter キーを 2 回押し、その後に F10 を押してから、「**Switched line list**」(6)、「**Remote network identifier**」(9)、「**Remote control point**」(10)、および「**LAN remote adapter address**」(16) に値を指定して、Enter キーを押します。

```

Create Ctl Desc (APPC) (CRTCTLAPPC)

Type choices, press Enter.

Controller description . . . . . > WINNTCP      Name
Link type . . . . . > *LAN      *FAX, *FR, *IDLC, *LAN...
Online at IPL . . . . . > *NO      *YES, *NO
APPN-capable . . . . . *YES      *YES, *NO
Switched line list . . . . . TOKENRINGL Name
+ for more values
Maximum frame size . . . . . *LINKTYPE 265-16393, 256, 265, 512...
Remote network identifier . . . NETID      Name, *NETATR, *NONE, *ANY
Remote control point . . . . . WINNTCP      Name, *ANY
Exchange identifier . . . . . 00000000-FFFFFFF
Initial connection . . . . . *DIAL      *DIAL, *ANS
Dial initiation . . . . . *LINKTYPE *LINKTYPE, *IMMED, *DELAY
LAN remote adapter address . . 10005AFC5D83 000000000001-FFFFFFF
APPN CP session support . . . . *YES      *YES, *NO
APPN node type . . . . . *ENDNODE *ENDNODE, *LENNODE...
APPN transmission group number 1      1-20, *CALC
More...
F3=Exit F4=Prompt F5=Refresh F12=Cancel F13=How to use this display
F24=More keys

```

b) 装置記述を作成します。

コマンド CRTDEVAPPC を入力して Enter キーを押し、「装置記述」(13)、「リモート・ロケーション」(11)、「ローカル・ロケーション」(3)、「リモート・ネットワーク ID」(9)、および「接続されたコントローラー」(12)の値を指定します。

```

Create Device Desc (APPC) (CRTDEVAPPC)

Type choices, press Enter.

Device description . . . . . WINNTLU      Name
Remote location . . . . . WINNTLU      Name
Online at IPL . . . . . *YES      *YES, *NO
Local location . . . . . AS400LU      Name, *NETATR
Remote network identifier . . . NETID      Name, *NETATR, *NONE
Attached controller . . . . . WINNTCP      Name
Mode . . . . . *NETATR      Name, *NETATR
+ for more values
Message queue . . . . . QSYSOPR      Name, QSYSOPR
Library . . . . . *LIBL      Name, *LIBL, *CURLIB
APPN-capable . . . . . *YES      *YES, *NO
Single session:
Single session capable . . . . *NO      *NO, *YES
Number of conversations . . . . 1-512

Bottom
F3=Exit F4=Prompt F5=Refresh F10=Additional parameters F12=Cancel
F13=How to use this display F24=More keys
Parameter DEVD required.      +

```

注: IBM i 自動構成サービスを利用することにより、コントローラー記述と装置記述の作成の手間を省くことができます。詳しくは、IBM i 資料を参照してください。

c) CPI-C サイド情報を作成します。

CRTCSI と入力して F10 を押し、「サイド情報」(14)、「リモート・ロケーション」(11)、「トランザクション・プログラム」(15)、「ローカル・ロケーション」(3)、「モード」、および「リモート・ネットワーク ID」(9)に値を指定して、Enter キーを押します。

```

Create Comm Side Information (CRTCSI)

Type choices, press Enter.

Side information . . . . . NTCPIC      Name
Library . . . . . *CURLIB      Name, *CURLIB
Remote location . . . . . WINNTLU    Name
Transaction program . . . . . MQSERIES

Text 'description' . . . . . *BLANK

Additional Parameters

Device . . . . . *LOC      Name, *LOC
Local location . . . . . AS400LU    Name, *LOC, *NETATR
Mode . . . . . #INTER      Name, *NETATR
Remote network identifier . . . . . NETID    Name, *LOC, *NETATR, *NONE
Authority . . . . . *LIBCRTAUT    Name, *LIBCRTAUT, *CHANGE...

Bottom
F3=Exit  F4=Prompt  F5=Refresh  F12=Cancel  F13=How to use this display
F24=More keys
Parameter CSI required.

```

d) APCC の通信項目を追加します。

コマンド行で、ADDCMNE と入力して Enter キーを押し、「**Subsystem description (サブシステム記述)**」(5) および「**Device (装置)**」(13) に値を指定して、もう一度 Enter キーを押します。

```

Add Communications Entry (ADDCMNE)

Type choices, press Enter.

Subsystem description . . . . . QCMN      Name
Library . . . . . *LIBL      Name, *LIBL, *CURLIB
Device . . . . . WINNTLU    Name, generic*, *ALL...
Remote location . . . . .      Name
Job description . . . . . *USRPRF    Name, *USRPRF, *SBSD
Library . . . . .      Name, *LIBL, *CURLIB
Default user profile . . . . . *NONE    Name, *NONE, *SYS
Mode . . . . . *ANY      Name, *ANY
Maximum active jobs . . . . . *NOMAX    0-1000, *NOMAX

Bottom
F3=Exit  F4=Prompt  F5=Refresh  F12=Cancel  F13=How to use this display
F24=More keys
Parameter SBSD required.

```

e) 構成リスト項目を追加します。

ADDCFGLE *APPNRMT と入力して F4 を押し、「**リモート・ロケーション名**」(11)、「**リモート・ネットワーク ID**」(9)、「**ローカル・ロケーション名**」(3)、「**リモート制御点**」(10) および「**制御点ネットワーク ID**」(9) に値を指定して、Enter キーを押します。

```

Add Configuration List Entries (ADDCFGLE)

Type choices, press Enter.

Configuration list type . . . . > *APPNRMT *APPNLCL, *APPNRMT...
APPN remote location entry:
Remote location name . . . . . WINNTLU Name, generic*, *ANY
Remote network identifier . . . NETID Name, *NETATR, *NONE
Local location name . . . . . AS400LU Name, *NETATR
Remote control point . . . . . WINNTCP Name, *NONE
Control point net ID . . . . . NETID Name, *NETATR, *NONE
Location password . . . . . *NONE
Secure location . . . . . *NO *YES, *NO
Single session . . . . . *NO *YES, *NO
Locally controlled session . . *NO *YES, *NO
Pre-established session . . . *NO *YES, *NO
Entry 'description' . . . . . *BLANK
Number of conversations . . . 10 1-512
+ for more values

Bottom
F3=Exit F4=Prompt F5=Refresh F12=Cancel F13=How to use this display
F24=More keys

```

次のタスク

これで LU 6.2 接続は確立されました。26 ページの『[IBM i でのチャンネルの構成](#)』で説明されているように、構成を完了する準備ができました。

関連タスク

24 ページの『[IBM i での TCP 接続の確立](#)』

TCP が既に構成されている場合、追加の構成タスクはありません。TCP/IP が構成されていない場合は、TCP/IP インターフェースを追加し、TCP/IP ループバック・インターフェースを追加し、デフォルト経路を追加する必要があります。

IBM i IBM i での LU 6.2 接続の構成パラメーター

LU 6.2 接続を使用して、IBM i システム上の IBM MQ から他のいずれかの IBM MQ プラットフォームへの通信をセットアップするために必要なパラメーター。

これらの表を、接続先のプラットフォームの表とともに使用します。

「参照」列に数字が表示されている場合は、その値がこのセクションの他の場所にある該当する表の値と一致する必要があることを示しています。15 ページの『[IBM i での LU 6.2 接続の確立](#)』のタスク・ステップでは、この表の「ID」列の値を参照します。

「パラメーター名」列の項目は、23 ページの『[表で使用される用語の説明](#)』で説明します。

ローカル・ノードの定義

ID	パラメーター名	参照	使用例
1	ローカル・ネットワーク ID		NETID
2	ローカル制御点名		AS400PU
3	LU 名		AS400LU
4	LAN 宛先アドレス		10005A5962EF
5	サブシステム記述		QCMN
6	回線記述		TOKENRINGL
7	リソース名		LIN041

表 7. ローカル・ノードの定義の構成例 (続き)			
ID	パラメーター名	参照	使用例
8	ローカル・トランザクション・プログラム名		MQSERIES

Windows 上の IBM MQ への接続

Windows

Windows

表 8. Windows 上の IBM MQ に接続するための構成例			
ID	パラメーター名	参照	使用例
9	ネットワーク ID	2	NETID
10	制御点名	3	WINNTCP
11	LU 名	5	WINNTLU
12	コントローラー記述		WINNTCP
13	デバイス		WINNTLU
14	サイド情報		NTCPIC
15	トランザクション・プログラム	7	MQSERIES
16	LAN アダプター・アドレス	9	08005AA5FAB9
17	モード (Mode)	17	#INTER

AIX 上の IBM MQ への接続

AIX

表 9. AIX システム上の IBM MQ に接続するための構成例			
ID	パラメーター名	参照	使用例
9	ネットワーク ID	1	NETID
10	制御点名	2	AIXPU
11	LU 名	4	AIXLU
12	コントローラー記述		AIXPU
13	デバイス		AIXLU
14	サイド情報		AIXCPIC
15	トランザクション・プログラム	6	MQSERIES
16	LAN アダプター・アドレス	8	123456789012
17	モード (Mode)	14	#INTER

Linux (x86 プラットフォーム) での IBM MQ への接続

Linux

表 10. Linux (x86 プラットフォーム) 上の IBM MQ に接続するための構成例

ID	パラメーター名	参照	使用例
9	ネットワーク ID	4	NETID
10	制御点名	2	LINUXPU
11	LU 名	5	LINUXLU
12	コントローラー記述		LINUXPU
13	デバイス		LINUXLU
14	サイド情報		LXCPIC
15	トランザクション・プログラム	7	MQSERIES
16	LAN アダプター・アドレス	8	08005AC6DF33
17	モード (Mode)	6	#INTER

IBM MQ for z/OS への接続

z/OS

表 11. IBM MQ for z/OS への接続の構成例

ID	パラメーター名	参照	使用例
9	ネットワーク ID	2	NETID
10	制御点名	3	MVSPU
11	LU 名	4	MVSLU
12	コントローラー記述		MVSPU
13	デバイス		MVSLU
14	サイド情報		MVSCPIC
15	トランザクション・プログラム	7	MQSERIES
16	LAN アダプター・アドレス	8	400074511092
17	モード (Mode)	6	#INTER

VSE/ESA システムへの接続

表 12. VSE/ESA システムへの接続の構成例

ID	パラメーター名	参照	使用例
9	ネットワーク ID	1	NETID
10	制御点名	2	VSEPU
11	LU 名	3	VSELU
12	コントローラー記述		VSEPU
13	デバイス		VSELU
14	サイド情報		VSECPIC
15	トランザクション・プログラム	4	MQ01
16	LAN アダプター・アドレス	5	400074511092

表 12. VSE/ESA システムへの接続の構成例 (続き)			
ID	パラメーター名	参照	使用例
17	モード (Mode)		#INTER

表で使用される用語の説明

1 2 3

構成された値を見つける方法については、[23 ページの『ネットワーク属性を見つける方法』](#)を参照してください。

4 LAN 宛先アドレス (LAN destination address)

IBM i システム・トークンリング・アダプターのハードウェア・アドレス。この値は、コマンド `DSPLIND Line description (6)` を使用して見つけることができます。

5 サブシステム記述

このパラメーターは、キュー・マネージャー使用中にアクティブになる IBM i サブシステムの名前です。QCMN という名前が使用されていますが、それは QCMN が IBM i 通信サブシステムであるためです。

6 回線記述

このパラメーターが指定されている場合、リソースの「リソース名」の「記述」フィールドにそのことが示されます。詳細については、[24 ページの『リソース名の値を見つける方法』](#)を参照してください。この値を指定しない場合は、回線記述を作成する必要があります。

7 リソース名

構成された値を見つける方法については、[24 ページの『リソース名の値を見つける方法』](#)を参照してください。

8 ローカル・トランザクション・プログラム名

このワークステーションとの会話を試行する IBM MQ アプリケーションは、受信側で実行するプログラムのシンボル名を指定します。この名前は、送信側のチャネル定義に定義されています。分かりやすいように、トランザクション・プログラム名には MQSERIES をできるだけ使用するようになっています。VSE/ESA へ接続している場合は、長さの上限が 4 バイトなので MQTP を使用するようになっています。

詳細については、[ローカル IBM i システムにおけるリモート・キュー・マネージャー・プラットフォーム用の設定値を参照してください。](#)

12 コントローラー記述

このパラメーターは、パートナー・システムの「制御点名」(または「ノード名」)の別名です。分かりやすいように、この例ではパートナーの実名を使用しています。

13 装置

このパラメーターは、パートナー・システムの LU の別名です。分かりやすいように、この例ではパートナーの LU 名を使用しています。

14 サイド情報

このパラメーターは、CPI-C サイド情報プロファイルに付ける名前です。独自の名前を 8 文字で指定します。

ネットワーク属性を見つける方法

ローカル・ノードは、IBM i インストールの一部として、すでに部分的に構成されています。現在のネットワーク属性を表示するには、コマンド `DSPNETA` を入力します。

これらの値を変更する必要がある場合は、コマンド `CHGNETA` を使用します。変更を適用するには、IPL が必要になる場合もあります。

```

Display Network Attributes
System: AS400PU
Current system name . . . . . : AS400PU
Pending system name . . . . . :
Local network ID . . . . . : NETID
Local control point name . . . . . : AS400PU
Default local location . . . . . : AS400LU
Default mode . . . . . : BLANK
APPN node type . . . . . : *ENDNODE
Data compression . . . . . : *NONE
Intermediate data compression . . . . . : *NONE
Maximum number of intermediate sessions . . . . . : 200
Route addition resistance . . . . . : 128
Server network ID/control point name . . . . . : NETID NETCP

```

More...
Press Enter to continue.

F3=Exit F12=Cancel

「ローカル・ネットワーク ID」(1)、「ローカル制御点名」(2)、および「デフォルト・ローカル・ロケーション」(3)の値が、表内の値、または独自の値(変更した場合)に対応していることを確認します。

リソース名の値を見つける方法

リソース名の値を見つけるには、WRKHDWRSC TYPE(*CMN)を入力して Enter キーを押します。

「Work with Communication Resources (通信リソースの処理)」パネルが表示されます。「リソース名」の値はトークンリング・ポートとして示されています。この例では、LIN041 となっています。

```

Work with Communication Resources
System: AS400PU
Type options, press Enter.
2=Edit 4=Remove 5=Work with configuration description
7=Add configuration description ...

```

```

Configuration
Opt Resource      Description Type Description
CC02              2636 Comm Processor
LIN04             2636 LAN Adapter
LIN041  TOKEN-RING 2636 Token-ring Port

```

Bottom
F3=Exit F5=Refresh F6=Print F11=Display resource addresses/statuses
F12=Cancel F23=More options

IBM i での TCP 接続の確立

TCP が既に構成されている場合、追加の構成タスクはありません。TCP/IP が構成されていない場合は、TCP/IP インターフェースを追加し、TCP/IP ループバック・インターフェースを追加し、デフォルト経路を追加する必要があります。

手順

1. TCP/IP インターフェースを追加します。

コマンド行で、ADDTCPIFC と入力して Enter キーを押し、マシンの「**IP アドレス**」と「**回線記述**」、および「**サブネット・マスク**」を指定して、もう一度 Enter キーを押しします。

```
Add TCP/IP Interface (ADDTCPIFC)

Type choices, press Enter.

Internet address . . . . . 19.22.11.55
Line description . . . . . TOKENRINGL Name, *LOOPBACK
Subnet mask . . . . . 255.255.0.0
Type of service . . . . . *NORMAL *MINDELAY, *MAXTHRPUT..
Maximum transmission unit . . . *LIND 576-16388, *LIND
Autostart . . . . . *YES *YES, *NO
PVC logical channel identifier 001-FFF
+ for more values
X.25 idle circuit timeout . . . 60 1-600
X.25 maximum virtual circuits . 64 0-64
X.25 DDN interface . . . . . *NO *YES, *NO
TRLAN bit sequencing . . . . . *MSB *MSB, *LSB

Bottom
F3=Exit F4=Prompt F5=Refresh F12=Cancel F13=How to use this display
F24=More keys
```

2. TCP/IP ループバック・インターフェースを追加します。

コマンド行で、ADDTCPIFC と入力して Enter キーを押し、「**IP アドレス**」、「**回線記述**」、および「**サブネット・マスク**」の値を指定します。

```
Add TCP Interface (ADDTCPIFC)

Type choices, press Enter.

Internet address . . . . . 127.0.0.1
Line description . . . . . *LOOPBACK Name, *LOOPBACK
Subnet mask . . . . . 255.0.0.0
Type of service . . . . . *NORMAL *MINDELAY, *MAXTHRPUT..
Maximum transmission unit . . . *LIND 576-16388, *LIND
Autostart . . . . . *YES *YES, *NO
PVC logical channel identifier 001-FFF
+ for more values
X.25 idle circuit timeout . . . 60 1-600
X.25 maximum virtual circuits . 64 0-64
X.25 DDN interface . . . . . *NO *YES, *NO
TRLAN bit sequencing . . . . . *MSB *MSB, *LSB

Bottom
F3=Exit F4=Prompt F5=Refresh F12=Cancel F13=How to use this display
F24=More keys
```

3. デフォルト経路を追加します。

コマンド行で、ADDTCP RTE と入力して Enter キーを押し、ご使用のネットワークに適した値を指定して Enter キーを押すと、デフォルトの経路項目が作成されます。

```
Add TCP Route (ADDTCPRTE)

Type choices, press Enter.

Route destination . . . . . *DFTRROUTE
Subnet mask . . . . . *NONE
Type of service . . . . . *NORMAL      *MINDELAY, *MAXTHRPUT.
Next hop . . . . . 19.2.3.4
Maximum transmission unit . . . 576      576-16388, *IFC

Bottom
F3=Exit F4=Prompt F5=Refresh F12=Cancel F13=How to use this display
F24=More keys
Command prompting ended when user pressed F12.
```

次のタスク

これで TCP 接続は確立されました。26 ページの『[IBM i でのチャンネルの構成](#)』で説明されているように、構成を完了する準備ができました。

関連タスク

15 ページの『[IBM i での LU 6.2 接続の確立](#)』

IBM i で LU 6.2 接続を確立するには、ローカル・ノードを構成し、それをパートナー・ノードに接続する必要があります。

IBM i IBM i でのチャンネルの構成

IBM i での構成例のために IBM MQ を構成するには、キュー・マネージャーの基本構成ステップを実行してから、送信側チャンネルと受信側チャンネルを構成します。

このタスクについて

WRKMQMQ コマンドを使用して、IBM MQ 構成メニューを表示します。

コマンド **STRMQLSR** を使用して、TCP チャンネル・リスナーを開始します。

コマンド **STRMQMCHL** **CHLNAME**(*channel_name*) を使用して、任意の送信側チャンネルを開始します。

注: AMQ* エラーは、エラーが発生したジョブに関連するログに記録されています。**WRKACTJOB** コマンドを使用して、ジョブのリストを表示します。QSYSWRK という名前のサブシステムのもとでジョブを探し出し、5 を入力しそのジョブの処理を行います。IBM MQ のログには接頭部 AMQ が付きます。

手順

1. キュー・マネージャーを作成します。
 - a) CRTMQM と入力して、Enter キーを押します。

```

Create Message Queue Manager (CRTMQM)

Type choices, press Enter.
Message Queue Manager name . . .
Text 'description' . . . . . *BLANK
Trigger interval . . . . . 999999999 0-999999999
Undelivered message queue . . . *NONE
Default transmission queue . . . *NONE
Maximum handle limit . . . . . 256 1-999999999
Maximum uncommitted messages . . 1000 1-10000
Default Queue manager . . . . . *NO *YES, *NO

Bottom
F3=Exit F4=Prompt F5=Refresh F12=Cancel F13=How to use this display
F24=More keys

```

- b) 「メッセージ・キュー・マネージャー名」 フィールドに AS400 と入力し、「未配布メッセージ・キュー」 フィールドに DEAD.LETTER.QUEUE と入力して Enter キーを押します。
- c) STRMQM MQMNAME (AS400) と入力して、キュー・マネージャーを開始します。
- d) 以下のパラメーターを使用して、未配布メッセージ・キューを作成します。

```

Local Queue
Queue name : DEAD.LETTER.QUEUE
Queue type : *LCL

```

キューの定義方法の詳細と例については、ステップ [27 ページの『2』](#) を参照してください。

2. キューを定義します。

- a) コマンド行で CRTMQMQ と入力します。

```

Create MQM Queue (CRTMQMQ)

Type choices, press Enter.
Queue name . . . . .
Queue type . . . . . *ALS, *LCL, *RMT

Bottom
F3=Exit F4=Prompt F5=Refresh F12=Cancel F13=How to use this display
F24=More keys
Parameter QNAME required.

```

- b) このパネルの 2 つのフィールドに値を入力して、Enter キーを押します。
次に、別のパネルが表示され、他のパラメーターの入力フィールドが示されます。他のすべてのキュー属性には、デフォルト値を設定できます。

3. チャンネルを定義します。

- a) コマンド行に CRTMQMCHL と入力して、「MQM チャンネルの作成」パネルを表示します。

```
Create MQM Channel (CRTMQMCHL)
Type choices, press Enter.
Channel name . . . . .
Channel type . . . . .          *RCVR, *SDR, *SVR, *RQSTR

Bottom
F3=Exit  F4=Prompt  F5=Refresh  F12=Cancel  F13=How to use this display
F24=More keys
Parameter CHLNAME required.
```

- b) このパネルの 2 つのフィールドに値を入力して、Enter キーを押します。

次に別のパネルが表示され、そこで前述の他のパラメーターの値を指定することができます。他のすべてのチャンネル属性には、デフォルト値を設定できます。

4. 構成例のチャンネルを構成します。

以下の例で使用されているパラメーターについては、[29 ページの『IBM i のチャンネル構成パラメーター』](#)を参照してください。

Windows 以下の例は、IBM i 上の IBM MQ を Windows 上の IBM MQ に接続するためのものです。別のプラットフォーム上の IBM MQ に接続するには、Windows の値ではなく、[29 ページの『IBM i のチャンネル構成パラメーター』](#)の表の適切な値を使用してください。

- a) 以下の例に示すように、送信側チャンネルを定義します。

- SNA の使用:

```
Local Queue
Queue name :   WINNT                F
Queue type :   *LCL
Usage :       *TMQ

Remote Queue
Queue name :   WINNT.REMOTEQ        D
Queue type :   *RMT
Remote queue : WINNT.LOCALQ         E
Remote Queue Manager : WINNT        C
Transmission queue : WINNT          F

Sender Channel
Channel Name : AS400.WINNT.SNA      G
Channel Type : *SDR
Transport type : *LU62
Connection name : WINNTCPIC         14
Transmission queue : WINNT          F
```

- TCP の使用:

```
Local Queue
Queue name :   WINNT                F
Queue type :   *LCL
Usage :       *TMQ
```

```

Remote Queue
  Queue name : WINNT.REMOTEQ           D
  Queue type : *RMT
  Remote queue : WINNT.LOCALQ         E
Remote Queue Manager : WINNT          C
Transmission queue : WINNT           F

Sender Channel
  Channel Name : AS400.WINNT.TCP      H
  Channel Type : *SDR
  Transport type : *TCP
  Connection name : WINNT.tcpip.hostname
  Transmission queue : WINNT          F

```

b) 以下の例に示すように、受信側チャンネルを定義します。

- SNA の使用:

```

Local Queue
  Queue name : AS400.LOCALQ           B
  Queue type : *LCL

Receiver Channel
  Channel Name : WINNT.AS400.SNA      I
  Channel Type : *RCVR
  Transport type : *LU62

```

- TCP の使用:

```

Local Queue
  Queue name : AS400.LOCALQ           B
  Queue type : *LCL

Receiver Channel
  Channel Name : WINNT.AS400.TCP      J
  Channel Type : *RCVR
  Transport type : *TCP

```

IBM i IBM i のチャンネル構成パラメーター

IBM i での構成例のチャンネルを構成するために必要なパラメーター。

26 ページの『[IBM i でのチャンネルの構成](#)』のステップ 28 ページの『4』では、5 ページの『[例: IBM MQ のクロスプラットフォーム通信のセットアップ](#)』で説明されているチャンネルを実装するために IBM i キュー・マネージャーで実行する構成について説明します。26 ページの『[IBM i でのチャンネルの構成](#)』の例は、IBM MQ for IBM i と IBM MQ for Windows を接続するためのものです。別のプラットフォーム上の IBM MQ に接続するには、Windows の値の代わりに該当する表の値を使用します。

注:

1. **太字**の語は推奨値であり、これらの例で使用されている IBM MQ オブジェクトの名前を反映していません。これらの値は製品のインストール済み環境で変更できますが、変更する場合は、このセクションの例を参照する際に必ず独自の値を使用してください。
2. IBM MQ チャンネル ping コマンド (**PNGMQMCHL**) は対話式に実行されますが、チャンネルを開始するとバッチ・ジョブが実行依頼されます。チャンネルの ping は正常に完了するのにチャンネルが開始しないときは、ネットワークと IBM MQ 定義は正しく、バッチ・ジョブの IBM i 環境の定義に誤りがあると思われます。例えば、QSYS2 が、個人用ライブラリー・リストにだけでなくライブラリー・リストのシステム用の部分にも組み込まれていることを確認してください。

表にリストされているオブジェクトの作成方法の詳細と例については、26 ページの『[IBM i でのチャンネルの構成](#)』を参照してください。

ローカル・ノードの定義

ID	パラメーター名	参照	使用例
A	キュー・マネージャー名		AS400
B	ローカル・キュー名		AS400.LOCALQ

Windows 上の IBM MQ への接続

Windows

表のこのセクションで指定する値は、42 ページの『Windows のチャンネル構成パラメーター』で使用した値と一致させる必要があります。

ID	パラメーター名	参照	使用例
C	リモート・キュー・マネージャー名	A	WINNT
D	リモート・キュー名		WINNT.REMOTEQ
E	リモート・システムのキュー名	B	WINNT.LOCALQ
F	伝送キュー名		WINNT
G	送信側 (SNA) チャンネル名		AS400.WINNT.SNA
H	送信側 (TCP/IP) チャンネル名		AS400.WINNT.TCP
I	受信側 (SNA) チャンネル名	G	WINNT.AS400.SNA
J	受信側 (TCP/IP) チャンネル名	H	WINNT.AS400.TCP

AIX 上の IBM MQ への接続

AIX

表のこのセクションで指定する値は、12 ページの『AIX のチャンネル構成パラメーター』で使用した値と一致させる必要があります。

ID	パラメーター名	参照	使用例
C	リモート・キュー・マネージャー名	A	AIX
D	リモート・キュー名		AIX.REMOTEQ
E	リモート・システムのキュー名	B	AIX.LOCALQ
F	伝送キュー名		AIX
G	送信側 (SNA) チャンネル名		AS400.AIX.SNA
H	送信側 (TCP/IP) チャンネル名		AS400AIX。TCP (TCP)
I	受信側 (SNA) チャンネル名	G	AIX.AS400.SNA
J	受信側 (TCP) チャンネル名	H	AIX.AS400.TCP

Linux 上の IBM MQ への接続

Linux

表のこのセクションで指定する値は、36 ページの『Linux のチャンネル構成パラメーター』で使用した値と一致させる必要があります。

表 16. Linux 上の IBM MQ に接続するための構成例

ID	パラメーター名	参照	使用例
C	リモート・キュー・マネージャー名	A	LINUX
D	リモート・キュー名		LINUX.REMOTEQ
E	リモート・システムのキュー名	B	LINUX.LOCALQ
F	伝送キュー名		LINUX
G	送信側 (SNA) チャンネル名		AS400.LINUX.SNA
H	送信側 (TCP/IP) チャンネル名		AS400.LINUX.TCP
I	受信側 (SNA) チャンネル名	G	LINUX.AS400.SNA
J	受信側 (TCP/IP) チャンネル名	H	LINUX.AS400.TCP

IBM MQ for z/OS への接続



表のこのセクションで指定する値は、47 ページの『Channel configuration parameters for z/OS』で使用した値と一致させる必要があります。

表 17. IBM MQ for z/OS に接続するための構成例

ID	パラメーター名	参照	使用例
C	リモート・キュー・マネージャー名	A	MVS
D	リモート・キュー名		MVS.REMOTEQ
E	リモート・システムのキュー名	B	MVS.LOCALQ
F	伝送キュー名		MVS
G	送信側 (SNA) チャンネル名		AS400.MVS.SNA
H	送信側 (TCP) チャンネル名		AS400.MVS.TCP
I	受信側 (SNA) チャンネル名	G	MVS.AS400.SNA
J	受信側 (TCP) チャンネル名	H	MVS.AS400.TCP

VSE/ESA システムへの接続

表のこのセクションで指定する値は、VSE/ESA システムで使用した値と一致させる必要があります。

表 18. VSE/ESA システムに接続するための構成例

ID	パラメーター名	参照	使用例
C	リモート・キュー・マネージャー名	A	VSE
D	リモート・キュー名		VSE.REMOTEQ
E	リモート・システムのキュー名	B	VSE.LOCALQ
F	伝送キュー名		VSE
G	送信側チャンネル名		AS400.VSE.SNA

表 18. VSE/ESA システムに接続するための構成例 (続き)			
ID	パラメーター名	参照	使用例
I	受信側チャンネル名	G	VSE.AS400.SNA

Linux 例: Linux での IBM MQ クロスプラットフォーム通信のセットアップ

この例は、Linux 上の IBM MQ から別のプラットフォーム上の IBM MQ への通信リンクをセットアップし、そのプラットフォームへの作業チャンネルを確立する方法を示しています。

始める前に

この例とその使用方法の背景情報については、5 ページの『例: IBM MQ のクロスプラットフォーム通信のセットアップ』および 7 ページの『クロスプラットフォーム通信の使用法の例』を参照してください。

このタスクについて

この例では、Linux 上の IBM MQ から以下のプラットフォームへのクロスプラットフォーム通信のセットアップについて説明します。

-  Windows
-  AIX
-  IBM i
-  z/OS

`MQ_INSTALLATION_PATH` は、IBM MQ がインストールされている上位ディレクトリーを表します。

LU 6.2 または TCP のいずれかを使用してネットワーク接続を確立します。

注: TCP の場合、一部の Linux ディストリビューションでは、inet デーモン (INETD) の代わりに拡張 inet デーモン (XINETD) を使用するようになりました。次に、inet デーモンまたは拡張 inet デーモンのいずれかを使用した TCP 接続の確立方法を説明します。

手順

1. LU6.2 を使用したネットワーク接続の確立

注: このセクションの情報は、IBM MQ for Linux (x86 プラットフォーム) のみに適用されます。IBM MQ for Linux (x86-64 プラットフォーム)、IBM MQ for Linux (zSeries s390x プラットフォーム)、または IBM MQ for Linux (Power プラットフォーム) には適用されません。

SNA over TCP/IP の構成に関する最新情報については、資料「[Communications Server for Data Center Deployment on Linux library](#)」から、ご使用のバージョンの Linux の管理ガイドを参照してください。

2. inet デーモン (INETD) を使用した TCP 接続の確立

a) /etc/services ファイルを編集します。

そのファイルに次の行がない場合は、それを以下のように追加します。

```
MQSeries    1414/tcp    # MQSeries channel listener
```

注: このファイルを編集するには、スーパーユーザーまたはルートとしてログインしなければなりません。

b) /etc/inetd.conf ファイルを編集します。

以下に示す行がこのファイルにない場合には、次のとおりに入力してファイルに追加してください。


```
MQSeries stream tcp nowait mqm MQ_INSTALLATION_PATH/bin/amqcrsta amqcrsta
[-m queue.manager.name ]
```

- c) 以下のコマンドにより、inetd のプロセス ID を調べる

```
ps -ef | grep inetd
```

- d) コマンドを実行します。

```
kill -1 inetd processid
```

システムに複数のキュー・マネージャーがあるために複数のサービスが必要な場合は、追加のキュー・マネージャーごとに 1 行を /etc/services と inetd.conf の両方に追加する必要があります。

以下に例を示します。

```
MQSeries1 1414/tcp
MQSeries2 1822/tcp
```

```
MQSeries1 stream tcp nowait mqm MQ_INSTALLATION_PATH/bin/amqcrsta amqcrsta -m QM1
MQSeries2 stream tcp nowait mqm MQ_INSTALLATION_PATH/bin/amqcrsta amqcrsta -m QM2
```

このようにすると、1 つの TCP ポートのキューに入る未解決の接続要求の数に制限がある場合、エラー・メッセージの生成が回避されます。未解決の接続要求の数については、[TCP リスナー・バックログ・オプションの使用](#)を参照してください。

Linux 上の inetd プロセスは、TCP ポートのインバウンド接続の速度を制限する場合があります。デフォルトは 60 秒間隔内で 40 の接続です。さらに高速であることが必要な場合は、ピリオド (.) に続けて新規制限を inetd.conf 内の適切なサービスの nowait パラメーターに付加して、60 秒間隔内のインバウンド接続の数に対する新規制限を指定します。例えば、60 秒間隔内で 500 の接続の制限の場合は、以下を使用します。

```
MQSeries stream tcp nowait.500 mqm / MQ_INSTALLATION_PATH/bin/amqcrsta amqcrsta -m QM1
```

MQ_INSTALLATION_PATH は、IBM MQ がインストールされている上位ディレクトリーを表します。

3. 拡張 inet デーモン (XINETD) を使用した TCP 接続の確立

次に、Red Hat Linux への拡張 inet デーモンの実装方法について説明します。別の Linux 配布版を使用している場合は、これらの説明を変更しなければならないことがあります。

- a) /etc/services ファイルを編集します。

そのファイルに次の行がない場合は、それを以下のように追加します。

```
MQSeries 1414/tcp # MQSeries channel listener
```

そのファイルに次の行がない場合は、それを以下のように追加します。

```
MQSeries 1414/tcp # MQSeries channel listener
```

- b) 以下のスタanzas をファイルに追加して、XINETD 構成ディレクトリー /etc/xinetd.d に IBM MQ というファイルを作成します。

```
# IBM MQ service for XINETD
service MQSeries
{
    disable          = no
    flags            = REUSE
```

```

socket_type    = stream
wait           = no
user           = mqm
server         = MQ_INSTALLATION_PATH/bin/amqcrsta
server_args    = -m queue.manager.name
log_on_failure += USERID
}

```

c) 次のコマンドを実行して、拡張 inet デーモンを再始動します。

```
/etc/rc.d/init.d/xinetd restart
```

システムに複数のキュー・マネージャーがあるために複数のサービスが必要な場合は、追加のキュー・マネージャーごとに /etc/services に 1 行追加する必要があります。サービスごとに /etc/xinetd.d ディレクトリーにファイルを作成するか、以前に作成した IBM MQ ファイルにスタンプを追加することができます。

Linux 上の xinetd プロセスは、TCP ポートのインバウンド接続の速度を制限する場合があります。デフォルトは 10 秒間隔内で 50 の接続です。さらに高速であることが必要な場合は、xinetd 構成ファイル内で 'CPS' 属性を指定することで、インバウンド接続の速度に対する新規制限を指定します。例えば、60 秒間隔内で 500 の接続の制限の場合は、以下を使用します。

```
cps = 500 60
```

4. TCP/IP 接続が確立されたので、構成を完了します。

34 ページの『Linux でのチャンネルの構成』に移動します。

Linux Linux でのチャンネルの構成

Linux での構成例のために IBM MQ を構成するには、キュー・マネージャーの基本構成ステップを実行してから、送信側チャンネルと受信側チャンネルを構成します。

始める前に

プロセスを開始する前に、まず mqm ユーザー ID と mqm グループを作成し、パスワードを設定したことを確認してください。

次のコマンドを使用して、チャンネルを開始します。

```
runmqchl -c channel.name
```

このタスクについて

注:

1. サンプル・プログラムは MQ_INSTALLATION_PATH/samp にインストールされています。ここで、MQ_INSTALLATION_PATH は IBM MQ がインストールされている上位のディレクトリーを表します。
2. エラー・ログは /var/mqm/qmgrs/ qmgrname /エラーに保管されます。
3. コマンド・インタプリタ **runmqsc** を使用して、管理コマンドを入力している場合、行の終わりにある + は、次の行が継続していることを示します。必ず最後のパラメーターと継続文字との間にスペースを入れるようにします。

手順

1. 基本構成をセットアップします。

- a) 以下のコマンドを使用して、UNIX プロンプトからキュー・マネージャーと一連のデフォルト・オブジェクトを作成します。

```
crtmqm -u dlqname -q linux
```

ここで、

linux

キュー・マネージャーの名前

-q

これがデフォルトのキュー・マネージャーになることを示す

-u dlqname

送達不能キューの名前を指定します。

b) 次のコマンドを使用して、UNIX プロンプトからキュー・マネージャーを開始します。

```
strmqm linux
```

ここで、*linux* は、作成時に、キュー・マネージャーに指定された名前です。

2. 構成例のチャンネルを構成します。

以下の例で使用されているパラメーターについて詳しくは、36 ページの『Linux のチャンネル構成パラメーター』を参照してください。いずれの場合も、この例では MQSC コマンドを示しています。Linux コマンド行から **runmqsc** を開始して各コマンドを順番に入力するか、コマンドをコマンド・ファイルにビルドします。

Windows 以下の例は、Linux 上の IBM MQ を Windows 上の IBM MQ に接続するためのものです。別のプラットフォーム上の IBM MQ に接続するには、Windows の値ではなく、36 ページの『Linux のチャンネル構成パラメーター』の表の適切な値を使用してください。

a) 以下の例に示すように、送信側チャンネルを定義します。

• SNA の使用

```
def q1 (WINNT) +                                     F
  usage(xmitq) +
  replace

def qr (WINNT.REMOTEQ) +                             D
  rname(WINNT.LOCALQ) +                             E
  rqmname(WINNT) +                                  C
  xmitq(WINNT) +                                     F
  replace

def chl (LINUX.WINNT.SNA) chltype(sdr) +            G
  triptype(lu62) +
  conname('WINNTCPIC') +                            14
  xmitq(WINNT) +                                     F
  replace
```

• TCP の使用

```
def q1 (WINNT) +                                     F
  usage(xmitq) +
  replace

def qr (WINNT.REMOTEQ) +                             D
  rname(WINNT.LOCALQ) +                             E
  rqmname(WINNT) +                                  C
  xmitq(WINNT) +                                     F
  replace

def chl (LINUX.WINNT.TCP) chltype(sdr) +            H
  triptype(tcp) +
  conname(remote_tcpip_hostname) +
  xmitq(WINNT) +                                     F
  replace
```

b) 以下の例に示すように、受信側チャンネルを定義します。

• SNA の使用:

```
def q1 (LINUX.LOCALQ) replace                        B
```

```
def chl (WINNT.LINUX.SNA) chltype(rcvr) +      I
  trptype(lu62) +
  replace
```

- TCP の使用:

```
def ql (LINUX.LOCALQ) replace                  B

def chl (WINNT.LINUX.TCP) chltype(rcvr) +     J
  trptype(tcp) +
  replace
```

Linux Linux のチャネル構成パラメーター

Linux での構成例のチャネルを構成するために必要なパラメーター。

34 ページの『Linux でのチャネルの構成』のステップ 35 ページの『2』では、5 ページの『例: IBM MQ のクロスプラットフォーム通信のセットアップ』で説明されているチャネルを実装するために Linux キュー・マネージャーで実行する構成について説明します。34 ページの『Linux でのチャネルの構成』の例は、IBM MQ for IBM i と IBM MQ for Windows を接続するためのものです。別のプラットフォーム上の IBM MQ に接続するには、Windows の値の代わりに該当する表の値を使用します。

注: 太字の語は推奨値であり、これらの例で使用されている IBM MQ オブジェクトの名前を反映しています。これらの値は製品インストール済み環境で変更できますが、変更する場合は、このセクションの例を参照する際に必ず独自の値を使用してください。

ローカル・ノードの定義

ID	パラメーター名	参照	使用例
A	キュー・マネージャー名		LINUX
B	ローカル・キュー名		LINUX.LOCALQ

Windows 上の IBM MQ への接続

Windows

表のこのセクションで指定する値は、42 ページの『Windows のチャネル構成パラメーター』で指定した値と一致させる必要があります。

ID	パラメーター名	参照	使用例
C	リモート・キュー・マネージャー名	A	WINNT
D	リモート・キュー名		WINNT.REMOTEQ
E	リモート・システムのキュー名	B	WINNT.LOCALQ
F	伝送キュー名		WINNT
G	送信側 (SNA) チャネル名		LINUX.WINNT.SNA
H	送信側 (TCP/IP) チャネル名		LINUX.WINNT.TCP
I	受信側 (SNA) チャネル名	G	WINNT.LINUX.SNA
J	受信側 (TCP) チャネル名	H	WINNT.LINUX.TCP

AIX 上の IBM MQ への接続

AIX

表のこのセクションで指定する値は、10 ページの『AIX でのチャンネルの構成』で指定した値と一致させる必要があります。

ID	パラメーター名	参照	使用例
C	リモート・キュー・マネージャー名	A	AIX
D	リモート・キュー名		AIX.REMOTEQ
E	リモート・システムのキュー名	B	AIX.LOCALQ
F	伝送キュー名		AIX
G	送信側 (SNA) チャンネル名		.LINUX.AIX.SNA
H	送信側 (TCP/IP) チャンネル名		LINUX.AIX.TCP
I	受信側 (SNA) チャンネル名	G	AIX.LINUX.SNA
J	受信側 (TCP/IP) チャンネル名	H	AIX.LINUX.TCP

IBM MQ for IBM i への接続

IBM i

表のこのセクションで指定する値は、29 ページの『IBM i のチャンネル構成パラメーター』で指定した値と一致させる必要があります。

ID	パラメーター名	参照	使用例
C	リモート・キュー・マネージャー名	A	AS400
D	リモート・キュー名		AS400.REMOTEQ
E	リモート・システムのキュー名	B	AS400.LOCALQ
F	伝送キュー名		AS400
G	送信側 (SNA) チャンネル名		LINUX.AS400.SNA
H	送信側 (TCP) チャンネル名		LINUX.AS400.TCP
I	受信側 (SNA) チャンネル名	G	AS400.LINUX.SNA
J	受信側 (TCP) チャンネル名	H	AS400.LINUX.TCP

IBM MQ for z/OS への接続

z/OS

表のこのセクションで指定する値は、47 ページの『Channel configuration parameters for z/OS』で指定した値と一致させる必要があります。

ID	パラメーター名	参照	使用例
C	リモート・キュー・マネージャー名	A	MVS
D	リモート・キュー名		MVS.REMOTEQ

表 23. IBM MQ for z/OS に接続するための構成例 (続き)

ID	パラメーター名	参照	使用例
E	リモート・システムのキュー名	B	MVS.LOCALQ
F	伝送キュー名		MVS
G	送信側 (SNA) チャンネル名		LINUX.MVS.SNA
H	送信側 (TCP) チャンネル名		LINUX.MVS.TCP
I	受信側 (SNA) チャンネル名	G	MVS.LINUX.SNA
J	受信側 (TCP) チャンネル名	H	MVS.LINUX.TCP

キュー共有グループを使用した IBM MQ for z/OS への接続

z/OS

表のこのセクションで指定する値は、55 ページの『Shared channel configuration parameters』で指定した値と一致させる必要があります。

表 24. キュー共有グループを使用して IBM MQ for z/OS に接続するための構成例

ID	パラメーター名	参照	使用例
C	リモート・キュー・マネージャー名	A	QSG
D	リモート・キュー名		QSG.REMOTEQ
E	リモート・システムのキュー名	B	QSG.SHAREDQ
F	伝送キュー名		QSG
G	送信側 (SNA) チャンネル名		LINUX.QSG.SNA
H	送信側 (TCP) チャンネル名		LINUX.QSG.TCP
I	受信側 (SNA) チャンネル名	G	QSG.LINUX.SNA
J	受信側 (TCP) チャンネル名	H	QSG.LINUX.TCP

Windows 例: Windows での IBM MQ クロスプラットフォーム通信のセットアップ

この例は、Windows 上の IBM MQ から別のプラットフォーム上の IBM MQ への通信リンクをセットアップし、そのプラットフォームへの作業チャンネルを確立する方法を示しています。

始める前に

この例とその使用方法の背景情報については、5 ページの『例: IBM MQ のクロスプラットフォーム通信のセットアップ』および 7 ページの『クロスプラットフォーム通信の使用法の例』を参照してください。

このタスクについて

この例では、Windows 上の IBM MQ から以下のプラットフォームへのクロスプラットフォーム通信のセットアップについて説明します。

-  AIX
-  IBM i
-  Linux

-  z/OS

手順

1. LU6.2 を使用してネットワーク接続を確立します。

AnyNet® SNA over TCP/IP の構成については、「[AnyNet SNA over TCP/IP](#)」および「[Communications Server for Windows](#)」を参照してください。

2. TCP を使用してネットワーク接続を確立します。

Windows システムとともに出荷される TCP スタックには、*inet* デーモンまたはそれに相当するものは組み込まれていません。

チャンネルを開始する前に、リスナーを明示的に開始する必要があります。これにより、インバウンド送信チャンネルからの要求に回答して受信チャンネルが自動的に開始されるようになります。

以下のコマンドを使用して、IBM MQ TCP リスナーを開始します。

```
runmqtsr -t tcp
```

3. NetBIOS を使用してネットワーク接続を確立します。

- a) チャンネルの両端で、IBM MQ チャンネル・プロセスが使用するローカル NetBIOS 名をキュー・マネージャー構成ファイル `qm.ini` に指定します。

例えば、送信側の Windows の NETBIOS スタンザは、一例として次のようになります。

```
NETBIOS:  
LocalName=WNTNETB1
```

受信側では、次のようになります。

```
NETBIOS:  
LocalName=WNTNETB2
```

各 IBM MQ プロセスが必ず別々のローカル NetBIOS 名を使用するようにします。システム名は既に Windows で使用されているので、NetBIOS 名として使用しないでください。

- b) チャンネルの送信側と受信側のシステムで使用している LAN アダプター番号を確認します。

IBM MQ for Windows の論理アダプター番号 0 のデフォルトは、インターネット・プロトコル・ネットワークで実行される NetBIOS です。ネイティブの NetBIOS を使用するには、論理アダプター番号 1 を選択する必要があります。[LAN アダプター番号の設定](#)を参照してください。

Windows レジストリーの NETBIOS スタンザに正しい LAN アダプター番号を指定します。以下に例を示します。

```
NETBIOS:  
AdapterNum=1
```

- c) 送信側のチャンネル開始が作動するように、次の MQNAME 環境変数でローカル NetBIOS 名を指定します。

```
SET MQNAME=WNTNETB1I
```

注：この名前は、固有の名前である必要があります。

- d) 送信側で、チャンネルの受信側で使用する NetBIOS 名を指定して、チャンネルを定義します。

以下に例を示します。

```
DEFINE CHANNEL (WINNT.OS2.NET) CHLTYPE(SDR) +  
TRPTYPE(NETBIOS) +  
CONNAME(WNTNETB2) +
```

```
XMITQ(OS2) +
MCATYPE(THREAD) +
REPLACE
```

Windows では、送信側チャンネルをスレッドとして実行する必要があるため、オプション MCATYPE(THREAD) を指定する必要があります。

- e) 受信側で、対応する受信側チャンネルを定義します。
以下に例を示します。

```
DEFINE CHANNEL (WINNT.OS2.NET) CHLTYPE(RCVR) +
TRPTYPE(NETBIOS) +
REPLACE
```

- f) チャンネル・イニシエーターを開始します。
各新規チャンネルは、新規プロセスとしてではなく、スレッドとして開始されます。

```
runmqchi
```

- g) 受信側で、IBM MQ リスナーを開始します。

```
runmqclsr -t netbios
```

オプションで、キュー・マネージャー名、NetBIOS ローカル名、セッション数、名前数、およびコマンド数に値を指定できます。NetBIOS 接続のセットアップの詳細については、[Windows での NetBIOS 接続の定義](#)を参照してください。

4. ネットワーク接続が確立されたので、構成を完了します。40 ページの『[Windows でのチャンネルの構成](#)』を参照してください。

Windows Windows でのチャンネルの構成

Windows での構成例のために IBM MQ を構成するには、キュー・マネージャーの基本構成ステップを実行してから、送信側チャンネルと受信側チャンネルを構成します。

このタスクについて

注:

1. サンプル・プログラム AMQSBCG を使用して、キュー内のすべてのメッセージのコンテンツとヘッダーを表示できます。以下に例を示します。

```
AMQSBCG q_name qmgr_name
```

これは、キュー・マネージャー *qmgr_name* で定義されるキュー *q_name* のコンテンツを表示します。
あるいは、IBM MQ エクスプローラー内でメッセージ・ブラウザーを使用できます。

2. コマンドを使用して、コマンド・プロンプトからすべてのチャンネルを開始させることができます。

```
runmqchl -c channel.name
```

3. エラー・ログは、ディレクトリー *MQ_INSTALLATION_PATH\qmgrs\qmgrname\errors* および *MQ_INSTALLATION_PATH\qmgrs\@system\errors* にあります。どちらの場合でも、最新メッセージは *amqerr01.log* の最後にあります。

MQ_INSTALLATION_PATH は、IBM MQ がインストールされている上位ディレクトリーを表します。

4. コマンド・インタープリター **runmqsc** を使用して、管理コマンドを入力している場合、行の終わりにある + は、次の行が継続していることを示します。必ず最後のパラメーターと継続文字との間にスペースを入れるようにします。

手順

1. コマンド・プロンプトを使用して基本構成をセットアップするには、以下の手順を実行します。

- a) 次のコマンドを使用して、キュー・マネージャーとデフォルト・オブジェクトのセットを作成します。

```
critmqm -u dlqname -q winnt
```

ここで、

winnt

キュー・マネージャーの名前

-q

これがデフォルトのキュー・マネージャーになることを示す

-u dlqname

未配布メッセージ・キューの名前を指定する

- b) 次のコマンドを使用して、キュー・マネージャーを起動します。

```
strmqm winnt
```

ここで、*winnt* は、作成時にキュー・マネージャーに指定した名前です。

2. 構成例のチャンネルを構成します。

以下の例で使用されているパラメーターについては、[42 ページの『Windows のチャンネル構成パラメーター』](#)を参照してください。いずれの場合も、この例では MQSC コマンドを示しています。Linux コマンド行から **runmqsc** を開始して各コマンドを順番に入力するか、コマンドをコマンド・ファイルにビルドします。これらの例は、IBM MQ for Windows と IBM MQ for AIX を接続するためのものです。別のプラットフォーム上の IBM MQ に接続するには、IBM MQ for AIX の値ではなく、[42 ページの『Windows のチャンネル構成パラメーター』](#)の表の適切な値を使用してください。

- a) 以下の例に示すように、送信側チャンネルを定義します。

• SNA の使用

```
def ql (AIX) +                               F
  usage(xmitq) +
  replace

def qr (AIX.REMOTEQ) +                       D
  rname(AIX.LOCALQ) +                       E
  rqmname(AIX) +                             C
  xmitq(AIX) +                               F
  replace

def chl (WINNT.AIX.SNA) chltype(sdr) +      G
  trptype(lu62) +
  conname(AIXCPIC) +                         18
  xmitq(AIX) +                               F
  replace
```

• TCP の使用

```
def ql (AIX) +                               F
  usage(xmitq) +
  replace

def qr (AIX.REMOTEQ) +                       D
  rname(AIX.LOCALQ) +                       E
  rqmname(AIX) +                             C
  xmitq(AIX) +                               F
  replace

def chl (WINNT.AIX.TCP) chltype(sdr) +      H
  trptype(tcp) +
  conname(remote_tcpip_hostname) +
```

```
xmitq(AIX) + replace F
```

b) 以下の例に示すように、受信側チャンネルを定義します。

- SNA の使用:

```
def ql (WINNT.LOCALQ) replace B
def chl (AIX.WINNT.SNA) chltype(rcvr) + I
  trptype(lu62) +
  replace
```

- TCP の使用:

```
def ql (WINNT.LOCALQ) replace B
def chl (AIX.WINNT.TCP) chltype(rcvr) + J
  trptype(tcp) +
  replace
```

次のタスク

自動始動

IBM MQ for Windows では、キュー・マネージャーとそのチャンネル・イニシエーター、チャンネル、リスナー、およびコマンド・サーバーの開始を自動化できます。

キュー・マネージャーのサービスを定義するには、IBM MQ Services スナップインを使用します。通信セットアップのテストが正常に終了したら、スナップイン内で該当するサービスを「**自動**」に設定します。このファイルは、システム始動時に、指定された IBM MQ サービスによって読み取られます。

詳しくは、[IBM MQ の管理](#)を参照してください。

プロセスまたはスレッドとしてのチャンネルの実行

IBM MQ for Windows は、送信チャンネルを Windows プロセスまたは Windows スレッドとして実行する柔軟性を備えています。どちらで実行するかは、送信側チャンネル定義の MCATYPE パラメーターに指定します。

ほとんどのインストール環境では、多数の同時チャンネル接続をサポートするために必要な実メモリと仮想メモリがより少なくすむので、送信チャンネルはスレッドとして実行されます。ただし、NetBIOS 接続では、送信メッセージ・チャンネル・エージェント用に別個プロセスが必要とされます。

Windows Windows のチャンネル構成パラメーター

Windows での構成例のチャンネルを構成するために必要なパラメーター。

40 ページの『[Windows でのチャンネルの構成](#)』のステップ 41 ページの『[2](#)』では、5 ページの『[例: IBM MQ のクロスプラットフォーム通信のセットアップ](#)』で説明されているチャンネルを実装するために Linux キュー・マネージャーで実行する構成について説明します。40 ページの『[Windows でのチャンネルの構成](#)』の例は、IBM MQ for Windows と IBM MQ for AIX を接続するためのものです。別のプラットフォーム上の IBM MQ に接続するには、Windows の値の代わりに該当する表の値を使用します。

注: 太字の語は推奨値であり、これらの例で使用されている IBM MQ オブジェクトの名前を反映しています。これらの値は製品のインストール済み環境で変更できますが、変更する場合は、このセクションの例を参照する際に必ず独自の値を使用してください。

それぞれの構成について該当する MQSC コマンドを示しています。コマンド・プロンプトから **runmqsc** を開始して、次に各コマンドを入力するか、またはコマンドをコマンド・ファイルに組み込みます。

IBM MQ for Windows と IBM MQ for AIX の接続のための例が示されています。他のプラットフォーム上の IBM MQ に接続するには、Windows 用の値の代わりに表から該当する一連の値を使用します。

ローカル・ノードの定義

ID	パラメーター名	参照	使用例
A	キュー・マネージャー名		WINNT
B	ローカル・キュー名		WINNT.LOCALQ

AIX 上の IBM MQ への接続

AIX

表のこのセクションで指定する値は、[12 ページの『AIX のチャンネル構成パラメーター』](#)で指定した値と一致させる必要があります。

	パラメーター名	参照	使用例
C	リモート・キュー・マネージャー名	A	AIX
D	リモート・キュー名		AIX.REMOTEQ
E	リモート・システムのキュー名	B	AIX.LOCALQ
F	伝送キュー名		AIX
G	送信側 (SNA) チャンネル名		WINNT.AIX.SNA
H	送信側 (TCP) チャンネル名		WINNT.AIX.TCP
I	受信側 (SNA) チャンネル名	G	AIX.WINNT.SNA
J	受信側 (TCP) チャンネル名	H	AIX.WINNT.TCP

IBM i 上の IBM MQ への接続

IBM i

表のこのセクションで指定する値は、[29 ページの『IBM i のチャンネル構成パラメーター』](#)で指定した値と一致させる必要があります。

ID	パラメーター名	参照	使用例
C	リモート・キュー・マネージャー名	A	AS400
D	リモート・キュー名		AS400.REMOTEQ
E	リモート・システムのキュー名	B	AS400.LOCALQ
F	伝送キュー名		AS400
G	送信側 (SNA) チャンネル名		WINNT.AS400.SNA
H	送信側 (TCP) チャンネル名		WINNT.AS400.TCP
I	受信側 (SNA) チャンネル名	G	AS400.WINNT.SNA
J	受信側 (TCP) チャンネル名	H	AS400.WINNT.TCP

IBM MQ for z/OS への接続

z/OS

表のこのセクションで指定する値は、47 ページの『Channel configuration parameters for z/OS』で指定した値と一致させる必要があります。

ID	パラメーター名	参照	使用例
C	リモート・キュー・マネージャー名	A	MVS
D	リモート・キュー名		MVS.REMOTEQ
E	リモート・システムのキュー名	B	MVS.LOCALQ
F	伝送キュー名		MVS
G	送信側 (SNA) チャンネル名		WINNT.MVS.SNA
H	送信側 (TCP) チャンネル名		WINNT.MVS.TCP
I	受信側 (SNA) チャンネル名	G	MVS.WINNT.SNA
J	受信側 (TCP) チャンネル名	H	MVS.WINNT.TCP

キュー共有グループを使用した IBM MQ for z/OS への接続

z/OS

表のこのセクションで指定する値は、55 ページの『Shared channel configuration parameters』で指定した値と一致させる必要があります。

ID	パラメーター名	参照	使用例
C	リモート・キュー・マネージャー名	A	QSG
D	リモート・キュー名		QSG.REMOTEQ
E	リモート・システムのキュー名	B	QSG.SHAREDQ
F	伝送キュー名		QSG
G	送信側 (SNA) チャンネル名		WINNT.QSG.SNA
H	送信側 (TCP) チャンネル名		WINNT.QSG.TCP
I	受信側 (SNA) チャンネル名	G	QSG.WINNT.SNA
J	受信側 (TCP) チャンネル名	H	QSG.WINNT.TCP

z/OS

Example: setting up IBM MQ cross-platform communication on

z/OS

This example shows how to set up communication links from IBM MQ on z/OS to IBM MQ on another platform and establish a working channel to that platform.

Before you begin

For background information about this example and how to use it, see “例: IBM MQ のクロスプラットフォーム通信のセットアップ” on page 5 and “クロスプラットフォーム通信の使用法の例” on page 7.

About this task

This example covers setting up cross platform communication from IBM MQ on z/OS to the following platforms:

- **Windows** Windows
- **AIX** AIX
- **Linux** Linux
- **IBM i** IBM i
- VSE/ESA

You can also connect any of the following:

- z/OS to z/OS
- z/OS to MVS
- MVS to MVS

Procedure

1. Establish a network connection.

- Establishing an LU 6.2 connection

For the latest information about configuring SNA over TCP/IP, refer to the following online IBM documentation: [Communications Server for z/OS](#).

- Establishing a TCP connection

Alter the queue manager object to use the correct distributed queuing parameters using the following command. You must add the name of the TCP address space to the TCPNAME queue manager attribute.

```
ALTER QMGR TCPNAME(TCPIP)
```

The TCP connection is now established. You are ready to complete the configuration.

2. Configure the channels.

See [“Configuring the channels on IBM MQ for z/OS”](#) on page 45 for details on how you configure the channels.

z/OS Configuring the channels on IBM MQ for z/OS

To configure IBM MQ for the example configuration on z/OS, start and configure the channels and listeners.

Procedure

1. Start the channel initiator using the command:

```
/cpf START CHINIT 1
```

2. Start an LU 6.2 listener using the command:

```
/cpf START LSTR LUNAME( M1 ) TRPTYPE(LU62)
```

The LUNAME of M1 refers to the symbolic name you gave your LU (5). You must specify TRPTYPE(LU62), otherwise the listener assumes that you want TCP.

3. Start a TCP listener using the command:

```
/cpf START LSTR
```

If you want to use a port other than 1414 (the default IBM MQ port), use the command:

```
/cpf START LSTR PORT( 1555 )
```

IBM MQ channels do not initialize successfully if the channel negotiation detects that the message sequence number is different at each end. You might need to reset these channels manually.

4. Configure the channels for the example configuration.

For more information about the parameters used in the following examples, see [“Channel configuration parameters for z/OS”](#) on page 47. These examples are for connecting IBM MQ for z/OS and IBM MQ for Windows. To connect to IBM MQ on another platform use the values from the appropriate table in [“Channel configuration parameters for z/OS”](#) on page 47 instead of the values for Windows.

a) Define the sender channel as shown in the following example:s

For LU 6.2:

```
Local Queue
  Object type : QLOCAL
  Name       : WINNT
  Usage      : X (XmitQ)           F

Remote Queue
  Object type : QREMOTE
  Name       : WINNT.REMOTEQ      D
Name on remote system : WINNT.LOCALQ E
Remote system name : WINNT        C
Transmission queue : WINNT        F

Sender Channel
  Channel name : MVS.WINNT.SNA     G
  Transport type : L (LU6.2)
Transmission queue name : WINNT    F
Connection name : M3              13
```

For TCP:

```
Local Queue
  Object type : QLOCAL
  Name       : WINNT
  Usage      : X (XmitQ)           F

Remote Queue
  Object type : QREMOTE
  Name       : WINNT.REMOTEQ      D
Name on remote system : WINNT.LOCALQ E
Remote system name : WINNT        C
Transmission queue : WINNT        F

Sender Channel
  Channel name : MVS.WINNT.TCP     H
  Transport type : T (TCP)
Transmission queue name : WINNT    F
Connection name : winnt.tcpip.hostname
```

b) Define the receiver channel as shown in the following examples:

For LU 6.2:

```
Local Queue
  Object type : QLOCAL
  Name       : MVS.LOCALQ         B
  Usage      : N (Normal)

Receiver Channel
  Channel name : WINNT.MVS.SNA     I
```

For TCP:

```

Local Queue
  Object type : QLOCAL
             Name : MVS.LOCALQ           B
             Usage : N (Normal)

Receiver Channel
  Channel name : WINNT.MVS.TCP         J

```

Channel configuration parameters for z/OS

The parameters needed to configure the channels for the example configuration on z/OS.

Step “4” on page 46 of “Configuring the channels on IBM MQ for z/OS” on page 45 describes the configuration to be performed on the z/OS queue manager to implement the channel described in “例: IBM MQ のクロスプラットフォーム通信のセットアップ” on page 5. The examples in “Configuring the channels on IBM MQ for z/OS” on page 45 are for connecting IBM MQ for z/OS and IBM MQ for Windows. To connect to IBM MQ on another platform use the values from the appropriate table in place of the values for Windows.

Note: The words in **bold** are suggested values and reflect the names of IBM MQ objects used throughout these examples. You can change them in your product installation but, if you do, make sure that you use your own values when working through the examples in this section

Definition for local node

Table 30. Configuration examples for the definition for the local node			
ID	Parameter Name	Reference	Example Used
A	Queue Manager Name		MVS
B	Local queue name		MVS.LOCALQ

Connection to IBM MQ on Windows

The values in this section of the table must match the values used in “Windows のチャンネル構成パラメーター” on page 42, as indicated.



Table 31. Configuration examples for connecting to IBM MQ on Windows			
ID	Parameter Name	Reference	Example Used
C	Remote queue manager name	A	WINNT
D	Remote queue name		WINNT.REMOTEQ
E	Queue name at remote system	B	WINNT.LOCALQ
F	Transmission queue name		WINNT
G	Sender (LU 6.2) channel name		MVS.WINNT.SNA
H	Sender (TCP) channel name		MVS.WINNT.TCP
I	Receiver (LU 6.2) channel name	G	WINNT.MVS.SNA
J	Receiver (TCP/IP) channel name	H	WINNT.MVS.TCP

Connection to IBM MQ on AIX

####

The values in this section of the table must match the values used in “[AIX のチャンネル構成パラメーター](#)” on page 12, as indicated.

Table 32. Configuration examples for connecting to IBM MQ on AIX

ID	Parameter Name	Reference	Example Used
  Connection to IBM MQ for AIX			
C	Remote queue manager name	A	AIX
D	Remote queue name		AIX.REMOTEQ
E	Queue name at remote system	B	AIX.LOCALQ
F	Transmission queue name		AIX
G	Sender (LU 6.2) channel name		MVS.AIX.SNA
H	Sender (TCP/IP) channel name		MVS.AIX.TCP
I	Receiver (LU 6.2) channel name	G	AIX.MVS.SNA
J	Receiver (TCP/IP) channel name	H	AIX.MVS.TCP

Connection to IBM MQ on Linux



The values in this section of the table must match the values used in “[Linux のチャンネル構成パラメーター](#)” on page 36, as indicated.

Table 33. Configuration examples for connecting to IBM MQ on Linux

ID	Parameter Name	Reference	Example Used
C	Remote queue manager name	A	LINUX
D	Remote queue name		LINUX.REMOTEQ
E	Queue name at remote system	B	LINUX.LOCALQ
F	Transmission queue name		LINUX
G	Sender (LU 6.2) channel name		MVS.LINUX.SNA
H	Sender (TCP) channel name		MVS.LINUX.TCP
I	Receiver (LU 6.2) channel name	G	LINUX.MVS.SNA
J	Receiver (TCP/IP) channel name	H	LINUX.MVS.TCP

Connection to IBM MQ on IBM i



The values in this section of the table must match the values used in “[IBM i のチャンネル構成パラメーター](#)” on page 29, as indicated.

Table 34. Configuration examples for connecting to IBM MQ on IBM i

ID	Parameter name	Reference	Example used
C	Remote queue manager name	A	AS400
D	Remote queue name		AS400.REMOTEQ
E	Queue name at remote system	B	AS400.LOCALQ

Table 34. Configuration examples for connecting to IBM MQ on IBM i (continued)

ID	Parameter name	Reference	Example used
F	Transmission queue name		AS400
G	Sender (LU 6.2) channel name		MVS.AS400.SNA
H	Sender (TCP/IP) channel name		MVS.AS400.TCP
I	Receiver (LU 6.2) channel name	G	AS400.MVS.SNA
J	Receiver (TCP/IP) channel name	H	AS400.MVS.TCP

z/OS Example: setting up IBM MQ cross-platform communication on z/OS using QSGs

This example shows how to set up communication links to a queue sharing group (QSG) from IBM MQ on Windows and AIX. You can also connect from z/OS to z/OS.

Before you begin

Setting up communication links from a queue sharing group to a platform other than z/OS is the same as described in [“Example: setting up IBM MQ cross-platform communication on z/OS”](#) on page 44.

For background information about this example and how to use it, see [“例: IBM MQ のクロスプラットフォーム通信のセットアップ”](#) on page 5 and [“クロスプラットフォーム通信の使用方の例”](#) on page 7.

Procedure

1. Establish a network connection using one of the following options.
 - Establish an LU 6.2 connection as described in [“Establishing an LU 6.2 connection into a queue sharing group”](#) on page 49.
 - Establish a TCP connection using Sysplex Distributor as described in [“Establishing a TCP connection using Sysplex Distributor”](#) on page 53.
2. Define some channels to complete the configuration after the connection is established. See [“Configuring shared channels on IBM MQ for z/OS ”](#) on page 54 for details of this process.

z/OS Establishing an LU 6.2 connection into a queue sharing group

There are two steps to establish an LU 6.2 connection. Defining yourself to the network and defining a connection to the partner.

About this task

Note: This example is for a connection to a Windows system but the task is the same for other platforms.

Procedure

1. Use VTAM Generic Resources to have one connection name to connect to the queue sharing group.
 - a) SYS1.PARMLIB(APPCPMxx) contains the start-up parameters for APPC. You must add a line to this file to tell APPC where to locate the sideinfo.

This line must be of the form:

```
SIDEINFO
  DATASET (APPC .APPCSI)
```

- b) Add another line to SYS1.PARMLIB(APPCPMxx) to define the local LU name you intend to use for the IBM MQ LU 6.2 group listener.

The line you add must take the form

```
LUADD ACBNAME(mvslu1)
      NOSCHED
      TPDATA(csq.appc1p)
      GRNAME(mvsg1)
```

Specify values for ACBNAME (9), TPDATA and GRNAME (10).

The NOSCHED parameter tells APPC that our new LU is not using the LU 6.2 scheduler (ASCH), but has one of its own. TPDATA refers to the Transaction Program data set in which LU 6.2 stores information about transaction programs. Again, IBM MQ does not use this parameter, but it is required by the syntax of the LUADD command.

- c) Start the APPC subsystem with the command:

```
START APPC, SUB=MSTR, APPC=xx
```

where *xx* is the suffix of the PARMLIB member in which you added the LU in step 1.

Note: If APPC is already running, it can be refreshed with the command:

```
SET APPC=xx
```

The effect of this is cumulative, that is, APPC does not lose its knowledge of objects already defined to it in this member or another PARMLIB member.

- d) Add the new LU to a suitable VTAM major node definition. These are typically in SYS1.VTAMLST. The APPL definition will look like the sample shown.

```
      MVSLU APPL ACBNAME=MVSLU1,      9
              APPXC=YES,
              AUTOSES=0,
              DDRAINL=NALLOW,
              DLOGMOD=#INTER,      6
              DMINWML=10,
              DMINWNR=10,
              DRESPL=NALLOW,
              DSESLIM=60,
              LMDENT=19,
              MODETAB=MTCICS,
              PARSESS=YES,
              VERIFY=NONE,
              SECACPT=ALREADYV,
              SRBEXIT=YES
```

- e) Activate the major node.

You can do this activation with the command:

```
V, NET, ACT, majornode
```

- f) Add entries defining your LU and generic resource name to the CPI-C side information data set, using the APPC utility program ATBSDFMU to do so.

Sample JCL is in *thlqual*.SCSQPROC(CSQ4SIDE) (where *thlqual* is the target library high-level qualifier for IBM MQ data sets in your installation.)

The entries you add will look like this example:

```
SIADD
      DESTNAME(G1)      11
      MODENAME(#INTER)
      TPNAME(MQSERIES)
```

```

PARTNER_LU(MVSLU1)    9
SIADD
  DESTNAME(G2)        12
  MODENAME(#INTER)
  TPNAME(MQSERIES)
  PARTNER_LU(MVSGR)   10

```

- g) Alter the queue manager object to use the correct distributed queuing parameters using the following command.

You must specify the local LU (9) assigned to your queue manager in the LUGROUP attribute of the queue manager.

```
ALTER QMGR LUGROUP(MVSLU1)
```

2. Define a connection to a partner by adding an entry to the CPI-C side information data set.

- a) Add an entry to the CPI-C side information data set to define the connection.

Sample JCL to do this definition is in *thlqual.SCSQPROC(CSQ4SIDE)*.

The entry you add looks like this:

```

SIADD
  DESTNAME(M3)        13
  MODENAME(#INTER)   14
  TPNAME(MQSERIES)   15
  PARTNER_LU(WINNTLU) 16

```

What to do next

The connection is now established. You are ready to complete the configuration.

Go to [“Configuring shared channels on IBM MQ for z/OS”](#) on page 54.

Configuration parameters for an LU 6.2 connection

The following table lists all the parameters required to set up communication from a z/OS system to IBM MQ on another platform.

The steps required to set up an LU 6.2 connection are described in [“Establishing an LU 6.2 connection into a queue sharing group”](#) on page 49, with numbered cross-references to the parameters in the example.

Numbers in the Reference column indicate that the value must match that in the appropriate example elsewhere in this section. The examples that follow in this section refer to the values in the ID column. The entries in the Parameter Name column are explained in [“Explanation of terms”](#) on page 52.

Definition for local node using generic resources

Table 35. Configuration examples for the definition for the local node using generic resources			
ID	Parameter name	Reference	Example used
1	Command prefix		/cpf
2	Network ID		NETID
3	Node name		MVSPU
6	Modename		#INTER
7	Local Transaction Program name		MQSERIES
8	LAN destination address		400074511092
9	Local LU name		MVSLU1
10	Generic resource name		MVSGR

Table 35. Configuration examples for the definition for the local node using generic resources (continued)

ID	Parameter name	Reference	Example used
11	Symbolic destination		G1
12	Symbolic destination for generic resource name		G2

Connection to IBM MQ on Windows

Windows

Table 36. Configuration examples for connecting to IBM MQ on Windows using LU 6.2

ID	Parameter name	Reference	Example used
13	Symbolic destination		M3
14	Modename	21	#INTER
15	Remote Transaction Program name	7	MQSERIES
16	Partner LU name	5	WINNTLU
21	Remote node ID	4	05D 30F65

Connection to IBM MQ on AIX

AIX

Table 37. Configuration examples for connecting to IBM MQ on AIX using LU 6.2

ID	Parameter name	Reference	Example used
13	Symbolic Destination		M4
14	Modename	18	#INTER
15	Remote Transaction Program name	6	MQSERIES
16	Partner LU name	4	AIXLU

Explanation of terms

1 Command prefix

This term is the unique command prefix of your IBM MQ for z/OS queue manager subsystem. The z/OS system programmer defines this value at installation time, in SYS1.PARMLIB(IEFSSNss), and can tell you the value.

2 Network ID

The VTAM startup procedure in your installation is partly customized by the ATCSTRxx member of the data set referenced by the DDNAME VTAMLST. The Network ID is the value specified for the NETID parameter in this member. For Network ID, you must specify the name of the NETID that owns the IBM MQ communications subsystem. Your network administrator can tell you the value.

3 Node name

VTAM, being a low-entry network node, does not have a Control Point name for Advanced Peer-to-Peer Networking (APPN) use. It does however have a system services control point name (SSCPNAME). For node name, you must specify the name of the SSCP that owns the IBM MQ communications subsystem. This value is defined in the same ATCSTRxx member as the Network ID. Your network administrator can tell you the value.

9 Local LU name

A logical unit (LU) is software that serves as an interface or translator between a transaction program and the network. It manages the exchange of data between transaction programs. The local LU name

is the unique VTAM APPLID of this IBM MQ subsystem. Your network administrator can tell you this value.

11 12 13 Symbolic destination

This term is the name you give to the CPI-C side information profile. You need a side information entry for each LU 6.2 listener.

6 14 Modename

This term is the name given to the set of parameters that control the LU 6.2 conversation. An entry with this name and similar attributes must be defined at each end of the session. In VTAM, this corresponds to a mode table entry. Your network administrator can assign this table entry to you.

7 15 Transaction Program name

IBM MQ applications trying to converse with this queue manager specify a symbolic name for the program to be run at the receiving end. This has been specified in the TPNAME attribute on the channel definition at the sender. For simplicity, wherever possible use a transaction program name of MQSERIES, or in the case of a connection to VSE/ESA, where the length is limited to 4 bytes, use MQTP.

See [Defining an LU6.2 connection for z/OS using APPC/MVS](#) for more information.

8 LAN destination address

This term is the LAN destination address that your partner nodes use to communicate with this host. When you are using a 3745 network controller, it is the value specified in the LOCADD parameter for the line definition to which your partner is physically connected. If your partner nodes use other devices such as 317X or 6611 devices, the address is set during the customization of those devices. Your network administrator can tell you this value.

10 Generic resource name

A generic resource name is a unique name assigned to a group of LU names used by the channel initiators in a queue sharing group.

16 Partner LU name

This term is the LU name of the IBM MQ queue manager on the system with which you are setting up communication. This value is specified in the side information entry for the remote partner.

21 Remote node ID

For a connection to Windows, this ID is the ID of the local node on the Windows system with which you are setting up communication.

Establishing a TCP connection using Sysplex Distributor

You can set up Sysplex distributor to use one connection name to connect to the queue sharing group.

Procedure

1. Define a Distributed DVIPA address as follows:

- a) Add a DYNAMICXCF statement to the IPCONFIG. This statement is used for inter-image connectivity using dynamically created XCF TCP/IP links.
- b) Use the VIPADYNAMIC block on each image in the Sysplex.

On the owning image, code a VIPADEFINE statement to create the DVIPA. Then code a VIPADISTRIBUTE statement to distribute it to all other or selected images.

On the backup image, code a VIPABACKUP statement for the DVIPA address.

2. Add the SHAREPORT option for the port to be shared in the PORT reservation list in the PROFILE data set if more than one channel initiator is to be started on any LPAR in the sysplex.

See [PORT statement](#) in the *z/OS Communications Server: IP Configuration Reference* for more information.

When you have completed these steps, the TCP connection is established. You are ready to complete the configuration.

What to do next

Go to “Configuring shared channels on IBM MQ for z/OS ” on page 54.

Configuring shared channels on IBM MQ for z/OS

Configure the shared channel by starting the channel initiator and issuing appropriate commands for your configuration.

About this task

There can be only one instance of the shared channel running at a time. If you try to start a second instance of the channel it fails (the error message varies depending on other factors). The shared synchronization queue tracks the channel status.

Important: IBM MQ channels do not initialize successfully if the channel negotiation detects that the message sequence number is different at each end. You might need to reset this manually.

Procedure

1. Start the channel initiator using the command:

```
/cpf START CHINIT
```

2. Start an LU6.2 group listener using the command:

```
/cpf START LSTR TRPTYPE(LU62) LUNAME( G1 ) INDISP(GROUP)
```

The LUNAME of G1 refers to the symbolic name you gave your LU (11).

3. Use the following command if you are using Virtual IP Addressing using Sysplex Distributor and want to listen on a specific address:

```
/cpf START LSTR TRPTYPE(TCP) PORT(1555) IPADDR( mvsvipa ) INDISP(GROUP)
```

4. Configure the channels for the example configuration.

For more information about the parameters used in the following examples, see “Shared channel configuration parameters” on page 55. These examples are for connecting IBM MQ for z/OS and Windows. To connect to IBM MQ on another platform, use the appropriate values from the tables in “Shared channel configuration parameters” on page 55 instead of the values for Windows.

- a) Define the shared sender channel as shown in the following examples.

Using LU 6.2:

```
Local Queue
  Object type : QLOCAL
  Name       : WINNT
  Usage      : X (XmitQ)
  Disposition : SHARED
  F

Remote Queue
  Object type : QREMOTE
  Name       : WINNT.REMOTEQ
  Name on remote system : WINNT.LOCALQ
  Remote system name : WINNT
  Transmission queue : WINNT
  Disposition : GROUP
  D
  E
  C
  F

Sender Channel
  Channel name : MVS.WINNT.SNA
  Transport type : L (LU6.2)
  Transmission queue name : WINNT
  Connection name : M3
  Disposition : GROUP
  G
  F
  13
```

Using TCP

```
Local Queue
  Object type : QLOCAL
  Name       : WINNT           F
  Usage     : X (XmitQ)
  Disposition : SHARED

Remote Queue
  Object type : QREMOTE
  Name       : WINNT.REMOTEQ   D
Name on remote system : WINNT.LOCALQ   E
Remote system name : WINNT       C
Transmission queue : WINNT       F
Disposition : GROUP

Sender Channel
  Channel name : QSG.WINNT.TCP   H
  Transport type : T (TCP)
Transmission queue name : WINNT   F
Connection name : winnt.tcpip.hostname
Disposition : GROUP
```

b) Define the shared receiver channel as shown in the following examples.

Using LU 6.2:

```
Local Queue
  Object type : QLOCAL
  Name       : QSG.SHAREDQ     B
  Usage     : N (Normal)
  Disposition : SHARED

Receiver Channel
  Channel name : WINNT.QSG.SNA   I
  Disposition : GROUP
```

Using TCP:

```
Local Queue
  Object type : QLOCAL
  Name       : QSG.SHAREDQ     B
  Usage     : N (Normal)
  Disposition : SHARED

Receiver Channel
  Channel name : WINNT.QSG.TCP   J
  Disposition : GROUP
```

Shared channel configuration parameters

The parameters needed to configure a shared channel for the example configuration on z/OS.

Step “4” on page 54 of “[Configuring shared channels on IBM MQ for z/OS](#)” on page 54 describes the configuration to be performed on the z/OS queue manager to implement the channel described in “[例: IBM MQ のクロスプラットフォーム通信のセットアップ](#)” on page 5. The examples in “[Configuring shared channels on IBM MQ for z/OS](#)” on page 54 are for connecting IBM MQ for z/OS and Windows. To connect to IBM MQ on another platform, use the values from the appropriate table in place of the values for Windows.

Note: The words in **bold** are suggested values and reflect the names of IBM MQ objects used throughout these examples. You can change them in your product installation but, if you do, make sure that you use your own values when working through the examples in this section.

Definition for local node

Table 38. Configuration examples for the definition for the local node

ID	Parameter Name	Reference	Example Used
A	Queue Manager Name		QSG
B	Local queue name		QSG.SHAREDQ

Connection to IBM MQ on Windows

Windows

The values in this section of the table must match the values used in “[Windows のチャネル構成パラメーター](#)” on page 42, as indicated.

Table 39. Configuration examples for connecting to to IBM MQ on Windows

ID	Parameter name	Reference	Example used
C	Remote queue manager name	A	WINNT
D	Remote queue name		WINNT.REMOTEQ
E	Queue name at remote system	B	WINNT.LOCALQ
F	Transmission queue name		WINNT
G	Sender (LU 6.2) channel name		QSG.WINNT.SNA
H	Sender (TCP) channel name		QSG.WINNT.TCP
I	Receiver (LU 6.2) channel name	G	WINNT.QSG.SNA
J	Receiver (TCP/IP) channel name	H	WINNT.QSG.TCP

Connection to IBM MQ on AIX

AIX

The values in this section of the table must match the values used in “[AIX のチャネル構成パラメーター](#)” on page 12, as indicated.

Table 40. Configuration examples for connecting to IBM MQ on AIX

ID	Parameter name	Reference	Example used
C	Remote queue manager name		AIX
D	Remote queue name		AIX.REMOTEQ
E	Queue name at remote system	B	AIX.LOCALQ
F	Transmission queue name		AIX
G	Sender (LU 6.2) channel name		QSG.AIX.SNA
H	Sender (TCP/IP) channel name		QSG.AIX.TCP
I	Receiver (LU 6.2) channel name	G	AIX.QSG.SNA
J	Receiver (TCP/IP) channel name	H	AIX.QSG.TCP

Example: setting up IBM MQ cross-platform communication for intra-group queuing on z/OS

This example shows how a typical payroll query application that currently uses distributed queuing to transfer small messages between queue managers could be migrated to use queue sharing groups and shared queues.

About this task

Three configurations are described to illustrate the use of distributed queuing, intra-group queuing with shared queues, and shared queues. The associated diagrams show only the flow of data in one direction, that is, from queue manager QMG1 to queue manager QMG3.

Procedure

1. Set up and run Configuration 1.
For more information, see [“Setting up and running configuration 1” on page 57.](#)
2. Set up and run Configuration 2.
For more information, see [“Setting up and running configuration 2” on page 60.](#)
3. Set up and run Configuration 3.
For more information, see [“Setting up and running configuration 3” on page 62.](#)

What to do next

You can expand the example in a number of ways by:

- Using channel triggering as well as application (PAYROLL and PAYROLL.REPLY queue) triggering.
- Configuring for communication using LU6.2.
- Configuring more queue managers to the queue sharing group. Then the server application can be cloned to run on other queue manager instances to provide multiple servers for the PAYROLL query queue.
- Increasing the number of instances of the payroll query requesting application to demonstrate the processing of requests from multiple clients.
- Using security (IGQAUT and IGQUSER).

Setting up and running configuration 1

Configuration 1 describes how distributed queuing is currently used to transfer messages between queue managers QMG1 and QMG3.

About this task

Configuration 1 shows a distributed queuing system that is used to transfer messages received by queue manager QMG1 from the payroll query to queue manager QMG2 and then finally on to queue manager QMG3, to be sent to the payroll server.

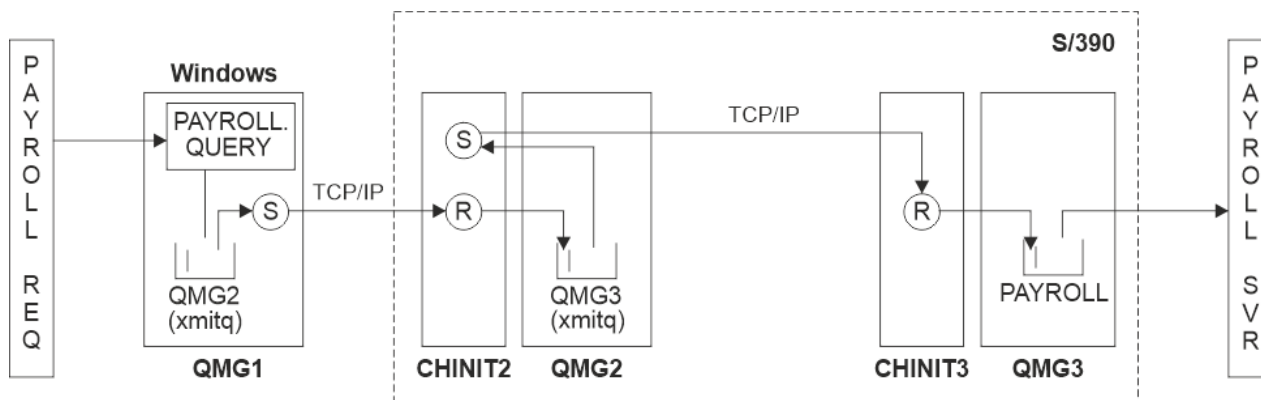


Figure 2. Configuration 1: z/OS using intra-group queuing

The flow of operations is as follows:

1. A query is entered using the payroll request application connected to queue manager QMG1.
2. The payroll request application puts the query on to remote queue PAYROLL.QUERY. As queue PAYROLL.QUERY resolves to transmission queue QMG2, the query is put on to transmission queue QMG2.
3. Sender channel (S) on queue manager QMG1 delivers the query to the partner receiver channel (R) on queue manager QMG2.
4. Receiver channel (R) on queue manager QMG2 puts the query on to queue PAYROLL on queue manager QMG3. As queue PAYROLL on QMG3 resolves to transmission queue QMG3, the query is put on to transmission queue QMG3.
5. Sender channel (S) on queue manager QMG2 delivers the query to the partner receiver channel (R) on queue manager QMG3.
6. Receiver channel (R) on queue manager QMG3 puts the query on to local queue PAYROLL.
7. The payroll server application connected to queue manager QMG3 retrieves the query from local queue PAYROLL, processes it, and generates a suitable reply.

The definitions required for Configuration 1 are as follows (note that the definitions do not take into account triggering, and that only channel definitions for communication using TCP/IP are provided).

Procedure

1. Procedure on QMG1:
 - a) Setup the remote queue definition:

```
DEFINE QREMOTE(PAYROLL.QUERY) DESCR('Remote queue for QMG3') REPLACE +
PUT(ENABLED) RNAME(PAYROLL) RQMNAME(QMG3) XMITQ(QMG2)
```

- b) Setup the transmission queue definition:

```
DEFINE QLOCAL(QMG2) DESCR('Transmission queue to QMG2') REPLACE +
PUT(ENABLED) USAGE(XMITQ) GET(ENABLED)
```

- c) Setup the sender channel definition using TCP/IP:

```
DEFINE CHANNEL(QMG1.TO.QMG2) CHLTYPE(SDR) TRPTYPE(TCP) REPLACE +
DESCR('Sender channel to QMG2') XMITQ(QMG2) CONNAME('MVSQMG2(1415)')
```

Note: Replace MVSQMG2(1415) with your queue manager connection name and port.

- d) Setup the receiver channel definition using TCP/IP:

```
DEFINE CHANNEL(QMG2.TO.QMG1) CHLTYPE(RCVR) TRPTYPE(TCP) +
REPLACE DESCR('Receiver channel from QMG2')
```

e) Setup the reply-to queue definition:

```
DEFINE QLOCAL(PAYROLL.REPLY) REPLACE PUT(ENABLED) GET(ENABLED) +
DESCR('Reply queue for replies to payroll queries sent to QMG3')
```

2. Procedure on QMG2:

a) Setup the transmission queue definition:

```
DEFINE QLOCAL(QMG1) DESCR('Transmission queue to QMG1') REPLACE +
PUT(ENABLED) USAGE(XMITQ) GET(ENABLED)

DEFINE QLOCAL(QMG3) DESCR('Transmission queue to QMG3') REPLACE +
PUT(ENABLED) USAGE(XMITQ) GET(ENABLED)
```

b) Setup the sender channel definitions using TCP/IP:

```
DEFINE CHANNEL(QMG2.TO.QMG1) CHLTYPE(SDR) TRPTYPE(TCP) REPLACE +
DESCR('Sender channel to QMG1') XMITQ(QMG1) CONNAME('WINTQMG1(1414)')
```

Note: Replace WINTQMG1(1414) with your queue manager connection name and port.

```
DEFINE CHANNEL(QMG2.TO.QMG3) CHLTYPE(SDR) TRPTYPE(TCP) REPLACE +
DESCR('Sender channel to QMG3') XMITQ(QMG3) CONNAME('MVSQMG3(1416)')
```

Note: Replace MVSQMG3(1416) with your queue manager connection name and port.

c) Setup the receiver channel definitions using TCP/IP:

```
DEFINE CHANNEL(QMG1.TO.QMG2) CHLTYPE(RCVR) TRPTYPE(TCP) +
REPLACE DESCR('Receiver channel from QMG1')

DEFINE CHANNEL(QMG3.TO.QMG2) CHLTYPE(RCVR) TRPTYPE(TCP) +
REPLACE DESCR('Receiver channel from QMG3')
```

3. Procedure on QMG3:

a) Setup the local queue definition:

```
DEFINE QLOCAL(PAYROLL) DESCR('Payroll query request queue') REPLACE +
PUT(ENABLED) USAGE(NORMAL) GET(ENABLED) SHARE

DEFINE QLOCAL(QMG2) DESCR('Transmission queue to QMG2') REPLACE +
PUT(ENABLED) USAGE(XMITQ) GET(ENABLED)
```

b) Setup the sender channel definition using TCP/IP:

```
DEFINE CHANNEL(QMG3.TO.QMG2) CHLTYPE(SDR) TRPTYPE(TCP) REPLACE +
DESCR('Sender channel to QMG2') XMITQ(QMG2) CONNAME('MVSQMG2(1415)')
```

Note: Replace MVSQMG2(1415) with your queue manager connection name and port.

c) Setup the receiver channel definition using TCP/IP:

```
DEFINE CHANNEL(QMG2.TO.QMG3) CHLTYPE(RCVR) TRPTYPE(TCP) +
REPLACE DESCR('Receiver channel from QMG2')
```

4. Use the following procedure to run configuration 1:

a) Start queue managers QMG1, QMG2, and QMG3.

b) Start channel initiators for QMG2 and QMG3.

- c) Start the listeners on QMG1 to listen to port 1414, QMG2 to listen on port 1415, and QMG3 to listen on port 1416.
- d) Start sender channels on QMG1, QMG2, and QMG3.
- e) Start the payroll query requesting application connected to QMG1.
- f) Start the payroll server application connected to QMG3.
- g) Submit a payroll query request to QMG3 and wait for the payroll reply.

z/OS Setting up and running configuration 2

Configuration 2 describes how queue sharing groups and intra-group queuing can be used, with no effect on the back-end payroll server application, to transfer messages between queue managers QMG1 and QMG3.

About this task

Configuration 2 shows a distributed queuing system that uses queue sharing groups and intra-group queuing to transfer messages from the payroll request application to the payroll server. This configuration removes the need for channel definitions between queue managers QMG2 and QMG3 because intra-group queuing is used to transfer messages between these two queue managers.

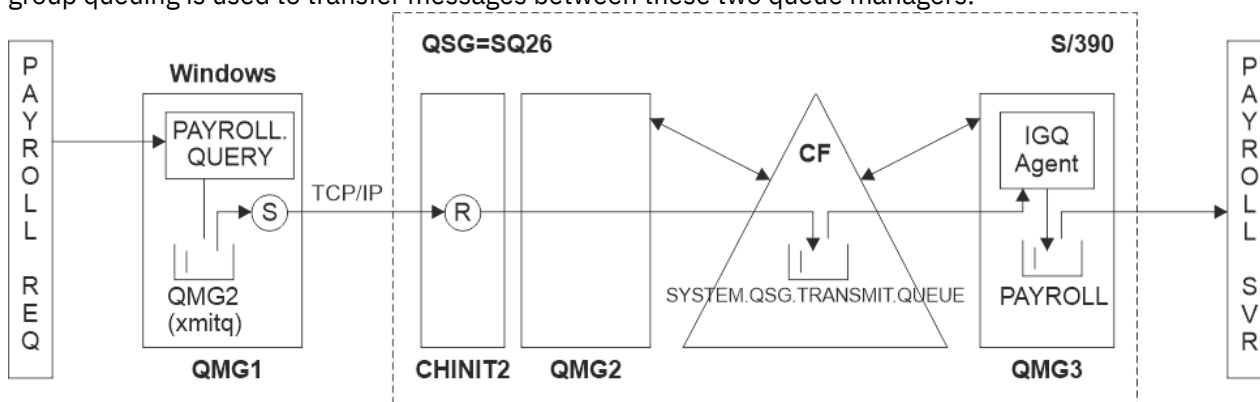


Figure 3. Configuration 2

The flow of operations is as follows:

1. A query is entered using the payroll request application connected to queue manager QMG1.
2. The payroll request application puts the query on to remote queue PAYROLL.QUERY. As queue PAYROLL.QUERY resolves to transmission queue QMG2, the query is put on to transmission queue QMG2.
3. Sender channel (S) on queue manager QMG1 delivers the query to the partner receiver channel (R) on queue manager QMG2.
4. Receiver channel (R) on queue manager QMG2 puts the query on to queue PAYROLL on queue manager QMG3. As queue PAYROLL on QMG3 resolves to shared transmission queue SYSTEM.QSG.TRANSMIT.QUEUE, the query is put on to shared transmission queue SYSTEM.QSG.TRANSMIT.QUEUE.
5. IGQ agent on queue manager QMG3 retrieves the query from shared transmission queue SYSTEM.QSG.TRANSMIT.QUEUE, and puts it on to local queue PAYROLL on queue manager QMG3.
6. The payroll server application connected to queue manager QMG3 retrieves the query from local queue PAYROLL, processes it, and generates a suitable reply.

Notes:

- The payroll query example transfers small messages only. If you need to transfer both persistent and non-persistent messages, you can establish a combination of Configuration 1 and Configuration 2, so

that large messages can be transferred using the distributed queuing route, while small messages can be transferred using the potentially faster intra-group queuing route.

- The definitions do not take into account triggering, and that only channel definitions for communication using TCP/IP are provided.
- The example assumes that you have already configured queue managers QMG2 and QMG3 to be members of the same queue sharing group.

Procedure

1. Procedure on QMG1:

a) Setup the remote queue definition:

```
DEFINE QREMOTE(PAYROLL.QUERY) DESCR('Remote queue for QMG3') REPLACE +  
PUT(ENABLED) RNAME(PAYROLL) RQMNAME(QMG3) XMITQ(QMG2)
```

b) Setup the transmission queue definition:

```
DEFINE QLOCAL(QMG2) DESCR('Transmission queue to QMG2') REPLACE +  
PUT(ENABLED) USAGE(XMITQ) GET(ENABLED)
```

c) Setup the sender channel definition for TCP/IP:

```
DEFINE CHANNEL(QMG1.TO.QMG2) CHLTYPE(SDR) TRPTYPE(TCP) REPLACE +  
DESCR('Sender channel to QMG2') XMITQ(QMG2) CONNAME('MVSQMG2(1415)')
```

Note: Replace MVSQMG2(1415) with your queue manager connection name and port.

d) Setup the receiver channel definition for TCP/IP:

```
DEFINE CHANNEL(QMG2.TO.QMG1) CHLTYPE(RCVR) TRPTYPE(TCP) +  
REPLACE DESCR('Receiver channel from QMG2')
```

e) Setup the reply-to queue definition:

```
DEFINE QLOCAL(PAYROLL.REPLY) REPLACE PUT(ENABLED) GET(ENABLED) +  
DESCR('Reply queue for replies to payroll queries sent to QMG3')
```

2. Procedure on QMG2:

a) Setup the transmission queue definition:

```
DEFINE QLOCAL(QMG1) DESCR('Transmission queue to QMG1') REPLACE +  
PUT(ENABLED) USAGE(XMITQ) GET(ENABLED)  
  
DEFINE QLOCAL(SYSTEM.QSG.TRANSMIT.QUEUE) QSGDISP(SHARED) +  
DESCR('IGQ Transmission queue') REPLACE PUT(ENABLED) USAGE(XMITQ) +  
GET(ENABLED) INDXTYPE(CORRELID) CFSTRUCT('APPLICATION1') +  
DEFSOPT(SHARED) DEFPSIST(NO)
```

Note: Replace APPLICATION1 with your defined CF structure name. Also, this queue being a shared queue, need only be defined on one of the queue managers in the queue sharing group.

b) Setup the sender channel definitions for TCP/IP:

```
DEFINE CHANNEL(QMG2.TO.QMG1) CHLTYPE(SDR) TRPTYPE(TCP) REPLACE +  
DESCR('Sender channel to QMG1') XMITQ(QMG1) CONNAME('WINTQMG1(1414)')
```

Note: Replace WINTQMG1(1414) with your queue manager connection name and port.

c) Setup the receiver channel definition for TCP/IP:

```
DEFINE CHANNEL(QMG1.TO.QMG2) CHLTYPE(RCVR) TRPTYPE(TCP) +
REPLACE DESCR('Receiver channel from QMG1')
```

d) Setup the queue manager definition:

```
ALTER QMGR IGQ(ENABLED)
```

3. Procedure on QMG3:

a) Setup the local queue definition:

```
DEFINE QLOCAL(PAYROLL) DESCR('Payroll query request queue') REPLACE +
PUT(ENABLED) USAGE(NORMAL) GET(ENABLED) SHARE
```

b) Setup the queue manager definition:

```
ALTER QMGR IGQ(ENABLED)
```

4. Use the following procedure to run configuration 2:

- Start queue managers QMG1, QMG2, and QMG3.
- Start the channel initiator for QMG2.
- Start the listeners on QMG1 to listen on port 1414, and QMG2 to listen on port 1415.
- Start the sender channel on QMG1 and QMG2.
- Start the payroll query requesting application connected to QMG1.
- Start the payroll server application connected to QMG3.
- Submit a payroll query request to QMG3 and wait for the payroll reply.

z/OS Setting up and running configuration 3

Configuration 3 describes how queue sharing groups and shared queues can be used, with no effect on the back-end payroll server application, to transfer messages between queue managers QMG1 and QMG3.

About this task

Configuration 3 shows a distributed queuing system that uses queue sharing groups and shared queues to transfer messages between queue manager QMG1 and queue manager QMG3.

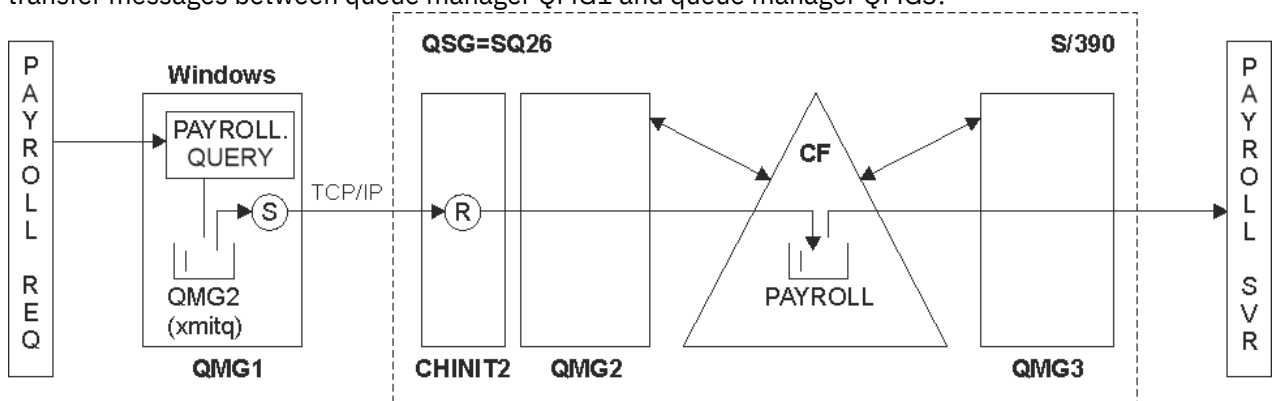


Figure 4. Configuration 3

The flow of operations is:

- A query is entered using the payroll request application connected to queue manager QMG1.

2. The payroll request application puts the query on to remote queue PAYROLL.QUERY. As queue PAYROLL.QUERY resolves to transmission queue QMG2, the query is put on to transmission queue QMG2.
3. Sender channel (S) on queue manager QMG1 delivers the query to the partner receiver channel (R) on queue manager QMG2.
4. Receiver channel (R) on queue manager QMG2 puts the query on to shared queue PAYROLL.
5. The payroll server application connected to queue manager QMG3 retrieves the query from shared queue PAYROLL, processes it, and generates a suitable reply.

This configuration is certainly the simplest to configure. However, you would need to configure distributed queuing or intra-group queuing to transfer replies (generated by the payroll server application connected to queue manager QMG3) from queue manager QMG3 to queue manager QMG2, and then on to queue manager QMG1.

For the configuration used to transfer replies back to the payroll request application, see [“Example: planning a message channel for z/OS using queue sharing groups”](#) on page 163.

Notes:

- Only channel definitions for communication using TCP/IP are provided.
- The example assumes that you have already configured queue managers QMG2 and QMG3 to be members of the same queue sharing group.
- No definitions are required on QMG3.

Procedure

1. Procedure on QMG1:

a) Setup the remote queue definition:

```
DEFINE QREMOTE(PAYROLL.QUERY) DESCR('Remote queue for QMG3') REPLACE +
PUT(ENABLED) RNAME(PAYROLL) RQMNAME(QMG3) XMITQ(QMG2)
```

b) Setup the transmission queue definition:

```
DEFINE QLOCAL(QMG2) DESCR('Transmission queue to QMG2') REPLACE +
PUT(ENABLED) USAGE(XMITQ) GET(ENABLED)
```

c) Setup the sender channel definition:

```
DEFINE CHANNEL(QMG1.TO.QMG2) CHLTYPE(SDR) TRPTYPE(TCP) +
REPLACE DESCR('Sender channel to QMG2') XMITQ(QMG2) CONNAME('MVSQMG2(1415)')
```

Note: Replace MVSQMG2(1415) with your queue manager connection name and port.

d) Setup the transmission channel definition:

```
DEFINE CHANNEL(QMG2.TO.QMG1) CHLTYPE(RCVR) TRPTYPE(TCP) +
REPLACE DESCR('Receiver channel from QMG2')
```

e) Setup the reply-to queue definition:

```
DEFINE QLOCAL(PAYROLL.REPLY) REPLACE PUT(ENABLED) GET(ENABLED) +
DESCR('Reply queue for replies to payroll queries sent to QMG3')
```

2. Procedure on QMG2:

a) Setup the transmission queue definition:

```
DEFINE QLOCAL(QMG1) DESCR('Transmission queue to QMG1') REPLACE +
PUT(ENABLED) USAGE(XMITQ) GET(ENABLED)
```

b) Setup the sender channel definitions for TCP/IP:

```
DEFINE CHANNEL(QMG2.TO.QMG1) CHLTYPE(SDR) TRPTYPE(TCP) +
REPLACE DESCR('Sender channel to QMG1') XMITQ(QMG1) CONNAME('WINTQMG1(1414)')
```

Note: Replace WINTQMG1(1414) with your queue manager connection name and port.

c) Setup the receiver channel definitions for TCP/IP:

```
DEFINE CHANNEL(QMG1.TO.QMG2) CHLTYPE(RCVR) TRPTYPE(TCP) +
REPLACE DESCR('Receiver channel from QMG1')
```

d) Setup the local queue definition:

```
DEFINE QLOCAL(PAYROLL) QSGDISP(SHARED) DESCR('Payroll query request queue') +
REPLACE PUT(ENABLED) USAGE(NORMAL) GET(ENABLED) SHARE +
DEFSOPT(SHARED) DEFPSIST(NO) CFSTRUCT(APPLICATION1)
```

Note: Replace APPLICATION1 with your defined CF structure name. Also this queue being a shared queue, need only be defined on one of the queue managers in the queue sharing group.

3. Use the following procedure to run configuration 3:

- a) Start queue managers QMG1, QMG2, and QMG3.
- b) Start the channel initiator for QMG2.
- c) Start the listeners on QMG1 to listen on port 1414, and QMG2 to listen on port 1415.
- d) Start sender channels on QMG1 and QMG2.
- e) Start the payroll query requesting application connected to QMG1.
- f) Start the payroll server application connected to QMG3.
- g) Submit a payroll query request to QMG3 and wait for the payroll reply.

Linux

IBM i

AIX

/var/mqm に適用される IBM MQ ファイル・

システム権限

以下の情報は、/var/mqm/ の下にあるファイルとディレクトリーに適用されるセキュリティーについて、およびファイル・システム権限がそのように設定される理由について説明します。IBM MQ の適切な運用を確保するには、IBM MQ によって設定されるファイル・システム権限を変更しないでください。

crtmqdir コマンド

何かの理由で /var/mqm のファイル権限を社内に変更した場合は、**crtmqdir** コマンドを使用してその権限を更新したりディレクトリーを追加したりできます。

AIX、Linux、および IBM i での IBM MQ ファイル・システム・セキュリティー

IBM MQ データ・ディレクトリー (/var/mqm) の下にあるファイルは、以下のデータの保管に使用されま

- IBM MQ 構成データ
- アプリケーション・データ (IBM MQ オブジェクトおよび IBM MQ メッセージ内に含まれるデータ)
- ランタイム制御情報
- モニタリング情報 (メッセージと FFST ファイル)

このデータへのアクセスは、ファイル・システム許可を使用して制御されます。一部のデータはすべてのユーザーがアクセスできますが、その他のデータは IBM MQ 管理者グループ 'mqm' (または IBM i 上の QMQM) のメンバーのみに制限されます。

アクセス権限は以下の 3 つのカテゴリーで付与されます。

mqm グループのみ

このカテゴリーのファイルとディレクトリーにアクセスできるのは、IBM MQ 管理者 (「mqm」グループのメンバー) と IBM MQ キュー・マネージャー・プロセスのみです。

これらのファイルとディレクトリーに対するファイル権限は次のとおりです。

```
-rwxrwx---   mqm:mqm      (UNIX and Linux)
-rwxrwx---   QMQMADM:QMQM (IBM i)
```

このカテゴリーのファイルおよびディレクトリーには、例えば以下のものがあります。

```
/var/mqm/qmgrs/QMGR/qm.ini
/var/mqm/qmgrs/QMGR/channel/
/var/mqm/qmgrs/QMGR/channel/SYSTEM!DEF!SCRVONN
/var/mqm/qmgrs/QMGR/queues/
/var/mqm/qmgrs/QMGR/queues/SYSTEM!DEFAULT!LOCAL!QUEUES/
/var/mqm/qmgrs/QMGR/errors/
/var/mqm/qmgrs/QMGR/errors/AMQERR01.LOG
/var/mqm/qmgrs/QMGR/ssl/
/var/mqm/qmgrs/QMGR/@qmgr/
/var/mqm/qmgrs/QMGR/@qmpersist/
...
```

全ユーザーに読み取り権限、mqm グループ・メンバーに読み取り/書き込み権限

このカテゴリーのファイルおよびディレクトリーはすべてのユーザーが読み取ることができますが、これらのファイルを変更してこれらのディレクトリーを操作できるのは、'mqm' グループのメンバーのみです。

これらのファイルとディレクトリーに対するファイル権限は次のとおりです。

```
-rwxrwxr-x   mqm:mqm      (UNIX and Linux)
-rwxrwxr-x   QMQMADM:QMQM (IBM i)
```

このカテゴリーのファイルおよびディレクトリーには、例えば以下のものがあります。

```
/var/mqm/mqs.ini
/var/mqm/exits/
/var/mqm/qmgrs/
/var/mqm/qmgrs/QMGR/
/var/mqm/qmgrs/QMGR/@app/
/var/mqm/qmgrs/QMGR/@ipcc/
```



重要: 実行可能ファイルとスクリプトにのみ実行権限を設定する必要があります。例えば、Linux では、**crtmqm** コマンドの実行時に以下のファイル権限が設定されます。

```
-rw-rw----   mqm mqm /var/mqm/qmgrs/QMGR/qm.ini
-rw-rw----   mqm mqm /var/mqm/qmgrs/QMGR/channel/SYSTEM!DEF!SCRVONN
-rw-rw----   mqm mqm /var/mqm/qmgrs/QMGR/errors/AMQERR01.LOG
-rw-rw-r--   mqm mqm /var/mqm/mqs.ini
```

IBM MQ 8.0:

```
/var/mqm/sockets/@SYSTEM
/var/mqm/sockets/QMGR/@app/hostname
/var/mqm/sockets/QMGR/@ipcc/hostname
```

全ユーザーに読み取り/書き込み権限

すべてのユーザーに読み取り/書き込み権限が与えられるファイル

IBM MQ には、全ユーザー書き込み可能ファイル許可 (777) を持つ標準ファイルがありません。ただし、全ユーザー書き込み可能ファイル許可を持つように見える特殊ファイルがいくつかあります。

これらの特殊ファイルはセキュリティ上の危険にさらされているわけではありません。アクセス権は 777 として表示されますが、標準ファイルではないため、直接書き込むことはできません。

このような特殊ファイルには、以下のものがあります。

シンボリック・リンク

シンボリック・リンクは、アクセス権の先頭にある「l」文字によって識別されます。シンボリック・リンクに対する権限は、どのユーザーがターゲット・ファイルにアクセスできるかに影響を与えません。これは、シンボリック・リンクのターゲットに対する権限によって、コマンドへのアクセスが制御されるためです。

ほとんどの AIX and Linux システムでは、シンボリック・リンクの許可を変更することはできないため、常に `lrwxrwxrwx` として表示されます。

ソケット・ファイル

ソケット・ファイルは、UNIX ドメイン・ソケットの作成プロセスの結果としてオペレーティング・システムによって作成される特殊ファイルです。これらのファイルは、ファイル・アクセス権の先頭にある「s」(`srwxrwxrwx`)によって識別できます。

ファイルに対する権限は、ファイル自体へのアクセス権限を与えるわけではなく、どのユーザーが UNIX ドメイン・ソケットに接続できるユーザーを定義するものです。

IBM MQ ではこのようなソケット・ファイルを多数使用し、権限は常に、どのユーザーがソケットとの通信を許可されるかに応じて設定されます。

以下のディレクトリーには、すべてのユーザーに対して読み取り/書き込み権限を持つソケット・ファイルが含まれています (`srwxrwxrwx`)。

IBM MQ 8.0:

```
/var/mqm/sockets/QMGR/zsocketEC/hostname/Zsocket_*
```

分離バインディングを使って IBM MQ に接続するアプリケーションで使用されるソケット・ファイル。

```
/var/mqm/sockets/QMGR/@ipcc/ssem/hostname/*
```

すべてのユーザーに読み取り/書き込み権限が与えられるディレクトリー

IBM MQ アプリケーションが IBM MQ データ・ディレクトリーの下にファイルを作成する必要があることがあります。必要な場合にアプリケーションがファイルを作成できるようにするために、いくつかのディレクトリーには全員書き込みアクセス権限が付与されています。つまり、システム上の任意のユーザーがそのディレクトリー内にファイルを作成できます。

'mqm' グループの任意のメンバーが書き込むことができるエラー・ログ・ファイルを除き、これらのディレクトリー内に作成されるすべてのファイルは、ファイル作成者の書き込み権限のみを許可する制限付き許可を使用して作成されます。これにより、システム管理者はこれらのディレクトリー内のファイルに書き込まれるすべてのデータのユーザー ID を追跡できます。

`/var/mqm/errors/`

このディレクトリーにはシステム・エラー・ログ・ファイルと FFST ファイルが含まれます。このディレクトリーの許可は '`drwxrwsrwt`' です。これは、システム上のすべてのユーザーがこのディレクトリーにファイルを作成できることを意味します。

SetGroupId ビット 's' は、このディレクトリーに作成されたすべてのファイルのグループ所有権が 'mqm'であることを示します。

このディレクトリーでは 't' スティッキー・ビットはデフォルトでは設定されませんが、IBM MQ 管理者はこれを明示的に設定して、作成したファイルのみをユーザーが削除できるようにすることができます。

注: **IBM i** この機能は IBM i では使用できません。

AMQERR0*.LOG

これらのエラー・ログ・ファイルは、group のメンバーのみが直接書き込むことができますが、すべてのユーザーがこれらのファイルに書き込まれたメッセージを読み取ることができます (許可: -rw-rw-r--)。

AMQnnnnn*.FDC

これらのファイルには、キュー・マネージャー、またはユーザー作成のアプリケーションでエラーが発生したときに書き込まれる FFST 情報が含まれます。これらのファイルは、許可 -rw-r----- を使用して作成されます。

/var/mqm/trace/

IBM MQ トレースが有効になっている場合、トレース・ファイルがこのディレクトリーに書き込まれます。IBM MQ トレースは、トレースが有効になっているキュー・マネージャーに関連付けられたすべてのプロセスによって書き込まれます。

このディレクトリーの許可は 'drwxrwsrwt' です。これは、システム上のすべてのユーザーがこのディレクトリーにファイルを作成できることを意味します。

SetGroupId ビット 's' は、このディレクトリーに作成されたすべてのファイルのグループ所有権が 'mqm' であることを示します。

このディレクトリーでは 't' スティッキー・ビットはデフォルトでは設定されませんが、IBM MQ 管理者はこれを明示的に設定して、作成したファイルのみをユーザーが削除できるようにすることができます。

注: **IBM i** この機能は IBM i では使用できません。

AMQnnnnn*.TRC

これらのファイルには、トレース中の各プロセスによって書き込まれ、アクセス権 -rw-r----- を指定して作成されたトレース・データが含まれています。

このディレクトリーの許可は drwxrwsrwt で、このディレクトリーに作成されるソケット・ファイルの許可は srwx----- です。

IBM MQ 8.0:

```
/var/mqm/sockets/QMGR/zsocketapp/hostname/
```

このディレクトリーは、分離されたバインディングを使用して IBM MQ キュー・マネージャーに接続するアプリケーションによって使用されます。接続処理中に、接続元のアプリケーションによってこのディレクトリー内にソケット・ファイルが作成されます。キュー・マネージャーへの接続が行われた後、ソケット・ファイルは削除されます。

このディレクトリーの許可は drwxrwsrwt で、このディレクトリーに作成されるソケット・ファイルの許可は srwx----- です。

このディレクトリーの SetGroupId ビット 's' は、このディレクトリー内で作成されるすべてのファイルに 'mqm' グループ所有権が設定されることを示します。

IBM i を除くすべてのプラットフォームで、このディレクトリーには 't' スティッキー・ビットも設定されるため、ユーザーは自分が所有者ではないファイルを削除できません。これにより、無許可ユーザーが自分の所有していないファイルを削除することを防止できます。

```
/var/mqm/sockets/QMGR/@ipcc/ssem/hostname/  
/var/mqm/sockets/QMGR/@app/ssem/hostname/
```

AIX 共有 バインディングを使用して IBM MQ に接続するプロセスの場合、UNIX ドメイン・ソケットを使用して、アプリケーションとキュー・マネージャーの間の同期化を行うことがで

きます。UNIX ドメイン・ソケットが使用される場合、それに関連するソケット・ファイルがこれらのディレクトリー内に作成されます。

これらのディレクトリーに対する権限は `drwxrwsrwt` であり、これらのディレクトリーに作成されるソケット・ファイルの権限は `srwxrwxrwx` です。

これらのディレクトリーに関する `SetGroupId` ビット '`s`' は、これらのディレクトリー内に作成されるすべてのファイルに '`mqm`' になりグループ所有権が必ず設定されるようにします。

IBM i を除くすべてのプラットフォームで、これらのディレクトリーには '`z`' スティッキー・ビットも設定されるため、ユーザーは自分が所有者ではないファイルを削除できません。これにより、無許可ユーザーが自分の所有していないファイルを削除することを防止できます。

HOME

`${HOME}/.mqm` ディレクトリーは、未登録または未インストールのバージョンの IBM MQ、例えば再配布可能クライアントを使用する際に作成されます。

このディレクトリーは、`sun_path` の長さ内に収まるパスを使用して、IBM MQ がソケット・ファイルにアクセスする信頼性の高い方法を使用するよう作成されます。IBM MQ が HOME ディレクトリーに書き込むことができない場合、エラー・メッセージが表示されます。

IBM MQ による System V IPC リソースの使用

IBM MQ は System V 共有メモリーおよびセマフォをプロセス間通信 (IPC) のために使用します。これらのリソースは、適切な所有権とアクセス権限を持つそれぞれのグループによってどのように使用されるかに応じてグループ化されます。

システム上のどの System V IPC リソースが IBM MQ に属するかを確認するには、次の方法に従うことができます。

- 所有権を確認します。

IBM MQ System V IPC リソースを所有するユーザーは、AIX and Linux プラットフォームでは常に '`mqm`' ユーザーです。IBM i では所有者ユーザーは '`QMQM`' です。

- IBM MQ 8.0 以降では、`amqspdbg` ユーティリティーを使用します。

IBM MQ に付属の `amqspdbg` ユーティリティーを使用して、特定のキュー・マネージャーの共有メモリーおよびセマフォ ID を表示できます。

IBM MQ によって作成される System V リソースの '`system`' グループに関してコマンドを一度発行する必要があります。

```
# amqspbg -z -I
```

さらに、システム上の各キュー・マネージャーに関して 4 度発行することにより、IBM MQ で使用される System V リソースの完全なリストを得ることができます。以下の例では、キュー・マネージャー名を `QMGR1` と想定しています。

```
# amqspdbg -i QMGR1 -I
# amqspdbg -q QMGR1 -I
# amqspdbg -p QMGR1 -I
# amqspdbg -a QMGR1 -I
```

IBM MQ によって作成される System V リソースに対するアクセス権限は、許可されたユーザーに適切なアクセス・レベルのみを付与するよう設定されます。IBM MQ によって作成される多くの System V IPC リソースは、マシン上のすべてのユーザーからアクセス可能であり、`-rw-rw-rw-` という権限が設定されます。

`crtmqm` コマンドで `-g ApplicationGroup` パラメーターを使用すると、キュー・マネージャーへのアクセス権限を、特定のオペレーティング・システム・グループのメンバーのみに限定できます。このグループ限定機能を使用することで、System V IPC リソースに関して付与される権限をさらに制限できます。

イル権限

ここでは、社内のセキュリティー・チームが、ディレクトリー・ツリー \$MQ_INSTALLATION_PATH 内の実行可能 IBM MQ ファイルがローカル・セキュリティー・ポリシーに違反していることを問題視しているという状況に対応する説明を行います。AIX ではデフォルトの場所は /usr/mqm ですが、その他の UNIX オペレーティング・システムでは /opt/mqm です。IBM MQ をデフォルト以外のディレクトリー (/opt/mqm90 など) にインストールした場合や、複数のインストール済み環境がある場合にも、このトピックの詳細情報が当てはまります。

問題の原因

セキュリティー・チームが \$MQ_INSTALLATION_PATH 下の次の領域の問題点を指摘しました。

1. /opt/mqm/bin ディレクトリー内のファイルは、それらが存在するディレクトリー・ツリーの所有者の setuid です。以下に例を示します。

```
dr-xr-xr-x  mqm  mqm  ${MQ_INSTALLATION_PATH}/bin
-r-sr-s---  mqm  mqm  ${MQ_INSTALLATION_PATH}/bin/addmqinf
-r-sr-s---  mqm  mqm  ${MQ_INSTALLATION_PATH}/bin/amqcrsta
-r-sr-s---  mqm  mqm  ${MQ_INSTALLATION_PATH}/bin/amqfcxba
...
```

2. ほとんどすべてのディレクトリーとファイルを "mqm:mqm" が所有していますが、例外として、次のものは root が所有しています。

```
dr-xr-x---  root  mqm  ${MQ_INSTALLATION_PATH}/bin/security
-r-sr-x---  root  mqm  ${MQ_INSTALLATION_PATH}/bin/security/amqoamax
-r-sr-x---  root  mqm  ${MQ_INSTALLATION_PATH}/bin/security/amqoampx
```

このサブディレクトリーは、root が所有する必要があります。IBM MQ クライアントのユーザーがパスワードを指定した場合に、そのパスワードが有効かどうかを確認するために、IBM MQ キュー・マネージャーはパスワードをオペレーティング・システムに渡しますが、そのときにオペレーティング・システムと対話する実行可能ファイルがこのサブディレクトリーに存在するからです。

3. ユーザーは /opt/mqm/lib/iconv ディレクトリー内のファイルを所有していません (このディレクトリーは AIX には存在しません)。以下に例を示します。

```
dr-xr-xr-x  mqm  mqm  ${MQ_INSTALLATION_PATH}/lib/iconv
-r--r--r--  bin  bin  ${MQ_INSTALLATION_PATH}/lib/iconv/002501B5.tbl
-r--r--r--  bin  bin  ${MQ_INSTALLATION_PATH}/lib/iconv/002501F4.tbl
-r--r--r--  bin  bin  ${MQ_INSTALLATION_PATH}/lib/iconv/00250333.tbl
...
```

4. RPM ベース Linux システムのフィックスバック保守ディレクトリー。フィックスバックがインストールされると、既存のファイルは、以下の例に示すような構造でこのディレクトリーの下に保存されます。ただし、この例では、V.R は IBM MQ のバージョンとリリース番号を表しており、表示されるサブディレクトリーは、インストールされるフィックスバックによって異なります。

```
drwx-----  root  root  ${MQ_INSTALLATION_PATH}/maintenance
drwxr-xr-x  root  root  ${MQ_INSTALLATION_PATH}/maintenance/V.R.0.1
drwxr-xr-x  root  root  ${MQ_INSTALLATION_PATH}/maintenance/V.R.0.3
drwxr-xr-x  root  root  ${MQ_INSTALLATION_PATH}/maintenance/V.R.0.4
...
```

問題の解決

UNIX システムの setuid プログラムに関する懸念事項の 1 つは、LD* などの環境変数 (LD_LIBRARY_PATH、AIX の LIBPATH など) の操作によってシステム・セキュリティーが脅かされるという点でした。しかし、この点は問題にならなくなりました。各種 UNIX オペレーティング・システムが、setuid プログラムのロード時にこのような LD* 環境変数を無視するようになったからです。

1. IBM MQ プログラムの一部が mqm-setuid や mqm-setgid であるのはなぜですか。

IBM MQ では、ユーザー ID "mqm" と "mqm" グループに含まれている ID は、IBM MQ の管理ユーザーです。

IBM MQ のキュー・マネージャー・リソースは、このユーザーの認証を行うことで保護されています。キュー・マネージャー・プロセスは、そうしたキュー・マネージャー・リソースを使用して変更するので、リソースにアクセスするために "mqm" 権限を必要とします。したがって、IBM MQ のキュー・マネージャー・サポート・プロセスは、"mqm" の実効ユーザー ID で実行するように設計されています。

IBM MQ オブジェクトにアクセスする非管理ユーザーのために、IBM MQ にはオブジェクト権限マネージャー (OAM) 機能が用意されています。この機能によって、非管理ユーザーで実行するアプリケーションの要件に合わせて権限の付与/取り消しを実行できます。

ユーザーにさまざまなレベルの認証を付与できるうえに、**setuid** プログラムと **setgid** プログラムは LD* 変数を見捨てるので、IBM MQ のバイナリー・ファイルやライブラリー・ファイルがシステムのセキュリティを脅かすことは一切ありません。

2. IBM MQ の機能を低下させずに、社内のセキュリティ・ポリシーに合わせて権限を変更することはできません。

IBM MQ バイナリーおよびライブラリーのアクセス権と所有権を変更してはなりません。そのような変更を行うと、IBM MQ の機能に問題が発生するおそれがあります。例えば、キュー・マネージャー・プロセスが一部のリソースにアクセスできなくなる可能性があります。

権限および所有権がシステムに対するセキュリティ上の脅威を引き起こすことはありません。

IBM MQ がインストールされている、または IBM MQ のデータが配置されている Linux ハード・ドライブ/ディスクは、nosuid オプションを使用してマウントしないでください。この構成では IBM MQ の機能が阻害される可能性があります。

詳しくは、64 ページの『[/var/mqm に適用される IBM MQ ファイル・システム権限](#)』を参照してください。

関連概念

[ファイル・システム](#)

Windows Windows での IBM MQ ファイル・システム権限

ここでは、Windows でファイルやディレクトリーに適用されるセキュリティについて説明します。IBM MQ の適切な運用を確保するには、IBM MQ によって設定されるファイル・システム権限を変更しないでください。

データ・ディレクトリー

注：このディレクトリーのルートに設定した権限が、ディレクトリー構造全体にわたって下位方向に継承されます。

データ・ディレクトリー (DATADIR) の下のディレクトリーには、一部の例外を除き、以下の権限が設定されます。例外については、この後の文章で詳しく説明します。

管理者

フル・コントロール

mqm グループ

フル・コントロール

SYSTEM

フル・コントロール

全員

読み取りと実行

ただし、次の例外があります。

DATADIR¥errors

全員のフル・コントロール

DATADIR\trace
全員のフル・コントロール

DATADIR\log
管理者
フル・コントロール
mqm グループ
フル・コントロール

SYSTEM
フル・コントロール
全員
読み取り

DATADIR\log\<qmgrname>\active

管理者
フル・コントロール
mqm グループ
フル・コントロール
SYSTEM
フル・コントロール

全員に与えられるアクセス権はありません。

AMQERR01.LOG などのエラー・ログ・ファイルは、そのディレクトリーからセキュリティ設定を継承せず、「全員：フル・コントロール (Everyone: Full Control)」に設定されます。

以前のリリースの製品

IBM MQ 8.0 より前のリリースの製品では、デフォルトのプログラム・ディレクトリーとデフォルトのデータ・ディレクトリーが同じ場所にありました。

元々 IBM MQ 8.0 より前にインストールされていたすべてのインストール済み環境。デフォルトの場所にインストールされてアップグレードされたデータ・ディレクトリーとプログラム・ディレクトリーは、同じ場所 (C:\Program Files\IBM\WebSphere MQ) に残ります。

データ・ディレクトリーとプログラム・ディレクトリーが同じ場所にある場合、前述の情報は、データ・ディレクトリーに属するディレクトリーにのみ当てはまります。プログラム・ディレクトリーに属するディレクトリーには当てはまりません。

キューの命名上の制約

キュー名の長さには制限があります。いくつかのキュー名は、キュー・マネージャーによって定義されたキュー用に予約済みです。

名前の長さの制限

キューの名前には最大 48 文字を使用できます。

予約キュー名

"SYSTEM." で始まる名前は、キュー・マネージャーによって定義されるキュー用に予約されています。ご使用のインストール・システムに合わせてこのキュー定義を変更したい場合は、**ALTER** コマンドまたは **DEFINE REPLACE** コマンドで変更できます。IBM MQ では、次の名前が定義されています。

キュー名	説明
SYSTEM.ADMIN.ACTIVITY.QUEUE	アクティビティー報告書のためのキュー

表 41. 予約キュー名と説明 (続き)	
キュー名	説明
SYSTEM.ADMIN.CHANNEL.EVENT	チャンネル・イベントのためのキュー
SYSTEM.ADMIN.COMMAND.EVENT	コマンド・イベントのためのキュー
SYSTEM.ADMIN.COMMAND.QUEUE	PCF コマンド・メッセージの送り先となるキュー
SYSTEM.ADMIN.CONFIG.EVENT	構成イベントのためのキュー
SYSTEM.ADMIN.PERFM.EVENT	パフォーマンス・イベントのためのキュー
SYSTEM.ADMIN.PUBSUB.EVENT	システムのパブリッシュ/サブスクライブに関連したイベント・キュー
SYSTEM.ADMIN.QMGR.EVENT	キュー・マネージャー・イベントのためのキュー
SYSTEM.ADMIN.TRACE.ROUTE.QUEUE	トレース経路の応答メッセージのためのキュー
SYSTEM.AUTH.DATA.QUEUE	キュー・マネージャー用のアクセス制御リストを保持するキュー (z/OS を除く)
SYSTEM.CHANNEL.INITQ	チャンネルの開始キュー
SYSTEM.CHANNEL.SYNCQ	チャンネルの同期データを保持するキュー
SYSTEM.CHLAUTH.DATA.QUEUE	IBM MQ のチャンネル認証データ・キュー
SYSTEM.CICS.INITIATION.QUEUE	トリガーに使用するキュー (z/OS を除く)
SYSTEM.CLUSTER.COMMAND.QUEUE	キュー・マネージャー間でリポジトリの変更をやり取りするために使用されるキュー
SYSTEM.CLUSTER.HISTORY.QUEUE	キューは、サービス目的でクラスター状態情報の履歴を保管するために使用されます。
SYSTEM.CLUSTER.REPOSITORY.QUEUE	リポジトリについての情報を保持するために使用されるキュー
SYSTEM.CLUSTER.TRANSMIT.MODEL.QUEUE	各クラスター送信側チャンネルの個々の送信キューを作成するために使用されるキュー。
SYSTEM.CLUSTER.TRANSMIT.QUEUE	クラスター・サポートで管理されるすべての宛先用の伝送キュー
SYSTEM.COMMAND.INPUT	z/OS でコマンド・メッセージの送り先となるキュー
SYSTEM.COMMAND.REPLY.MODEL	コマンド応答用のモデル・キュー定義 (z/OS 用)
SYSTEM.DEAD.LETTER.QUEUE	送達不能キュー (z/OS を除く)
SYSTEM.DEFAULT.ALIAS.QUEUE	デフォルトの別名キュー定義
SYSTEM.DEFAULT.INITIATION.QUEUE	指定したプロセスを起動するためのキュー (z/OS を除く)
SYSTEM.DEFAULT.LOCAL.QUEUE	デフォルトのローカル・キュー定義
SYSTEM.DEFAULT.MODEL.QUEUE	デフォルトのモデル・キュー定義
SYSTEM.DEFAULT.REMOTE.QUEUE	デフォルトのリモート・キュー定義
SYSTEM.DURABLE.SUBSCRIBER.QUEUE	キュー・マネージャー内の永続サブスクリプションの持続コピーを保持するために使用するローカル・キュー
SYSTEM.HIERARCHY.STATE	パブリッシュ/サブスクライブ階層におけるキュー・マネージャー間の関係についての状態情報を保持するためのキュー

表 41. 予約キュー名と説明 (続き)	
キュー名	説明
SYSTEM.JMS.TEMPQ.MODEL	JMS 一時キューのモデル
SYSTEM.INTERNAL.REPLY.QUEUE	IBM MQ 内部応答キュー (z/OS を除く)
SYSTEM.INTER.QMGR.CONTROL	パブリッシュ/サブスクライブ階層で、プロキシー・サブスクリプションを作成するためにリモート・キュー・マネージャーから要求を受け取るのに使用するキュー
SYSTEM.INTER.QMGR.PUBS	パブリッシュ/サブスクライブ階層で、リモート・キュー・マネージャーからパブリケーションを受け取るために使用するキュー
SYSTEM.INTER.QMGR.FANREQ	パブリッシュ/サブスクライブ階層で、リモート・キュー・マネージャー上にプロキシー・サブスクリプションを作成するための要求を処理するのに使用するキュー
SYSTEM.MQEXPLORER.REPLY.MODEL	IBM MQ Explorer への応答用のモデル・キュー定義
SYSTEM.MQSC.REPLY.QUEUE	MQSC コマンド応答用のモデル・キュー定義 (z/OS を除く)
SYSTEM.QSG.CHANNEL.SYNCQ	共有チャネルの同期情報を含むメッセージの保管に使用される共有ローカル・キュー (z/OS のみ)
SYSTEM.QSG.TRANSMIT.QUEUE	同じキュー共有グループ内のキュー・マネージャー間でメッセージを送信する際に、グループ内のキューイング・エージェントで使用される共有ローカル・キュー (z/OS のみ)
SYSTEM.RETAINED.PUB.QUEUE	キュー・マネージャーに保存されたそれぞれのパブリケーションのコピーを保持するために使用するローカル・キュー
SYSTEM.SELECTION.EVALUATION.QUEUE	IBM MQ 内部選択評価キュー (z/OS を除く)
SYSTEM.SELECTION.VALIDATION.QUEUE	IBM MQ 内部選択検証キュー (z/OS を除く)

その他のオブジェクトの命名上の制約

オブジェクト名の長さには制限があります。いくつかのオブジェクト名は、キュー・マネージャーによって定義されたオブジェクト用に予約済みです。

名前の長さの制限

プロセス、名前リスト、クラスター、トピック、サービス、および認証情報オブジェクトには、最大 48 文字の名前を付けることができます。



チャネルの名前に使用できるのは最大 20 文字です。

ストレージ・クラスには 8 文字までの長さの名前を付けることができます。

CF 構造体には、12 文字までの長さの名前を付けることができます。

予約オブジェクト名

SYSTEM で始まる名前。はキュー・マネージャーによって定義されたオブジェクト用に予約済みです。ご使用のインストール・システムに合わせてこのオブジェクト定義を変更する場合は、**ALTER** コマンドまたは **DEFINE REPLACE** コマンドを使用できます。IBM MQ では、次の名前が定義されています。

表 42. 予約オブジェクト名と説明	
オブジェクト名	説明
SYSTEM.ADMIN.SVRCONN	キュー・マネージャーのリモート管理に使用されるサーバー接続チャンネル
SYSTEM.AUTO.RECEIVER	自動定義のためのデフォルトの受信側チャンネル (AIX, Linux, and Windows システムのみ)
SYSTEM.AUTO.SVRCONN	自動定義のためのデフォルトのサーバー接続チャンネル (Multiplatforms のみ)
SYSTEM.BASE.TOPIC	ASPARENT を解決する基本トピック。特定の管理トピック・オブジェクトに親管理トピック・オブジェクトがない場合、ASPARENT 属性があればそれがこのオブジェクトから継承されます。
SYSTEM.DEF.CLNTCONN	デフォルトのクライアント接続チャンネル定義
SYSTEM.DEF.CLUSRCVR	デフォルトのクラスター受信側チャンネル定義
SYSTEM.DEF.CLUSSDR	デフォルトのクラスター送信側チャンネル定義
SYSTEM.DEF.RECEIVER	デフォルトの受信側チャンネル定義
SYSTEM.DEF.REQUESTER	デフォルトの要求側チャンネル定義
SYSTEM.DEF.SENDER	デフォルトの送信側チャンネル定義
SYSTEM.DEF.SERVER	デフォルトのサーバー・チャンネル定義
SYSTEM.DEF.SVRCONN	デフォルトのサーバー接続チャンネル定義
SYSTEM.DEFAULT.AUTHINFO.CRLLDAP	タイプ CRLLDAP の認証情報オブジェクトを定義するための、デフォルトの認証情報オブジェクト定義
SYSTEM.DEFAULT.AUTHINFO.OCSP	タイプ OCSP の認証情報オブジェクトを定義するための、デフォルトの認証情報オブジェクト定義
SYSTEM.DEFAULT.LISTENER.LU62	デフォルトの SNA リスナー (Windows のみ)
SYSTEM.DEFAULT.LISTENER.NETBIOS	デフォルトの NetBIOS リスナー (Windows のみ)
SYSTEM.DEFAULT.LISTENER.SPX	デフォルトの SPX リスナー (Windows のみ)
SYSTEM.DEFAULT.LISTENER.TCP	デフォルトの TCP/IP リスナー (Multiplatforms のみ)
SYSTEM.DEFAULT.NAMELIST	デフォルト名前リスト定義
SYSTEM.DEFAULT.PROCESS	デフォルトのプロセス定義
SYSTEM.DEFAULT.SERVICE	デフォルトのサービス (Multiplatforms のみ)
SYSTEM.DEFAULT.TOPIC	デフォルトのトピック定義
SYSTEM.QPUBSUB.QUEUE.NAMELIST	モニターするキューに入れられたパブリッシュ/サブスクライブ・インターフェースに対するキューのリスト
  SYSTEMST	デフォルトのストレージ・クラス定義 (z/OS のみ)

キュー名の解決

大規模なネットワークでは、キュー・マネージャーを使用することは、他の通信形式に比べていくつかの利点があります。それらの利点は、チャンネルの送信側と受信側の両方のキュー・マネージャーでキュー名の解決が実行されるといふ分散キュー管理の名前解決機能に起因するものです。

この方式の主な利点は次のとおりです。

- アプリケーションが経路判断を行う必要がない
- アプリケーションがネットワーク構造体を認識している必要がない
- ネットワーク・リンクはシステム管理者によって作成される
- ネットワーク構造はネットワーク計画者によって制御される
- トラフィックを分割するためにノード間で複数のチャンネルを使用できる

以下の図は、キュー名の解決の例を示しています。図には、ネットワーク内の2つのマシンが示されています。1つは書き込みアプリケーションを実行し、もう1つは読み取りアプリケーションを実行しています。これらのアプリケーションは、MCAによって制御されるIBM MQチャンネルを経由して相互に通信します。

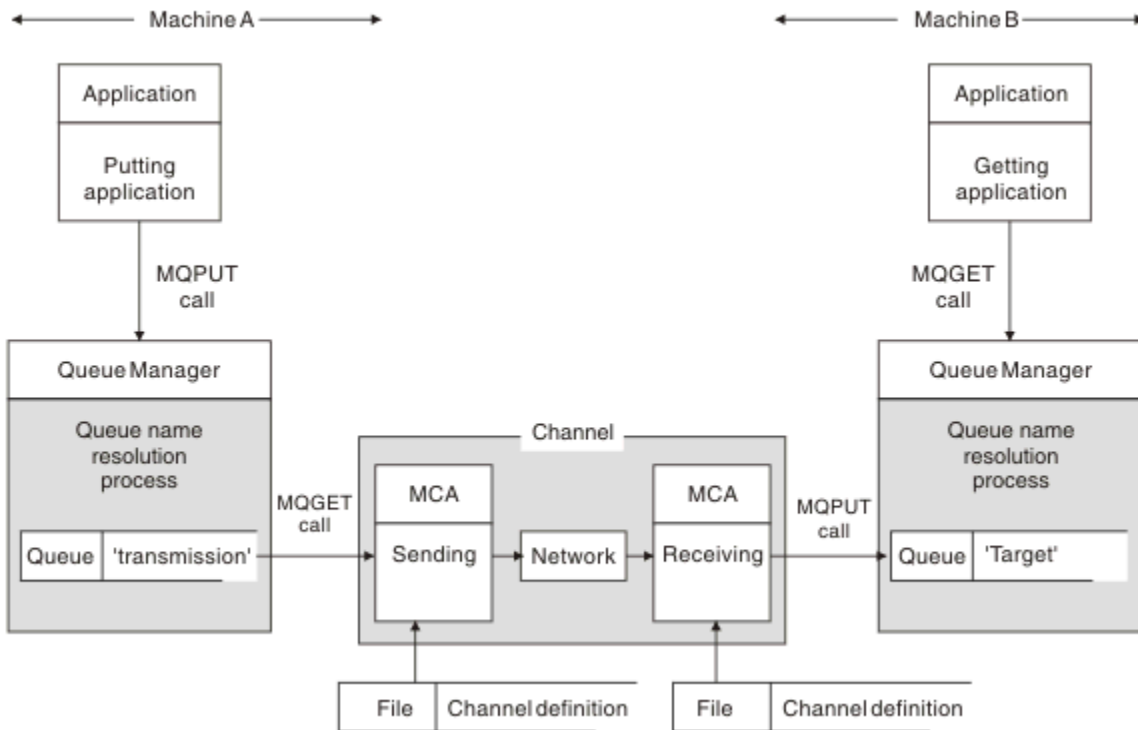


図 5. 名前の解決

75 ページの図 5 に示したように、リモート・キューにメッセージを入れるための基本メカニズムは、アプリケーションに関する限り、ローカル・キューにメッセージを入れる場合と同じです。

- メッセージを書き込むアプリケーションが MQOPEN および MQPUT 呼び出しを発行して、ターゲット・キューにメッセージを入れます。
- メッセージを読み取るアプリケーションが MQOPEN および MQGET 呼び出しを発行して、ターゲット・キューからメッセージを取り出します。

両方のアプリケーションが同じキュー・マネージャーに接続されている場合には、キュー・マネージャー間通信を行う必要はなく、ターゲット・キューが両方のアプリケーションに対してローカルなキューとして記述されます。

ただし、アプリケーションが異なるキュー・マネージャーに接続されている場合は、図に示したように、2つの MCA とそれらの関連ネットワーク接続が転送のために必要になります。この場合、ターゲット・キューは書き込みを行うアプリケーションにとってリモート・キューであると見なされます。

イベントの順序は次のとおりです。

1. 書き込みを行うアプリケーションが、ターゲット・キューにメッセージを入れるために MQOPEN および MQPUT 呼び出しを発行します。

2. MQOPEN 呼び出し中に、名前の解決機能は、ターゲット・キューがローカルではないことを検出し、適切な伝送キューを決定します。そのあとで、MQOPEN 呼び出しに関連する MQPUT 呼び出しで、すべてのメッセージがこの伝送キューに入れられます。
3. 送信側 MCA が伝送キューからメッセージを入手し、それをリモート側: 計算機の受信側 MCA に渡します。
4. 受信側 MCA がメッセージを1つまたは複数のターゲット・キューに書き込みます。
5. 読み取りを行うアプリケーションが、MQOPEN および MQGET 呼び出しを発行してターゲット・キューからメッセージを入手します。

注: アプリケーション・コードに関連するのは、ステップ1とステップ5だけであり、ステップ2から4まではローカル・キュー・マネージャーと MCA プログラムによって実行されます。書き込みを行うアプリケーションは、ターゲット・キューの位置を認識しません(ターゲット・キューは、同じプロセッサにある場合も、海外の別のプロセッサにある場合もあります)。

送信側 MCA、ネットワーク接続、および受信側 MCA の組み合わせはメッセージ・チャンネルと呼ばれ、単一方向の装置です。通常は、メッセージを両方向に移動させる必要があるため、各方向ごとに1つずつ、合計2つのチャンネルがこの移動のためにセットアップされます。

関連タスク

[リモート・キューへのメッセージの書き込み](#)

キュー名解決について

キュー名の解決は分散キュー管理の非常に重要な機能です。この機能のおかげで、アプリケーションはキューの物理的な場所を認識する必要がなくなるため、ネットワークの詳細をアプリケーションから切り離すことができます。

システム管理者は、キュー・マネージャーからキュー・マネージャーにキューを移動し、キュー・マネージャーをつなぐ経路を変更できますが、アプリケーションがこのことを認識する必要はありません。

データの正確な転送パスをアプリケーション設計から切り離すために、アプリケーションでターゲット・キューの参照に使用する名前と、転送に使用されるチャンネルの名前を間接的に対応付ける層が必要になります。これには、キュー名解決機能を使用します。

実際には、アプリケーションがキュー名を参照すると、解決機能によってこの名前が伝送キューまたはそれ以外のローカル・キューのいずれかにマップされます。伝送キューにマップする場合には、宛先で2回目の名前の解決を行う必要があり、受信されたメッセージはアプリケーション設計者が意図したターゲット・キューに入ります。アプリケーションは、メッセージの移動に使用された伝送キューとチャンネルについては関知しません。

注: キューとチャンネルの定義はシステム管理者によって行われ、オペレーターまたはシステム管理ユーティリティーは、アプリケーションを変更することなくこの定義を変更できます。

メッセージ・フローのシステム管理のための重要な条件として、キュー・マネージャー間で代替パスを用意する必要があります。例えば、業務の必要上から、異なるサービス・クラスを別のチャンネルを介して同一の宛先に送ることがあります。この決定はシステム管理者によって行われ、キュー名解決機構を使用して、柔軟性のある方法で実現できます。「アプリケーション・プログラミング・ガイド」に詳しい説明がありますが、基本となるのは、送信側キュー・マネージャーでキュー名解決を使用して、アプリケーションによって提供されたキュー名を、関連するトラフィックのタイプに適した伝送キューにマップするという考え方です。同様に受信側でも、キュー名解決によって、メッセージ記述子内の名前が(伝送キュー以外の)ローカル・キュー、または再度適切な伝送キューにマップされます。

あるキュー・マネージャーから別のキュー・マネージャーへのトラフィック・パスをさまざまなタイプのトラフィックに分割できるだけでなく、アウトバウンド・メッセージの応答先キュー定義に送られる戻りメッセージにも、同じトラフィック分割を使用できます。キュー名解決がこの条件を満たすため、アプリケーション設計者はこういったトラフィック分割の判断に関与する必要がありません。

送信側と受信側の両方のキュー・マネージャーでマッピングが行われることは、名前の解決における重要な働きの1つです。このマッピングにより、書き込みを行うアプリケーションが提供したキュー名が送信側キュー・マネージャーのローカル・キューまたは伝送キューにマップされ、また受信側キュー・マネージャーのローカル・キューまたは伝送キューにも再びマップされるようになります。

受信側のアプリケーションまたは MCA からの応答メッセージでも、同じ方法で名前の解決が行われ、経路上のすべてのキュー・マネージャーのキュー定義を使用して、特定のパスを介した戻り経路が確立されま
す。

別名キュー、リモート・キュー、クラスター・キューの宛先オブジェクト属性の解決方法

アプリケーション API 呼び出しのために名前解決が実行されると、オブジェクトの使用に影響を及ぼす属性が、最初に指定されたオブジェクト、"path" (74 ページの『キュー名の解決』を参照)、解決されたターゲット・オブジェクトの組み合わせによって解決されます。キュー・マネージャー・クラスターでは、対象の"指定されたオブジェクト"は、クラスター・オブジェクト(キューまたはトピック)の定義になります。これは、キュー・マネージャー間で共有され、表示されるオブジェクト属性のサブセットです。例えば、**DISPLAY QCLUSTER** などです。

アプリケーションで開く指定されたオブジェクトに属性を定義できる場合は、その属性が優先されます。例えば、別名キューやリモート・キューの定義には、すべての DEF**** 属性(デフォルトの永続性、優先順位、非同期書き込み応答)を構成できます。そうした属性は、解決された宛先キューや伝送キューではなく、別名キューやリモート・キューをアプリケーションで開いたときに有効になります。

アプリケーションとターゲット・オブジェクトの間の対話を制限することを意図した属性は、通常、指定されたオブジェクト(リモート・キュー定義や別名)では定義できません。例えば、**MAXMSGL** と **MAXDEPTH** は、リモート・キュー定義や別名では設定できません。キュー・マネージャー・クラスターのメンバー間で受け渡すこともできません。そのため、こうした属性は、解決されたキュー(ローカル・キュー、該当する伝送キュー、SYSTEM.CLUSTER.TRANSMIT.QUEUE など)から取得されます。リモート・キュー・マネージャーに到着したときに、2 番目の制約がターゲット・キューへの配送に適用される可能性があり、その結果、メッセージが送達不能キューに書き込まれたり、チャンネルが強制停止させられる場合があります。

PUT と **GET** の有効化は、属性解決の特殊なケースになります。どちらの属性も、キューのパスに **DISABLED** が出現すると、全体として解決される属性が **DISABLED** になります。

システムおよびデフォルト・オブジェクト

crtmqm により作成されるシステム・オブジェクトおよびデフォルト・オブジェクトをリストします。

crtmqm 制御コマンドを使用してキュー・マネージャーを作成する場合、システム・オブジェクトおよびデフォルト・オブジェクトが自動的に作成されます。

- システム・オブジェクトは、キュー・マネージャーまたはチャンネルの操作に必要な IBM MQ オブジェクトです。
- デフォルト・オブジェクトは、オブジェクトのすべての属性を定義します。ローカル・キューなどのオブジェクトを作成する場合、明示的に指定していない属性はすべてデフォルト・オブジェクトから継承されます。

次の表は、**crtmqm** により作成されるシステムおよびデフォルトのオブジェクトを示しています。

注: 表には含まれていませんが、これらの他に 2 つのデフォルト・オブジェクトがあります。それは、キュー・マネージャー・オブジェクトとオブジェクト・カタログです。これらは、ログに記録され、リカバリ可能であるという意味でオブジェクトです。

- [システムおよびデフォルト・オブジェクト: キュー](#)
- [システムおよびデフォルト・オブジェクト: トピック](#)
- [システムおよびデフォルトのオブジェクト: サーバー・チャンネル](#)
- [システムおよびデフォルトのオブジェクト: クライアント・チャンネル](#)
- [システムおよびデフォルトのオブジェクト: 認証情報](#)
- [システムおよびデフォルトのオブジェクト: 公益/情報通信情報](#)
- [システムおよびデフォルト・オブジェクト: リスナー](#)
- [システムおよびデフォルト・オブジェクト: 名前リスト](#)
- [システムおよびデフォルト・オブジェクト: プロセス](#)

・システムおよびデフォルト・オブジェクト: サービス

表 43. システムおよびデフォルト・オブジェクト: キュー	
オブジェクト名	説明
SYSTEM.ADMIN.ACCOUNTING.QUEUE	アプリケーションがキュー・マネージャーから切断されたときに生成される、アカウントング・メッセージ・データのキュー。
SYSTEM.ADMIN.ACTIVITY.QUEUE	戻されたアクティビティー・レポート・メッセージを保持するキュー。
SYSTEM.ADMIN.CHANNEL.EVENT	チャンネルのためのイベント・キュー
SYSTEM.ADMIN.COMMAND.EVENT	コマンド・イベントのイベント・キュー
SYSTEM.ADMIN.COMMAND.QUEUE	管理者コマンド・キュー。リモート MQSC コマンドおよび PCF コマンドに使用されます。
SYSTEM.ADMIN.CONFIG.EVENT	構成イベントのイベント・キュー
SYSTEM.ADMIN.LOGGER.EVENT	ロガー・イベント (ジャーナル・レシーバー) メッセージのイベント・キュー。
SYSTEM.ADMIN.PERFM.EVENT	パフォーマンス・イベントのイベント・キュー
SYSTEM.ADMIN.PUBSUB.EVENT	システムのパブリッシュ/サブスクライブに関連したイベント・キュー
SYSTEM.ADMIN.QMGR.EVENT	キュー・マネージャー・イベントのイベント・キュー
SYSTEM.ADMIN.STATISTICS.QUEUE	MQI、キュー、およびチャンネルの統計のモニタリング・データを保持するキュー。
SYSTEM.ADMIN.TRACE.ACTIVITY.QUEUE	トレース・アクティビティーを表示するキュー。
SYSTEM.ADMIN.TRACE.ROUTE.QUEUE	戻されたトレース経路応答メッセージを保持するキュー。
SYSTEM.AMQP.COMMAND.QUEUE	AMQP の IBM MQ 管理コマンド・キュー
SYSTEM.AUTH.DATA.QUEUE	キュー・マネージャー用のアクセス制御リストを保持するキュー オブジェクト権限マネージャー (OAM) によって使用されます。
SYSTEM.BROKER.ADMIN.STREAM	キューに入れられたパブリッシュ/サブスクライブ・インターフェースの管理ストリーム
SYSTEM.BROKER.CONTROL.QUEUE	パブリッシュ/サブスクライブ・インターフェースの制御キュー。
SYSTEM.BROKER.DEFAULT.STREAM	キューに入れられたパブリッシュ/サブスクライブ・インターフェースによって使用されるデフォルトのストリーム。
SYSTEM.BROKER.INTER.BROKER.COMMUNICATIONS	ブローカー間の通信キュー。
SYSTEM.CHANNEL.INITQ	チャンネル開始キュー
SYSTEM.CHANNEL.SYNCQ	チャンネルの同期データを保持するキュー
SYSTEM.CHLAUTH.DATA.QUEUE	IBM MQ のチャンネル認証データ・キュー
SYSTEM.CICS.INITIATION.QUEUE	デフォルト CICS 開始キュー。

表 43. システムおよびデフォルト・オブジェクト: キュー (続き)

オブジェクト名	説明
SYSTEM.CLUSTER.COMMAND.QUEUE	リポジトリ・キュー・マネージャーにメッセージを転送するために使用されるキュー
SYSTEM.CLUSTER.HISTORY.QUEUE	サービス目的でクラスターの状態情報のヒストリーを格納するために使用されるキュー。
SYSTEM.CLUSTER.REPOSITORY.QUEUE	すべてのリポジトリ情報を保管するために使用されるキュー
SYSTEM.CLUSTER.TRANSMIT.MODEL.QUEUE	各クラスター送信側チャネルの個々の送信キューを作成するために使用されるキュー。
SYSTEM.CLUSTER.TRANSMIT.QUEUE	すべてのクラスターに対するすべてのメッセージのための伝送キュー
SYSTEM.DEAD.LETTER.QUEUE	送達不能 (未配布メッセージ) キュー
SYSTEM.DEFAULT.ALIAS.QUEUE	デフォルト別名キュー
SYSTEM.DEFAULT.INITIATION.QUEUE	デフォルト開始キュー
SYSTEM.DEFAULT.LOCAL.QUEUE	デフォルト・ローカル・キュー
SYSTEM.DEFAULT.MODEL.QUEUE	デフォルト・モデル・キュー
SYSTEM.DEFAULT.REMOTE.QUEUE	デフォルト・リモート・キュー
SYSTEM.DOTNET.XARECOVERY.QUEUE	IBM MQ .NET XA リカバリー・キュー
SYSTEM.DURABLE.MODEL.QUEUE	管理される永続サブスクリプションのモデルとして使用されるキュー。
SYSTEM.DURABLE.SUBSCRIBER.QUEUE	キュー・マネージャー内の永続サブスクリプションの永続コピーを保持するために使用されるキュー。
SYSTEM.HIERARCHY.STATE	IBM MQ 分散パブリッシュ/サブスクライブ階層関係の状態。
SYSTEM.INTER.QMGR.CONTROL	IBM MQ 分散パブリッシュ/サブスクライブの制御キュー。
SYSTEM.INTER.QMGR.FANREQ	IBM MQ 分散パブリッシュ/サブスクライブの内部プロキシー・サブスクリプション多分岐プロセスの入力キュー。
SYSTEM.INTER.QMGR.PUBS	IBM MQ 分散パブリッシュ/サブスクライブのパブリケーション
SYSTEM.INTERNAL.REPLY.QUEUE	
SYSTEM.INTERNAL.REQUEST.QUEUE	
SYSTEM.JMS.TEMPQ.MODEL	JMS 一時キューのモデル
SYSTEM.MQEXPLORER.REPLY.MODEL	IBM MQ Explorer 応答先キュー。これは、IBM MQ Explorer に対する応答の一時動的キューを作成するモデル・キューです。
SYSTEM.MQSC.REPLY.QUEUE	MQSC コマンド応答先キュー。これは、リモート MQSC コマンドに回答するための一時動的キューを作成するモデル・キューです。

表 43. システムおよびデフォルト・オブジェクト: キュー (続き)

オブジェクト名	説明
SYSTEM.NDURABLE.MODEL.QUEUE	管理対象非永続サブスクリプションのモデルとして使用されるキュー。
SYSTEM.PENDING.DATA.QUEUE	JMS における据え置きメッセージをサポートします。
SYSTEM.PROTECTION.ERROR.QUEUE	IBM MQ メッセージ保護エラー・キュー。
SYSTEM.PROTECTION.POLICY.QUEUE	IBM MQ メッセージ保護ポリシー・キュー。
SYSTEM.REST.REPLY.QUEUE	
SYSTEM.RETAINED.PUB.QUEUE	それぞれの保存パブリケーションのコピーをキュー・マネージャーに保持するために使用されるキュー。
SYSTEM.SELECTION.EVALUATION.QUEUE	
SYSTEM.SELECTION.VALIDATION.QUEUE	

表 44. システムおよびデフォルト・オブジェクト: トピック

オブジェクト名	説明
SYSTEM.ADMIN.TOPIC	管理トピック。
SYSTEM.BASE.TOPIC	ASPARENT を解決する基本トピック。特定のトピックに親管理トピック・オブジェクトがない場合、またはこれらの親オブジェクトに ASPARENT もある場合は、残りのどの ASPARENT 属性もこのオブジェクトから継承されます。
SYSTEM.BROKER.ADMIN.STREAM	キューに入れられたパブリッシュ/サブスクライブ・インターフェースによって使用される管理ストリーム。
SYSTEM.BROKER.DEFAULT.STREAM	キューに入れられたパブリッシュ/サブスクライブ・インターフェースによって使用されるデフォルトのストリーム。
SYSTEM.BROKER.DEFAULT.SUBPOINT	キューに入れられたパブリッシュ/サブスクライブ・インターフェースによって使用されるデフォルトのサブポイント。
SYSTEM.DEFAULT.TOPIC	デフォルトのトピック定義

表 45. システムおよびデフォルトのオブジェクト: サーバー・チャネル

オブジェクト名	説明
SYSTEM.AUTO.RECEIVER	動的受信側チャネル
SYSTEM.AUTO.SVRCONN	動的サーバー接続チャネル
SYSTEM.DEF.AMQP	デフォルト AMQP チャネル。オブジェクトは定義されていますが、AMQP サービスはサポートされていないことに注意してください。
SYSTEM.DEF.CLUSRCVR	クラスターのためのデフォルト受信側チャネル。クラスター内のキュー・マネージャー上に CLUSRCVR チャネルを作成するときに指定しなかった属性のデフォルト値を提供するために使用されます。

表 45. システムおよびデフォルトのオブジェクト: サーバー・チャンネル (続き)

オブジェクト名	説明
SYSTEM.DEF.CLUSSDR	クラスターのためのデフォルト送信側チャンネル。クラスター内のキュー・マネージャー上に CLUSSDR チャンネルを作成するときに指定しなかった属性のデフォルト値を提供するために使用されます。
SYSTEM.DEF.RECEIVER	デフォルト受信側チャンネル
SYSTEM.DEF.REQUESTER	デフォルト要求側チャンネル
SYSTEM.DEF.SENDER	デフォルト送信側チャンネル
SYSTEM.DEF.SERVER	デフォルト・サーバー・チャンネル
SYSTEM.DEF.SVRCONN	デフォルト・サーバー接続チャンネル
SYSTEM.DEFAULT.AUTHINFO.IDPWLDAP	
SYSTEM.DEFAULT.AUTHINFO.IDPWOS	

表 46. システムおよびデフォルトのオブジェクト: クライアント・チャンネル

オブジェクト名	説明
SYSTEM.DEF.CLNTCONN	デフォルト・クライアント接続チャンネル




表 47. システムおよびデフォルトのオブジェクト: 認証情報

オブジェクト名	説明
SYSTEM.DEFAULT.AUTHINFO.CRLLDAP	CRLLDAP タイプの認証情報オブジェクトを定義するためのデフォルト認証情報オブジェクト。
SYSTEM.DEFAULT.AUTHINFO.OCSP	OCSP タイプの認証情報オブジェクトを定義するためのデフォルト認証情報オブジェクト。

表 48. システムおよびデフォルトのオブジェクト: 公益/情報通信情報

オブジェクト名	説明
SYSTEM.DEFAULT.COMMINFO.MULTICAST	マルチキャストのためのデフォルトの通信情報オブジェクト。

表 49. システムおよびデフォルト・オブジェクト: リスナー

オブジェクト名	説明
SYSTEM.DEFAULT.LISTENER.TCP	TCP トランスポートのためのデフォルト・リスナー。
 SYSTEM.DEFAULT.LISTENER.LU62	デフォルトの LU62 リスナー。
 SYSTEM.DEFAULT.LISTENER.NETBIOS	デフォルトの NETBIOS リスナー。
 SYSTEM.DEFAULT.LISTENER.SPX	デフォルトの SPX リスナー。

オブジェクト名	説明
SYSTEM.DEFAULT.NAMELIST	デフォルト名前リスト定義
SYSTEM.QPUBSUB.QUEUE.NAMELIST	キューに入れられたパブリッシュ/サブスクライブ・インターフェースによってモニターされるキュー名のリスト。
SYSTEM.QPUBSUB.SUBPOINT.NAMELIST	トピック・オブジェクトをサブスクリプション・ポイントと一致させるために、キューに入れられたパブリッシュ/サブスクライブ・インターフェースによって使用されるトピック・オブジェクトのリスト。

オブジェクト名	説明
SYSTEM.DEFAULT.PROCESS	デフォルトのプロセス定義



オブジェクト名	説明
SYSTEM.AMQP.SERVICE	MQ Light API サービス。オブジェクトは定義されていますが、サービスはサポートされていないことに注意してください。
SYSTEM.DEFAULT.SERVICE	デフォルト・サービス

SYSTEM.BASE.TOPIC

ASPARENT を解決する基本トピック。特定のトピックに親管理トピック・オブジェクトがない場合、またはこれらの親オブジェクトに ASPARENT もある場合は、残りのどの ASPARENT 属性もこのオブジェクトから継承されます。

パラメーター	値
TOPICSTR	"
V9.4.0 CAPEXPY (CAPEXPY)	NOLIMIT
CLROUTE	DIRECT
CLUSTER	デフォルト値は空ストリングです。
COMMINFO	SYSTEM.DEFAULT.COMMINFO.MULTICAST
DEFPRESP	SYNC
DEFPRTY	0
DEFPSIST	NO
DESCR	「属性の解決のための基本トピック」
DURSUB	YES
MCAST	DISABLED
MDURMDL	SYSTEM.DURABLE.MODEL.QUEUE

表 53. SYSTEM.BASE.TOPIC のデフォルト値 (続き)

パラメーター	値
MNDURMDL	SYSTEM.NDURABLE.MODEL.QUEUE
NPMSGDLV	ALLAVAIL
PMSGDLV	ALLDUR
PROXYSUB	FIRSTUSE
PUB	ENABLED
PUBSCOPE	ALL
  QSGDISP (z/OS プラットフォームのみ)	QMGR
SUB	ENABLED
SUBSCOPE	ALL
USEDLQ	YES
WILDCARD	PASSTHRU

このオブジェクトが存在しない場合でも、IBM MQ は、トピック・ツリーのうち、さらに上位の親トピックによって解決されない ASPARENT 属性に対して、そのデフォルト値を使用します。

SYSTEM.BASE.TOPIC の PUB 属性または SUB 属性を DISABLED に設定すると、トピック・ツリーに含まれているトピックのパブリッシュまたはサブスクライブの操作をアプリケーションが実行できなくなります。ただし、以下の 2 つの例外があります。

1. トピック・ツリーに含まれているトピック・オブジェクトのうち、PUB または SUB が明示的に ENABLE に設定されているトピック・オブジェクト。アプリケーションは、これらのトピックとその子トピックのパブリッシュまたはサブスクライブの操作を実行できます。
2. SYSTEM.BASE.TOPIC の PUB 属性や SUB 属性を DISABLED に設定しても、SYSTEM.BROKER.ADMIN.STREAM のパブリケーションやサブスクリプションは無効になりません。


PUB パラメーターの特殊な処理も参照してください。



構成ファイル・スタanzas情報



ここではスタanzas内の情報を構成する際に役立つ情報と、mqs.ini、qm.ini、およびmqclient.iniファイルの内容の一覧を示します。

スタanzasの構成

以下のリンクは、企業内で 1 つ以上のシステムを構成する際に役立ちます。

- [mqs.ini ファイルのスタanzasと属性](#) は、以下の構成に役立ちます。
 - AllQueueManagers スタanzas
 - DefaultQueueManager スタanzas
 - ExitProperties スタanzas
 - LogDefaults スタanzas
 - qm.ini ファイルの Security スタanzas
- [qm.ini ファイルのスタanzasおよび属性](#) は、以下の構成に役立ちます。
 -  AccessMode スタanzas (Windows のみ)

- Service スタンザ - インストール可能なサービス用
- Log スタンザ
-   RestrictedMode スタンザ (AIX and Linux システムのみ)
- XAResourceManager スタンザ
- TCP、LU62、および NETBIOS スタンザ
- ExitPath スタンザ
- QMErrorLog スタンザ
- SSL スタンザ
- ExitPropertiesLocal スタンザ
- 構成サービスおよびコンポーネントは、以下の構成に役立ちます。
 - Service スタンザ
 - ServiceComponent スタンザ

また AIX, Linux, and Windows プラットフォームでさまざまなサービスにどのように使用されるかを示すリンクも含まれています。
- API 出口の構成は、以下の構成に役立ちます。
 - AllActivityTrace スタンザ
 - ApplicationTrace スタンザ
- アクティビティー・トレース動作の構成ファイルは、以下の構成に役立ちます。
 - ApiExitCommon スタンザ
 - ApiExitTemplate スタンザ
 - ApiExitLocal スタンザ
- IBM MQ MQI client 構成ファイル mqclient.ini は、以下の構成に役立ちます。
 - CHANNELS スタンザ
 - ClientExitPath スタンザ
 -  LU62、NETBIOS、および SPX スタンザ (Windows のみ)
 - MessageBuffer スタンザ
 - SSL スタンザ
 - TCP スタンザ
 -  Trace スタンザ (IBM MQ .NET および XMS .NET でのみ使用)
- 86 ページの『分散キューイング用の構成ファイル・スタンザ』は、以下の構成に役立ちます。
 - CHANNELS スタンザ
 - TCP スタンザ
 - LU62 スタンザ
 - netbios
 - ExitPath スタンザ
- キューに入れられたパブリッシュ/サブスクライブのメッセージ属性の設定は、以下の構成に役立ちます。
 - PersistentPublishRetry 属性
 - NonPersistentPublishRetry 属性
 - PublishBatchSize 属性
 - PublishRetryInterval 属性

これらは、Broker スタンザに含まれています。



重要: Broker スタンザが必要な場合は、それを作成する必要があります。

- 自動構成を使用すると、以下のスタンザを簡単に構成できます。
 - AutoConfig スタンザ
 - AutoCluster スタンザ
 - Variables スタンザ

構成ファイル

以下を参照してください。

- [mqs.ini](#) ファイル
- [qm.ini](#) ファイル
- [mqclient.ini](#) ファイル

これらには、各構成ファイルで可能なスタンザの一覧が示されています。

Linux

AIX

mqs.ini ファイル

[AIX and Linux システム用の IBM MQ 構成ファイルの例](#)に、`mqs.ini` ファイルの例を示しています。

`mqs.ini` ファイルには、以下のスタンザを含めることができます。

- [AllQueue](#) マネージャー
- [DefaultQueue](#) マネージャー
- [ExitProperties](#)
- [LogDefaults](#)

さらに、キュー・マネージャーごとに1つの [QueueManager](#) スタンザがあります。

qm.ini ファイル

[IBM MQ for AIX or Linux システム用のキュー・マネージャー構成ファイルの例](#)に、`qm.ini` ファイルの例を示しています。

`qm.ini` ファイルには、以下のスタンザを含めることができます。

- [ExitPath](#)
- [ログ](#)
- [QMErrorLog](#)
- [QueueManager](#)
- [セキュリティ](#)
- [ServiceComponent](#)

Multi

[InstallableServices](#) を構成するには、`Service` および `ServiceComponent` スタンザを使用します。

- 接続 ([DefaultBind](#) タイプ の場合)




重要: `Connection` スタンザが必要な場合は、それを作成する必要があります。

- [SSL](#) および [TLS](#)
- [TCP](#)、[LU62](#)、および [NETBIOS](#)
- [XAResourceManager](#)

さらに、以下のものを変更できます。

- **Windows** [AccessMode](#) (Windows のみ)

-  *RestrictedMode* (AIX and Linux システムのみ)

この操作には `crtmqm` コマンドを使用します。

mqclient.ini ファイル

mqclient.ini ファイルには、以下のスタンザを含めることができます。

- [チャンネル](#)
- [ClientExit](#) パス
- [LU62](#)、[NETBIOS](#)、および [SPX](#)
- [MessageBuffer](#)
- [SSL](#)
- [TCP](#)

さらに、[PreConnect](#) スタンザを使用して PreConnect 出口を構成する必要がある場合もあります。

分散キューイング用の構成ファイル・スタンザ

分散キューイングに関連した、キュー・マネージャー構成ファイル `qm.ini` のスタンザの説明。

このトピックでは、分散キューイングに関連するキュー・マネージャー構成ファイル内のスタンザを示しています。これは、IBM MQ for Multiplatforms 用のキュー・マネージャー構成ファイルに適用されます。すべてのプラットフォームで、このファイルは `qm.ini` と呼ばれます。

分散キューイングに関連するスタンザとして次のスタンザがあります。

- チャンネル
- TCP
- LU62
- NETBIOS
- EXITPATH

87 ページの [図 6](#) には、これらのスタンザを使用して設定できる値を示しています。どれか 1 つのスタンザを定義する場合、各項目ごとに改行する必要はありません。コメントを入れる場合は、セミコロン (;) またはハッシュ文字 (#) を使用できます。

```

CHANNELS:
MAXCHANNELS=n           ; Maximum number of channels allowed, the
                        ; default value is 100.
MAXACTIVECHANNELS=n    ; Maximum number of channels allowed to be active at
                        ; any time, the default is the value of MaxChannels.
MAXINITIATORS=n        ; Maximum number of initiators allowed, the default
                        ; and maximum value is 3.
MQIBINDTYPE=type       ; Whether the binding for applications is to be
                        ; "fastpath" or "standard".
                        ; The default is "standard".
PIPELINELENGTH=n       ; The maximum number of concurrent threads a channel will use.
                        ; The default is 1. Any value greater than 1 is treated as 2.
ADOPTNEWMCA=chltype    ; Stops previous process if channel fails to start.
                        ; The default is "NO".
ADOPTNEWMCATIMEOUT=n   ; Specifies the amount of time that the new
                        ; process should wait for the old process to end.
                        ; The default is 60.
ADOPTNEWMCACHECHECK=   ; Specifies the type checking required.
  typecheck             ; The default is "NAME", "ADDRESS", and "QM".
CHLAUTHEARLYADOPT=Y/N ; The order in which connection authentication and channel
authentication rules are ; processed. If not present in the qm.ini file the default is "N".

From MQ9.0.4 all
PASSWORDPROTECTION=    ; queue managers are created with a default of "Y"
than using TLS.        ; From MQ8.0, set protected passwords in the MQCSP structure, rather
options                ; The options are "compatible", "always", "optional" and "warn"
                        ; The default is "compatible".
IGNORESEQNUMBERMISMATCH ; How the queue manager handles a sequence number mismatch during
channel startup.
=Y/N                   ; The options are "Y" and "N" with the default being "N".
CHLAUTHIGNOREUSERCASE  ; Enables a queue manager to make username matching within CHLAUTH
rules case-insensitive.
=Y/N                   ; The options are "Y" and "N" with the default being "N".
CHLAUTHISSUEWARN=Y     ; If you want message AMQ9787 to be generated when you set theWARN=YES
attribute
                        ; on the SET CHLAUTH command.
TCP:                   ; TCP entries
PORT=n                 ; Port number, the default is 1414
KEEPALIVE=Yes          ; Switch TCP/IP KeepAlive on
LU62:
LIBRARY2=DLLName2     ; Used if code is in two libraries
EXITPATH:1 Location of user exits
EXITPATHS=             ; String of directory paths.

```

図 6. 分散キューイングの *qm.ini* スタンザ

注:

1. EXITPATH は、次のプラットフォームにのみ適用されます。

-  AIX
-  Windows

関連タスク

構成

 z/OS の構成

Multiplatforms での IBM MQ 構成情報の変更

チャネルの属性

このセクションでは、チャネル定義に保持されるチャネル属性について説明します。

チャネルごとに特定の一組の環境に最適のチャネル属性を選択してください。ただし、チャネルが実行されているときには、開始折衝中に実際の値が変更された可能性があります。[チャネルの作成](#)を参照してください。

多くの属性にはデフォルト値が設定されており、ほとんどのチャネルではこれらの値を使用できます。ただし、デフォルト値が最適な値ではない環境の場合、このセクションを参照して正しい値を選択するための説明をお読みください。

注：IBM MQ for IBM iでは、ほとんどの属性は*SYSDFTCHLとして指定できます。これは、使用するシステムのシステム・デフォルト・チャンネルから値が取られることを表します。

IBM MQ チャンネル属性のチャンネル・タイプを、MQSC コマンド・パラメーターの順序で以下の表にリストします。

注：クラスター・チャンネル(表の CLUSSDR 列と CLUSRCVR 列)では、可能であれば両方のチャンネルに属性を設定し、設定を確実に同じにします。これらの設定が一致していない場合、CLUSRCVR チャンネルで指定した設定の方が使用されます。これについては、[クラスター・チャンネル](#)で説明しています。


属性フィールド	MQSC コマンド・パラメーター	SDR	SVR	RCV R	RQST R	CLNT-CONN	SVR-CONN	CLUS-SDR	CLUS-RCVR	AMQP
Connection affinity (接続アフィニティー)	AFFINITY					はい				
Alter date (変更日付)	ALTDATE	はい	はい	はい	はい	はい	はい	はい	はい	はい
Alter time (変更時刻)	ALTTIME	はい	はい	はい	はい	はい	はい	はい	はい	はい
 AMQP キープアライブ	AMQPKA									はい
Batch heartbeat interval (バッチ・ハートビート間隔)	BATCHHB	はい	はい					はい	はい	
Batch interval (バッチ間隔)	BATCHINT	はい	はい					はい	はい	
バッチ制限	BATCHLIM	はい	はい					はい	はい	
バッチのサイズ	BATCHSZ	はい	はい	はい	はい			はい	はい	
証明書ラベル	CERTLABL	はい	はい	はい	はい	はい	はい	はい 92 ページの『1』	はい	はい
チャンネル名	CHANNEL	はい	はい	はい	はい	はい	はい	はい	はい	はい
チャンネル・タイプ	CHLTYPE	はい	はい	はい	はい	はい	はい	はい	はい	はい
クライアント・チャンネルの重み	CLNTWGH T					はい				
クラスター名前リスト	CLUSNL							はい	はい	
クラスター	CLUSTER							はい	はい	
Cluster workload priority (クラスター・ワークロードの優先順位)	CLWLPRT Y							はい	はい	

表 54. チャネル属性とチャネル・タイプ (続き)

属性フィールド	MQSC コマンド・パラメータ	SDR	SVR	RCV R	RQST R	CLNT-CONN	SVR-CONN	CLUS-SDR	CLUS-RCVR	AMQP
<u>Cluster workload rank</u> (クラスター・ワークロードのランク)	CLWLRANK							はい	はい	
<u>Cluster workload weight</u> (クラスター・ワークロード・ウェイト)	CLWLWGH T							はい	はい	
<u>ヘッダー圧縮</u>	COMPHDR	はい	はい	はい	はい	はい	はい	はい	はい	
<u>Data compression</u> (データ圧縮)	COMPMSG	はい	はい	はい	はい	はい	はい	はい	はい	
<u>接続名</u>	CONNNAME	はい	はい		はい	はい		はい	はい	
<u>メッセージ変換</u>	CONVERT	はい	はい					はい	はい	
<u>デフォルト再接続</u>	DEFRECON					はい				
<u>説明</u>	DESCR	はい	はい	はい	はい	はい	はい	はい	はい	はい
<u>切断インターバル</u>	DISCINT	はい	はい				z/OS はい Multi いいえ	はい	はい	
<u>ハートビート間隔</u>	HBINT	はい	はい	はい	はい	はい	はい	はい	はい	
<u>キープアライブ間隔</u>	KAINT	はい	はい	はい	はい	はい	はい	はい	はい	
<u>Local address</u> (ローカル・アドレス)	LOCLADDR	はい	はい		はい	はい		はい	はい	はい
<u>長期再試行カウント</u>	LONGRTY	はい	はい					はい	はい	
<u>長期再試行間隔</u>	LONGTMR	はい	はい					はい	はい	
<u>最大インスタンス数</u>	MAXINST						はい			はい
<u>Maximum instances per client</u> (クライアントごとの最大インスタンス数)	MAXINST C						はい			
<u>Maximum message length</u> (最大メッセージ長)	MAXMSG L	はい	はい	はい	はい	はい	はい	はい	はい	はい

表 54. チャネル属性とチャネル・タイプ (続き)

属性フィールド	MQSC コマンド・パラメーター	SDR	SVR	RCV R	RQST R	CLNT-CONN	SVR-CONN	CLUS-SDR	CLUS-RCVR	AMQP
<u>メッセージ・チャネル・エージェント名</u>	MCANAME	はい	はい		はい			はい	はい	
<u>Message channel agent type (メッセージ・チャネル・エージェント・タイプ)</u>	MCATYPE	はい	はい		はい			はい	はい	
<u>メッセージ・チャネル・エージェント・ユーザー</u>	MCAUSER	はい	はい	はい	はい		はい	はい	はい	はい
<u>LU 6.2 モード名</u>	MODENAME	はい	はい		はい	はい		はい	はい	
<u>モニター</u>	MONCHL	はい	はい	はい	はい		はい	はい	はい	
<u>Message-retry exit user data (メッセージ再試行出口ユーザー・データ)</u>	MRDATA			はい	はい				はい	
<u>メッセージ再試行出口名</u>	MREXIT			はい	はい				はい	
<u>メッセージ再試行カウント</u>	MRRTY			はい	はい				はい	
<u>メッセージ再試行間隔</u>	MRTMR			はい	はい				はい	
<u>メッセージ出口ユーザー・データ</u>	MSGDATA	はい	はい	はい	はい			はい	はい	
<u>メッセージ出口名</u>	MSGEXIT	はい	はい	はい	はい			はい	はい	
<u>Network-connection priority (ネットワーク接続優先順位)</u>	NETPRTY								はい	
<u>非持続性メッセージ速度</u>	NPMSPEED	はい	はい	はい	はい			はい	はい	
<u>PASSWORD</u>	PASSWORD	はい	はい		はい	はい		はい		
<u>ポート番号</u>	PORT									はい
<u>MQGMO の PROPCTL チャネル・オプション</u>	PROPCTL	はい	はい					はい	はい	

表 54. チャネル属性とチャネル・タイプ (続き)

属性フィールド	MQSC コマンド・パラメーター	SDR	SVR	RCV R	RQST R	CLNT-CONN	SVR-CONN	CLUS-SDR	CLUS-RCVR	AMQP
PUT authority (PUT 権限)	PUTAUT			はい	はい		z/OS はい Multi いいえ		はい	
キュー・マネージャー名	QMNAME					はい				
z/OS 後処理	QSGDISP	はい	はい	はい	はい	はい	はい	はい	はい	
受信出口ユーザー・データ	RCVDATA	はい	はい	はい	はい	はい	はい	はい	はい	
受信出口ユーザー名	RCVEXIT	はい	はい	はい	はい	はい	はい	はい	はい	
セキュリティー出口ユーザー・データ	SCYDATA	はい	はい	はい	はい	はい	はい	はい	はい	
セキュリティー出口名	SCYEXIT	はい	はい	はい	はい	はい	はい	はい	はい	
送信出口ユーザー・データ	SENDDATA	はい	はい	はい	はい	はい	はい	はい	はい	
送信出口名	SENDEXIT	はい	はい	はい	はい	はい	はい	はい	はい	
シーケンス番号の折り返し	SEQWRAP	はい	はい	はい	はい			はい	はい	
共用接続	SHARECNV					はい	はい			
短期再試行カウント	SHORTRTY	はい	はい					はい	はい	
短期再試行間隔	SHORTTMR	はい	はい					はい	はい	
z/OS セキュリティー・ポリシー保護	SPLPROT	はい	はい	はい	はい					
SSL クライアント認証	SSLCAUTH		はい	はい	はい		はい		はい	はい
SSL 暗号仕様	SSLCIPH	はい	はい	はい	はい	はい	はい	はい	はい	はい
SSL ピア	SSLPEER	はい	はい	はい	はい	はい	はい	はい	はい	はい
チャネル統計	STATCHL	はい	はい	はい	はい			はい	はい	

表 54. チャネル属性とチャネル・タイプ (続き)

属性フィールド	MQSC コマンド・パラメータ	SDR	SVR	RCV R	RQST R	CLNT-CONN	SVR-CONN	CLUS-SDR	CLUS-RCVR	AMQP
<u>LU 6.2 トランザクション・プログラム名</u>	TPNAME	はい	はい		はい	はい		はい	はい	
<u>トピック・ルート</u>	TPROOT									はい
<u>トランスポート・タイプ</u>	TRPTYPE	はい	はい	はい	はい	はい	はい	はい	はい	
<u>クライアント ID を使用</u>	USECLTID									はい
<u>送達不能キューの使用</u>	USEDLQ	はい	はい	はい	はい			はい	はい	
<u>ユーザー ID</u>	ユーザー ID	はい	はい		はい	はい		はい		
<u>伝送キュー名</u>	XMITQ	はい	はい							

注:

- この属性を照会したり CLUSSDR チャネルに対して設定したりできる管理インターフェースはありません。MQRCCF_WRONG_CHANNEL_TYPE メッセージを受け取ります。ただし、この属性は CLUSSDR チャネル・オブジェクト (MQCD 構造体を含む) に存在しており、必要な場合は CHAD 出口でプログラムマチックに設定できます。

一部のプラットフォームの IBM MQ では、このセクションに示されるすべての属性を使用できない場合があります。例外とプラットフォームによる違いについては、関連する個々の属性の記述内で説明しています。

各属性の名前を大括弧で囲んで示します。

属性は、グループ内でアルファベット順に配置されます。

関連資料

- MQSC コマンド
- ALTER CHANNEL
- DEFINE CHANNEL

MQSC キーワード (A-B) のチャネル属性

MQSC キーワードのチャネル属性のアルファベット順リスト。文字は A または B で始まります。

AFFINITY (接続アフィニティ)

この属性は、同じキュー・マネージャー名を使用して複数回接続するクライアント・アプリケーションが同じクライアント・チャネルを使用するかどうかを指定します。

この属性 (MQIACH_CONNECTION_AFFINITY) は、複数の適用可能なチャネル定義が選択できる場合に使用します。

指定できる値は以下のとおりです。

PREFERRED

クライアント・チャンネル定義テーブル (CCDT) を読み取るプロセスの最初の接続によって、該当する定義のリストが作成されます。そのリストは、クライアント・チャンネル・ウェイトの値に基づいています。ウェイトの値が 0 の定義が先頭に来て、各定義がアルファベット順で並びます。プロセス内の各接続は、リスト内の最初の定義を使用して接続を試行します。接続が失敗した場合は、次の定義が使用されます。失敗した定義のうち、クライアント・チャンネル・ウェイトの値が 0 以外の定義は、リストの末尾に移動します。クライアント・チャンネル・ウェイトの値が 0 の定義は、リストの先頭に残り、各接続で最初に選択されます。

同じホスト名を持つ各クライアント・プロセスは、常に同じリストを作成します。

C、C++、または .NET プログラミング・フレームワーク (完全管理の .NET を含む) で作成されたクライアント・アプリケーション、および IBM MQ classes for Java と IBM MQ classes for JMS を使用するアプリケーションについては、リストの作成以降に CCDT が変更された場合、リストは更新されます。

この値がデフォルトで、値は 1 です。

NONE

CCDT を読み取るプロセス内の最初の接続が、適用可能な定義のリストを作成します。プロセス内のすべての接続が、クライアント・チャンネル・ウェイトの値に基づいて該当する定義を選択します。アルファベット順で、ウェイトの値が 0 の定義を最初に選択します。

C、C++、または .NET プログラミング・フレームワーク (完全管理の .NET を含む) で作成されたクライアント・アプリケーション、および IBM MQ classes for Java と IBM MQ classes for JMS を使用するアプリケーションについては、リストの作成以降に CCDT が変更された場合、リストは更新されます。

この属性は、クライアント接続チャンネル・タイプでのみ有効です。

ALTDATE (日付変更)

この属性は、定義が最後に変更された日付を yyyy-mm-dd の形式で示し、すべてのチャンネル・タイプに有効です。

変更時刻 (ALTTIME)

この属性は、定義が最後に変更された時刻を hh.mm.ss の形式で示し、すべてのチャンネル・タイプに有効です。

AMQPKA (AMQP キープアライブ)

Multi

AMQPKA 属性を使用すると、AMQP クライアント接続のキープアライブ時間を指定できます。AMQP クライアントがキープアライブ間隔内にフレームをまったく送信しなかった場合、接続は閉じられます。

AMQPKA 属性は、IBM MQ から AMQP クライアントに送信されるアイドル・タイムアウト属性の値が決まります。この属性は、ミリ秒単位の期間です。

AMQPKA が > 0 の値に設定された場合、IBM MQ はアイドル・タイムアウト属性としてその半分の値を送ります。例えば、値が 10000 の場合、キュー・マネージャーはアイドル・タイムアウト値として 5000 を送信します。クライアントは、少なくとも 10000 ミリ秒ごとにデータが IBM MQ に送信されるようにする必要があります。その期間内に IBM MQ がデータを受け取らなかった場合、IBM MQ はクライアントとの接続が失われたと想定して、amqp:resource-limit-exceeded エラー条件により接続を強制的にクローズします。

値の AUTO または 0 は、IBM MQ がアイドル・タイムアウト属性を AMQP クライアントに送らないことを意味します。

その場合でも、AMQP クライアントは独自のアイドル・タイムアウト値を送ることが可能です。そのようにした場合、IBM MQ はデータ (または空の AMQP フレーム) を少なくともその頻度で送り、使用可能であることをクライアントに通知します。

BATCHHB (バッチ・ハートビート間隔)

この属性を使用すると、送信側チャンネルは、メッセージのバッチをコミットする直前に、受信側チャンネルがまだアクティブであることを確認できます。

したがって、バッチ・ハートビート間隔を使用すると、受信側チャンネルがアクティブではない場合に、バッチをバックアウトできるため、バッチが未確定とならずに済みます。バッチをバックアウトすることによって、メッセージは処理可能な状態にとどまるので、例えば、メッセージを別のチャンネルにリダイレクトできます。

バッチ・ハートビート間隔内に、送信側チャンネルに受信側チャンネルからの通信があった場合、受信側チャンネルはアクティブであると見なされます。その他の場合、検査のために「ハートビート」が受信側チャンネルに送信されます。送信側チャンネルは、チャンネル・ハートビート間隔 (HBINT) 属性で指定された秒数に基づいて、チャンネルの受信側からの応答をその間隔待機します。

値はミリ秒単位で、0 から 999999 までの範囲内でなければなりません。ゼロの値は、バッチ・ハートビートが使用されないことを示します。

この属性は、以下のチャンネル・タイプで有効です。

- 送信側
- サーバー
- クラスター送信側
- クラスター受信側

バッチ間隔 (BATCHINT)

この属性は、伝送キューにメッセージが無くてもチャンネルがバッチをオープンにしておく時間 (ミリ秒) です。

0 から 999 999 999 までの任意の数をミリ秒数として指定できます。デフォルト値はゼロです。

バッチ間隔を指定しない場合、次の条件のいずれかが満たされたときにバッチがクローズします。

- BATCHSZ に指定された数のメッセージが送信された。
- BATCHLIM に指定されたバイト数が送信された。
- 伝送キューが空である。

伝送キューが頻繁に空になる負荷の軽いチャンネルでは、効率的なバッチ・サイズは BATCHSZ よりかなり小さい場合があります。

BATCHINT 属性を使用して、ショート・バッチ数を減らし、チャンネルをさらに効率的にすることができます。ただし、バッチが長くなるとメッセージがコミットされていないままの時間も長くなるので、応答に時間がかかる可能性があることに注意してください。

BATCHINT を指定した場合、バッチは以下の条件のいずれかと一致したときだけクローズします。

- BATCHSZ に指定された数のメッセージが送信された。
- BATCHLIM に指定されたバイト数が送信された。
- 伝送キューにはもうメッセージが残っておらず、(バッチの最初のメッセージが検索されてから) メッセージを待機する間に BATCHINT の時間間隔が経過した。

注: BATCHINT は、メッセージの待機時間の合計を指定します。伝送キューで既に使用可能になっているメッセージの検索時間もメッセージの転送時間も、BATCHINT には含まれません。

この属性は、以下のチャンネル・タイプで有効です。

- 送信側
- サーバー
- クラスター送信側
- クラスター受信側

バッチ限界 (BATCHLIM)

この属性では、同期点を取る前にチャンネルを介して送信可能なデータ量の限度 (キロバイト単位) を指定します。

限度に達した際のメッセージがチャンネルを通過して送信された後に、同期点が取られます。

値は 0 から 999999 の範囲でなければなりません。デフォルト値は 5000 です。

この属性の値がゼロの場合、それはこのチャンネルに対するバッチに適用されるデータ限度がないことを意味します。

バッチは、以下のいずれかの条件が満たされると終了します。

- BATCHSZ メッセージが送信された。
- BATCHLIM バイトが送信された。
- 伝送キューが空で、BATCHINT が経過した。

この属性は、以下のチャンネル・タイプで有効です。

- 送信側
- サーバー
- クラスタ送信側
- クラスタ受信側

BATCHSZ (バッチ・サイズ)

この属性は、同期点を取る前に送信されるメッセージの最大数です。

バッチ・サイズは、チャンネルがメッセージを転送する方法には影響を与えません。メッセージは常に個別に転送されますが、メッセージのコミットまたはバックアウトはバッチ単位で行われます。

パフォーマンスを向上させるために、バッチ・サイズを設定して、2つの同期点でやりとりされるメッセージの最大数を定義することができます。使用されるバッチ・サイズはチャンネルの開始時に折衝され、2つのチャンネル定義のうちの低いほうの値が採用されます。実現方法によっては、2つのチャンネル定義の低い方の値と2つのキュー・マネージャーの MAXUMSGS 値からバッチ・サイズが計算されます。バッチの実際のサイズはこの値より小さい場合があります。例えば、伝送キューにメッセージが残されていないかバッチ間隔が時間切れになると、バッチは完了してしまいます。

バッチ・サイズに大きな値を設定するとスループットが向上しますが、バックアウトや再送信の対象となるメッセージの数も多くなるため、リカバリーにかかる時間も長くなります。BATCHSZ のデフォルトは 50 であり、この値をまず試してみることをお勧めします。通信の信頼性が低く、リカバリーが必要になることが多い場合には、BATCHSZ の値を小さくした方がよいかもしれません。

同期点手順では、バッチ・コミット手順を調整するために、同期点が取られるたびにリンクを介して固有の論理作業単位 ID を交換する必要があります。

同期化されたバッチ・コミット手順が中断されると、未確定状態が発生する可能性があります。未確定状態は、メッセージ・チャンネルの開始時に自動的に解決されます。この方法で解決しない場合は、RESOLVE コマンドを使用した手操作による介入が必要になる可能性があります。

バッチ・サイズの値を選択する際には、以下の点を考慮してください。

- 値が大きすぎると、リンクの両側で占有されるキュー空間が過大になります。メッセージは、コミットされていないときにはキュー空間を占有し、コミットされるまでキューから除去することはできません。
- メッセージのフローが安定していることが予想される場合は、バッチ・サイズを増やすことによってチャンネルのパフォーマンスを向上できます。これは、同じ数のバイトを転送するのに多くの確認フローは必要ないためです。
- メッセージ・フローの特性から、メッセージが断続的に到着することが予想される場合は、切断時間間隔を比較的大きくしてバッチ・サイズに 1 を指定すると、良好なパフォーマンスが得られます。
- この値の範囲は 1 から 9999 までです。
- 高速チャンネルでの非永続メッセージは、同期点を待たない場合でも、バッチ・サイズに数えられます。

この属性は、以下のチャンネル・タイプで有効です。

- 送信側
- サーバー
- 受信側
- リクエスター
- クラスター送信側
- クラスター受信側

MQSC キーワードのチャンネル属性 (C)

MQSC キーワードのチャンネル属性のアルファベット順リスト。文字は C で始まります。

CERTLABL (証明書ラベル)

この属性は、チャンネル定義の証明書ラベルを指定します。

ラベルにより、鍵リポジトリに含まれているどの個人証明書をリモート・ピアに送信するかを指定します。証明書は、[デジタル証明書ラベル](#)に記載されているように定義されます。

インバウンド・チャンネル (チャンネル RCVR、RQSTR、CLUSRCVR、修飾されていない SERVER、および SVRCONN を含む) は、リモート・ピアの IBM MQ のバージョンが証明書ラベル構成を完全にサポートして、チャンネルが TLS CipherSpec を使用している場合にのみ、構成済みの証明書を送信します。

そうでない場合、送信される証明書は、キュー・マネージャーの **CERTLABL** 属性によって決定されます。この制限があるのは、インバウンド・チャンネルの証明書ラベルの選択メカニズムが依存する TLS プロトコル拡張が、すべてのケースでサポートされるわけではないためです。特に、Java クライアントおよび JMS クライアントは、必要なプロトコル拡張をサポートしておらず、チャンネル固有のラベル設定に関係なく、キュー・マネージャーの **CERTLABL** 属性によって構成された証明書のみを受信します。

修飾されていないサーバー・チャンネルとは、CONNNAME フィールドが設定されていないチャンネルです。

この属性を照会したり CLUSSDR チャンネルに対して設定したりできる管理インターフェースはありません。MQRCCF_WRONG_CHANNEL_TYPE メッセージを受け取ります。ただし、この属性は CLUSSDR チャンネル・オブジェクト (MQCD 構造体を含む) に存在しており、必要な場合は CHAD 出口でプログラマチックに設定できます。

証明書ラベルに含まれる情報については詳しくは、[デジタル証明書ラベルの要件に関する説明](#)を参照してください。

この属性は、すべてのチャンネル・タイプで有効です。

注: SSL/TLS では、CERTLABL は QMGR 定義で定義する必要があります。オプションで、CHANNEL 定義で CERTLABL を設定することができます。

CHANNEL 定義で CERTLABL を設定している場合であっても、キュー・マネージャー CERTLABL はチェックされ、有効な個人証明書である必要があります。

チャンネル名 (CHANNEL)

この属性は、チャンネル定義の名前を指定します。

この名前には 20 文字まで使用できますが、メッセージ・チャンネルの両端で同じ名前を指定する必要があります。また実現方法によってはサイズが制限されている場合があるため、実際の文字数はより小さくしなければならないことがあります。

可能な場合、チャンネル名は、相互接続されたキュー・マネージャーのネットワーク内にある任意の 2 つのキュー・マネージャーにおいて、1 つのチャンネルに固有の名前にします。

この名前の指定には、以下の文字を使用してください。

英字 (A から Z、a から z。大文字と小文字には意味があります)

数字	(0 から 9)
ピリオド	(.)
スラッシュ	(/)
下線	(_)
パーセント記号	(%)

注:

1. ブランクを埋め込むことは許されず、また先行ブランクは無視されます。
 2. EBCDIC カタカナを使用するシステムでは、小文字を使用することはできません。
- この属性は、すべてのチャンネル・タイプで有効です。

CHLTYPE (チャンネル・タイプ)

この属性は、定義中のチャンネルのタイプを指定します。


指定できるチャンネル・タイプは次のとおりです。

メッセージ・チャンネル・タイプ:

- 送信側
- サーバー
- 受信側
- リクエスター
- クラスター送信側
- クラスター受信側

MQI チャンネル・タイプ:

- クライアント接続 (AIX, Linux, and Windows のみ)

注:  クライアント接続チャンネルを、その他のプラットフォームで使用するために z/OS でも定義することができます。

- サーバー接続
- AMQP

チャンネルの両端は、同じ名前をもち、次のように互換性を備えたタイプでなければなりません。

- 送信側と受信側
- 要求側とサーバー
- 要求側と送信側 (コールバックの場合)
- サーバーと受信側 (サーバーが送信側として使用されます)
- クライアント接続とサーバー接続
- クラスター送信側とクラスター受信側
- AMQP と AMQP

CLNTWGHT (クライアント・チャンネル・ウェイト)

この属性は、どのクライアント接続チャンネル定義を使用するかに影響を与える加重を指定します。

クライアント・チャンネル加重属性を使用すると、複数の適切な定義が選択可能な場合に、クライアント・チャンネル定義を加重に基づいてランダムに選択できます。

先頭がアスタリスクのキュー・マネージャー名を指定して (複数のキュー・マネージャー間でクライアント・ウェイト・balancingが可能になる)、クライアントが MQCONN 要求接続をキュー・マネージャー・

グループに対して発行し、複数の適切なチャンネル定義がクライアント・チャンネル定義テーブル (CCDT) で選択可能な場合、使用する定義は加重に基づいてランダムに選択されます。適用可能な任意の CLNTWGHT(0) の定義が、アルファベット順に従って最初に選択されます。

注: JSON CCDT を使用する場合は、同じ名前のチャンネルが複数存在することがあります。同じ名前のチャンネルが複数存在する場合は、CLNTWGHT(0) になり、JSON CCDT で定義されている順序でチャンネルが選択されます。

0 から 99 の範囲の値を指定します。デフォルトは 0 です。

値として 0 を指定すると、ロード・バランシングが実行されず、該当する定義がアルファベット順で選択されます。ロード・バランシングを有効にするには、1 から 99 までの範囲の値を選択します (1 が最低の加重値、99 が最高の加重値です)。非ゼロの加重を持つ 2 つ以上のチャンネル間での接続の分散は、それらの加重の比率に比例したものになります。例えば、CLNTWGHT 値として 2、4、および 14 を持つ 3 つのチャンネルは、ほぼ 10%、20%、および 70% の時間の割合で選択されます。この分散は保証されているわけではありません。接続の AFFINITY 属性が PREFERRED に設定されている場合、最初の接続はクライアントの加重に応じてチャンネル定義を選択し、後続の接続は同じチャンネル定義を使用し続けます。

この属性は、クライアント接続チャンネル・タイプでのみ有効です。

クラスターの名前リスト (CLUSNL)

この属性は、そのチャンネルが所属するクラスターのリストを指定した名前リスト名です。

CLUSTER および CLUSNL の結果値は、そのどちらか一方だけ非ブランクにすることができます。一方の値を非ブランクにした場合、もう一方の値は必ずブランクにしてください。

この属性は、以下のチャンネル・タイプで有効です。

- クラスター送信側
- クラスター受信側

クラスター (CLUSTER)

この属性は、チャンネルが属するクラスターの名前です。

最大長は 48 文字で、IBM MQ オブジェクトの命名規則に従います。

CLUSTER および CLUSNL の結果値は、そのどちらか一方だけ非ブランクにすることができます。一方の値を非ブランクにした場合、もう一方の値は必ずブランクにしてください。

この属性は、以下のチャンネル・タイプでのみ有効です。

- クラスター送信側
- クラスター受信側

CLWLPRTY (クラスター・ワークロード優先順位)

CLWLPRTY チャンネル属性は、クラスター・ワークロード分散のために、チャンネルの優先順位を指定します。この値は、0 から 9 の範囲でなければなりません。0 が最低、9 が最高の優先度です。

CLWLPRTY チャンネル属性は、使用可能なクラスター宛先の優先順位を設定するために使用します。IBM MQ は、クラスターの宛先の優先度が最も低い宛先を選択する前に、優先度が最も高い宛先を選択します。優先度が同じである複数の宛先が存在する場合は、最も長期間未使用になっている宛先を選択します。

有効な宛先が 2 つある場合は、この属性を使用して、フェイルオーバーを許可できます。メッセージは、最高の優先順位のチャンネルを持つキュー・マネージャーに移動されます。このキュー・マネージャーが使用不可になった場合、メッセージはその次に高い優先順位のキュー・マネージャーに移動されます。低い優先順位のキュー・マネージャーは、予備としての役割を果たします。

IBM MQ は、チャンネルに優先順位付けする前に、チャンネルの状況を検査します。使用可能なキュー・マネージャーのみが選択候補になります。

注:

- ・ クラスター受信側チャンネルのターゲット・キュー・マネージャーで、この属性を指定します。一致するクラスター送信側チャンネルで指定したすべてのバランシングは、無視される可能性が高くなります。[クラスター・チャンネル](#)を参照してください。
- ・ リモート・キュー・マネージャーの可用性は、そのキュー・マネージャーに対するチャンネルの状況に基づきます。チャンネルを開始する際に、チャンネルの状態は何度か変わります。その中には、クラスターのワークロード管理アルゴリズムの結果より優先度の低い状態もあります。これは、実際に、優先順位がより高い(プライマリー)宛先に対するチャンネルが開始されていても、優先順位がより低い(バックアップ)宛先が選択される可能性があることを意味します。
- ・ バックアップ宛先に対してメッセージが送信されないようにする必要がある場合は、CLWLPRTY を使用しないでください。個別のキューを使用するか、または CLWLPRTY を使用してプライマリーからバックアップに手動で切り替えることを検討してください。

CLWLPRTY (クラスター・ワークロード・ランク)

CLWLPRTY チャンネル属性は、クラスター・ワークロード分散に使用するチャンネルのランクを指定します。値は 0 から 9 の範囲でなければなりません (0 が最低ランク、9 が最高ランク)。

CLWLPRTY チャンネル属性は、別のクラスター内のキュー・マネージャーに送信されるメッセージの最終宛先を制御する場合に使用します。クラスターの交点にあるゲートウェイ・キュー・マネージャーにキュー・マネージャーを接続するチャンネルのランクを設定することによって、最終宛先の選択を制御します。

CLWLPRTY を設定すると、メッセージは、相互接続されたクラスターを経由してランクの高い宛先に向かう指定経路を取ります。例えば、ランク 1 とランク 2 のチャンネルを使用する 2 つのキュー・マネージャーのどちらかにメッセージを送信できるゲートウェイ・キュー・マネージャーにメッセージが到着したとします。これらのメッセージは、最高ランクのチャンネル (この場合はランク 2 のキュー・マネージャーへのチャンネル) で接続されたキュー・マネージャーに自動的に送信されます。

IBM MQ は、チャンネル状況を検査する前にチャンネルのランクを取得します。チャンネル状況を検査する前にランクを取得するということは、アクセスできないチャンネルでも選択の対象になることを意味します。この方法は、最終宛先が使用不可であったとしても、ネットワークを介してメッセージがルーティングされることを可能にします。

注:

- ・ クラスター受信側チャンネルのターゲット・キュー・マネージャーで、この属性を指定します。一致するクラスター送信側チャンネルで指定したすべてのバランシングは、無視される可能性が高くなります。[クラスター・チャンネル](#)を参照してください。
- ・ 優先度属性 **CLWLPRTY** も使用した場合、IBM MQ は使用可能な宛先から選択します。最高ランクの宛先がチャンネルを使用できない場合、メッセージは伝送キューに保持されます。チャンネルが使用可能になると解放されます。ランク順で次に使用可能な宛先にメッセージが送信されるわけではありません。

CLWLWGT (クラスター・ワークロード・ウェイト)

CLWLWGT チャンネル属性は、クラスター・ワークロード分散用に、CLUSDR および CLUSRCVR チャンネルに適用される重みを指定します。この値は、1 から 99 の範囲でなければなりません。1 が最低、99 が最高の重みです。

CLWLWGT を使用して、より多くの処理能力があるサーバーに、より多くのメッセージを送信します。チャンネルの重みが高いほど、そのチャンネルを通して多くのメッセージが送信されます。

注:

- ・ クラスター受信側チャンネルのターゲット・キュー・マネージャーで、この属性を指定します。一致するクラスター送信側チャンネルで指定したすべてのバランシングは、無視される可能性が高くなります。[クラスター・チャンネル](#)を参照してください。
- ・ いずれかのチャンネルで **CLWLWGT** がデフォルトの 50 から変更された場合、ワークロード・バランシングはいずれかのクラスター・キューに送信されたメッセージのために各チャンネルが合計何回選択されたかに依存します。詳しくは、[146 ページ](#)の『[クラスター・ワークロード管理アルゴリズム](#)』を参照してください。

COMPHDR (ヘッダー圧縮)

この属性は、チャンネルでサポートされるヘッダー・データ圧縮技法のリストです。

送信側、サーバー、クラスター送信側、クラスター受信側、およびクライアント接続の各チャンネルでは、指定された値が設定の順序に並び、チャンネルのリモート・エンドでサポートされる最初の圧縮手法が使用されます。チャンネルで相互にサポートされる圧縮手法は、送信側チャンネルのメッセージ出口に渡されます。そこでは、使用される圧縮手法をメッセージごとに変更できます。圧縮により、送信および受信出口に渡されたデータが変更されます。

指定可能な値は以下のとおりです。

NONE

ヘッダー・データ圧縮は実行されません。この値がデフォルト値です。

SYSTEM

ヘッダー・データ圧縮が実行されます。

この属性は、すべてのチャンネル・タイプで有効です。

COMPMSG (データ圧縮)

この属性は、チャンネルでサポートされるメッセージ・データ圧縮技法のリストです。

送信側チャンネル、サーバー・チャンネル、クラスター送信側チャンネル、クラスター受信側チャンネル、およびクライアント接続チャンネルの場合、望ましい順に値が指定されます。チャンネルのリモート・エンドでサポートされる最初の圧縮手法が使用されます。チャンネルで相互にサポートされる圧縮手法は、送信側チャンネルのメッセージ出口に渡されます。そこでは、使用される圧縮手法をメッセージごとに変更できます。圧縮により、送信および受信出口に渡されたデータが変更されます。メッセージ・ヘッダーの圧縮については、[100 ページの『COMPHDR \(ヘッダー圧縮\)』](#)を参照してください。

指定できる値は以下のとおりです。

NONE

メッセージ・データ圧縮は実行されません。この値がデフォルト値です。

RLE

ラン・レングス・エンコードを使用してメッセージ・データ圧縮が実行されます。

ZLIBFAST

zlib 圧縮手法を使用してメッセージ・データ圧縮が実行されます。高速圧縮時間を推奨します。

ZLIBFAST はオプションで zEnterprise® データ圧縮ファシリティにオフロードできます。詳細については、[zEDC Express 機能](#)を参照してください。

ZLIBHIGH

zlib 圧縮手法を使用してメッセージ・データ圧縮が実行されます。ハイレベル圧縮を推奨します。

V 9.4.0

LZ4FAST

メッセージ・データ圧縮は、LZ4 圧縮技法を使用して実行されます。高速圧縮時間を推奨します。

V 9.4.0

LZ4HIGH

メッセージ・データ圧縮は、LZ4 圧縮技法を使用して実行されます。ハイレベル圧縮を推奨します。

ANY

チャンネルで、キュー・マネージャーがサポートするすべての圧縮手法のサポートが可能になります。サポート対象は受信側、リクエスター、サーバー接続の各チャンネルのみです。

この属性は、すべてのチャンネル・タイプで有効です。

AIX

IBM MQ 9.3.0 以降では、ZLIBFAST および ZLIBHIGH の技法で、IBM MQ for AIX 上のハードウェア・アクセラレーションされた zlibNX ライブラリーを使用することができます (このライブラリーがインストールされている場合)。zlibNX ライブラリーは zlib 圧縮ライブラリーの拡張バージョンで、IBM POWER9 プロセッサ・ベース・サーバー上のネスト・アクセラレーター (NX) と呼ばれるコプロセッサを使用することによって、ハードウェア・アクセラレーションされたデータの圧縮および圧縮解除をサポートします。zlibNX ライブラリーは、IBM AIX 7.2 with Technology Level 4 Expansion Pack 以降で使用可

能です。zlibNX ライブラリーの使用を選択することにより CPU 使用率が削減されることから益を得る可能性が最も高いのは、高率での圧縮が可能な 2KB を超えるサイズのメッセージです。メッセージング・チャンネル・エージェント (MCA) で zlibNX ライブラリーを使用できるようにするためには、環境変数 AMQ_USE_ZLIBNX を設定します。

CONNNAME (接続名)

この属性は、通信接続 ID です。チャンネルが使用する特定の通信リンクを指定します。

サーバー・チャンネルが起動されていない場合、サーバー・チャンネルではオプションです。その場合、接続名を指定する必要があります。

CONNNAME を、指定された **TRPTYPE** のマシンの名前のコンマ区切りリストとして指定します。通常、必要なマシン名は 1 つだけです。複数のマシン名を指定して、同じプロパティーで複数の接続を構成することができます。接続は、通常は正常に確立されるまで、接続リストに指定された順序で試行されます。

CLNTWGHT 属性が指定されている場合、クライアントの順序が変更されます。どの接続も成功しなかった場合、チャンネルの属性によって決められたとおりに、チャンネルは再接続を試みます。クライアント・チャンネルでは、キュー・マネージャー・グループの代わりに、接続リストを使用して複数接続を構成することができます。メッセージ・チャンネルでは、複数インスタンス・キュー・マネージャーの代替アドレスへの接続を構成するために、接続リストが使用されます。

Multi マルチプラットフォームでは、クラスター受信側チャンネルの TCP/IP 接続名パラメーターはオプションです。接続名をブランクにすると、IBM MQ はデフォルト・ポートを想定し、システムの現行 IP アドレスを使用して接続名を自動的に生成します。デフォルト・ポート番号をオーバーライドしても、システムの現行 IP アドレスを引き続き使用できます。各接続名について、IP 名をブランクにして、次のように括弧で囲んだポート番号を指定してください。

(1415)

生成される **CONNNAME** は常にドット 10 進 (IPv4) 形式または 16 進 (IPv6) 形式であり、英数字の DNS ホスト名の形式ではありません。

名前の最大長はプラットフォームによって異なります。

- ▶ **Multi** 264 文字。
- ▶ **z/OS** 48 文字 (注 1 を参照)。

トランスポート・タイプが TCP の場合

CONNNAME は、ホスト名、またはリモート・マシン (あるいはクラスター受信側チャンネルのローカル・マシン) のネットワーク・アドレスのいずれかです。例えば、(ABC.EXAMPLE.COM)、(2001:DB8:0:0:0:0:0:0)、(127.0.0.1) などです。ポート番号を含めることができます (例: (MACHINE(123)))。

▶ **z/OS** またこれには、動的 DNS グループまたはネットワーク・ディスパッチャーの入力ポートの IP_name を入れることもできます。

IPv4 のみをサポートするネットワークで IPv6 アドレスを使用する場合、接続名は解決されません。IPv4 および IPv6 の両方を使用するネットワークでは、接続名とローカル・アドレスの相互作用によって、どちらの IP スタックを使用するかが決定されます。詳細については、[106 ページの『ローカル・アドレス \(LOCLADDR\)』](#)を参照してください。

トランスポート・タイプが LU 6.2 の場合

▶ **Multi** TPNAME および MODENAME を指定する場合は、パートナー LU の完全修飾名を指定します。TPNAME および MODENAME がブランクの場合は、特定のプラットフォームの CPI-C サイド情報オブジェクト名を指定します。

▶ **z/OS** 次の 2 とおりの形式を使用して値を指定します。

- 論理装置 (LU) 名

キュー・マネージャーの論理装置名。論理装置名、TP 名、およびオプション・モード名で構成されます。次の 3 通りの形式のどれかを使用して、この名前を指定します。

表 55. 論理装置名と形式	
形式	例
luname	IGY12355
luname/TPname	IGY12345/APING
luname/TPname/modename	IGY12345/APINGD/#INTER

最初の形式を使用する場合は、TPNAME 属性と MODENAME 属性に対して、それぞれ TP 名とモード名を指定する必要があります。それ以外の形式を使用する場合は、これらの属性は必ずブランクにしてください。クライアント接続チャンネルでは、最初の形式しか使用できません。

- シンボル名

キュー・マネージャーの論理装置名を表すシンボリック宛先名。この名前はサイド情報データ・セットに定義されています。TPNAME 属性と MODENAME 属性は、必ずブランクにしてください。クラスター受信側チャンネルでは、サイド情報はクラスター内の他のキュー・マネージャーにあることに注意してください。この場合には、チャンネル自動定義出口による名前解決処理の結果、ローカル・キュー・マネージャーに適した論理装置情報になるような名前にすることができます。

指定する LU 名または暗黙の LU 名は、VTAM 汎用リソース・グループの名前にすることができます。

伝送プロトコルが NetBIOS である場合

CONNNAME はリモート・マシンで定義された NetBIOS 名です。

伝送プロトコルが SPX である場合

CONNNAME は、4 バイトのネットワーク・アドレス、6 バイトのノード・アドレス、および 2 バイトのソケット番号から構成される SPX 形式のアドレスです。ピリオドでネットワーク・アドレスとノード・アドレスを区切り、ソケット数を大括弧で囲み、16 進数でこれらの値を入力してください。以下に例を示します。

```
CONNNAME('0a0b0c0d.804abcde23a1(5e86)')
```

ソケット数が省略された場合、デフォルトの IBM MQ SPX ソケット数が使用されます。デフォルトは X'5E86' です。

この属性は、以下のチャンネル・タイプで有効です。

- 送信側
- サーバー

サーバー・チャンネルが起動されていない場合、サーバー・チャンネルではオプションです。その場合、接続名を指定する必要があります。

- リクエスター
- クライアント接続
- クラスター送信側
- クラスター受信側

注:

1. 名前の長さについては、以下のいずれかの方法で、48 文字の長さ制限を回避できます。
 - 短いホスト名を使用するように (例えば、「myserver.location.company.com」の代わりに「myserver」というホスト名) DNS サーバーをセットアップする。
 - IP アドレスを使用する。
2. 伝送プロトコルの定義は、[TRPTYPE](#) に含まれています。

CONVERT (メッセージの変換)

この属性は、伝送する前にメッセージを受信側のシステムが要求する形式に変換しなければならないことを指定します。

アプリケーション・メッセージ・データは通常、受信側のアプリケーションによって変換されます。しかし、リモート・キュー・マネージャーのあるプラットフォームがデータ変換をサポートしていない場合は、このチャンネル属性を使用して、伝送する **前に** メッセージを受信側のシステムが要求する形式に変換しなければならないことを指定します。

指定可能できる値は **yes** と **no** です。 **yes** を指定すると、標準装備の形式名の 1 つが指定されているか、またはユーザー定義の形式用に変換出口が使用可能である場合に ([データ変換出口の作成](#)を参照)、メッセージ内のアプリケーション・データは送信前に変換されます。 **no** を指定すると、メッセージ中のアプリケーション・データは送信前に変換されません。

この属性は、以下のチャンネル・タイプで有効です。

- 送信側
- サーバー
- クラスター送信側
- クラスター受信側

MQSC キーワード (D-L) のチャンネル属性

MQSC キーワードのチャンネル属性のアルファベット順リスト。 *D* から *L* の文字で始まります。

DEFRECON (デフォルト再接続)

クライアント接続がクライアント・アプリケーションへの接続から切断した場合に、自動的に再接続するかどうかを指定します。

指定できる値は以下のとおりです。

NO (デフォルト)

MQCONN によってオーバーライドされない限り、クライアントは自動的に再接続されません。

はい

MQCONN によってオーバーライドされない限り、クライアントは自動的に再接続します。

QMGR

MQCONN によってオーバーライドされない限り、クライアントは、同じキュー・マネージャーに対してのみ自動的に再接続します。 **QMGR** オプションには **MQCNO_RECONNECT_Q_MGR** と同じ効果があります。

disabled

MQCONN **MQI** 呼び出しを使用してクライアント・プログラムによって要求された場合でも、再接続は無効になります。

この属性はクライアント接続チャンネルにのみ有効です。

DESCR (説明)

この属性は、チャンネル定義を記述します。最大 64 バイトのテキストが入ります。

注: システムが 2 バイト文字セット (DBCS) を使用している場合、最大文字数が少なくなります。

テキストを別のメッセージ・キュー・マネージャーに送信する場合には、テキストが正しく変換されるように、メッセージ・キュー・マネージャー用のコード化文字セット ID (CCSID) で識別された文字セットの中の文字を使用してください。

この属性は、すべてのチャンネル・タイプで有効です。

DISCINT (切断間隔)

この属性は、到達するメッセージがない場合にチャンネルがクローズされるまでの時間の長さを指定します。

この属性は、以下のチャンネル・タイプで有効です。

- 送信側
- サーバー
- TCP プロトコルのみを使用したサーバー接続
- クラスター送信側
- クラスター受信側

この属性は、秒単位で指定されるタイムアウトです。

サーバー間メッセージ・チャンネル (サーバー、クラスター送信側、送信側、およびクラスター受信側のチャンネル・タイプ) の場合、間隔は、バッチが終了した時点 (バッチ・サイズに達した時点、またはバッチ間隔が満了して伝送キューが空になった時点) から測定されます。指定の時間間隔内に伝送キューに到達するメッセージがない場合には、チャンネルはクローズされます。(この時間は概数です。)

サーバー間メッセージ・チャンネルの両端の間の制御データのクローズ・ダウン交換には、クローズの理由が示されます。これにより、チャンネルの相手側が再び開始できる状態に保たれます。

0 から 999 999 までの任意の秒数を指定できます。値 0 は切断なしを意味し、無期限に待機します。

デフォルトは、サーバー間メッセージ・チャンネルの場合は 6000 秒 (100 分) で、サーバー接続チャンネルの場合は 0 (タイムアウトなし) です。デフォルトのチャンネル・オブジェクトを変更することにより、作成する新規チャンネルに使用されるデフォルト値を変更できます。例えば、SYSTEM.DEF.SENDER。定義する新しい送信側チャンネルの新しいデフォルトを指定します。

TCP プロトコルを使用するサーバー接続チャンネルでは、この間隔はクライアントの非アクティブ切断の値 (秒単位) を表します。サーバー接続チャンネル・プログラムは、この期間にパートナー・クライアントから通信を受信しなかった場合、接続を終了します。

サーバー接続の非アクティブ間隔は、クライアントからの IBM MQ API 呼び出し間で適用されます。

注: 待機呼び出しを伴う長時間実行される可能性がある MQGET は、非アクティブとして分類されないため、DISCINT の有効期限が切れた結果としてタイムアウトになることはありません。



重要: 切断間隔に指定する値は、パフォーマンスに影響を与えます。

低い値 (例えば、数秒) は、チャンネルを絶えず停止して再始動することにより、システム・パフォーマンスに悪影響を及ぼす可能性があります。値が大きい (1 時間を超える) 場合は、システム・リソースが無益に消費されている可能性があります。ハートビート間隔を指定することで、伝送キューにメッセージがないときには、送信側の MCA が受信側の MCA にハートビート・フローを送信し、切断間隔が経過するまで待たずに受信側の MCA がチャンネルを静止できるようにすることも可能です。これら 2 つの値を共に有効にするには、ハートビート間隔値を切断間隔値よりもかなり小さい値にする必要があります。

サーバー間メッセージ・チャンネルのデフォルトの DISCINT 値は 6000 秒 (100 分) です。ただし、パフォーマンスに影響を与えたり、不必要に長時間にわたってチャンネルを実行し続けたりしないようにするには、多くの場合、この値を数分に設定するのが適切です。ご使用の環境に適している場合は、個々のチャンネルで、またはデフォルトのチャンネル定義 (SYSTEM.DEF.SENDER (送信側チャンネルの場合) (独自のチャンネルを作成する前))。

詳しくは、[チャンネルの停止および静止](#)を参照してください。

HBINT (ハートビート間隔)

この属性によって、伝送キューにメッセージがなくなったときに送信メッセージ・チャンネル・エージェント (MCA) からハートビート・フローが渡される間の時間の近似値を指定することができます。

ハートビート・フローは、メッセージの到着または切断インターバルの満了を待機している受信 MCA を非ブロック化します。受信 MCA が非ブロック化された場合には、そこで切断インターバルが満了するのを待たずにチャンネルを切断することができます。ハートビート・フローは、また、大きなメッセージのために

割り振られているすべてのストレージ・バッファを解放し、チャンネルの受信終了時にオープンされたままになっていたすべてのキューをクローズします。

値は秒単位であり、範囲は 0 から 999 999 でなければなりません。値 0 は、ハートビート・フローが送信されないことを意味します。デフォルト値は 300 です。最も有効にするには、この値が切断インターバル値より大幅に低いものでなければなりません。

IBM MQ classes for Java、JMS、または .NET API を使用するアプリケーションでは、HBINT の値が次のいずれかの方法で決定されます。

- アプリケーションによって使用される SVRCONN チャンネル上の値。
- アプリケーションが CCDT を使用するよう構成されている場合は、CLNTCONN チャンネル上の値。

サーバー接続チャンネルおよびクライアント接続チャンネルでは、ハートビートはサーバー・サイドおよびクライアント・サイドの両方から個々に流れることがあります。ハートビート間隔の時間内にチャンネルを通してデータが転送されないと、クライアント接続 MQI エージェントはハートビート・フローを送信し、それに対してサーバー接続 MQI エージェントは別のハートビート・フローで応答します。これは、チャンネルの状態に関係なく発生します。例えば、API の呼び出し中にチャンネルが非アクティブであっても、クライアント・ユーザーからの入力の待機中にチャンネルが非アクティブであっても関係なく発生します。サーバー接続 MQI エージェントも、同様にチャンネルの状態に関係なく、クライアントへのハートビートを開始することができます。サーバー接続 MQI エージェントとクライアント接続 MQI エージェントが同時に互いにハートビートを送信しないようにするために、サーバーのハートビートは、ハートビート間隔より 5 秒長い時間内にチャンネルを通してデータが転送されなかった場合に流れます。

IBM WebSphere® MQ 7.0 より前のチャンネル・モードで動作するサーバー接続およびクライアント接続の各チャンネルでは、サーバー MCA が、WAIT オプションを指定して、クライアント・アプリケーションに代わって発行した MQGET コマンドを待機しているときにのみハートビート・フローが送信されます。

MQI チャンネルを 2 つのモードで動作させる方法については、[SharingConversations \(MQLONG\)](#) を参照してください。

KAINT (キープアライブ間隔)

この属性は、チャンネルのタイムアウト値を指定するために使用します。

キープアライブ間隔属性は、チャンネルのキープアライブ・タイミングを指定する通信スタックに渡される値です。このため、チャンネルごとに別々の KeepAlive 値を指定できます。

チャンネルのキープアライブ間隔 (KAINT) 属性はチャンネルごとに設定することができます。

Multi マルチプラットフォームでは、このパラメーターへのアクセスおよび変更が可能ですが、保管されて転送されるだけです。機能的にはこのパラメーターは実装されていません。KAINT パラメーターの機能が必要な場合、[104 ページの『HBINT \(ハートビート間隔\)』](#)で説明しているとおり、Heartbeat Interval (HBINT) パラメーターを使用してください。

この属性を有効にするには、TCP/IP キープアライブを使用可能にする必要があります。

- **z/OS** z/OS でキープアライブを使用可能にするには、ALTER QMGR TCPKEEP(YES) MQSC コマンドを発行します。
- **Multi** マルチプラットフォームでこれを行うには、KEEPALIVE=YES パラメーターを分散キューイング構成ファイル qm.ini の TCP スタンザで指定するか、IBM MQ Explorer エクスプローラーを使用して指定します。

TCP プロファイル構成データ・セットを使用して、TCP/IP 自体の中でもキープアライブを使用可能にする必要があります。

この値は秒単位の時間を表し、範囲は 0 から 99999 でなければなりません。キープアライブ間隔の値 0 は、チャンネル固有のキープアライブがチャンネルに対して有効になっておらず、TCP/IP で設定されたシステム全体のキープアライブ値だけが使用されることを示しています。KAINT の値を AUTO (この値がデフォルト) に設定することもできます。KAINT を AUTO に設定すると、キープアライブ値は、以下に示すとおり、折衝ハートビート間隔 (HBINT) の値に基づいたものとなります。

表 56. 折衝された HBINT 値および対応する KAIN T 値	
折衝された HBINT	KAIN T
>0	折衝された HBINT + 60 秒
0	0

この属性は、すべてのチャンネル・タイプで有効です。

この値は、TransportType (TRPTYPE) が TCP または SPX 以外のすべてのチャンネルでは無視されます。

ローカル・アドレス (LOCLADDR)

この属性は、チャンネルのローカル通信アドレスを指定します。

注: AMQP チャンネルは、他の IBM MQ チャンネルと同じ形式の LOCLADDR をサポートしません。詳しくは、108 ページの『#unique_51/unique_51_Connect_42_locladdr_amqp』を参照してください。

AMQP チャンネル以外のすべてのチャンネルの LOCLADDR

この属性は、トランスポート・タイプ (TRPTYPE) が TCP/IP の場合にのみ適用されます。その他のすべてのトランスポート・タイプでは、これは無視されます。

LOCLADDR 値を指定すると、停止され再始動されるチャンネルは、引き続き LOCLADDR で指定される TCP/IP アドレスを使用します。リカバリーのシナリオでは、チャンネルがファイアウォールを介して通信を行っている場合に、この属性が役に立つ可能性があります。これは、チャンネルをその接続先の TCP/IP スタックの IP アドレスを使用して再始動することによる問題がなくなるからです。LOCLADDR を使用すると、デュアル・スタック・システム上で IPv4 または IPv6 スタックを使用するように、またはシングル・スタック・システム上でデュアル・モード・スタックを使用するようにチャンネルを強制することもできます。

この属性は、以下のチャンネル・タイプで有効です。

- 送信側
- サーバー
- リクエスター
- クライアント接続
- クラスター送信側
- クラスター受信側

LOCLADDR にネットワーク・アドレスが含まれている場合、そのアドレスは、チャンネルが実行されるシステム上のネットワーク・インターフェースに属するネットワーク・アドレスでなければなりません。例えば、MSQC コマンドを使用して、キュー・マネージャー ALPHA の送信側チャンネルをキュー・マネージャー BETA に定義する場合は、次のようにします。

```
DEFINE CHANNEL(TO.BETA) CHLTYPE(SDR) CONNAME(192.0.2.0) XMITQ(BETA) LOCLADDR(192.0.2.1)
```

この LOCLADDR アドレスは、IPv4 アドレス 192.0.2.1 です。この送信側チャンネルはキュー・マネージャー ALPHA のシステム上で実行されるため、IPv4 アドレスは、そのシステムのネットワーク・インターフェースのいずれかに属していなければなりません。

値は、アウトバウンド TCP/IP 通信に使用されるオプションの IP アドレスおよびオプションのポートまたはポート範囲です。この情報の形式は、次のとおりです。

```
LOCLADDR([ip-addr][(low-port[,high-port])][, [ip-addr][(low-port[,high-port])]])
```

複数のアドレスを含む **LOCLADDR** の最大長は `MQ_LOCAL_ADDRESS_LENGTH` です。

LOCLADDR を省略すると、ローカル・アドレスが自動的に割り振られます。

クライアント・チャンネル定義テーブル (CCDT) を使用して C クライアントに対して **LOCLADDR** を設定できることに注意してください。

どのパラメーターも指定は任意です。アドレスの `ip-addr` 部分を省略しておくと、IP ファイアウォール用に固定ポート番号を設定できて便利です。ポート番号を省略すると、固有のローカル・ポート番号を指定しなくても、特定のネットワーク・アダプターを選択できます。TCP/IP スタックで固有のポート番号が生成されます。

追加のローカル・アドレスごとに `[, [ip-addr][(low-port[,high-port])]]` を複数回指定します。複数のローカル・アドレスは、ローカル・ネットワーク・アダプターの特定のサブセットを指定する場合に使用します。`[, [ip-addr][(low-port[,high-port])]]` を使用して、マルチインスタンス・キュー・マネージャー構成の一部である、さまざまなサーバー上の特定のローカル・ネットワーク・アドレスを表すこともできます。

ip-addr

`ip-addr` は、次の 3 つの形式のいずれかで指定できます。

IPv4 ドット 10 進数

例えば、192.0.2.1

IPv6 16 進表記

例えば、2001:DB8:0:0:0:0:0:0

英数字のホスト名書式

例えば `WWW.EXAMPLE.COM`

low-port および high-port

`low-port` および `high-port` は、括弧で囲まれたポート番号です。

次の表は、**LOCLADDR** パラメーターを使用する方法を示しています。

LOCLADDR	意味
9.20.4.98	チャンネルは、ローカル側でこのアドレスにバインドします。
9.20.4.98, 9.20.4.99	チャンネルは、いずれかの IP アドレスにバインドします。このアドレスは、1 つのサーバーの 2 つのネットワーク・アダプターであるか、複数インスタンス構成された 2 つの別個のサーバーの各ネットワーク・アダプターであるかもしれません。
9.20.4.98(1000)	チャンネルは、このアドレスおよびポート 1000 にローカルにバインドします。
9.20.4.98(1000,2000)	チャンネルは、このアドレスにバインドし、1000 から 2000 の範囲のポートをローカル側で使用します。
(1000)	チャンネルは、ローカル側でポート 1000 にバインドします。
(1000,2000)	チャンネルは、ローカル側で 1000 から 2000 の範囲のポートにバインドします。

チャンネルが開始されると、接続名 (CONNAME) およびローカル・アドレス (LOCLADDR) に指定した値により、通信に使用される IP スタックが決定します。使用される IP スタックは、以下のように決定します。

- システムに IPv4 スタックのみ構成されている場合、常時 IPv4 スタックを使用します。ローカル・アドレス (LOCLADDR) または接続名 (CONNAME) が IPv6 ネットワーク・アドレスとして指定されていると、エラーが生成され、チャンネルの開始は失敗します。
- システムに IPv6 スタックのみ構成されている場合、常時 IPv6 スタックを使用します。ローカル・アドレス (LOCLADDR) が IPv4 ネットワーク・アドレスとして指定されていると、エラーが生成され、チャンネルの開始は失敗します。

ルの開始は失敗します。IPv6 マップ・アドレッシングをサポートするプラットフォームでは、接続名 (CONNAME) が IPv4 ネットワーク・アドレスとして指定されていると、その IPv4 アドレスは IPv6 アドレスにマップされます。例えば、xxx.xxx.xxx.xxx が ::ffff:xxx.xxx.xxx.xxx にマップされます。マップされたアドレスを使用するには、プロトコル変換プログラムが必要となる場合があります。マップされたアドレスはできるだけ使用しないようにしてください。

- ローカル・アドレス (LOCLADDR) がチャンネルの IP アドレスとして指定されている場合、その IP アドレスのスタックが使用されます。ローカル・アドレス (LOCLADDR) が、IPv4 アドレスと IPv6 アドレスの両方に解決されるホスト名として指定されている場合、使用するスタックは、接続名 (CONNAME) によって決定されます。ローカル・アドレス (LOCLADDR) および接続名 (CONNAME) の両方が、IPv4 アドレスと IPv6 アドレスの両方に解決されるホスト名として指定されている場合、使用するスタックは、キュー・マネージャー属性 IPADDRV によって決定されます。
- システムに IPv4 および IPv6 のデュアル・スタックが構成され、チャンネルのローカル・アドレス (LOCLADDR) が指定されていない場合、使用するスタックはチャンネルで指定された接続名 (CONNAME) によって決定されます。接続名 (CONNAME) が、IPv4 アドレスと IPv6 アドレスの両方に解決されるホスト名として指定されている場合、使用するスタックは、キュー・マネージャー属性 IPADDRV によって決定されます。

Multi マルチプラットフォームでは、ローカル・アドレスが定義されていないすべての送信側チャンネルで使用される、デフォルトのローカル・アドレス値を設定することができます。このデフォルト値を定義するには、キュー・マネージャーの開始前に MQ_LCLADDR 環境変数を設定します。この値の形式は、MQSC 属性の LOCLADDR の形式と同じです。

クラスター送信側チャンネルのローカル・アドレス

クラスター送信側チャンネルは、常に、宛先キュー・マネージャーに定義された対応するクラスター受信側チャンネルの構成を継承します。これは、同じ名前のクラスター送信側チャンネルがローカルに定義されている場合でも行われます。その場合、手動定義は初期通信にしか使用されません。

したがって、クラスター受信側チャンネルに定義された LOCLADDR に依存することはできません。この LOCLADDR の IP アドレスは、クラスター送信側が作成されるシステムには存在しない可能性があるからです。そのため、クラスター受信側で LOCLADDR は使用すべきではありませんが、想定されるすべてのクラスター送信側の IP アドレスではなくポートのみを制限する理由があり、クラスター送信側チャンネルが作成される可能性のあるすべてのシステムでそれらのポートが使用可能であることが既知である場合は例外です。

クラスターで LOCLADDR を使用してアウトバウンド通信チャンネルを特定の IP アドレスにバインドする必要がある場合は、[チャンネル自動定義出口](#)を使用するか、または可能であればキュー・マネージャーのデフォルトの LOCLADDR を使用してください。チャンネル出口を使用すると、出口の LOCLADDR 値が、自動的に定義される CLUSSDR チャンネルに強制的に適用されます。

出口を使用してクラスター送信側チャンネルでデフォルト以外の LOCLADDR を使用する場合も、デフォルト値を使用する場合も、手動で定義された一致するクラスター送信側チャンネルがある場合は (完全リポジトリのキュー・マネージャーに対するチャンネルなど)、そのチャンネルによる初期通信を可能にするためにそのチャンネルにも LOCLADDR 値が設定されている必要があります。

注: LOCLADDR で指定されているポート (あるいは、ポート範囲が指定されている場合はすべてのポート) に関してオペレーティング・システムからバインド・エラーが返された場合は、チャンネルが開始されず、システムはエラー・メッセージを出します。

AMQP チャンネルの LOCLADDR

AMQP チャンネルは、他の IBM MQ チャンネルとは異なる形式の LOCLADDR をサポートします。

LOCLADDR (*ip-addr*)

LOCLADDR は、チャンネルのローカル通信アドレスです。このパラメーターは、特定の IP アドレスの使用をクライアントに強制する必要がある場合に使用します。LOCLADDR は、チャンネルで IPv4 または IPv6 アドレスを使用したり (選択可能な場合)、複数のネットワーク・アダプターがあるシステムにおいて特定のネットワーク・アダプターを使用したりすることを強制する場合に役立ちます。

LOCLADDR の最大長は MQ_LOCAL_ADDRESS_LENGTH です。

LOCLADDR を省略すると、ローカル・アドレスが自動的に割り振られます。

ip-addr

ip-addr は、次の 3 つの形式のいずれかで指定される単一のネットワーク・アドレスです。

IPv4 ドット 10 進数

例えば 192.0.2.1

IPv6 16 進表記

例えば 2001:DB8:0:0:0:0:0:0

英数字のホスト名書式

例えば WWW.EXAMPLE.COM

IP アドレスを入力すると、アドレス・フォーマットのみが妥当性検査されます。IP アドレス自体は妥当性検査されません。

追加情報については、[自動定義クラスター送信側チャンネルの操作](#) を参照してください。

長期再試行カウント (LONGRTY)

この属性は、チャンネルがそのパートナーへのセッションの割り振りを試行する最大回数を指定します。

long retry count 属性に設定できる値は 0 から 999 999 999 です。

この属性は、以下のチャンネル・タイプで有効です。

- 送信側
- サーバー
- クラスター送信側
- クラスター受信側

初期の割り振りの試みに失敗した場合、短期再試行カウント番号が減り、チャンネルは短期再試行カウントの残りの回数を再試行します。それでも失敗した場合、長期再試行カウントの回数を、各試行間の長期再試行間隔の間隔で再試行します。それでも割り振りが成功しない場合には、チャンネルは停止します。このチャンネルは、後で、コマンドを使用して再始動する必要があります。チャンネル・イニシエーターによって自動的に開始されることはありません。

z/OS z/OS では、チャンネルの最大数 (MAXCHL) を超えると、チャンネルが再試行を開始できなくなります。

Multi マルチプラットフォームでは、再試行を行うためには、チャンネル・イニシエーターが実行されている必要があります。チャンネル・イニシエーターは、チャンネルが使用している伝送キューの定義に指定されている開始キューをモニターしている必要があります。

チャンネルの再試行中にチャンネル・イニシエーター (z/OS の場合) またはチャンネル (マルチプラットフォームの場合) が停止すると、チャンネル・イニシエーターまたはチャンネルの再始動時、またはメッセージが正常に送信側チャンネルに書き込まれたときに、短期再試行カウント および 長期再試行カウント がリセットされます。ただし、チャンネル・イニシエーター (z/OS の場合) または、キュー・マネージャー (Multiplatforms では) がシャットダウンされて再始動された場合、短期再試行カウント および 長期再試行カウント はリセットされません。チャンネルは、キュー・マネージャーの再始動またはメッセージの送信の前に保持していた再試行カウント値を保持します。

Multi On マルチプラットフォーム:

1. チャンネルが RETRYING 状態から RUNNING 状態になる際、短期再試行カウント および 長期再試行カウント はすぐにはリセットされません。チャンネルが RUNNING 状態になったあとの最初のメッセージがチャンネルを通して正常に流れたときにだけリセットされます。つまり、ローカルのチャンネルが、相手側に送信したメッセージの数を確認した時点でリセットされます。
2. 短期再試行カウント および 長期再試行カウント は、チャンネルが再始動されるとリセットされます。

LONGTMR (長期再試行間隔)

この属性は、長期再試行モードのときに、チャンネルが接続の確立を再試行するまでに待機するおよその間隔 (秒単位) です。

チャンネルがアクティブになるのを待つ必要がある場合、再試行間隔が延長されることがあります。

チャンネルは、短期再試行間隔で短期再試行カウント回数を試みた後、この長期隔で長期再試行カウントの回数、接続を試みます。

この属性に設定できる値は 0 から 999 999 です。

この属性は、以下のチャンネル・タイプで有効です。

- 送信側
- サーバー
- クラスター送信側
- クラスター受信側

MQSC キーワードのチャンネル属性 (M)

MQSC キーワードのチャンネル属性のアルファベット順リスト。文字は M で始まります。

MAXINST (最大インスタンス数)

この属性は、サーバー接続チャンネルまたは AMQP チャンネルの、開始可能な同時インスタンスの最大数を指定します。

サーバー接続チャンネル接続の最大インスタンス数

サーバー接続チャンネルの場合、この属性は、1 つのサーバー接続チャンネルで開始可能な同時インスタンスの最大数を指定します。

この属性に設定できる値は 0 から 999 999 999 です。ゼロの値は、このチャンネルでクライアント接続が許可されないことを示します。デフォルト値は 999 999 999 です。

値が現在実行中のサーバー接続チャンネルのインスタンス数より少ない数まで引き下げられる場合でも、実行中のチャンネルは影響を受けません。しかし、新規インスタンスは、十分な数の既存のインスタンスが実行を終了するまで開始できません。

AMQP チャンネル接続の最大インスタンス数

AMQP チャンネルの場合、この属性は 1 つの AMQP チャンネルで開始可能な同時インスタンスの最大数を指定します。

この属性に設定できる値は 0 から 999 999 999 です。ゼロの値は、このチャンネルでクライアント接続が許可されないことを示します。デフォルト値は 999 999 999 です。

クライアントが接続を試行した場合、接続済みクライアント数が MAXINST に達していると、チャンネルはクローズ・フレームを伴って接続を閉じます。クローズ・フレームには以下のメッセージが含まれます。

```
amqp:resource-limit-exceeded
```

既に接続済みの ID を使ってクライアントが接続した場合 (つまりクライアント・テークオーバーを実行した場合)、接続済みクライアント数が MAXINST に達しているかどうかにかかわらず、テークオーバーが成功します。

追加情報については、[サーバー接続チャンネルの制限](#) を参照してください。

クライアントあたりの最大インスタンス数 (MAXINSTC)

この属性は、サーバー接続チャンネルの、単一クライアントから開始可能な同時インスタンスの最大数を指定します。

この属性に設定できる値は 0 から 999 999 999 です。ゼロの値は、このチャンネルでクライアント接続が許可されないことを示します。デフォルト値は 999 999 999 です。

値が個々のクライアントから現在実行中のサーバー接続チャンネルのインスタンス数より少ない数まで引き下げられる場合でも、実行中のチャンネルは影響を受けません。しかし、それらのクライアントからの新規インスタンスは、十分な数の既存のインスタンスが実行を終了するまで開始できません。

この属性はサーバー接続チャンネルにのみ有効です。

追加情報については、[サーバー接続チャンネルの制限](#)を参照してください。

MAXMSGL (最大メッセージ長)

この属性は、チャンネルで伝送可能な最大メッセージ長を指定します。

Multi マルチプラットフォームでは、0 からキュー・マネージャーの最大メッセージ長までの範囲の値を指定します。詳細は、[ALTER QMGR](#) で説明されている ALTER QMGR コマンドの MAXMSGL パラメーターを参照してください。

z/OS IBM MQ for z/OS では、ゼロ以上 104 857 600 バイト (100 MB) 以下の値を指定してください。

プラットフォームごとに IBM MQ システムの実装環境は異なるため、一部のアプリケーションでメッセージ処理に使用可能なサイズが制限されることがあります。その数値には、システムが滞りなく処理できるサイズが反映されます。チャンネルが開始すると、チャンネルの両端で指定された 2 つの数値のうち小さい方の値が採用されます。

注: チャンネルには最大メッセージ・サイズ 0 を使用できます。これは、サイズがローカル・キュー・マネージャーの最大値に設定されることを意味します。

メッセージにデジタル署名と鍵を追加することで、[Advanced Message Security](#) ではメッセージの長さが増します。

この属性は、すべてのチャンネル・タイプで有効です。

MCANAME (メッセージ・チャンネル・エージェント名)

この属性は、指定する必要があるのがブランクのみで、最大長が 20 文字の場合に予約されます。

MCATYPE (メッセージ・チャンネル・エージェント・タイプ)

この属性は、メッセージ・チャンネル・エージェントをプロセスまたはスレッドとして指定できます。

プロセスとして実行することには、次のような利点があります。

- 各チャンネルが分離しているため、安全性が向上する
- チャンネルごとに固有のジョブ権限を指定できる
- ジョブ・スケジュールを制御できる

スレッドには、次のような利点があります。

- 使用するストレージが少なく済む
- コマンド行からの入力で簡単に構成できる
- 実行速度の向上 - プロセスを開始するようオペレーティング・システムに指示するよりも、スレッドを開始する方が早い

注:送信側、サーバー、要求側のチャンネル・タイプの場合のデフォルトは `process` です。クラスター送信側およびクラスター受信側のチャンネル・タイプの場合のデフォルトは `thread` です。これらのデフォルトはインストール時に変更される可能性があります。

チャンネル定義に `process` を指定すると、`RUNMQCHL` プロセスが開始します。`thread` を指定すると、MCA は `AMQRMPPA` プロセスのスレッド、または `MQNOREMPOOL` が指定されている場合は `RUNMQCHI` プロセスのスレッドで実行されます。インバウンド割り振りを受け取るマシンでは、`RUNMQLSR` を使用する場合、MCA はスレッドとして実行されます。`inetd` を使用する場合は、プロセスとして実行されます。

z/OS IBM MQ for z/OS では、この属性がサポートされているのはチャンネル・タイプがクラスター受信側であるチャンネルだけです。

Multi 他のプラットフォームでは、この属性は以下のチャンネル・タイプで有効です。

- 送信側
- サーバー
- リクエスター
- クラスター送信側
- クラスター受信側

MCAUSER (メッセージ・チャンネル・エージェント・ユーザー ID)

この属性は、MCA が IBM MQ リソースへのアクセスの許可に使用するユーザー ID (ストリング) です。

注:チャンネルの実行に使用するユーザー ID を提供するための代替手段としては、チャンネル認証の記録を使用するという方法があります。チャンネル認証レコードを使用すると、複数の異なる接続で、それぞれ異なる資格情報を使用して、同一のチャンネルを使用することができます。チャンネルで MCAUSER が設定されており、かつチャンネル認証レコードが同じチャンネルに適用されている場合、チャンネル認証レコードが優先されます。チャンネル定義での MCAUSER は、チャンネル認証レコードが `USERSRC(CHANNEL)` を使用する場合にのみ使用されます。

この許可には、(PUT 権限が DEF である場合) 受信側チャンネルまたは要求側チャンネルの宛先キューへのメッセージ書き込みが含まれます。

IBM MQ for Windows の場合、ユーザー ID をドメイン修飾して `user@domain` の形式で表すことができます。この `domain` は、ローカル・システムの Windows システム・ドメインか、またはトラステッド・ドメインのどちらかでなければなりません。

この属性を空白にした場合、MCA はデフォルトのユーザー ID を使用します。詳しくは、[DEFINE CHANNEL](#) を参照してください。

この属性は、以下のチャンネル・タイプで有効です。

- 受信側
- リクエスター
- サーバー接続
- クラスター受信側

LU 6.2 モード名 (MODENAME)

この属性は、LU 6.2 接続に使用します。これは、通信セッションの割り振りが実行されるときに、接続のセッションの特性について追加の定義を提供します。

サイド情報を SNA 通信で使用するときは、モード名は CPI-C 通信サイド・オブジェクトか APPC サイド情報に定義し、この属性は空白のままにしておく必要があります。そうでない場合は、SNA モード名に設定する必要があります。

名前は 1 から 8 文字の英数字でなければなりません。

この属性は、以下のチャンネル・タイプでのみ有効です。

- 送信側
- サーバー
- リクエスター
- クライアント接続
- クラスター送信側
- クラスター受信側

MONCHL (モニター)

この属性は、オンライン・モニター・データの収集を制御します。

指定可能な値は以下のとおりです。

QMGR

オンライン・モニター・データの収集は、キュー・マネージャー・オブジェクトの MONCHL 属性の設定から継承されます。この値がデフォルト値です。

OFF

このチャンネルのオンライン・モニター・データ収集は無効になります。

LOW

パフォーマンスへの影響が最も少ない低いデータ収集率。ただし、表示されるモニター結果は最新になるとは限りません。

MEDIUM

システムのパフォーマンスへの影響が制限される適度なデータ収集率。

HIGH

パフォーマンスに影響する高いデータ収集率。ただし、表示されるモニター結果は最新のものです。

この属性は、以下のチャンネル・タイプで有効です。

- 送信側
- サーバー
- 受信側
- リクエスター
- サーバー接続
- クラスター送信側
- クラスター受信側

モニター・データの詳細については、[キューおよびチャンネルのモニター・データの表示](#)を参照してください。

MRDATA (メッセージ再試行出口ユーザー・データ)

この属性は、チャンネル・メッセージ再試行出口が呼び出されたときにその出口に渡されるデータを指定します。

この属性は、以下のチャンネル・タイプで有効です。

- 受信側
- リクエスター
- クラスター受信側

MREXIT (メッセージ再試行出口名)

この属性は、メッセージ再試行ユーザー出口によって実行されるユーザー出口プログラムの名前を指定します。

有効なメッセージ再試行出口プログラムがない場合は、ブランクにしておきます。

名前の形式および最大長は、[118 ページの『受信出口名 \(RCVEXIT\)』](#)で説明するように、プラットフォームによって異なります。ただし、メッセージ再試行出口は1つしか指定できません。

この属性は、以下のチャンネル・タイプで有効です。

- 受信側
- リクエスター
- クラスター受信側

MRRTY (メッセージ再試行カウント)

この属性は、チャンネルがメッセージを再配信しようとする回数を指定します。

この属性は、メッセージ再試行出口名がブランクの場合にのみ MCA の処置を制御します。出口名がブランクではないときは、MRRTY の値は出口に引き渡されます。ただし、再試行される場合、その回数はこの属性ではなく、出口が制御します。

値の範囲は 0 から 999 999 999 でなければなりません。値 0 は、追加の再試行が行われないことを意味します。デフォルトは 10 です。

この属性は、以下のチャンネル・タイプで有効です。

- 受信側
- リクエスター
- クラスター受信側

メッセージ再試行間隔 (MRTMR)

この属性は、チャンネルが MQPUT 操作を再試行できるようになるまでの最小時間間隔をミリ秒単位で指定します。

この属性は、メッセージ再試行出口名がブランクの場合にのみ MCA の処置を制御します。出口名がブランクではない場合、MRTMR の値は、その出口に渡され、使用されます。ただし、再試行間隔は、この属性ではなく出口が制御します。

値の範囲は 0 から 999 999 999 でなければなりません。値 0 は、再試行が可能になるとただちに再試行されることを意味します (ただし、MRRTY の値が 0 より大きい場合)。デフォルトは 1000 です。

この属性は、以下のチャンネル・タイプで有効です。

- 受信側
- リクエスター
- クラスター受信側

MSGDATA (メッセージ出口ユーザー・データ)

この属性は、チャンネル・メッセージ出口に渡されるユーザー・データを指定します。

一連のメッセージ出口を実行できます。ユーザー・データ長の制限および複数の出口に対する MSGDATA の指定方法は、RCVDATA について示されているものと同じです。[118 ページの『RCVDATA \(受信出口ユーザー・データ\)』](#)を参照してください。

この属性は、以下のチャンネル・タイプで有効です。

- 送信側
- サーバー
- 受信側
- リクエスター
- クラスター送信側
- クラスター受信側

MSGEXIT (メッセージ出口名)

この属性は、チャンネル・メッセージ出口によって実行されるユーザー出口プログラムの名前を指定します。

この属性を、順番に実行されるプログラムの名前のリストにすることができます。有効なチャンネル・メッセージ出口がない場合は、ブランクにしておきます。

この属性の形式および最大長は、118 ページの『受信出口名 (RCVEXIT)』で説明するように、プラットフォームによって異なります。

この属性は、以下のチャンネル・タイプで有効です。

- 送信側
- サーバー
- 受信側
- リクエスター
- クラスター送信側
- クラスター受信側

MQSC キーワード (N-R) のチャンネル属性

MQSC キーワードのチャンネル属性のアルファベット順のリストで、文字 *N* から *R* で始まります。

NETPRTY (ネットワーク接続優先順位)

NETPRTY チャンネル属性では、CLUSRCVR チャンネルの優先順位を指定します。この値は、0 から 9 の範囲でなければなりません。0 が最低、9 が最高の優先度です。

NETPRTY 属性は、あるネットワークを主ネットワークにし、別のネットワークをバックアップ・ネットワークにする場合に使用します。同じランクの 1 組みのチャンネルが存在すると仮定すると、複数のパスが使用可能な場合、クラスター化では最高の優先順位を持つパスを選択します。

NETPRTY チャンネル属性を使用する典型的な例としては、異なるコストまたは速度を持ち、また同じ複数の宛先を接続する 2 つのネットワークを区別するケースです。

注：クラスター受信側チャンネルのターゲット・キュー・マネージャーで、この属性を指定します。一致するクラスター送信側チャンネルで指定したすべてのバランシングは、無視される可能性が高くなります。[クラスター・チャンネル](#)を参照してください。

非持続メッセージ速度 (NPMSPEED)

この属性は、非持続メッセージが送信される速度を指定します。

指定可能な値は以下のとおりです。

NORMAL


チャンネル上の非持続メッセージはトランザクションの中で転送されます。

FAST

チャンネル上の非持続メッセージは、トランザクションの中では転送されません。

デフォルトは FAST です。この方法には、同期点に達するまで待機しないため、非持続メッセージをより迅速に取り出せるという利点があります。欠点は、トランザクションの一部ではないため、伝送障害があった場合やメッセージの転送中にチャンネルが停止した場合にデータが失われることがある点です。[メッセージの安全性](#)を参照してください。

注：

1.  IBM MQ for z/OS のアクティブ・リカバリー・ログの切り替えおよびアーカイブの頻度が予想よりも高い場合、チャンネルを介して送信されるメッセージが非持続メッセージであるのであれば、チャンネルの送信側と受信側の両方で NPMSPEED (FAST) を設定することにより、SYSTEM.CHANNEL.SYNCQ が更新されました。

2. SYSTEM.CHANNEL.SYNCQ への更新に関連して高い CPU 使用率が見られる場合は、NPMSPEED(FAST) を設定して CPU 使用率を大幅に下げることができます。

この属性は、以下のチャンネル・タイプで有効です。

- 送信側
- サーバー
- 受信側
- リクエスター
- クラスター送信側
- クラスター受信側

パスワード (Password)

この属性は、リモート MCA とのセキュア LU 6.2 セッションの開始を試行するときに MCA が使用できるパスワードを指定します。

最大 12 文字の長さのパスワードを指定できます。ただし、最初の 10 文字だけが使用されます。

z/OS IBM MQ for z/OS では、この属性はクライアント接続チャンネルでのみ有効です。

Multi 他のプラットフォームでは、この属性は以下のチャンネル・タイプで有効です。

- 送信側
- サーバー
- リクエスター
- クライアント接続
- クラスター送信側

PORT (ポート番号)

AMQP クライアントへの接続に使用するポート番号を指定します。

AMQP 1.0 接続のデフォルト・ポートは 5672 です。ポート 5672 を既に使用している場合は、異なるポートを指定できます。

PUT 権限 (PUTAUT)

この属性を使用して、MCA で実行されるセキュリティー処理のタイプを指定します。

この属性を使用して、以下の実行時に MCA で行われるセキュリティー処理の種類を選びます。

- 宛先キューへの MQPUT コマンド (メッセージ・チャンネルの場合)、または
- MQI 呼び出し (MQI チャンネルの場合)

z/OS z/OS では、検査されるユーザー ID と検査されるユーザー ID の数は、MQADMIN RACF® クラス hlq.RESLEVEL プロファイルの設定により異なります。hlq.RESLEVEL に対してチャンネル・イニシエーターのユーザー ID が持つアクセスのレベルに応じて、0、1、または 2 個のユーザー ID が検査されます。検査されるユーザー ID の数については、[RESLEVEL およびチャンネル・イニシエーター接続を参照してください](#)。どのユーザー ID が検査されるかについては、[チャンネル・イニシエーターで使用されるユーザー ID を参照してください](#)。

以下のいずれかを選択することができます。

プロセス・セキュリティー。デフォルトの権限 (DEF) とも呼ばれます。

デフォルトのユーザー ID が使用されます。

Multi Multiplatforms では、キューのオープン権限を検査するために使用されるユーザー ID は、メッセージ・チャンネルの受信側で MCA を実行しているプロセスまたはユーザーのユーザー ID です。

z/OS z/OSでは、チェックされるユーザー ID の数に応じて、ネットワークから受信したユーザー ID と、**MCAUSER** から得られたユーザー ID の両方が使用される場合があります。

キューはこのユーザー ID とオープン・オプション **MQOO_SET_ALL_CONTEXT** でオープンされます。

コンテキスト・セキュリティ (CTX)

メッセージに関連付けられたコンテキスト情報からのユーザー ID が、代替ユーザー ID として使用されます。

メッセージ記述子内の *UserIdentifier* はオブジェクト記述子内の *AlternateUserId* フィールドへ転送されます。キューは、オープン・オプション **MQOO_SET_ALL_CONTEXT** と **MQOO_ALTERNATE_USER_AUTHORITY** でオープンされます。

Multi Multiplatforms では、**MQOO_SET_ALL_CONTEXT** および **MQOO_ALTERNATE_USER_AUTHORITY** のキューのオープン権限を検査するために使用されるユーザー ID は、メッセージ・チャンネルの受信側で **MCA** を実行しているプロセスまたはユーザーのユーザー ID です。**MQOO_OUTPUT** のキュー上のオープン権限をチェックするのに使用するユーザー ID は、メッセージ記述子の *UserIdentifier* です。

z/OS z/OSでは、チェックされるユーザー ID の数に応じて、ネットワークから受信したユーザー ID、または **MCAUSER** から得られたユーザー ID だけでなく、メッセージ記述子内のコンテキスト情報からのユーザー ID も使用される場合があります。

コンテキスト・セキュリティ (CTX) はサーバー接続チャンネル上ではサポートされていません。

z/OS **MCA** セキュリティのみ (**ONLYMCA**)

MCAUSER から得られたユーザー ID が使用されます。

キューはオープン・オプション **MQOO_SET_ALL_CONTEXT** でオープンされます。

この値は z/OS にのみ適用されます。

z/OS 代替 **MCA** セキュリティ (**ALTMCA**)

チェックされるユーザー ID の数に応じて、**MCAUSER** から得られたユーザー ID だけでなく、メッセージ記述子内のコンテキスト情報 (*UserIdentifier* フィールド) からのユーザー ID も使用される場合があります。

この値は z/OS にのみ適用されます。

コンテキスト・フィールドとオープン・オプションの詳細については、[コンテキスト情報の制御](#)を参照してください。

セキュリティの詳細については、以下を参照してください。

- [セキュリティ](#)
- **ALW** [AIX, Linux, and Windows でのセキュリティのセットアップ](#)
- **IBM i** [IBM i でのセキュリティのセットアップ](#)
- **z/OS** [z/OS でのセキュリティのセットアップ](#)

この属性は、以下のチャンネル・タイプで有効です。

- 受信側
- リクエスター
- **z/OS** サーバー接続 (z/OS のみ)
- クラスタ受信側

QMNAME (キュー・マネージャー名)

この属性は、IBM MQ MQI client アプリケーションが接続を要求できるキュー・マネージャーまたはキュー・マネージャー・グループの名前を指定します。

この属性は、以下のチャンネル・タイプで有効です。

- ・クライアント接続

QSGDISP (後処理)

z/OS

この属性は、キュー共有グループでのチャンネルの処理を指定します。これは、z/OS でのみ有効です。

値は次のとおりです。

QMGR

チャンネルはコマンドを実行するキュー・マネージャーのページ・セットに定義されます。この値はデフォルトです。

GROUP

チャンネルは共有リポジトリに定義されます。この値は、共有キュー・マネージャー環境が存在する場合にのみ有効です。チャンネルを QSGDISP(GROUP) で定義すると、コマンドの DEFINE CHANNEL(name) NOREPLACE QSGDISP(COPY) が自動的に生成されて、すべてのアクティブなキュー・マネージャーに送信されます。これにより、すべてのアクティブなキュー・マネージャーが、ページ・セット 0 にローカル・コピーを作成します。アクティブでないキュー・マネージャー、または後日キュー共有グループに参加するキュー・マネージャーでは、このコマンドはそのキュー・マネージャーの開始時に生成されます。

COPY

チャンネルはコマンドを実行するキュー・マネージャーのページ・セットに定義されます。その定義は同じ名前の QSGDISP(GROUP) チャンネルからコピーされます。この値は、共有キュー・マネージャー環境が存在する場合にのみ有効です。

この属性は、すべてのチャンネル・タイプで有効です。

RCVDATA (受信出口ユーザー・データ)

この属性は、受信出口に渡されるユーザー・データを指定します。

一連の受信出口を実行できます。一連の出口に対するユーザー・データのストリングは、コンマ、スペース、またはその両方で分離する必要があります。以下に例を示します。

```
RCVDATA(exit1_data exit2_data)
MSGDATA(exit1_data,exit2_data)
SENDDATA(exit1_data, exit2_data)
```

ALW

IBM MQ for UNIX システム、および Windows システムでは、出口名のストリングとユーザー・データのストリングの長さは 500 文字に制限されています。

IBM i

IBM MQ for IBM i では、最大 10 個の出口名を指定でき、それぞれのユーザー・データの長さは、32 文字に制限されています。

z/OS

IBM MQ for z/OS では、それぞれの長さが 32 文字のユーザー・データのストリングを、最大 8 個まで指定できます。

この属性は、すべてのチャンネル・タイプで有効です。

受信出口名 (RCVEXIT)

この属性は、チャンネル受信ユーザー出口で実行されるユーザー出口プログラムの名前を指定します。

この属性を、順番に実行されるプログラムの名前前のリストにすることができます。有効なチャンネル受信ユーザー出口がない場合には、ブランクのままにしておいてください。

この属性の形式および最大長はプラットフォームによって異なります。

- ▶ **z/OS** z/OS では、これはロード・モジュール名で、最大長が 8 文字になります。例外として、クライアント接続チャンネルは最大長が 128 文字です。
- ▶ **IBM i** IBM i では、次の形式です。

```
libname/progname
```

これは CL コマンドに指定するときの形式です。

IBM MQ コマンド (MQSC) に指定するときは、次の形式になります。

```
progname libname
```

ここで、*progname* は最初の 10 文字を使用し、*libname* はその次の 10 文字を使用します (いずれも必要に応じて右側にブランクを埋め込みます)。ストリングの最大長は 20 文字です。

- ▶ **Linux** ▶ **AIX** AIX and Linux では、次の形式です。

```
libraryname(functionname)
```

ストリングの最大長は 40 文字です。

- ▶ **Windows** Windows では、次の形式です。

```
dllname(functionname)
```

この *dllname* は、接尾部 .DLL を付けずに指定します。ストリングの最大長は 40 文字です。

▶ **z/OS** z/OS でのクラスター送信側のチャンネル自動定義中に、チャンネル出口名は z/OS の形式に変換されます。出口名の変換方法を制御したい場合は、チャンネル自動定義出口を作成できます。詳細については、[チャンネル自動定義出口プログラム](#)を参照してください。

受信、送信、またはメッセージ出口プログラム名のリストを指定できます。名前は、コンマ、スペース、またはその両方で分離する必要があります。以下に例を示します。

```
RCVEXIT(exit1 exit2)  
MSGEXIT(exit1,exit2)  
SENDEXIT(exit1, exit2)
```

特定タイプの出口のユーザー・データ出口名および出口ストリングのストリング全長は、500 文字に制限されています。

- ▶ **IBM i** IBM MQ for IBM i では、最大で 10 個までの出口名を指定できます。
- ▶ **z/OS** IBM MQ for z/OS では、最大で 8 個までの出口名を指定できます。

この属性は、すべてのチャンネル・タイプで有効です。

MQSC キーワードのチャンネル属性 (S)

MQSC キーワードのチャンネル属性のアルファベット順リスト。文字は S で始まります。

SCYDATA (セキュリティ出口ユーザー・データ)

この属性は、セキュリティ出口に渡されるユーザー・データを指定します。

最大長は 32 文字です。

この属性は、すべてのチャンネル・タイプで有効です。

セキュリティー出口名 (SCYEXIT)

この属性は、チャンネル・セキュリティー出口によって実行される出口プログラムの名前を指定します。

有効なチャンネル・セキュリティー出口がない場合は、これを空白のままにします。

RCVEXIT の場合のように、名前の形式と最大長はプラットフォームによって異なります。ただし、指定できるセキュリティー出口は 1 つだけです。

セキュリティー出口について詳しくは、[チャンネル・セキュリティー出口プログラム](#)を参照してください。

この属性は、すべてのチャンネル・タイプで有効です。

SENDDATA (送信出口ユーザー・データ)

この属性は、送信出口に渡されるユーザー・データを指定します。

一連の送信出口を実行できます。ユーザー・データ長の制限および複数の出口に対する SENDDATA の指定方法は、RCVDATA について示されているものと同じです。[RCVDATA](#) を参照してください。

この属性は、すべてのチャンネル・タイプで有効です。

SENDEXIT (送信出口名)

この属性は、チャンネル送信出口によって実行される出口プログラムの名前を指定します。

この属性として、順次実行されるプログラムの名前のリストを指定できます。有効なチャンネル送信出口がない場合には、空白のままにしておいてください。

RCVEXIT のように、この属性の形式と最大長はプラットフォームによって異なります。

この属性は、すべてのチャンネル・タイプで有効です。

SEQWRAP (順序番号折り返し)

この属性は、メッセージ・シーケンス番号が 1 で再始動する前に到達する最大数を指定します。

前のメッセージがこの数値を使用している間に数値が再発行されることを防ぐため、この数値には十分大きな値を指定する必要があります。チャンネルの両端は、チャンネルの開始時に同じシーケンス番号のラップ値を保持している必要があります。そうでない場合は、エラーが発生します。

値は、100 から 999 999 999 の間で設定できます。

この属性は、以下のチャンネル・タイプで有効です。

- 送信側
- サーバー
- 受信側
- リクエスト
- クラスター送信側
- クラスター受信側

SHORTRTY (短期再試行カウント)

この属性は、チャンネルがそのパートナーへのセッションの割り振りを試行する最大回数を指定します。

SHORTRTY 属性に設定できる値はゼロから 999 999 999 までです。

チャンネル内で複数の IP アドレスが定義されており、再接続が必要な場合、IBM MQ はチャンネル定義を評価し、接続が正常に確立されるか、すべてのアドレスが試行されるまで、定義された順序で各 IP アドレスへの接続を試みます。

この場合、SHORTRTY は、個々の IP アドレスではなく、チャンネル全体が再接続を試行する合計試行回数に関連します。

初期割り振りの試行に失敗した場合、試行するたびに短期再試行カウントが減り、チャンネルは、**short retry interval** 属性で定義された間隔で残りの回数を再試行します。それでも失敗した場合、長期再試行カウントの回数を、各試行間の長期再試行間隔の間隔で再試行します。それでも割り振りが成功しない場合には、チャンネルは停止します。

z/OS z/OS では、チャンネルの最大数 (MAXCHL) を超えると、チャンネルが再試行を開始できなくなります。

Multi マルチプラットフォームでは、再試行を行うためには、チャンネル・イニシエーターが実行されている必要があります。チャンネル・イニシエーターは、チャンネルが使用している伝送キューの定義に指定されている開始キューをモニターしている必要があります。

チャンネル・イニシエーター (z/OS の場合) または、チャンネルの再試行中にチャンネル (マルチプラットフォームの場合) が停止すると、チャンネル・イニシエーターまたはチャンネルが再始動したとき、またはメッセージが送信側チャンネルに正常に書き込まれたときに、短期再試行カウント および 長期再試行カウント がリセットされます。ただし、チャンネル・イニシエーター (z/OS の場合) または、キュー・マネージャー (Multiplatforms) がシャットダウンされて再始動された場合、短期再試行カウント および 長期再試行カウント はリセットされません。チャンネルは、キュー・マネージャーの再始動またはメッセージの送信の前に保持していた再試行カウント値を保持します。

Multi On マルチプラットフォーム:

1. チャンネルが RETRYING 状態から RUNNING 状態になる際、短期再試行カウント および 長期再試行カウント はすぐにはリセットされません。チャンネルが RUNNING 状態になったあとの最初のメッセージがチャンネルを通して正常に流れたときにだけリセットされます。つまり、ローカルのチャンネルが、相手側に送信したメッセージの数を確認した時点でリセットされます。
2. 短期再試行カウント および 長期再試行カウント は、チャンネルが再始動されるとリセットされます。

この属性は、以下のチャンネル・タイプで有効です。

- 送信側
- サーバー
- クラスター送信側
- クラスター受信側

SHORTTMR (短期再試行間隔)

この属性は、短期間の再試行モード中に、接続の確立を再試行するまでチャンネルが待機する間隔の概数を秒単位で指定します。

チャンネルがアクティブになるのを待機する必要がある場合、再試行間隔が延長されることがあります。

この属性に設定できる値は 0 から 999 999 です。

チャンネル内で複数の IP アドレスが定義されており、再接続が必要な場合、IBM MQ はチャンネル定義を評価し、接続が正常に確立されるか、すべてのアドレスが試行されるまで、定義された順序で各 IP アドレスへの接続を試みます。

この場合、SHORTTMR は、個々の IP アドレスではなく、チャンネル全体が接続プロセスの再開を待機する時間に関連します。

この属性は、以下のチャンネル・タイプで有効です。

- 送信側
- サーバー

- クラスター送信側
- クラスター受信側

SPLPROT (セキュリティー・ポリシー保護)

z/OS

この属性は、AMS がアクティブで該当ポリシーが存在する場合にサーバー間メッセージング・チャンネル・エージェントでメッセージ保護をどう処理するかを指定します。

この属性は以下の値に設定できます。

PASSTHRU

送信側、サーバー、受信側、およびリクエスター・チャンネルの場合

REMOVE

送信側およびサーバー・チャンネルの場合

ASPOLICY

受信側およびリクエスター・チャンネルの場合

この属性は、以下のチャンネル・タイプで有効です。

- 送信側
- サーバー
- 受信側
- リクエスター

SSLCAUTH (SSL クライアント認証)

SSLCAUTH 属性は、チャンネルが TLS クライアントから TLS 証明書を受信して認証する必要があるかどうかを指定します。

SSLCAUTH はオプションの属性です。この属性で可能な値は以下のとおりです。

オプション

ピア TLS クライアントが証明書を送信する場合、証明書は通常どおりに処理されますが、証明書が送信されなくても認証は失敗しません。

REQUIRED

TLS クライアントが証明書を送信しない場合、認証は失敗します。

デフォルト値は **REQUIRED** です。

非 TLS チャンネル定義で **SSLCAUTH** の値を指定できます。つまり、**SSLCIPH** 属性が欠落しているかブランクになっているチャンネル定義です。

SSLCAUTH について詳しくは、>[DEFINE CHANNEL \(MQTT\)](#) および [セキュリティー](#) を参照してください。

SSLCAUTH 属性は、チャンネル開始フローを受信できるすべてのチャンネル・タイプに有効です (送信側チャンネルは除く)。この属性は、以下のチャンネル・タイプで有効です。

- サーバー
- 受信側
- リクエスター
- サーバー接続
- クラスター受信側

SSLCIPH (SSL 暗号仕様)

SSLCIPH 属性は、TLS 接続用の別名または単一の名前付きの CipherSpec を指定します。

各 IBM MQ チャネル定義には、**SSLCIPH** 属性が含まれています。この値は、最大長が 32 文字のストリングです。

SSLCIPH 属性は、トランスポート・タイプ (**TRPTYPE**) が TCP のチャネルにのみ有効です。**TRPTYPE** が TCP でない場合、データは無視され、エラー・メッセージは出されません。

注:

- **SSLCIPH** 属性は空白値にすることができます。この場合、TLS を使用しません。チャネルの片方の **SSLCIPH** 属性が空白値の場合は、もう片方の **SSLCIPH** 属性も空白値である必要があります。

▶ **Multi** **SecureCommsOnly** が有効になっている場合、プレーン・テキスト通信はサポートされず、チャネルを開始できません。

- また、**SSLCIPH** に非空白値が含まれている場合、値は別名または名前付きの CipherSpec のいずれかです。チャネルは、チャネルの両端でサポートされる最も強力な CipherSpec をネゴシエーションします。
- フルマネージドの .NET クライアントでは、特殊値 *NEGOTIATE を指定できます。このオプションにより、チャネルは .NET フレームワークでサポートされている最新バージョンのプロトコルを選択し、サーバーでサポートされる CipherSpec とネゴシエーションすることができます。

SSLCIPH 属性は、トランスポート・タイプ (**TRPTYPE**) が TCP のチャネルにのみ有効です。**TRPTYPE** が TCP でない場合、データは無視され、エラー・メッセージは出されません。

SSLCIPH について詳しくは、[DEFINE CHANNEL](#) および [CipherSpec](#) の指定を参照してください。

SSLPEER (SSL ピア)

SSLPEER 属性は、IBM MQ チャネルの相手側にあるピア・キュー・マネージャーまたはピア・クライアントからの証明書の識別名 (DN) を検査するために使用されます。

注: TLS サブジェクト識別名との突き合わせによってチャネルへの接続を制限する別の方法は、チャネル認証レコードを使用することです。チャネル認証レコードを使用すると、TLS のサブジェクト識別名のさまざまなパターンを同じチャネルに適用することができます。チャネルで **SSLPEER** が設定されており、かつチャネル認証レコードが同じチャネルに適用されている場合、接続するには、インバウンド証明書が両方のパターンと一致する必要があります。

ピアから受信した DN が **SSLPEER** 値と一致しない場合、チャネルは開始しません。

SSLPEER はオプションの属性です。値を指定しない場合は、チャネル開始時にピア DN は検査されません。

SSLPEER 属性の最大長は、プラットフォームによって異なります。

- ▶ **z/OS** z/OS では、属性の最大長は 256 バイトです。
- ▶ **Multi** それ以外のすべてのプラットフォームでは、1024 バイトです。

チャネル認証レコードにより、**SSLPEER** の使用時に柔軟性が大幅に向上し、すべてのプラットフォームで最大長の 1024 バイトがサポートされます。

SSLPEER 属性値の検査は、プラットフォームによって異なります。

- ▶ **z/OS** z/OS では、使用される属性値は検査されません。正しくない値を入力すると、チャネルは開始時に失敗し、チャネルの両端のエラー・ログにエラー・メッセージが書き込まれます。チャネル SSL エラー・イベントも、チャネルの両端で生成されます。
- ▶ **Multi** **SSLPEER** をサポートする [マルチプラットフォーム](#) では、ストリングの妥当性が最初に入力されたときに検査されます。

非 TLS チャネル定義の **SSLPEER** には、**SSLCIPH** 属性が欠落しているか空白になっている値を指定できます。これは、デバッグを行うためにいったん TLS パラメーターをクリアし、あとで再入力しなくてもよいように、TLS を一時的に使用不可にするために使用できます。

SSLPEER 属性は、すべてのチャンネル・タイプに有効です。

SSLPEER の使用について詳しくは、[SET CHLAUTH](#)、[Securing](#)、および [チャンネル認証レコード](#) を参照してください。

STATCHL (チャンネル統計)

この属性は、チャンネルの統計データの収集方法を制御します。

指定できる値は以下のとおりです。

QMGR

このチャンネルの統計データ収集は、キュー・マネージャー属性 **STATCHL** の設定に基づいて行われます。この値がデフォルト値です。

OFF

このチャンネルの統計データ収集は、無効になります。

LOW

このチャンネルの統計データ収集は、低いデータ収集率で有効になります。


MEDIUM

このチャンネルの統計データ収集は、中程度のデータ収集率で有効になります。

HIGH

このチャンネルの統計データ収集は、高いデータ収集率で有効になります。

チャンネル統計の詳細については、[モニター・リファレンス](#) を参照してください。

 **z/OS** の上 z/OS システムでは、このパラメータを有効にすると、選択した値に関係なく、統計データの収集がオンになるだけです。LOW、MEDIUM、または HIGH のどれを指定しても、結果に違いはありません。チャンネル・アカウント・レコードを収集するには、このパラメータを有効にしなければなりません。

この属性は、以下のチャンネル・タイプで有効です。

- 送信側
- サーバー
- 受信側
- リクエスター
- クラスター送信側
- クラスター受信側

MQSC キーワード (T-Z) のチャンネル属性

MQSC キーワードのチャンネル属性のアルファベット順のリスト。文字 T から Z で始まります。

TPNAME (LU 6.2 トランザクション・プログラム名)

この属性は、LU 6.2 接続に使用します。これは、リンクの末端で実行するトランザクション・プログラム (MCA) の名前または総称名です。

サイド情報を SNA 通信で使用するときは、トランザクション・プログラム名は CPI-C 通信サイド・オブジェクトか APPC サイド情報に定義し、この属性はブランクのままにしておく必要があります。そうしないと、送信側チャンネルおよびリクエスター・チャンネルでこの名前が必要になります。

この名前は最大 64 文字まで許容されます。

CONNNAME にサイド・オブジェクト名が指定されていない場合 (この場合ブランクに設定される)、この名前は SNA トランザクション・プログラム名に設定されます。その代わりに、実際の名前は CPI-C コミュニケーション・サイド・オブジェクト、つまり APPC サイド情報データ・セットから取得されます。

この情報は、プラットフォームごとに異なる方法で設定されます。ご使用のプラットフォームにおける通信の設定方法について詳しくは、[分散キューイングの構成](#) を参照してください。

この属性は、以下のチャンネル・タイプで有効です。

- 送信側
- サーバー
- リクエスター
- クライアント接続
- クラスター送信側
- クラスター受信側

TPROOT (トピック・ルート)

この属性は AMQP チャンネルのトピック・ルートを指定します。

TPROOT 属性を使用して、AMQP チャンネルのトピック・ルートを指定することができます。この属性を使用すると、MQ Light アプリケーションがキュー・マネージャーにデプロイされたとき、他のアプリケーションで使用されているトピック・ツリー領域との間でメッセージのパブリッシュやサブスクライブが行われないようにすることができます。

TPROOT のデフォルト値は SYSTEM.BASE.TOPIC です。この値を設定した場合、AMQP クライアントがパブリッシュまたはサブスクライブに使用するトピック・ストリングに接頭部が付かず、クライアントは他の MQ パブリッシュ/サブスクライブ・アプリケーションとの間でメッセージを交換できます。トピック接頭部の下で AMQP クライアントにパブリッシュ/サブスクライブさせるには、まず、トピック・ストリングを適切な接頭部に設定して MQ トピック・オブジェクトを作成した後、AMQP チャンネル TPROOT 属性の値を、既に作成した MQ トピック・オブジェクトの名前に変更します。以下の例は、AMQP チャンネル MYAMQP 用に APPGROUP1.BASE.TOPIC に設定されたトピック・ルートを示しています。

```
DEFINE CHANNEL(MYAMQP) CHLTYPE(AMQP) TPROOT(APPGROUP1.BASE.TOPIC) PORT(5673)
```

注：TPROOT 属性値またはその基礎となるトピック・ストリングが変更された場合、既存の AMQP トピックとそのメッセージが孤立する可能性があります。

TRPTYPE (トランスポート・タイプ)

この属性は、使用されるトランスポート・タイプを指定します。

指定できる値は以下のとおりです。

値	トランスポート・タイプ
LU62	LU 6.2
TCP	TCP/IP
NETBIOS	NetBIOS 125 ページの『1』
SPX	SPX 125 ページの『1』

注：

1. Windows 上で使用。z/OS でも、Windows で使用するクライアント接続チャンネルを定義するために使用できます。

この属性は、すべてのチャンネル・タイプで有効ですが、応答するメッセージ・チャンネル・エージェントによって無視されます。

クライアント ID の使用 (USECLTID)

AMQP チャンネルの接続にクライアント ID を使用するかどうかを指定します。Yes または No に設定します。

USEDLQ (送達不能キューの使用)

この属性は、チャンネルでメッセージが配信できない場合に、送達不能キュー (または未配布メッセージ・キュー) を使用するかどうかを判別します。

指定可能な値は以下のとおりです。

NO

チャンネルによって送信できないメッセージは、失敗したものとして扱われます。NPMSPEED の設定に従って、チャンネルがこれらのメッセージを破棄するか、チャンネルが終了します。

YES (デフォルト)

キュー・マネージャー DEADQ の属性に送達不能キューの名前が指定されている場合は、そのキューが使用されます。指定されていない場合は、NO が指定されたときの動作になります。


ユーザー ID (USERID)


この属性は、リモート MCA との保護 SNA セッションの開始を試みるときに MCA が使用するユーザー ID を指定します。

20 文字のタスク・ユーザー ID を指定できます。

受信側でパスワードが暗号形式で保持され、LU 6.2 ソフトウェアが別の暗号方式を使用している場合、チャンネルを開始しようとする、セキュリティの詳細が無効なために失敗します。この失敗は、受信側の SNA 構成を次のいずれかに変更することによって回避できます。

- パスワード置換をオフにする。
- セキュリティー・ユーザー ID およびパスワードを定義する。

 IBM MQ for z/OS では、この属性はクライアント接続チャンネルでのみ有効です。

 他のプラットフォームでは、この属性は以下のチャンネル・タイプで有効です。

- 送信側
- サーバー
- リクエスター
- クライアント接続
- クラスター送信側

XMITQ (伝送キュー名)

この属性は、メッセージの検索元の伝送キューの名前を指定します。

チャンネルの相手側のキュー・マネージャーに対応した、この送信側チャンネルまたはサーバー・チャンネルと関連付けられる伝送キューの名前を指定します。伝送キューに、リモート・エンドのキュー・マネージャーと同じ名前を指定できます。

この属性は、タイプが送信側またはサーバーのチャンネルには必須であり、他のチャンネル・タイプには無効です。

IBM MQ クラスターのコマンドおよび属性

MQSC および PCF クラスター・コマンドを使用して、クラスターをリフレッシュまたはリセットしたり、クラスター・キュー・マネージャーを表示、再開、または中断したりすることができます。さらに、チャンネル、キュー、およびキュー・マネージャーを定義する MQSC コマンドおよび PCF コマンドには、クラスターに適用される属性があります。これらの属性の一部は、クラスター・ワークロード管理アルゴリズムによって使用されます。

MQSC コマンド

MQSC コマンドは、システム管理者がコマンド・コンソールで入力するとおりに示されています。しかし、これが唯一の入力方法ではありません。プラットフォームに応じて、他の複数の方法があります。例を以下に示します。

- ▶ **IBM i** IBM MQ for IBM i では、**WRKMQM** のオプション 26 から MQSC コマンドを対話式に実行できます。また、CL コマンドを使用することも、MQSC コマンドをファイルに保管しておいて、**STRMQMMQSC** CL コマンドを使用することもできます。
- ▶ **z/OS** z/OS では、**CSQUTIL** ユーティリティの **COMMAND** 機能、操作パネルと制御パネル、または z/OS コンソールを使用できます。
- 他のすべてのプラットフォームでは、コマンドをファイルに保管して **runmqsc** を使用できます。

MQSC コマンドでは、**CLUSTER** 属性を使用して指定できるクラスター名の長さは最大 48 文字です。

CLUSNL 属性を使用して指定されたクラスター名のリストには、最大 256 個の名前を含めることができます。クラスター名前リストを作成するには、**DEFINE NAMELIST** コマンドを使用します。

IBM MQ Explorer

IBM MQ Explorer GUI は、IBM WebSphere MQ for z/OS 6.0 以降のリポジトリ・キュー・マネージャーを持つクラスターを管理できます。別個のシステムで追加のリポジトリを指名する必要はありません。以前のバージョンの IBM MQ for z/OS では、IBM MQ Explorer は、リポジトリ・キュー・マネージャーを含むクラスターを管理できません。そのため、IBM MQ Explorer が管理できるシステム上で、さらに別のリポジトリを指定する必要があります。

IBM MQ for Windows および IBM MQ for Linux では、IBM MQ Explorer を使用してクラスターを操作することもできます。また、スタンドアロンの IBM MQ Explorer ・クライアントを使用することもできます。

IBM MQ Explorer を使用すると、クラスター・キューを表示して、クラスター送信側チャンネルとクラスター受信側チャンネルの状況を照会できます。IBM MQ Explorer には 2 種類のウィザードがあり、以下の作業のガイドとして使用できます。

- クラスターを作成する
- 独立キュー・マネージャーをクラスターに結合する

クラスターを処理するための MQSC コマンドの、PCF での同等のコマンド

MQSC コマンド	同等の PCF コマンド
DISPLAY CLUSQMGR	MQCMD_INQUIRE_CLUSTER_Q_MGR
REFRESH CLUSTER	MQCMD_REFRESH_CLUSTER
RESET CLUSTER	MQCMD_RESET_CLUSTER
RESUME QMGR	MQCMD_RESUME_Q_MGR_CLUSTER
SUSPEND QMGR	MQCMD_SUSPEND_Q_MGR_CLUSTER

関連情報

[クラスター化: REFRESH CLUSTER の使用に関するベスト・プラクティス](#)

チャンネル定義コマンドで使用可能なクラスター属性

チャンネル定義コマンドで指定できるクラスター属性

DEFINE CHANNEL、**ALTER CHANNEL**、および **DISPLAY CHANNEL** コマンドには、クラスター用の 2 つの特定 **CHLTYPE** パラメーター、**CLUSRCVR** および **CLUSSDR** があります。クラスター受信側チャンネルを定義するには、**DEFINE CHANNEL** コマンドで **CHLTYPE (CLUSRCVR)** を指定します。クラスター受信側チャンネル定義の属性の多くは、受信側チャンネル定義または送信側チャンネルの定義の属性と同じです。クラスター

送信側チャンネルを定義するには、DEFINE CHANNEL コマンドで CHLTYPE (CLUSDR) と指定し、送信側チャンネルの定義に使用するのと同じ属性の多くを指定します。

クラスター送信側チャンネルを定義する際に、完全リポジトリ・キュー・マネージャーの名前を指定する必要はなくなりました。クラスター内のチャンネルに使用する命名規則を知っている場合、+QMNAME+ 構造を使用して CLUSSDR 定義を作成できます。+QMNAME+ 構造は z/OS ではサポートされていません。接続後に IBM MQ はチャンネルの名前を変更し、+QMNAME+ を正しい完全リポジトリ・キュー・マネージャー名に置換します。結果として得られるチャンネル名は 20 文字に切り捨てられます。

命名規則の詳細については、[クラスターの命名規則](#)を参照してください。

この手法が機能するのは、チャンネルの命名の規則にキュー・マネージャーの名前が含まれている場合だけです。例えば、CLUSTER1 というクラスターの QM1 という完全リポジトリ・キュー・マネージャーを、CLUSTER1.QM1.ALPHA というクラスター受信側チャンネルで定義するとします。他のキュー・マネージャーはすべて、このキュー・マネージャーに対して、チャンネル名である CLUSTER1.+QMNAME+.ALPHA を使用して、クラスター送信側チャンネルを定義することができます。

同じ命名規則をすべてのチャンネルに使用する場合、一度に存在できる +QMNAME+ 定義は 1 つだけであることに注意してください。

DEFINE CHANNEL コマンドと ALTER CHANNEL コマンドの次の属性は、クラスター・チャンネルに固有です。

CLUSTER

CLUSTER 属性は、このチャンネルが関連付けられるクラスターの名前を指定します。または、CLUSNL 属性を使用します。

CLUSNL

CLUSNL 属性はクラスター名の名前リストを指定します。

NETPRTY

クラスター受信側だけです。

NETPRTY 属性は、チャンネルのネットワーク優先度を指定します。NETPRTY はワークロード管理ルーチンの働きを助けます。宛先への可能な経路が 2 つ以上ある場合、ワークロード管理ルーチンは優先度が最も高いチャンネルへの経路を選択します。

CLWLPRTY

CLWLPRTY パラメーターは、ワークロード管理のために、同じ宛先へのチャンネルに優先度係数を適用します。このパラメーターは、クラスター・ワークロードを分散するために、チャンネルの優先度を指定します。値の範囲はゼロ (最低の優先度) から 9 (最高の優先度) でなければなりません。

CLWLRANK

CLWLRANK パラメーターは、ワークロード管理のために、チャンネルにランキング係数を適用します。このパラメーターは、クラスター・ワークロードを分散するために、チャンネルのランクを指定します。値の範囲はゼロ (最低ランク) から 9 (最高ランク) でなければなりません。

CLWLWGHT

CLWLWGHT パラメーターは、ワークロード管理のためにチャンネルに加重係数を適用します。CLWLWGHT は、そのチャンネルを経由して送信されるメッセージの比率を制御できるように、チャンネルを重み付けします。クラスター・ワークロード・アルゴリズムは CLWLWGHT を使用して、特定のチャンネルを介してより多くのメッセージを送信できるように宛先の選択を偏らせます。デフォルトでは、すべてのチャンネルの重み属性が同じデフォルト値です。重み属性では、強力な UNIX マシン上のチャンネルには、小さいデスクトップ PC 上の別のチャンネルより大きい重みを割り振ることができます。より大きい重みを割り振ることにより、クラスター・ワークロード・アルゴリズムは、メッセージの宛先として、PC より UNIX マシンをより頻繁に選択します。

CONNAME

クラスター受信側チャンネル定義に指定された CONNAME は、キュー・マネージャーのネットワーク・アドレスを識別するためにクラスター全体で使用されます。CONNAME パラメーターの値は、IBM MQ クラスター全体を通じて解決されるため、注意して選択してください。総称名は使用できません。クラスター受信側チャンネルで指定された値は、対応するクラスター送信側チャンネルに指定されたすべての値に優先することに注意してください。

DEFINE CHANNEL コマンドおよび ALTER CHANNEL コマンドのこれらの属性は、DISPLAY CHANNEL コマンドにも適用されます。

注: 自動的に定義されたクラスター送信側チャンネルの属性は、受信側のキュー・マネージャーの対応するクラスター受信側チャンネル定義の属性を基にして設定されます。手動で定義されたクラスター送信側チャンネルがあっても、その属性は対応するクラスター受信側の定義に指定された属性に合わせて自動的に修正されます。例えば、ポート番号を指定せずに CONNAME パラメーターに CLUSRCVR を定義することができます。ただし、ポート番号を指定する CLUSSDR を手操作で定義することになりますので注意してください。手操作で定義した CLUSSDR が自動定義されたもので置き換えられると、ポート番号 (CLUSRCVR から設定された) はブランクになります。デフォルトのポート番号が使用され、チャンネルには障害が発生します。

注: DISPLAY CHANNEL コマンドでは、自動定義チャンネルは表示されません。ただし、DISPLAY CLUSQMGR コマンドを使用して、自動定義クラスター送信側チャンネルの属性を調べることができます。

DISPLAY CHSTATUS コマンドを使用すると、クラスター送信側チャンネルまたはクラスター受信側チャンネルの状況を表示できます。このコマンドでは、手作業で定義したチャンネルと自動的に定義されたチャンネルの両方の状況が表示されます。

同等の PCF は、MQCMD_CHANGE_CHANNEL、MQCMD_COPY_CHANNEL、MQCMD_CREATE_CHANNEL、および MQCMD_INQUIRE_CHANNEL です。

CLUSRCVR 定義で CONNAME 値を省略する

状況によっては、CLUSRCVR 定義で CONNAME 値を省略することができます。z/OS では、CONNAME 値を省略してはなりません。

Multi マルチプラットフォームでは、クラスター受信側チャンネルの TCP/IP 接続名パラメーターはオプションです。接続名をブランクにすると、IBM MQ はデフォルト・ポートを想定し、システムの現行 IP アドレスを使用して接続名を自動的に生成します。デフォルト・ポート番号をオーバーライドしても、システムの現行 IP アドレスを引き続き使用できます。各接続名について、IP 名をブランクにして、次のように括弧で囲んだポート番号を指定してください。

(1415)

生成される **CONNAME** は常にドット 10 進 (IPv4) 形式または 16 進 (IPv6) 形式であり、英数字の DNS ホスト名の形式ではありません。

この機能は、動的ホスト構成プロトコル (DHCP) を使用するマシンがある場合に役立ちます。CLUSRCVR チャンネル上の CONNAME の値を指定しない場合、CLUSRCVR 定義に変更を加える必要はありません。DHCP により新しい IP アドレスが割り振られます。

CLUSRCVR 定義で CONNAME にブランクを指定した場合は、IBM MQ はシステムの IP アドレスから CONNAME を生成します。生成された CONNAME だけがリポジトリに格納されます。クラスター内の他のキュー・マネージャーは、CONNAME が当初はブランクだったことを知りません。

DISPLAY CLUSQMGR コマンドを実行すると、生成された CONNAME が表示されます。しかし、ローカル・キュー・マネージャーから DISPLAY CHANNEL コマンドを実行する場合には、CONNAME がブランクとして表示されます。

キュー・マネージャーが停止し、別の IP アドレスで再始動した場合には、DHCP のために、IBM MQ は CONNAME を再生成し、それに従ってリポジトリを更新します。

関連概念

クラスターでのワークロード・バランシング

1つのクラスターに同じキューのインスタンスが複数含まれる場合、IBM MQ はメッセージを送付するキュー・マネージャーを選択します。また、クラスター・ワークロード管理アルゴリズムと、クラスター・ワークロード固有のいくつかの属性を使用して、使用するのに最適なキュー・マネージャーを判別します。

z/OS Asynchronous behavior of CLUSTER commands on z/OS

The command issuer of a cluster command on z/OS receives confirmation a command has been sent, but not that it has completed successfully.

関連資料

キュー定義コマンドで使用可能なクラスター属性
キュー定義コマンドに指定できるクラスター属性です。

キュー・マネージャー定義コマンドで使用可能なクラスター属性
キュー・マネージャー定義コマンドに指定できるクラスター属性です。

DISPLAY CLUSQMGR

DISPLAY CLUSQMGR コマンドは、クラスター内にあるキュー・マネージャーに関するクラスター情報を表示する場合に使用します。

REFRESH CLUSTER

REFRESH CLUSTER コマンドは、クラスターに関するローカルに保持されたすべての情報を廃棄する場合にキュー・マネージャーから発行します。例外的な状況を除いて、このコマンドを使用する必要はほとんどありません。

RESET CLUSTER: クラスターからキュー・マネージャーを強制的に除去する

RESET CLUSTER コマンドは、例外的な状況にあるクラスターからキュー・マネージャーを強制的に削除する場合に使用します。

SUSPEND QMGR、RESUME QMGR、およびクラスター

SUSPEND QMGR および RESUME QMGR コマンドは、このキュー・マネージャーへのインバウンド・クラスター・アクティビティを一時的に (例えば、このキュー・マネージャーの保守を行う前に) 削減し、後で復元するために使用します。

141 ページの『クラスター・ワークロード・バランシング-チャンネル属性』

クラスター・ワークロード・バランシングで使用されるチャンネル属性のアルファベット順リスト。

キュー定義コマンドで使用可能なクラスター属性

キュー定義コマンドに指定できるクラスター属性です。

コマンド DEFINE QLOCAL、DEFINE QREMOTE、および DEFINE QALIAS

DEFINE QLOCAL、DEFINE QREMOTE、および DEFINE QALIAS コマンドのクラスター属性、および同等の3つの ALTER コマンドは、以下のとおりです。

CLUSTER

キューが属するクラスターの名前を指定します。

CLUSNL

クラスター名の名前リストを指定します。

DEFBIND

アプリケーションで MQOPEN 呼び出しに MQOO_BIND_AS_Q_DEF を指定している場合に使用するバインディングを指定します。この属性のオプションは、以下のとおりです。

- キューがオープンされる際にキュー・ハンドルをクラスター・キューの特定のインスタンスにバインドするには、DEFBIND(OPEN) を指定します。DEFBIND(OPEN) がこの属性のデフォルトです。
- キュー・ハンドルがクラスター・キューのすべてのインスタンスと結合しないようにするには、DEFBIND(NOTFIXED) を指定します。
- メッセージのグループをすべて同じ宛先インスタンスに割り振るようにアプリケーションで要求できるようにするには、DEFBIND(GROUP) を指定します。

同じ名前の複数のキューがキュー・マネージャー・クラスター内で公示されている場合、アプリケーションでは、このアプリケーションから単一インスタンスにすべてのメッセージを送信する (MQOO_BIND_ON_OPEN) か、ワークロード管理アルゴリズムでメッセージごとに最も適切な宛先の選択を可能にする (MQOO_BIND_NOT_FIXED) か、またはアプリケーションでメッセージのグループをすべて同じ宛先インスタンスに割り当てるように要求することを可能にする (MQOO_BIND_ON_GROUP)

かを選択することができます。ワークロード・バランシングは、メッセージのグループ間を再度実行します(キューの MQCLOSE および MQOPEN を要求しないで)。

キュー定義に DEFBIND を指定する場合、MQBND_BIND_ON_OPEN、MQBND_BIND_NOT_FIXED、または MQBND_BIND_ON_GROUP のいずれかの属性がキューに定義されます。どちらか MQBND_BIND_ON_OPEN または MQBND_BIND_ON_GROUP クラスターを含むグループを使用する場合は指定する必要があります。

DEFBIND 属性は、同じクラスター・キューのすべてのインスタンスで同じ値に設定する必要があります。

CLWLRANK

ワークロード管理のためにランキング因子をキューに適用します。CLWLRANK パラメーターは、モデル・キューではサポートされません。クラスター・ワークロード・アルゴリズムは、ランクが最も高い宛先キューを選択します。デフォルトでは、すべてのキューの CLWLRANK がゼロに設定されます。

最終宛先が異なるクラスター上のキュー・マネージャーの場合、隣接するクラスターの交差ですべての中間ゲートウェイ・キュー・マネージャーのランクを設定できます。ランク付けされた中間キュー・マネージャーに関し、クラスター・ワークロード・アルゴリズムでは、最終宛先により近い宛先キュー・マネージャーを正しく選択します。

同じロジックが別名キューに適用されます。ランクの選択は、チャンネル状況の検査前に行われるため、アクセス不可能なキュー・マネージャーでも選択できます。これには、(優先順位を指定する場合のように)2つの有効な宛先の中から選択するのではなく、メッセージをネットワークを介して経路指定できるという効果があります。そのため、ランクが示す場所でチャンネルが開始されていない場合、メッセージは次に高いランクに経路指定されません。ただし、チャンネルがその宛先で使用可能になるまで待機します(メッセージは伝送キューに保持されます)。

CLWLPRTY

ワークロード管理のために優先順位因子をキューに適用します。クラスター・ワークロード・アルゴリズムは、優先順位が最も高い宛先キューを選択します。デフォルトでは、すべてのキューの優先順位がゼロに設定されます。

有効な宛先キューが2つある場合は、この属性を使用して、一方の宛先をもう一方の宛先にフェイルオーバーできます。優先順位の選択は、チャンネル状況の検査後に行われます。すべてのメッセージは、優先順位が最も高いキューに送信されます。ただし、その宛先へのチャンネルの状況が、他の宛先へのチャンネルの状況ほど良好ではない場合を除きます。つまり、アクセスできる可能性が最も高い宛先のみを選択できます。これは、どれも使用可能である複数の宛先の間で優先順位付けを行えるという効果があります。

CLWLUSEQ

あるキューの MQPUT 操作の動作を指定します。このパラメーターでは、ターゲット・キューにローカル・インスタンスがあり、少なくとも1つのリモート・クラスター・インスタンスがある場合の MQPUT 操作の動作を指定します(MQPUT がクラスター・チャンネルから発信されている場合を除く)。このパラメーターは、ローカル・キューにのみ有効です。

指定可能な値は、QMGR(この動作は、キュー・マネージャー定義の CLWLUSEQ パラメーターで指定されたとおりになります)、ANY(キュー・マネージャーは、ワークロードを分散するために、ローカル・キューをクラスター・キューの別のインスタンスとして処理します)、LOCAL です(ローカル・キューが書き込み可能であれば、ローカル・キューが MQPUT 操作の唯一の宛先です)。MQPUT の動作は、クラスター・ワークロード管理アルゴリズムによって異なります。

コマンド DISPLAY QUEUE および DISPLAY QCLUSTER

DEFINE QLOCAL、DEFINE QREMOTE、および DEFINE QALIAS コマンドの属性は、DISPLAY QUEUE コマンドにも適用されます。

クラスター・キューに関する情報を表示するには、キュー・タイプの QCLUSTER またはキーワードの CLUSINFO を DISPLAY QUEUE コマンドに指定するか、あるいは DISPLAY QCLUSTER コマンドを使用します。

DISPLAY QUEUE コマンドと DISPLAY QCLUSTER コマンドは、キューをホストするキュー・マネージャーの名前(複数のキュー・インスタンスがある場合は、すべてのキュー・マネージャーの名前)を返します。

また、キューのホストである各キュー・マネージャーのシステム名、キューのタイプ、および定義がローカル・キュー・マネージャーに使用可能になった日時を返します。この情報は、CLUSQMGR、QMID、CLUSQT、CLUSDATE、および CLUSTIME の各属性を使用して返されます。

キュー・マネージャーのシステム名 (QMID) は、キュー・マネージャーに対してシステムによって生成される固有の名前です。

共有キューでもあるクラスター・キューを定義できます。次のような例を挙げてみます。z/OS では、次のように定義できます。

```
DEFINE QLOCAL(MYQUEUE) CLUSTER(MYCLUSTER) QSGDISP(SHARED) CFSTRUCT(STRUCTURE)
```

同等の PCF は、MQCMD_CHANGE_Q、MQCMD_COPY_Q、MQCMD_CREATE_Q、および MQCMD_INQUIRE_Q です。

関連概念

クラスターでのワークロード・バランシング

1つのクラスターに同じキューのインスタンスが複数含まれる場合、IBM MQ はメッセージを送付するキュー・マネージャーを選択します。また、クラスター・ワークロード管理アルゴリズムと、クラスター・ワークロード固有のいくつかの属性を使用して、使用するのに最適なキュー・マネージャーを判別します。

Asynchronous behavior of CLUSTER commands on z/OS

The command issuer of a cluster command on z/OS receives confirmation a command has been sent, but not that it has completed successfully.

関連資料

チャンネル定義コマンドで使用可能なクラスター属性

チャンネル定義コマンドで指定できるクラスター属性

キュー・マネージャー定義コマンドで使用可能なクラスター属性

キュー・マネージャー定義コマンドに指定できるクラスター属性です。

DISPLAY CLUSQMGR

DISPLAY CLUSQMGR コマンドは、クラスター内にあるキュー・マネージャーに関するクラスター情報を表示する場合に使用します。

REFRESH CLUSTER

REFRESH CLUSTER コマンドは、クラスターに関するローカルに保持されたすべての情報を廃棄する場合にキュー・マネージャーから発行します。例外的な状況を除いて、このコマンドを使用する必要はほとんどありません。

RESET CLUSTER: クラスターからキュー・マネージャーを強制的に除去する

RESET CLUSTER コマンドは、例外的な状況にあるクラスターからキュー・マネージャーを強制的に削除する場合に使用します。

SUSPEND QMGR、RESUME QMGR、およびクラスター

SUSPEND QMGR および RESUME QMGR コマンドは、このキュー・マネージャーへのインバウンド・クラスター・アクティビティーを一時的に (例えば、このキュー・マネージャーの保守を行う前に) 削減し、後で復元するために使用します。

143 ページの『クラスター・ワークロード・バランシング-キュー属性』

クラスター・ワークロード・バランシングで使用されるキュー属性のアルファベット順リスト。

キュー・マネージャー定義コマンドで使用可能なクラスター属性

キュー・マネージャー定義コマンドに指定できるクラスター属性です。

キュー・マネージャーがクラスターのフル・リポジトリを保持することを指定するには、属性

REPOS(*clustername*)を指定して **ALTER QMGR** コマンドを使用します。複数のクラスター名のリス

トを指定するには、クラスター名リストを定義してから、**ALTER QMGR** コマンドで属性 **REPOSNL** (*namelist*) を指定します。

```
DEFINE NAMELIST(CLUSTERLIST)
  DESCR('List of clusters whose repositories I host')
  NAMES(CLUS1, CLUS2, CLUS3)
ALTER QMGR REPOSNL(CLUSTERLIST)
```

ALTER QMGR コマンドには、追加のクラスター属性を指定することができます。

CLWLEXIT(*name*)

メッセージがクラスター・キューに書き込まれる際に呼び出すユーザー出口の名前を指定します。

CLWLDATA(*data*)

クラスター・ワークロード・ユーザー出口に渡されるデータを指定します。

CLWLLEN(*length*)

クラスター・ワークロード・ユーザー出口に渡されるメッセージ・データの最大量を指定します。

CLWLMRUC(*channels*)

アウトバウンド・クラスター・チャンネルの最大数を指定します。

CLWLMRUC は、クラスター全体には伝搬されないローカル・キュー・マネージャー属性です。これは、クラスター・ワークロード出口で使用可能で、メッセージの宛先を選択するクラスター・ワークロード・アルゴリズムです。

CLWLUSEQ(LOCAL|ANY)

ターゲット・キューにローカル・インスタンスと少なくとも 1 つのリモート・クラスター・インスタンスの両方がある場合の MQPUT の動作を指定します。書き込みの発生元がクラスター・チャンネルである場合、この属性は適用されません。CLWLUSEQ は、キュー属性とキュー・マネージャー属性の両方として指定することが可能です。

ANY を指定した場合、ローカル・キューとリモート・キューの両方が MQPUT の指定可能なターゲットになります。

LOCAL を指定した場合、ローカル・キューが MQPUT の唯一のターゲットになります。

同等の PCF は MQCMD_CHANGE_Q_MGR と MQCMD_INQUIRE_Q_MGR です。

関連概念

[クラスターでのワークロード・バランシング](#)

1 つのクラスターに同じキューのインスタンスが複数含まれる場合、IBM MQ はメッセージを送付するキュー・マネージャーを選択します。また、クラスター・ワークロード管理アルゴリズムと、クラスター・ワークロード固有のいくつかの属性を使用して、使用するのに最適なキュー・マネージャーを判別します。

Asynchronous behavior of CLUSTER commands on z/OS

The command issuer of a cluster command on z/OS receives confirmation a command has been sent, but not that it has completed successfully.

関連資料

[チャンネル定義コマンドで使用可能なクラスター属性](#)

[チャンネル定義コマンドで指定できるクラスター属性](#)

[キュー定義コマンドで使用可能なクラスター属性](#)

[キュー定義コマンドに指定できるクラスター属性](#)です。

DISPLAY CLUSQMGR

DISPLAY CLUSQMGR コマンドは、クラスター内にあるキュー・マネージャーに関するクラスター情報を表示する場合に使用します。

REFRESH CLUSTER

REFRESH CLUSTER コマンドは、クラスターに関するローカルに保持されたすべての情報を廃棄する場合にキュー・マネージャーから発行します。例外的な状況を除いて、このコマンドを使用する必要はほとんどありません。

[RESET CLUSTER: クラスターからキュー・マネージャーを強制的に除去する](#)

RESET CLUSTER コマンドは、例外的な状況にあるクラスターからキュー・マネージャーを強制的に削除する場合に使用します。

SUSPEND QMGR、RESUME QMGR、およびクラスター

SUSPEND QMGR および **RESUME QMGR** コマンドは、このキュー・マネージャーへのインバウンド・クラスター・アクティビティーを一時的に (例えば、このキュー・マネージャーの保守を行う前に) 削減し、後で復元するために使用します。

145 ページの『クラスター・ワークロード・バランシング-キュー・マネージャー属性』

クラスター・ワークロード・バランシングで使用されるキュー・マネージャー属性のアルファベット順リスト。

DISPLAY CLUSQMGR

DISPLAY CLUSQMGR コマンドは、クラスター内にあるキュー・マネージャーに関するクラスター情報を表示する場合に使用します。

このコマンドをフル・リポジトリを保持するキュー・マネージャーから発行した場合、返される情報はクラスター内のすべてのキュー・マネージャーに当てはまります。フル・リポジトリを保持しないキュー・マネージャーから発行した場合、返される情報は、関係のあるキュー・マネージャーのみに当てはまります。つまり、メッセージを送信しようとした送信先のキュー・マネージャー、およびフル・リポジトリを保持しているすべてのキュー・マネージャーに関する情報が返されます。

この情報には、クラスター送信側チャンネルとクラスター受信側チャンネルに適用される、ほとんどのチャンネル属性が含まれています。また、以下の属性も表示することができます。

CHANNEL

キュー・マネージャーのクラスター受信側チャンネル名。

CLUSDATE

定義がローカル・キュー・マネージャーに使用可能になった日付。

CLUSTER

キュー・マネージャーがどのクラスターに属しているか。

CLUSTIME

定義がローカル・キュー・マネージャーに有効になった時刻。

DEFTYPE

キュー・マネージャーがどのように定義されたか。DEFTYPE は、次のいずれかになります。

CLUSDR

クラスター送信側チャンネルが管理者によってローカル・キュー・マネージャーに定義されましたが、まだターゲット・キュー・マネージャーには認識されていません。こうした状態になるのは、ローカル・キュー・マネージャーで手動によりクラスター送信側チャンネルを定義したが、受信側キュー・マネージャーがそのクラスター情報をまだ受け入れていない場合です。この原因としては、使用可能な状態ではなかったためにチャンネルが確立されなかったか、送信側と受信側で **CLUSTER** プロパティの定義が一致していないなどのクラスター送信側の構成にエラーがあることが考えられます。これは一時的な状態またはエラー状態であり、調査する必要があります。

CLUSDRA

この値は自動的に検出されたクラスター・キュー・マネージャーを表します。ローカルにはクラスター送信側チャンネルは定義されていません。これは、ローカル・キュー・マネージャーにはローカル構成は存在しないが、ローカル・キュー・マネージャーに通知されたことのあるクラスター・キュー・マネージャーを表す DEFTYPE です。例えば

- ローカル・キュー・マネージャーが完全リポジトリ・キュー・マネージャーの場合、これは、クラスター内のすべての部分リポジトリ・キュー・マネージャーの DEFTYPE 値になります。
- ローカル・キュー・マネージャーが部分リポジトリの場合、これは、このローカル・キュー・マネージャーから使用されるクラスター・キュー、または、このキュー・マネージャーに連携するように指定した 2 番目の完全リポジトリ・キュー・マネージャーから使用されるクラスター・キューのホストになります。

DEFTYPE 値が CLUSSDRA であり、ローカル・キュー・マネージャーとリモート・キュー・マネージャーの両方が特定のクラスターの完全リポジトリである場合、この構成は正しくありません。ローカルに定義するクラスター送信側チャンネルで、これを CLUSSDRB の DEFTYPE に変換するように定義する必要があるからです。

CLUSSDRB

クラスター送信側チャンネルが管理者によってローカル・キュー・マネージャー上に定義され、ターゲット・キュー・マネージャーにも有効なクラスター・チャンネルとして受け入れられました。これは、部分リポジトリ・キュー・マネージャーの手動構成された完全リポジトリ・キュー・マネージャーに予期される DEFTYPE です。これは、クラスター内の 1 つの完全リポジトリから別の完全リポジトリに対する CLUSQMGR の DEFTYPE でもあります。手動クラスター送信側チャンネルを、複数の部分リポジトリに対して構成することはできません。また、1 つの部分リポジトリ・キュー・マネージャーから複数の完全リポジトリに対して構成することもできません。このような状況で CLUSSDRB である DEFTYPE が見られた場合は、調査して修正する必要があります。

CLUSRCVR

ローカル・キュー・マネージャー上にクラスター受信側チャンネルとして管理者によって定義されました。これはクラスター内のローカル・キュー・マネージャーを表します。

注：どの CLUSQMGR がクラスターの完全リポジトリ・キュー・マネージャーであるかを特定するには、QMTYPE プロパティを調べます。

クラスター・チャンネルの定義について詳しくは、『クラスター・チャンネル』を参照してください。

QMTYPE

完全リポジトリを持つか、部分リポジトリのみを持つか。

状況

このキュー・マネージャーのクラスター送信側チャンネルの状況。

SUSPEND

キュー・マネージャーが中断されているかどうか。

VERSION

のバージョン IBM MQ クラスター キュー マネージャーが関連付けられているインストール。

バージョンは形式 VVRRMMFF です。

- VV: バージョン
- RR: リリース
- MM: 保守レベル
- FF: 修正レベル

XMITQ

キュー・マネージャーにより使用されるクラスター伝送キュー。

また、DISPLAY QCLUSTER コマンドも参照してください。これについては、DISPLAY QUEUE、および 130 ページの『キュー定義コマンドで使用可能なクラスター属性』の DISPLAY QUEUE コマンドと DISPLAY QCLUSTER コマンドのセクションで簡単に説明します。DISPLAY QCLUSTER の使用例については、“DISPLAY QCLUSTER” および “DIS QCLUSTER” に関する情報を検索してください。

関連概念

クラスターでのワークロード・バランシング

1 つのクラスターに同じキューのインスタンスが複数含まれる場合、IBM MQ はメッセージを送付するキュー・マネージャーを選択します。また、クラスター・ワークロード管理アルゴリズムと、クラスター・ワークロード固有のいくつかの属性を使用して、使用するのに最適なキュー・マネージャーを判別します。

Asynchronous behavior of CLUSTER commands on z/OS

The command issuer of a cluster command on z/OS receives confirmation a command has been sent, but not that it has completed successfully.

関連資料

チャンネル定義コマンドで使用可能なクラスター属性
チャンネル定義コマンドで指定できるクラスター属性

キュー定義コマンドで使用可能なクラスター属性
キュー定義コマンドに指定できるクラスター属性です。

キュー・マネージャー定義コマンドで使用可能なクラスター属性
キュー・マネージャー定義コマンドに指定できるクラスター属性です。

REFRESH CLUSTER

REFRESH CLUSTER コマンドは、クラスターに関するローカルに保持されたすべての情報を廃棄する場合にキュー・マネージャーから発行します。例外的な状況を除いて、このコマンドを使用する必要はほとんどありません。

RESET CLUSTER: クラスターからキュー・マネージャーを強制的に除去する

RESET CLUSTER コマンドは、例外的な状況にあるクラスターからキュー・マネージャーを強制的に削除する場合に使用します。

SUSPEND QMGR, RESUME QMGR, およびクラスター

SUSPEND QMGR および RESUME QMGR コマンドは、このキュー・マネージャーへのインバウンド・クラスター・アクティビティーを一時的に (例えば、このキュー・マネージャーの保守を行う前に) 削減し、後で復元するために使用します。

MQSC command **DISPLAY CLUSQMGR**

REFRESH CLUSTER

REFRESH CLUSTER コマンドは、クラスターに関するローカルに保持されたすべての情報を廃棄する場合にキュー・マネージャーから発行します。例外的な状況を除いて、このコマンドを使用する必要はほとんどありません。

このコマンドの形式には、以下の3つがあります。

REFRESH CLUSTER(clustername) REPOS(NO)

これがデフォルトです。キュー・マネージャーには、ローカルで定義されているすべてのクラスター・キュー・マネージャーとクラスター・キュー、およびフル・リポジトリーのすべてのクラスター・キュー・マネージャーに関する情報が保持されます。また、キュー・マネージャーがクラスターのフル・リポジトリーの場合は、クラスター内の他のクラスター・キュー・マネージャーに関する情報も保持されます。他のすべてのものはリポジトリーのローカル・コピーから除去され、クラスター内の他の完全リポジトリーから再作成されます。REPOS(NO) を使用すると、クラスター・チャンネルは停止されません。フル・リポジトリーでは CLUSSDR チャンネルを使用して、そのリフレッシュが完了したことをクラスターの他の部分に通知します。

REFRESH CLUSTER(clustername) REPOS(YES)

デフォルトの動作に加えて、フル・リポジトリーのクラスター・キュー・マネージャーを表すオブジェクトもリフレッシュされます。キュー・マネージャーがフル・リポジトリーの場合、このオプションを使うことは適切ではありません。使用した場合、コマンドはエラー AMQ9406/CSQX406E をログして失敗します。キュー・マネージャーがフル・リポジトリーである場合、まずは、問題のクラスターのフル・リポジトリーではなくなるように、変更する必要があります。完全リポジトリーの場合は、手作業で定義された CLUSSDR 定義から回復されます。REPOS(YES) によるリフレッシュが発行された後は、キュー・マネージャーに変更を加えることができ、必要な場合はそれを再びフル・リポジトリーにすることができます。

REFRESH CLUSTER(*)

キュー・マネージャーは、そのキュー・マネージャーがメンバーであるすべてのクラスターでリフレッシュされます。REPOS(YES) REFRESH CLUSTER(*) と共に使用すると、そのキュー・マネージャーによるローカルの CLUSSDR 定義内の情報に基づくフル・リポジトリーの検索を強制的に再開するというその他の効果もあります。この検索は、CLUSSDR チャンネルでこのキュー・マネージャーを複数のクラスターに接続している場合でも実行されます。

注: 大規模クラスターでは、稼働中のクラスターに **REFRESH CLUSTER** コマンドを使用すると、そのクラスターに悪影響が及ぶ可能性があります。その後、クラスター・オブジェクトが 27 日間隔で対象のキュー・マネージャーすべてに状況の更新を自動的に送信する際にも同様のことが起こり得ます。大規模クラスターでのリフレッシュはクラスターのパフォーマンスと可用性に影響を与える可能性があるを参照してください。

関連概念

[クラスターでのワークロード・バランシング](#)

1つのクラスターに同じキューのインスタンスが複数含まれる場合、IBM MQ はメッセージを送付するキュー・マネージャーを選択します。また、クラスター・ワークロード管理アルゴリズムと、クラスター・ワークロード固有のいくつかの属性を使用して、使用するのに最適なキュー・マネージャーを判別します。

Asynchronous behavior of CLUSTER commands on z/OS

The command issuer of a cluster command on z/OS receives confirmation a command has been sent, but not that it has completed successfully.

関連資料

[チャンネル定義コマンドで使用可能なクラスター属性](#)

[チャンネル定義コマンドで指定できるクラスター属性](#)

[キュー定義コマンドで使用可能なクラスター属性](#)

[キュー定義コマンドに指定できるクラスター属性](#)です。

[キュー・マネージャー定義コマンドで使用可能なクラスター属性](#)

[キュー・マネージャー定義コマンドに指定できるクラスター属性](#)です。

DISPLAY CLUSQMGR

DISPLAY CLUSQMGR コマンドは、クラスター内にあるキュー・マネージャーに関するクラスター情報を表示する場合に使用します。

[RESET CLUSTER: クラスターからキュー・マネージャーを強制的に除去する](#)

RESET CLUSTER コマンドは、例外的な状況にあるクラスターからキュー・マネージャーを強制的に削除する場合に使用します。

[SUSPEND QMGR、RESUME QMGR、およびクラスター](#)

SUSPEND QMGR および RESUME QMGR コマンドは、このキュー・マネージャーへのインバウンド・クラスター・アクティビティーを一時的に (例えば、このキュー・マネージャーの保守を行う前に) 削減し、後で復元するために使用します。

関連情報

[クラスター化: REFRESH CLUSTER の使用に関するベスト・プラクティス](#)

RESET CLUSTER: クラスターからキュー・マネージャーを強制的に除去する

RESET CLUSTER コマンドは、例外的な状況にあるクラスターからキュー・マネージャーを強制的に削除する場合に使用します。

例外的な状況を除いて、このコマンドを使用する必要はほとんどありません。

RESET CLUSTER コマンドは、フル・リポジトリのキュー・マネージャーからのみ発行することができます。このコマンドは、キュー・マネージャーを名前または ID のいずれかで参照するかによって 2 つの形式があります。

1.

```
RESET CLUSTER( clustername
) QMNAME( qmname ) ACTION(FORCEREMOVE) QUEUES(NO)
```

2.

```
RESET CLUSTER( clustername
) QMID( qmid ) ACTION(FORCEREMOVE) QUEUES(NO)
```

QMNAME と QMID の両方を指定することはできません。QMNAME を使用した場合、その名前を持つキュー・マネージャーがクラスター内に複数あると、このコマンドは実行されません。**RESET CLUSTER** コマンドを確実に実行するには、QMNAME ではなく QMID を使用してください。

RESET CLUSTER コマンドで QUEUES(NO) を指定することがデフォルトです。QUEUES(YES) を指定すると、キュー・マネージャーによって所有されているクラスター・キューへの参照がクラスターから削除されます。つまり、キュー・マネージャーがクラスター自体から削除されるだけでなく、参照も削除されます。

参照は、クラスター・キュー・マネージャーがクラスター内で表示されない場合にも削除されます。これは、おそらくこのクラスター・キュー・マネージャーが前に **QUEUES** オプションなしで強制的に削除されているためです。

例えば、キュー・マネージャーは削除済みであるものの、クラスター受信側チャンネルが依然クラスターに定義されている場合、**RESET CLUSTER** コマンドを使用できます。これらの定義の IBM MQ による削除（この削除は自動的に実行されます）を待つのではなく、**RESET CLUSTER** コマンドを発行してすぐに整理することができます。この場合、次にクラスター内の他のすべてのキュー・マネージャーに、このキュー・マネージャーが使用できなくなったことが通知されます。

キュー・マネージャーが一時的に損傷を受けている場合、クラスター内の他のキュー・マネージャーがそのキュー・マネージャーにメッセージの送信を試みる前に、それらのキュー・マネージャーに通知を行うことがあります。**RESET CLUSTER** では、損傷を受けたキュー・マネージャーが削除されます。後で、損傷を受けたキュー・マネージャーを再び実行する場合は、**REFRESH CLUSTER** コマンドを使用して **RESET CLUSTER** の効果を取り消し、キュー・マネージャーをクラスターに戻します。キュー・マネージャーがパブリッシュ/サブスクライブ・クラスター内にある場合は、必要なプロキシ・サブスクリプションをすべて復元する必要があります。パブリッシュ/サブスクライブ・クラスターでの REFRESH CLUSTER についての考慮事項を参照してください。

注: 大規模クラスターでは、稼働中のクラスターに **REFRESH CLUSTER** コマンドを使用すると、そのクラスターに悪影響が及ぶ可能性があります。その後、クラスター・オブジェクトが 27 日間隔で対象のキュー・マネージャーすべてに状況の更新を自動的に送信する際にも同様のことが起こり得ます。大規模クラスターでのリフレッシュはクラスターのパフォーマンスと可用性に影響を与える可能性があるを参照してください。

自動定義されたクラスター送信側チャンネルを削除する唯一の方法は、**RESET CLUSTER** コマンドを使用することです。

重要: 除去する自動定義チャンネルが未確定の場合、**RESET CLUSTER** はそのチャンネルを即時に除去しません。この状態では、**RESET CLUSTER** コマンドの前に **RESOLVE CHANNEL** コマンドを発行する必要があります。

このコマンドは、通常環境で必要になることはまずありません。IBM サポートは、クラスター・キュー・マネージャーによって保持されているクラスター情報を整理するために、このコマンドを発行するようお勧めする場合があります。このコマンドを、クラスターからキュー・マネージャーを削除するための簡単な方法として使用しないでください。クラスターからキュー・マネージャーを削除する正しい方法は、クラスターからのキュー・マネージャーの除去で説明されています。

リポジトリでは 90 日間だけ情報を保持しているため、その期間以降は強制的に削除されたキュー・マネージャーをクラスターに再接続することができます。キュー・マネージャーが削除されていない限り、それは自動的に再接続します。キュー・マネージャーがクラスターに再び参加しないようにするには、適切なセキュリティ対策を講じる必要があります。

DISPLAY CLUSQMgr 以外のクラスター・コマンドは、すべて非同期で機能します。クラスター化を含むオブジェクト属性を変更するコマンドでは、オブジェクトが更新され、リポジトリ・プロセッサに要求が送信されます。クラスターを操作するコマンドは、構文が検査されてから要求がリポジトリ・プロセッサに送信されます。

リポジトリ・プロセッサに送信された要求は、クラスターの他のメンバーから受信されたクラスター要求とともに、非同期で処理されます。この処理は、要求が正常に実行されたかどうかを判別するために、要求がクラスター全体に伝搬されなければならない場合、相当な時間がかかる場合があります。

関連概念

クラスターでのワークロード・バランシング

1つのクラスターに同じキューのインスタンスが複数含まれる場合、IBM MQ はメッセージを送付するキュー・マネージャーを選択します。また、クラスター・ワークロード管理アルゴリズムと、クラスター・ワークロード固有のいくつかの属性を使用して、使用するのに最適なキュー・マネージャーを判別します。

Asynchronous behavior of CLUSTER commands on z/OS

The command issuer of a cluster command on z/OS receives confirmation a command has been sent, but not that it has completed successfully.

関連資料

[チャンネル定義コマンドで使用可能なクラスター属性](#)

[チャンネル定義コマンドで指定できるクラスター属性](#)

[キュー定義コマンドで使用可能なクラスター属性](#)

[キュー定義コマンドに指定できるクラスター属性](#)です。

[キュー・マネージャー定義コマンドで使用可能なクラスター属性](#)

[キュー・マネージャー定義コマンドに指定できるクラスター属性](#)です。

DISPLAY CLUSQMGR

DISPLAY CLUSQMGR コマンドは、クラスター内にあるキュー・マネージャーに関するクラスター情報を表示する場合に使用します。

REFRESH CLUSTER

REFRESH CLUSTER コマンドは、クラスターに関するローカルに保持されたすべての情報を廃棄する場合にキュー・マネージャーから発行します。例外的な状況を除いて、このコマンドを使用する必要はほとんどありません。

SUSPEND QMGR、RESUME QMGR、およびクラスター

SUSPEND QMGR および RESUME QMGR コマンドは、このキュー・マネージャーへのインバウンド・クラスター・アクティビティーを一時的に (例えば、このキュー・マネージャーの保守を行う前に) 削減し、後で復元するために使用します。

RESET CLUSTER (クラスターのリセット)

SUSPEND QMGR、RESUME QMGR、およびクラスター

SUSPEND QMGR および RESUME QMGR コマンドは、このキュー・マネージャーへのインバウンド・クラスター・アクティビティーを一時的に (例えば、このキュー・マネージャーの保守を行う前に) 削減し、後で復元するために使用します。

クラスターのキュー・マネージャーが中断状態になっているとき、そのクラスター内の代替キュー・マネージャーに同じ名前の使用可能キューがある場合、停止中のキュー・マネージャーは、そこでホストするクラスター・キューのメッセージを受け取りません。ただし、このキュー・マネージャーが明示的にメッセージのターゲットになっている場合や、このターゲット・キューを使用できるのがこのキュー・マネージャーだけである場合、メッセージは引き続きこのキュー・マネージャーに送信されます。

このクラスターのクラスター受信側チャンネルを停止すると、キュー・マネージャーが中断状態になっているときにこれ以上インバウンド・メッセージを受け取らないように設定できます。クラスターのクラスター受信側チャンネルを停止するには、[SUSPEND QMGR](#) コマンドの FORCE モードを使用します。

関連概念

[クラスターでのワークロード・バランシング](#)

1つのクラスターに同じキューのインスタンスが複数含まれる場合、IBM MQ はメッセージを送付するキュー・マネージャーを選択します。また、クラスター・ワークロード管理アルゴリズムと、クラスター・ワークロード固有のいくつかの属性を使用して、使用するのに最適なキュー・マネージャーを判別します。

[Asynchronous behavior of CLUSTER commands on z/OS](#)

The command issuer of a cluster command on z/OS receives confirmation a command has been sent, but not that it has completed successfully.

関連タスク

[キュー・マネージャーの保守](#)

関連資料

[チャンネル定義コマンドで使用可能なクラスター属性](#)

[チャンネル定義コマンドで指定できるクラスター属性](#)

[キュー定義コマンドで使用可能なクラスター属性](#)

[キュー定義コマンドに指定できるクラスター属性](#)です。

[キュー・マネージャー定義コマンドで使用可能なクラスター属性](#)

[キュー・マネージャー定義コマンドに指定できるクラスター属性](#)です。

DISPLAY CLUSQMGR

DISPLAY CLUSQMGR コマンドは、クラスター内にあるキュー・マネージャーに関するクラスター情報を表示する場合に使用します。

REFRESH CLUSTER

REFRESH CLUSTER コマンドは、クラスターに関するローカルに保持されたすべての情報を廃棄する場合にキュー・マネージャーから発行します。例外的な状況を除いて、このコマンドを使用する必要はほとんどありません。

RESET CLUSTER: クラスターからキュー・マネージャーを強制的に除去する

RESET CLUSTER コマンドは、例外的な状況にあるクラスターからキュー・マネージャーを強制的に削除する場合に使用します。

SUSPEND QMGR

RESUME QMGR

クラスターでのワークロード・バランシング

1つのクラスターに同じキューのインスタンスが複数含まれる場合、IBM MQ はメッセージを送付するキュー・マネージャーを選択します。また、クラスター・ワークロード管理アルゴリズムと、クラスター・ワークロード固有のいくつかの属性を使用して、使用するのに最適なキュー・マネージャーを判別します。

適切な宛先は、クラスター・ワークロード管理アルゴリズムによって、キュー・マネージャーとキューの可用性、およびチャンネル、キュー、およびキュー・マネージャーに関連付けられたいくつかのクラスター・ワークロード固有の属性に基づいて選択されます。これらの属性については、サブトピックで説明しています。

クラスターのワークロード固有の属性を構成した後に、その構成が予期したとおりに動作しない場合は、キュー・マネージャーを選択するアルゴリズムの仕組みの詳細を調べてください。146 ページの『[クラスター・ワークロード管理アルゴリズム](#)』を参照。このアルゴリズムの結果がニーズに合っていない場合は、クラスター・ワークロードのユーザー出口プログラムを作成し、この出口を使用してクラスター内の適切なキューにメッセージをルーティングすることができます。[クラスター・ワークロード出口の作成とコンパイル](#)を参照してください。

属性名	説明
チャンネルの属性	
CLWLPRTY	クラスター・ワークロード分散のチャンネルの優先順位を指定します。
CLWLRANK	クラスター・ワークロード分散のチャンネルのランクを指定します。
CLWLWGHT	クラスター・ワークロード分散のために CLUSSDR および CLUSRCVR チャンネルに適用される重みを指定します。
NETPRTY	CLUSRCVR チャンネルの優先順位を指定します。
キューの属性	
CLWLPRTY	クラスター・ワークロード配布のためのローカル・キュー、リモート・キュー、または別名キューの優先順位を指定します。
CLWLRANK	クラスター・ワークロード分散のためのローカル・キュー、リモート・キュー、または別名キューのランクを指定します。

表 59. クラスター・ワークロード固有の属性の要約 (続き)

属性名	説明
CLWLUSEQ	キューのローカル・インスタンスに宛先としてクラスター内の他のインスタンスよりも優先権を与えるかどうかを指定します。
キュー・マネージャーの属性	
CLWLMRUC	最後に選択されたチャンネルの数を設定します。
CLWLUSEQ	キューのローカル・インスタンスが、クラスター内のキューの他のインスタンスよりも宛先として優先されるかどうかを指定します。

関連概念

Asynchronous behavior of CLUSTER commands on z/OS

The command issuer of a cluster command on z/OS receives confirmation a command has been sent, but not that it has completed successfully.

関連資料

[チャンネル定義コマンドで使用可能なクラスター属性](#)

[チャンネル定義コマンドで指定できるクラスター属性](#)

[キュー定義コマンドで使用可能なクラスター属性](#)

キュー定義コマンドに指定できるクラスター属性です。

[キュー・マネージャー定義コマンドで使用可能なクラスター属性](#)

キュー・マネージャー定義コマンドに指定できるクラスター属性です。

DISPLAY CLUSQMGR

DISPLAY CLUSQMGR コマンドは、クラスター内にあるキュー・マネージャーに関するクラスター情報を表示する場合に使用します。

REFRESH CLUSTER

REFRESH CLUSTER コマンドは、クラスターに関するローカルに保持されたすべての情報を廃棄する場合にキュー・マネージャーから発行します。例外的な状況を除いて、このコマンドを使用する必要はほとんどありません。

[RESET CLUSTER: クラスターからキュー・マネージャーを強制的に除去する](#)

RESET CLUSTER コマンドは、例外的な状況にあるクラスターからキュー・マネージャーを強制的に削除する場合に使用します。

SUSPEND QMGR、RESUME QMGR、およびクラスター

SUSPEND QMGR および RESUME QMGR コマンドは、このキュー・マネージャーへのインバウンド・クラスター・アクティビティを一時的に (例えば、このキュー・マネージャーの保守を行う前に) 削減し、後で復元するために使用します。

クラスター・ワークロード・バランシング-チャンネル属性

クラスター・ワークロード・バランシングで使用されるチャンネル属性のアルファベット順リスト。

注: クラスター・ワークロードのチャンネル属性は、ターゲット・キュー・マネージャーのクラスター受信側チャンネルで指定します。対応するクラスター送信側チャンネルで指定するバランシングは無視される可能性があります。 [クラスター・チャンネル](#)を参照してください。

CLWLPRTY (クラスター・ワークロード優先順位)

CLWLPRTY チャンネル属性は、クラスター・ワークロード分散のために、チャンネルの優先順位を指定します。この値は、0 から 9 の範囲でなければなりません。0 が最低、9 が最高の優先度です。

CLWLPRTY チャンネル属性は、使用可能なクラスター宛先の優先順位を設定するために使用します。IBM MQ は、クラスターの宛先の優先度が最も低い宛先を選択する前に、優先度が最も高い宛先を選択します。優先度が同じである複数の宛先が存在する場合は、最も長期間未使用になっている宛先を選択します。

有効な宛先が2つある場合は、この属性を使用して、フェイルオーバーを許可できます。メッセージは、最高の優先順位のチャンネルを持つキュー・マネージャーに移動されます。このキュー・マネージャーが使用不可になった場合、メッセージはその次に高い優先順位のキュー・マネージャーに移動されます。低い優先順位のキュー・マネージャーは、予備としての役割を果たします。

IBM MQ は、チャンネルに優先順位付けする前に、チャンネルの状況を検査します。使用可能なキュー・マネージャーのみが選択候補になります。

注：

- クラスター受信側チャンネルのターゲット・キュー・マネージャーで、この属性を指定します。一致するクラスター送信側チャンネルで指定したすべてのバランシングは、無視される可能性が高くなります。[クラスター・チャンネル](#)を参照してください。
- リモート・キュー・マネージャーの可用性は、そのキュー・マネージャーに対するチャンネルの状況に基づきます。チャンネルを開始する際に、チャンネルの状態は何度か変わります。その中には、クラスターのワークロード管理アルゴリズムの結果より優先度の低い状態もあります。これは、実際に、優先順位がより高い(プライマリー)宛先に対するチャンネルが開始されていても、優先順位がより低い(バックアップ)宛先が選択される可能性があることを意味します。
- バックアップ宛先に対してメッセージが送信されないようにする必要がある場合は、CLWLPRTY を使用しないでください。個別のキューを使用するか、または CLWLPRTY を使用してプライマリーからバックアップに手動で切り替えることを検討してください。

CLWLPRTY (クラスター・ワークロード・ランク)

CLWLPRTY チャンネル属性は、クラスター・ワークロード分散に使用するチャンネルのランクを指定します。値は0から9の範囲でなければなりません(0が最低ランク、9が最高ランク)。

CLWLPRTY チャンネル属性は、別のクラスター内のキュー・マネージャーに送信されるメッセージの最終宛先を制御する場合に使用します。クラスターの交点にあるゲートウェイ・キュー・マネージャーにキュー・マネージャーを接続するチャンネルのランクを設定することによって、最終宛先の選択を制御します。

CLWLPRTY を設定すると、メッセージは、相互接続されたクラスターを経由してランクの高い宛先に向かう指定経路を取ります。例えば、ランク1とランク2のチャンネルを使用する2つのキュー・マネージャーのどちらかにメッセージを送信できるゲートウェイ・キュー・マネージャーにメッセージが到着したとします。これらのメッセージは、最高ランクのチャンネル(この場合はランク2のキュー・マネージャーへのチャンネル)で接続されたキュー・マネージャーに自動的に送信されます。

IBM MQ は、チャンネル状況を検査する前にチャンネルのランクを取得します。チャンネル状況を検査する前にランクを取得するということは、アクセスできないチャンネルでも選択の対象になることを意味します。この方法は、最終宛先が使用不可であったとしても、ネットワークを介してメッセージがルーティングされることを可能にします。

注：

- クラスター受信側チャンネルのターゲット・キュー・マネージャーで、この属性を指定します。一致するクラスター送信側チャンネルで指定したすべてのバランシングは、無視される可能性が高くなります。[クラスター・チャンネル](#)を参照してください。
- 優先度属性 CLWLPRTY も使用した場合、IBM MQ は使用可能な宛先から選択します。最高ランクの宛先がチャンネルを使用できない場合、メッセージは伝送キューに保持されます。チャンネルが使用可能になると解放されます。ランク順で次に使用可能な宛先にメッセージが送信されるわけではありません。

CLWLWGT (クラスター・ワークロード・ウェイト)

CLWLWGT チャンネル属性は、クラスター・ワークロード分散用に、CLUSDR および CLUSRCVR チャンネルに適用される重みを指定します。この値は、1から99の範囲でなければなりません。1が最低、99が最高の重みです。

CLWLWGHT を使用して、より多くの処理能力があるサーバーに、より多くのメッセージを送信します。チャンネルの重みが高いほど、そのチャンネルを通して多くのメッセージが送信されます。

注:

- クラスター受信側チャンネルのターゲット・キュー・マネージャーで、この属性を指定します。一致するクラスター送信側チャンネルで指定したすべてのバランシングは、無視される可能性が高くなります。[クラスター・チャンネル](#)を参照してください。
- いずれかのチャンネルで CLWLWGHT がデフォルトの 50 から変更された場合、ワークロード・バランシングはいずれかのクラスター・キューに送信されたメッセージのために各チャンネルが合計何回選択されたかに依存します。詳しくは、[146 ページの『クラスター・ワークロード管理アルゴリズム』](#)を参照してください。

NETPRTY (ネットワーク接続優先順位)

NETPRTY チャンネル属性では、CLUSRCVR チャンネルの優先順位を指定します。この値は、0 から 9 の範囲でなければなりません。0 が最低、9 が最高の優先度です。

NETPRTY 属性は、あるネットワークを主ネットワークにし、別のネットワークをバックアップ・ネットワークにする場合に使用します。同じランクの 1 組みのチャンネルが存在すると仮定すると、複数のパスが使用可能な場合、クラスター化では最高の優先順位を持つパスを選択します。

NETPRTY チャンネル属性を使用する典型的な例としては、異なるコストまたは速度を持ち、また同じ複数の宛先を接続する 2 つのネットワークを区別するケースです。

注: クラスター受信側チャンネルのターゲット・キュー・マネージャーで、この属性を指定します。一致するクラスター送信側チャンネルで指定したすべてのバランシングは、無視される可能性が高くなります。[クラスター・チャンネル](#)を参照してください。

関連概念

[クラスター・ワークロード管理アルゴリズム](#)

ワークロード管理アルゴリズムは、ワークロード・バランシング属性と多数の規則を使用して、クラスター・キューに書き込むメッセージの最終宛先を選択します。

関連資料

[クラスター・ワークロード・バランシング-キュー属性](#)

クラスター・ワークロード・バランシングで使用されるキュー属性のアルファベット順リスト。

[クラスター・ワークロード・バランシング-キュー・マネージャー属性](#)

クラスター・ワークロード・バランシングで使用されるキュー・マネージャー属性のアルファベット順リスト。

[127 ページの『チャンネル定義コマンドで使用可能なクラスター属性』](#)

チャンネル定義コマンドで指定できるクラスター属性

クラスター・ワークロード・バランシング-キュー属性

クラスター・ワークロード・バランシングで使用されるキュー属性のアルファベット順リスト。

CLWLPRTY

CLWLPRTY キュー属性は、クラスター・ワークロードの分散用にローカル・キュー、リモート・キュー、または別名キューの優先度を指定します。この値は、0 から 9 の範囲でなければなりません。0 が最低、9 が最高の優先度です。

CLWLPRTY キュー属性を使用して、宛先キューの設定を行います。IBM MQ は、クラスターの宛先の優先度が最も低い宛先を選択する前に、優先度が最も高い宛先を選択します。優先度が同じである複数の宛先が存在する場合は、最も長期間未使用になっている宛先を選択します。

IBM MQ は、チャンネル状況を検査した後で、キュー・マネージャーの優先度を取得します。使用可能なキュー・マネージャーのみが選択候補になります。

注:

リモート・キュー・マネージャーの可用性は、そのキュー・マネージャーに対するチャンネルの状況に基づきます。チャンネルを開始する際に、チャンネルの状態は何度か変わります。その中には、クラスターのワークロード管理アルゴリズムの結果より優先度の低い状態もあります。これは、実際に、優先順位がより高い(プライマリー)宛先に対するチャンネルが開始されていても、優先順位がより低い(バックアップ)宛先が選択される可能性があることを意味します。

バックアップ宛先に対してメッセージが送信されないようにする必要がある場合は、**CLWLPRTY**を使用しないでください。個別のキューを使用するか、または**CLWLRANK**を使用してプライマリーからバックアップに手動で切り替えることを検討してください。

有効な宛先が2つある場合は、この属性を使用して、フェイルオーバーを許可できます。優先度が最も高いキュー・マネージャーは要求を受け取り、優先度が低いキュー・マネージャーは予備として機能します。優先度が最も高いキュー・マネージャーが失敗した場合、次に優先度が高い使用可能なキュー・マネージャーが引き継ぎます。

CLWLRANK

CLWLRANK キュー属性は、クラスター・ワークロードの分散用に、ローカル・キュー、リモート・キュー、または別名キューのランクを指定します。値は0から9の範囲でなければなりません(0が最低ランク、9が最高ランク)。

CLWLRANK キュー属性は、別のクラスター内のキュー・マネージャーに送信されるメッセージの最終宛先を制御する場合に使用します。**CLWLRANK**を設定すると、メッセージは、相互接続されたクラスターを経由してランクの高い宛先に向かう指定経路を取ります。

例えば、ゲートウェイの可用性を高めるために、同等に構成された2つのゲートウェイ・キュー・マネージャーを定義したとします。クラスターで定義されたローカル・キュー用に、ゲートウェイでクラスター別名キューを定義したとします。ローカル・キューが使用不可になった場合には、キューが再度使用可能になるまで、メッセージがゲートウェイのいずれかで保留されるようにするとします。キューをゲートウェイで保持するには、ゲートウェイのクラスター別名キューよりも高いランクでローカル・キューを定義する必要があります。

キュー別名と同じランクでローカル・キューを定義して、ローカル・キューが使用不可になった場合、メッセージはゲートウェイ間を移動します。ローカル・キューが使用不可であると検出されると、最初のゲートウェイ・キュー・マネージャーはもう一方のゲートウェイにメッセージを経路指定します。もう一方のゲートウェイはターゲット・ローカル・キューに再度メッセージを送信しようとします。ローカル・キューが使用不可のままである場合、最初のゲートウェイにメッセージが戻されます。メッセージは、ターゲット・ローカル・キューが再度使用可能になるまで、ゲートウェイ間を移動し続けます。ローカル・キューに高位ランクを指定することによって、ローカル・キューが使用不可の場合でも、メッセージが下位ランクの宛先に転送されなくなります。

IBM MQ は、チャンネル状況を検査する前にキューのランクを取得します。チャンネル状況を検査する前にランクを取得するという事は、アクセスできないキューでも選択の対象になることとなります。この方法は、最終宛先が使用不可であったとしても、ネットワークを介してメッセージがルーティングされることを可能にします。

優先度属性を使用した場合、IBM MQ は使用可能な宛先から選択します。最高ランクの宛先がチャンネルを使用できない場合、メッセージは伝送キューに保持されます。チャンネルが使用可能になると解放されます。ランク順で次に使用可能な宛先にメッセージが送信されるわけではありません。

CLWLUSEQ

CLWLUSEQ キュー属性では、キューのローカル・インスタンスに宛先としてクラスター内の他のインスタンスよりも優先権を与えるかどうかを指定します。

CLWLUSEQ キュー属性は、ローカル・キューにのみ有効です。これは、メッセージがアプリケーション、あるいはクラスター・チャンネルではないチャンネルによって書き込まれる場合にのみ適用されます。

local

ローカル・キューは、書き込み可能であれば、MQPUT の唯一のターゲットです。MQPUT の動作は、クラスター・ワークロード管理によって異なります。

QMGR

この動作は、**CLWLUSEQ** キュー・マネージャー属性で指定されたとおりになります。

ANY

MQPUT は、ワークロード分散用に、ローカル・キューをクラスター内のキューのその他のインスタンスと同じように見なします。

関連概念

[クラスター・ワークロード管理アルゴリズム](#)

ワークロード管理アルゴリズムは、ワークロード・バランシング属性と多数の規則を使用して、クラスター・キューに書き込むメッセージの最終宛先を選択します。

関連資料

[クラスター・ワークロード・バランシング-チャンネル属性](#)

クラスター・ワークロード・バランシングで使用されるチャンネル属性のアルファベット順リスト。

[クラスター・ワークロード・バランシング-キュー・マネージャー属性](#)

クラスター・ワークロード・バランシングで使用されるキュー・マネージャー属性のアルファベット順リスト。

130 ページの『[キュー定義コマンドで使用可能なクラスター属性](#)』

キュー定義コマンドに指定できるクラスター属性です。

クラスター・ワークロード・バランシング-キュー・マネージャー属性

クラスター・ワークロード・バランシングで使用されるキュー・マネージャー属性のアルファベット順リスト。

CLWLMRUC

CLWLMRUC キュー・マネージャー属性は、最後に選択されたチャンネルの数を設定します。クラスター・ワークロード管理アルゴリズムは **CLWLMRUC** を使用して、アクティブなアウトバウンド・クラスター・チャンネルの数を制限します。値は 1 から 999 999 999 の範囲でなければなりません。

初期デフォルト値は 999 999 999 です。

CLWLUSEQ

CLWLUSEQ キュー・マネージャー属性は、クラスター内のキューの他のインスタンスよりも、キューのローカル・インスタンスを宛先として優先するかどうかを指定します。この属性は、**CLWLUSEQ** キュー属性が QMGR に設定されている場合に適用されます。

CLWLUSEQ キュー属性は、ローカル・キューにのみ有効です。これは、メッセージがアプリケーション、あるいはクラスター・チャンネルではないチャンネルによって書き込まれる場合にのみ適用されます。

local

ローカル・キューは、MQPUT の唯一のターゲットです。LOCAL がデフォルトです。

ANY

MQPUT は、ワークロード分散用に、ローカル・キューをクラスター内のキューのその他のインスタンスと同じように見なします。

関連概念

[クラスター・ワークロード管理アルゴリズム](#)

ワークロード管理アルゴリズムは、ワークロード・バランシング属性と多数の規則を使用して、クラスター・キューに書き込むメッセージの最終宛先を選択します。

関連資料

[クラスター・ワークロード・バランシング-チャンネル属性](#)

クラスター・ワークロード・バランシングで使用されるチャンネル属性のアルファベット順リスト。

[クラスター・ワークロード・バランシング-キュー属性](#)

クラスター・ワークロード・バランシングで使用されるキュー属性のアルファベット順リスト。

キュー・マネージャー定義コマンドに指定できるクラスター属性です。

クラスター・ワークロード管理アルゴリズム

ワークロード管理アルゴリズムは、ワークロード・バランシング属性と多数の規則を使用して、クラスター・キューに書き込むメッセージの最終宛先を選択します。

ワークロード管理アルゴリズムは宛先の選択が必要になるたびに毎回実行されます。

- MQOO_BIND_ON_OPEN オプションを使用してクラスター・キューが開かれた時点で使用されます。
- MQOO_BIND_NOT_FIXED を指定してクラスター・キューが開かれた場合に、メッセージがそのクラスター・キューに書き込まれるたびに使用されます。
- MQOO_BIND_ON_GROUP を使用してクラスター・キューが開かれた場合に、新規メッセージ・グループが開始されるたびに使用されます。
- トピック・ホスト・ルーティングでは、メッセージがクラスター・トピックにパブリッシュされるたびに使用されます。ローカル・キュー・マネージャーがそのトピックのホストではない場合、このアルゴリズムを使用して、メッセージのルーティング経由となるホスト・キュー・マネージャーが選択されません。

以下のセクションでは、クラスター・キューに書き込むメッセージの最終宛先を決定する際に使用されるワークロード管理アルゴリズムについて説明します。これらの規則は、キュー、キュー・マネージャー、およびチャネルの以下の属性に適用される設定の影響を受けます。

表 60. クラスター・ワークロード管理の属性		
キュー	キュー・マネージャー	チャネル
<ul style="list-style-type: none"> • <u>CLWLPRTY</u>¹ • <u>CLWLRANK</u>¹ • <u>CLWLUSEQ</u>¹ • PUT/PUB 	<ul style="list-style-type: none"> • <u>CLWLMRUC</u> • <u>CLWLUSEQ</u>¹ 	<ul style="list-style-type: none"> • <u>CLWLPRTY</u> • <u>CLWLRANK</u> • <u>CLWLWGHT (CLWLWGHT)</u> • <u>NETPRTY (NETPRTY)</u>

最初に、キュー・マネージャーは以下の2つの手順から有効な宛先のリストを構築します。

- ターゲットの ObjectName および ObjectQmgrName と、このキュー・マネージャーと同じクラスターで共有されるキュー・マネージャー別名定義を突き合わせます。
 - ObjectName という名前のキューをホストしているキュー・マネージャーへの固有の経路、つまりチャネルのうち、このキュー・マネージャーがメンバーになっているクラスター内にあるものを検索します。
- このアルゴリズムでは、以下の各ルールを順次適用して有効な宛先のリストから宛先を除去していきます。

1. ローカル・キュー・マネージャーとクラスターを共用しないキューまたはトピックまたはリモート CLUSRCVR チャネルのリモート・インスタンスは除去されます。
2. キューまたはトピック名が指定されている場合、キューまたはトピックと同じクラスター内にないリモート CLUSRCVR チャネルは除去されます。

注：この段階で残っているすべてのキュー、トピック、およびチャネルは、構成されていれば、クラスター・ワークロード出口で使用可能になります。

3. キュー・マネージャーに対するチャネルまたはキュー・マネージャー別名のうち、CLWLRANK が残りのすべてのチャネルまたはキュー・マネージャー別名の最大ランクを下回るものはすべて除去されます。
4. CLWLRANK が残りのすべてのキューの最大ランクを下回るキュー (キュー・マネージャー別名ではない) は、すべて除去されます。

¹ この属性は、クラスター・キューを選択する場合にのみ該当します。トピックを選択する場合は該当しません。

5. キュー、トピック、またはキュー・マネージャーの別名のインスタンスが複数残っている場合、パブリッシュ書き込みが有効なものがあれば、書き込みが無効にされたものはすべて除去されます。

注：書き込みが無効にされたインスタンスのみが残っている場合、照会操作のみが成功し、他のすべての操作は MQRC_CLUSTER_PUT_INHIBITED で失敗します。
6. キューを選択するときに、結果のキューの集合にキューのローカル・インスタンスが含まれている場合は、通常、そのローカル・インスタンスが使用されます。以下の条件のいずれかが満たされる場合は、キューのローカル・インスタンスが使用されます。
 - キューの「キューの使用」属性 CLWLUSEQ が LOCAL に設定されている。
 - 次の記述の両方に当てはまる場合:
 - キューの「キューの使用」属性 CLWLUSEQ が QMGR に設定されている。
 - キュー・マネージャーの「キューの使用」属性 CLWLUSEQ が LOCAL に設定されている。
 - メッセージは、ローカル・アプリケーションによって書き込まれずに、クラスター・チャンネルを介して受信される。
 - CLWLUSEQ(ANY) を使用してローカルに定義されたキュー、またはキュー・マネージャーから同じ設定を継承しているローカルに定義されたキューの場合、より広範囲の適用条件において、次の事項が該当します。
 - ローカル・キューは、このキューと同じクラスター内にあるローカル定義の CLUSRCVR チャンネルの status に基づいて選択されます。この状況が、メッセージを同名のリモート定義キューに送信する CLUSSDR チャンネルの状況と比較されます。

例えば、キューと同じクラスター内に1つの CLUSRCVR が存在するとします。この CLUSRCVR の状況は STOPPING であり、クラスター内の同名のキューの状況が RUNNING または INACTIVE であるとします。この場合、リモート・チャンネルが選択され、ローカル CLUSSDR チャンネルは使用されません。
 - ローカル・キューは、同名のリモート定義キューにメッセージを送信する同じ状況の CLUSSDR チャンネルと比較した CLUSRCVR チャンネルの number に基づいて選択されます。

例えば、キューと同じクラスターに4つの CLUSRCVR チャンネルがあり、また1つの CLUSSDR チャンネルがあるとします。すべてのチャンネルの状況が同一 (INACTIVE または RUNNING) であるとします。この場合、選択可能な5つのチャンネルと、2つのキュー・インスタンスが存在します。メッセージの4/5 (80%) はローカル・キューに入れられます。
7. 複数のキュー・マネージャーが残っている場合、中断されていないキュー・マネージャーがあれば、中断されているものはすべて除去されます。
8. キューまたはトピックの複数のリモート・インスタンスが残っている場合は、非アクティブまたは実行状態のチャンネルすべてが含まれます。以下の状態定数がリストされます。
 - MQCHS_INACTIVE
 - MQCHS_RUNNING
9. キューまたはトピックのリモート・インスタンスが1つも残っていない場合は、バインディング、初期化、開始、または停止のいずれかの処理中状態であるすべてのチャンネルが含まれます。以下の状態定数がリストされます。
 - MQCHS_BINDING
 - MQCHS_INITIALIZING
 - MQCHS_STARTING
 - MQCHS_STOPPING
10. キューまたはトピックのリモート・インスタンスが1つも残っていない場合は、再試行中のすべてのチャンネルが含まれます。以下の状態定数がリストされます。
 - MQCHS_RETRYING
11. キューまたはトピックのリモート・インスタンスが1つも残っていない場合は、要求中、休止、または停止状態のすべてのチャンネルが含まれます。以下の状態定数がリストされます。

- MQCHS_REQUESTING
 - MQCHS_PAUSED
 - MQCHS_STOPPED
 - MQCHS_SWITCHING
12. いずれかのキュー・マネージャーでキューまたはトピックの複数のリモート・インスタンスが残っている場合は、各キュー・マネージャーで最高の NETPRTY 値を持つチャンネルが選択されます。
 13. 優先度 (CLWLPRTY) が最も高いチャンネルと別名以外の、残りのチャンネルとキュー・マネージャー別名がすべて除去されます。キュー・マネージャー別名が残っている場合は、キュー・マネージャーに対するチャンネルは保持されます。
 14. キューが選択される場合は、次のようになります。
 - 優先度 (CLWLPRTY) が最も高いキュー以外のキューはすべて除去され、チャンネルは保持されます。
 15. 残りのチャンネルは、最も低い MQWDR.DestSeqNumber の値を持つチャンネルを除去することにより、最近使用されたチャンネル CLWLNRUC の最大許容数を超えないように削減されます。
- 注：内部クラスター制御メッセージは、同じクラスター・ワークロード・アルゴリズムを使用して送信されます (適切な場合)。

有効な宛先リストが算出された後、以下のロジックを使用して、それらの宛先の間でメッセージのワークロード・バランシングが行われます。

- 宛先のリモート・インスタンスが複数残っていて、その宛先に対するすべてのチャンネルの CLWLWGHT がデフォルトの設定値 50 に設定されている場合は、最も使用頻度の少ないチャンネルが選択されます。これは、複数のリモート・インスタンスが存在する場合のラウンドロビン方式のワークロード・バランシングとほぼ同一です。
- 宛先のリモート・インスタンスが複数残っていて、それらのキューに対する 1 つ以上のチャンネルの CLWLWGHT がデフォルト以外の設定値に設定されている場合 (それらすべてがデフォルト以外の同じ値に設定されている場合も含む)、ルーティングは、各チャンネルの相対ウェイトと、各チャンネルがこれまでメッセージの送信時に選択された合計回数によって決まります。
- 複数のインスタンスが存在する単一クラスター・キューのメッセージ分散を観察したときに、一部のキュー・インスタンスの間で分散が平衡化されなくなるように見える場合があります。この原因は、そのキューのメッセージ・トラフィックだけではなく、このキュー・マネージャーの各クラスター送信側チャンネルの過去の使用も平衡化されているからです。この動作が望ましくない場合は、以下の手順のいずれかを実行してください。
 - 均一な分散が必要な場合は、すべてのクラスター受信側チャンネルの CLWLWGHT を 50 に設定します。
 - または、特定のキュー・インスタンスに他とは異なるウェイトを設定する必要がある場合は、それらのキューを、専用のクラスター受信側チャンネルが定義された専用のクラスターに定義します。この操作により、これらのキューのワークロード・バランシングは、クラスター内の他のものとは分離されます。
- 使用可能なクラスター受信側チャンネルのクラスター・ワークロード属性が変更されたり、クラスター受信側チャンネルの状況が使用可能になると、チャンネルの平衡化のために使用される履歴データはリセットされます。手動で定義されたクラスター送信側チャンネルのワークロード属性を変更しても、履歴データはリセットされません。
- クラスター・ワークロード出口のロジックについて検討している場合、選択されるチャンネルは、MQWDR.DestSeqFactor が最も小さいチャンネルです。チャンネルが選択されるたびに、この値は約 1000/CLWLWGHT 増分されます。最も小さい値を持つチャンネルが複数ある場合は、MQWDR.DestSeqNumber 値が最も小さいチャンネルのいずれかが選択されます。

クラスターの管理や保守の際には、メッセージがチャンネルを介して流れるため、ユーザー・メッセージの分散は必ずしも正確ではありません。その結果、ユーザー・メッセージが不均一に分散され、一定になるまでに時間がかかることがあります。管理とユーザー・メッセージが混在しているので、ワークロード・バランシング中にメッセージが正確に分散されるとは限りません。

関連資料

[クラスター・ワークロード・バランシング-チャンネル属性](#)

クラスター・ワークロード・バランシングで使用されるチャンネル属性のアルファベット順リスト。

クラスター・ワークロード・バランシング-キュー属性

クラスター・ワークロード・バランシングで使用するキュー属性のアルファベット順リスト。

クラスター・ワークロード・バランシング-キュー・マネージャー属性

クラスター・ワークロード・バランシングで使用するキュー・マネージャー属性のアルファベット順リスト。

Asynchronous behavior of CLUSTER commands on z/OS

The command issuer of a cluster command on z/OS receives confirmation a command has been sent, but not that it has completed successfully.

For both REFRESH CLUSTER and RESET CLUSTER, message CSQM130I is sent to the command issuer indicating that a request has been sent. This message is followed by message CSQ9022I to indicate that the command has completed successfully, in that a request has been sent. It does not indicate that the cluster request has been completed successfully.

Any errors are reported to the z/OS console on the system where the channel initiator is running, they are not sent to the command issuer.

The asynchronous behavior is in contrast to CHANNEL commands. A message indicating that a channel command has been accepted is issued immediately. At some later time, when the command has been completed, a message indicating either normal or abnormal completion is sent to the command issuer.

Related concepts

クラスターでのワークロード・バランシング

1つのクラスターに同じキューのインスタンスが複数含まれる場合、IBM MQ はメッセージを送付するキュー・マネージャーを選択します。また、クラスター・ワークロード管理アルゴリズムと、クラスター・ワークロード固有のいくつかの属性を使用して、使用するのに最適なキュー・マネージャーを判別します。

Related tasks

Checking that async commands for distributed networks have finished

Related reference

チャンネル定義コマンドで使用可能なクラスター属性

チャンネル定義コマンドで指定できるクラスター属性

キュー定義コマンドで使用可能なクラスター属性

キュー定義コマンドに指定できるクラスター属性です。

キュー・マネージャー定義コマンドで使用可能なクラスター属性

キュー・マネージャー定義コマンドに指定できるクラスター属性です。

DISPLAY CLUSQMGR

DISPLAY CLUSQMGR コマンドは、クラスター内にあるキュー・マネージャーに関するクラスター情報を表示する場合に使用します。

REFRESH CLUSTER

REFRESH CLUSTER コマンドは、クラスターに関するローカルに保持されたすべての情報を廃棄する場合にキュー・マネージャーから発行します。例外的な状況を除いて、このコマンドを使用する必要はほとんどありません。

RESET CLUSTER: クラスターからキュー・マネージャーを強制的に除去する

RESET CLUSTER コマンドは、例外的な状況にあるクラスターからキュー・マネージャーを強制的に削除する場合に使用します。

SUSPEND QMGR、RESUME QMGR、およびクラスター

SUSPEND QMGR および RESUME QMGR コマンドは、このキュー・マネージャーへのインバウンド・クラスター・アクティビティを一時的に(例えば、このキュー・マネージャーの保守を行う前に)削減し、後で復元するために使用します。

チャンネル・プログラム

このセクションでは、チャンネルで使用可能なさまざまなタイプのチャンネル・プログラム (MCA) について検討します。

MCA の名前が、以下の表に示されています。

プログラム名	接続の方向	通信
amqrmppa		任意
runmqslr	インバウンド	任意
amqcrs6a	インバウンド	LU 6.2
amqcrsta	インバウンド	TCP
runmqchl	アウトバウンド	任意
runmqchi	アウトバウンド	任意

runmqslr (IBM MQ リスナーの実行)、runmqchl (IBM MQ チャンネルの実行)、および runmqchi (IBM MQ チャンネル・イニシエーターの実行) は、コマンド行で入力できる制御コマンドです。

amqcrsta は、AIX and Linux システム上のリスナーが開始されていない TCP チャンネルに対して、inetd を使用して呼び出されます。

LU6.2 を使用する場合、amqcrs6a はトランザクション・プログラムとして起動されます。

IBM i IBM i における相互通信ジョブ

以下のジョブは、IBM i 上での相互通信と関連付けられています。これらのジョブの名前を、次の表に示します。

ジョブ名	説明
AMQCLMAA	非スレッド化リスナー
AMQCRSTA	非スレッド化応答側ジョブ
AMQRMPPA	チャンネル・プール・ジョブ
RUNMQCHI	チャンネル・イニシエーター
RUNMQCHL	チャンネル・ジョブ
RUNMQLSR	スレッド化リスナー

IBM i IBM i におけるチャンネルの状態

チャンネルの状態は、「Work with Channels (チャンネルの処理)」パネルに表示されます。

状態名	意味
STARTING	チャンネルは、ターゲット MCA との折衝準備が整っている。
BINDING	セッションと初期データ変更を確立する。
REQUESTING	要求側チャンネルが接続を開始する。
実行中	転送または転送動作可能にする。
PAUSED	メッセージ再試行間隔の間待機する。

状態名	意味
STOPPING	再試行または停止を確立する。
RETRYING	次の再試行を試みるまで待機する。
STOPPED	エラーまたはチャンネル終了コマンドが出されたためチャンネルが停止した。
INACTIVE	チャンネルが正常に処理を終了したか、チャンネルが開始しない。
*なし	状態なし (サーバー接続チャンネルのみ)

ALW 例: AIX, Linux, and Windows でのメッセージ・チャンネルの計画

ここでは、2つのキュー・マネージャーを接続して、それらの間でメッセージを送信できるようにする方法の詳細な例を示します。

このタスクについて

例ではすべて、MQSC コマンドはコマンドのファイル内にあるものとして、またはコマンド行に入力されたものとして示されます。これらの2つの方法は同じに見えますが、コマンド行でコマンドを実行するには、まず、`runmqsc` (デフォルト・キュー・マネージャーの場合) または `runmqsc qmname` (`qmname` は必要なキュー・マネージャーの名前) を入力する必要があります。その後で、例に示されているように任意の数のコマンドを入力します。

別の方法として、これらのコマンドの入ったファイルを作成することがあります。こうすることで、コマンド内のどのエラーも簡単に訂正することができます。ファイル `MQSC.in` を呼び出してから、それをキュー・マネージャー `QMNAME` で実行するためには、次のように入力します。

```
runmqsc QMNAME < mqsc.in > mqsc.out
```

ファイル内のコマンドは、実行する前に、次のコマンドを使用して検査することができます。

```
runmqsc -v QMNAME < mqsc.in > mqsc.out
```

移植性を考慮して、コマンド行の長さは、最高 72 文字に限定されています。1 行を超える場合は、連結文字を使用してください。Windows では、`Ctrl-z` を使用してコマンド行の入力を終了させます。AIX and Linux システムでは、`Ctrl-d` を使用します。あるいは、`end` コマンドを使用します。

151 ページの図 7 はシナリオ例を示しています。

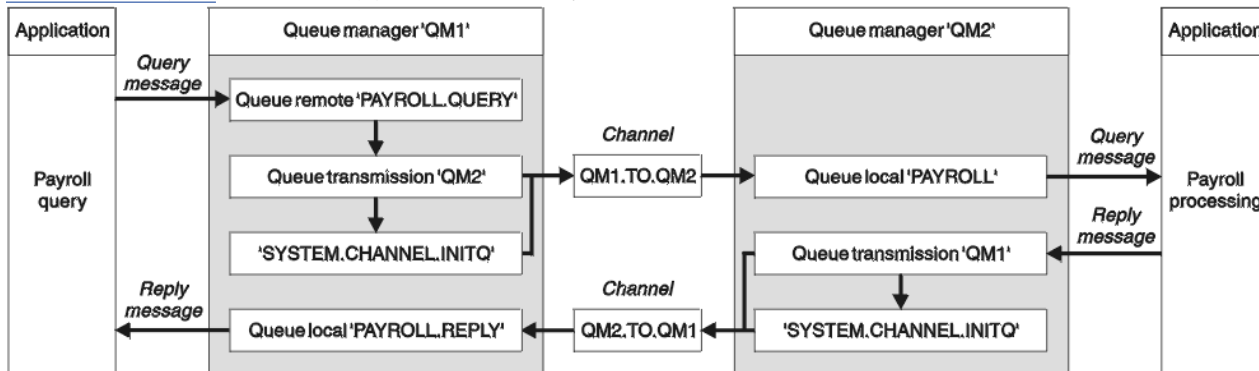


図 7. AIX, Linux, and Windows システムのメッセージ・チャンネルの例

例には、キュー・マネージャー QM1 に接続された給与計算照会アプリケーションが含まれますが、これは、キュー・マネージャー QM2 で実行している給与計算処理アプリケーションに給与計算照会メッセージ

を送信します。給与計算照会アプリケーションの照会に対する応答が QM1 に戻されなければなりません。給与計算照会メッセージは QM1.TO.QM2 と呼ばれる送信側 - 受信側チャンネルを介して、QM1 から QM2 へ送信され、その応答メッセージは QM2.TO.QM1 と呼ばれる別の送信側 - 受信側チャンネルを介して、QM2 から QM1 へと戻されます。これらのチャンネルはいずれも、他のキュー・マネージャーに送信するメッセージを得るとただちに、それをトリガーとして始動します。

給与計算照会アプリケーションは、QM1 で定義されているリモート・キュー "PAYROLL.query" に照会メッセージを入れます。このリモート・キュー定義は、QM2 でローカル・キュー "PAYROLL" として解決されます。さらに、給与計算照会アプリケーションでは、照会に対する応答が QM1 のローカル・キュー "PAYROLL.reply" に送信されるように指定されています。給与計算処理アプリケーションは、QM2 のローカル・キュー "PAYROLL" からメッセージを取得して、必要な宛先へ応答を送信します。この例では、QM1 のローカル・キュー "PAYROLL.REPLY" が宛先になります。

TCP/IP の定義の例では、QM1 は、ホスト・アドレスが 192.0.2.0 で、ポート 1411 で listen 中であり、QM2 はホスト・アドレスが 192.0.2.1 で、ポート 1412 で listen 中です。この例では、これらがすでに運用システムで定義済みであり、使用可能であるものと想定しています。

QM1 で作成する必要があるオブジェクト定義は次のとおりです。

- リモート・キュー定義 PAYROLL.QUERY
- 伝送キュー定義 QM2 (デフォルト値 = リモート・キュー・マネージャー名)
- 送信側チャンネル定義 QM1.TO.QM2
- 受信側チャンネル定義 QM2.TO.QM1
- 応答先キュー定義 PAYROLL.REPLY

QM2 で作成する必要があるオブジェクト定義は次のとおりです。

- ローカル・キュー定義 PAYROLL
- 伝送キュー定義 QM1 (デフォルト値 = リモート・キュー・マネージャー名)
- 送信側チャンネル定義 QM2.TO.QM1
- 受信側チャンネル定義 QM1.TO.QM2

接続の詳細は、送信側チャンネル定義の CONNAME 属性内に示されています。

この配置を表すダイアグラムが [151 ページの図 7](#) に示されています。

手順

以下を参照してください。

- メッセージ・チャンネルのセットアップについて詳しくは、[152 ページの『AIX, Linux, and Windows でのメッセージ・チャンネルの例のセットアップ』](#)を参照してください。
- 他の製品 (CICS など) を使用する方法、およびより多くのアプリケーションとユーザー出口を接続する方法に関する提案については、[154 ページの『AIX, Linux, and Windows の例の実行および展開』](#)を参照してください。

ALW AIX, Linux, and Windows でのメッセージ・チャンネルの例のセットアップ

These object definitions allow applications connected to queue manager QM1 to send request messages to a queue called PAYROLL on QM2, and to receive replies on a queue called PAYROLL.REPLY on QM1, and allow applications connected to queue manager QM2 to retrieve request messages from a local queue called PAYROLL, and to put replies to these request messages to a queue called PAYROLL.REPLY on queue manager QM1.

このタスクについて

どのオブジェクト定義にも DESCR と REPLACE の属性が指定されています。提供されているその他の属性は、例を機能させるために必要な最小限の属性です。提供されていない属性は、キュー・マネージャー QM1 および QM2 のデフォルト値を取ります。

リモート・キュー定義を用意して、応答を QM1 に戻す必要はありません。ローカル・キュー PAYROLL から取り出されるメッセージのメッセージ記述子には、応答先キューと応答先キュー・マネージャーの両方の名前が入っています。したがって、QM2 が応答先キュー・マネージャー名を、キュー・マネージャー QM2 の伝送キューの応答先キュー・マネージャー名として解決できるのであれば、応答メッセージを送信することができます。この例では、応答先キュー・マネージャー名は QM1 であるため、キュー・マネージャー QM2 には同じ名前の伝送キューが必要です。

手順

- キュー・マネージャー QM1 で以下のコマンドを実行します。

- a) リモート・キュー定義をセットアップします。

```
DEFINE QREMOTE(PAYROLL.QUERY) DESCR('Remote queue for QM2') REPLACE +  
PUT(ENABLED) XMITQ(QM2) RNAME(PAYROLL) RQMNAME(QM2)
```

注: リモート・キュー定義は物理キューではありませんが、伝送キュー QM2 にメッセージを送って、それらをキュー・マネージャー QM2 に送信できるようにします。

- b) 伝送キュー定義をセットアップします。

```
DEFINE QLOCAL(QM2) DESCR('Transmission queue to QM2') REPLACE +  
USAGE(XMITQ) PUT(ENABLED) GET(ENABLED) TRIGGER TRIGTYPE(FIRST) +  
INITQ(SYSTEM.CHANNEL.INITQ) PROCESS(QM1.TO.QM2.PROCESS)
```

最初のメッセージがこの伝送キューに書き込まれると、開始キュー SYSTEM.CHANNEL.INITQ にトリガー・メッセージが送信されます。チャンネル・イニシエーターは、開始キューからメッセージを入手して、指定されたプロセス内に示されているチャンネルを始動させます。

- c) 送信側チャンネル定義をセットアップします。

```
DEFINE CHANNEL(QM1.TO.QM2) CHLTYPE(SDR) TRPTYPE(TCP) +  
REPLACE DESCR('Sender channel to QM2') XMITQ(QM2) +  
CONNAME('192.0.2.1(1412)')
```

- d) 受信側チャンネル定義をセットアップします。

```
DEFINE CHANNEL(QM2.TO.QM1) CHLTYPE(RCVR) TRPTYPE(TCP) +  
REPLACE DESCR('Receiver channel from QM2')
```

- e) reply-to_queue 定義をセットアップします。

```
DEFINE QLOCAL(PAYROLL.REPLY) REPLACE PUT(ENABLED) GET(ENABLED) +  
DESCR('Reply queue for replies to query messages sent to QM2')
```

応答先キューは PUT(ENABLED) として定義されます。これによって、応答メッセージが確実にキューに書き込まれるようになります。応答先キューに応答を入れることができない場合、応答は、QM1 の送達不能キューに送信されるか、あるいは、このキューが使用できない場合は、キュー・マネージャー QM2 の伝送キュー QM1 に残されます。このキューは GET(ENABLED) として定義され、応答メッセージを検索することができます。

- キュー・マネージャー QM2 で以下のコマンドを実行します。

- a) ローカル・キュー定義をセットアップします。

```
DEFINE QLOCAL(PAYROLL) REPLACE PUT(ENABLED) GET(ENABLED) +
DESCR('Local queue for QM1 payroll details')
```

このキューは、キュー・マネージャー QM1 における応答先キュー定義の場合と同じ理由で、PUT(ENABLED) および GET(ENABLED) として定義されます。

b) 伝送キュー定義をセットアップします。

```
DEFINE QLOCAL(QM1) DESCR('Transmission queue to QM1') REPLACE +
USAGE(XMITQ) PUT(ENABLED) GET(ENABLED) TRIGGER TRIGTYPE(FIRST) +
INITQ(SYSTEM.CHANNEL.INITQ) PROCESS(QM2.TO.QM1.PROCESS)
```

最初のメッセージがこの伝送キューに書き込まれると、開始キュー SYSTEM.CHANNEL.INITQ にトリガー・メッセージが送信されます。チャンネル・イニシエーターは、開始キューからメッセージを入手して、指定されたプロセス内に示されているチャンネルを始動させます。

c) 送信側チャンネル定義をセットアップします。

```
DEFINE CHANNEL(QM2.TO.QM1) CHLTYPE(SDR) TRPTYPE(TCP) +
REPLACE DESCR('Sender channel to QM1') XMITQ(QM1) +
CONNAME('192.0.2.0(1411)')
```

d) 受信側チャンネル定義をセットアップします。

```
DEFINE CHANNEL(QM1.TO.QM2) CHLTYPE(RCVR) TRPTYPE(TCP) +
REPLACE DESCR('Receiver channel from QM1')
```

ALW

AIX, Linux, and Windows の例の実行および展開

チャンネル・イニシエーターおよびリスナーの開始に関する情報と、このシナリオの展開についての提案を示します。

このタスクについて

これらの定義が作成されたら、以下に示すことを行う必要があります。

- 各キュー・マネージャーに対してチャンネル・イニシエーターを開始させます。
- 各キュー・マネージャーに対してリスナーを開始させます。

例を展開することもできます。

手順

1. チャンネル・イニシエーターおよびリスナーを開始します。

[Windows の通信のセットアップ](#) および [AIX and Linux システムでの通信のセットアップ](#) を参照してください。

2. この例を展開するには、以下のようになります。

- LU 6.2 通信を使用して、CICS システムとの相互接続およびトランザクション処理を行います。
- キュー、プロセス、チャンネル定義をさらに追加して、他のアプリケーションが2つのキュー・マネージャー間でメッセージを送信できるようにします。
- チャンネルにユーザー出口プログラムを追加して、リンク暗号化、セキュリティチェック、または追加のメッセージ処理を行うことができますようにします。
- キュー・マネージャーの別名および応答先キューの別名を使用して、使用されているキュー・マネージャーのネットワーク構成において、それらをどのように使用できるかを理解します。

2つの IBM i キュー・マネージャーを接続して、それらの間でメッセージを送信できるようにする方法の詳細な例。例には、キュー・マネージャー QM1 に接続された給与計算照会アプリケーションが含まれますが、これは、キュー・マネージャー QM2 で実行している給与計算処理アプリケーションに給与計算照会メッセージを送信します。給与計算照会アプリケーションの照会に対する応答が QM1 に戻されなければなりません。

このタスクについて

この例は、アプリケーションがキュー・マネージャー QM1 を使用してメッセージをキュー・マネージャー QM2 のキューに入れるようにするために必要な準備を示しています。QM2 で実行されるアプリケーションは、これらのメッセージを検索して、QM1 の応答キューに応答を送信することができます。

この例では、TCP/IP 接続を介しての通信を想定しています。また、各チャネルは、対応する伝送キューに最初のメッセージが到達したときに、それをトリガーとして始動するものと想定しています。

この例では、開始キューとして SYSTEM.CHANNEL.INITQ を使用します。このキューは既に IBM MQ で定義されています。異なる開始キューを使用できますが、それを自分で定義し、STRMQMCHLI コマンドを使用してチャネル・イニシエーターの新規インスタンスを開始し、それに開始キューの名前を付ける必要があります。チャネルのトリガー操作の詳細については、[チャネルのトリガー操作](#)を参照してください。

注: **MQ Adv.** **CD** TCP/IP を使用するメッセージ・チャネルは、ネットワーク・スループットを大幅に向上させることができる高速 TCP/IP トンネルを提供する IBM Aspera® faspio Gateway を指すことができます。Linux または Windows での Aspera gateway 接続の定義を参照してください。

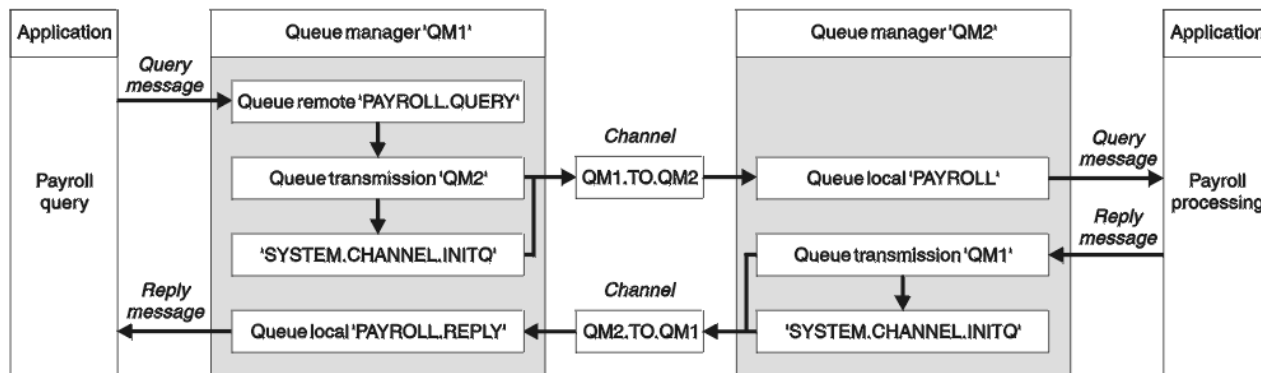


図 8. IBM MQ for IBM i のメッセージ・チャネルの例

給与計算照会メッセージは QM1.TO.QM2 と呼ばれる送信側 - 受信側チャネルを介して、QM1 から QM2 へ送信され、その応答メッセージは QM2.TO.QM1 と呼ばれる別の送信側 - 受信側チャネルを介して、QM2 から QM1 へと戻されます。これらのチャネルはいずれも、他のキュー・マネージャーに送信するメッセージを得るとただちに、それをトリガーとして始動します。

給与計算照会アプリケーションは、QM1 で定義されているリモート・キュー "PAYROLL.query" に照会メッセージを入れます。このリモート・キュー定義は、QM2 でローカル・キュー "PAYROLL" として解決されます。さらに、給与計算照会アプリケーションでは、照会に対する応答が QM1 のローカル・キュー "PAYROLL.reply" に送信されるように指定されています。給与計算処理アプリケーションは、QM2 のローカル・キュー "PAYROLL" からメッセージを取得して、必要な宛先へ応答を送信します。この例では、QM1 のローカル・キュー "PAYROLL.REPLY" が宛先になります。

いずれのキュー・マネージャーも IBM i 上で実行されるものと想定しています。この定義の例では、QM1 は、ホスト・アドレスが 192.0.2.0 で、ポート 1411 で listen 中です。QM2 は、ホスト・アドレスが 192.0.2.1 で、ポート 1412 で listen 中です。この例では、これらのキュー・マネージャーがすでに IBM i システムで定義済みであり、使用可能であるものと想定しています。

QM1 で作成する必要があるオブジェクト定義は次のとおりです。

- リモート・キュー定義 PAYROLL.QUERY
- 伝送キュー定義 QM2 (デフォルト値 = リモート・キュー・マネージャー名)

- 送信側チャンネル定義 QM1.TO.QM2
- 受信側チャンネル定義 QM2.TO.QM1
- 応答先キュー定義 PAYROLL.REPLY

QM2 で作成する必要があるオブジェクト定義は次のとおりです。

- ローカル・キュー定義 PAYROLL
- 伝送キュー定義 QM1 (デフォルト値 = リモート・キュー・マネージャー名)
- 送信側チャンネル定義 QM2.TO.QM1
- 受信側チャンネル定義 QM1.TO.QM2

接続の詳細は、送信側チャンネル定義の CONNAME 属性内に示されています。

この配置を表すダイアグラムが [155 ページの図 8](#) に示されています。

手順

以下を参照してください。

- メッセージ・チャンネルのセットアップについて詳しくは、[156 ページの『IBM iでのメッセージ・チャンネル・エージェントのセットアップ』](#)を参照してください。
- より多くのアプリケーションおよびユーザー出口を接続する方法に関する提案については、[159 ページの『IBM iの例の実行および展開』](#)を参照してください。

IBM i IBM iでのメッセージ・チャンネル・エージェントのセットアップ

The following object definitions allow applications connected to queue manager QM1 to send request messages to a queue called PAYROLL on QM2, to receive replies on a queue called PAYROLL.REPLY on QM1, allow applications connected to queue manager QM2 to retrieve request messages from a local queue called PAYROLL, and to put replies to these request messages to a queue called PAYROLL.REPLY on queue manager QM1.

このタスクについて

どのオブジェクト定義にも TEXT 属性が与えられています。それ以外には、この例を実施するための最小限の必須属性が与えられています。提供されていない属性は、キュー・マネージャー QM1 および QM2 のデフォルト値を取ります。

リモート・キュー定義を用意して、応答を QM1 に戻す必要はありません。ローカル・キュー PAYROLL から取り出されるメッセージのメッセージ記述子には、応答先キューと応答先キュー・マネージャーの両方の名前が入っています。したがって、QM2 が応答先キュー・マネージャー名を、キュー・マネージャー QM2 の伝送キューの応答先キュー・マネージャー名として解決できるのであれば、応答メッセージを送信することができます。この例では、応答先キュー・マネージャー名は QM1 であるため、キュー・マネージャー QM2 には同じ名前の伝送キューが必要です。

手順

- キュー・マネージャー QM1 で以下のコマンドを実行します。
 - a) 以下の属性を指定した CRTMQMQ コマンドを使用して、リモート・キュー定義をセットアップします。

QNAME	'PAYROLL.QUERY'
QTYPE	*RMT
TEXT	'Remote queue for QM2'
PUTENBL	*YES
TMQNAME	'QM2' (デフォルト = リモート・キュー・マネージャー名)

RMTQNAME 'PAYROLL'
RMTMQMNAME 'QM2'

注：リモート・キュー定義は物理キューではありませんが、伝送キュー QM2 にメッセージを送って、それらをキュー・マネージャー QM2 に送信できるようにします。

- b) 以下の属性を指定した CRTMQMQ コマンドを使用して、伝送キュー定義をセットアップします。

QNAME QM2
QTYPE *LCL
TEXT 'Transmission queue to QM2'
USAGE *TMQ
PUTENBL *YES
GETENBL *YES
TRGENBL *YES
TRGTYPE *FIRST
INITQNAME SYSTEM.CHANNEL.INITQ
TRIGDATA QM1.TO.QM2

最初のメッセージがこの伝送キューに書き込まれると、開始キュー SYSTEM.CHANNEL.INITQ にトリガー・メッセージが送信されます。チャンネル・イニシエーターは、開始キューからメッセージを入力して、指定されたプロセス内に示されているチャンネルを始動させます。

- c) 以下の属性を指定した CRTMQCHL コマンドを使用して、送信側チャンネル定義をセットアップします。

CHLNAME QM1.TO.QM2
CHLTYPE *SDR
TRPTYPE *TCP
TEXT 'Sender channel to QM2'
TMQNAME QM2
CONNNAME '192.0.2.1(1412)'

- d) 以下の属性を指定した CRTMQCHL コマンドを使用して、受信側チャンネル定義をセットアップします。

CHLNAME QM2.TO.QM1
CHLTYPE *RCVR
TRPTYPE *TCP
TEXT 'Receiver channel from QM2'

- e) 以下の属性を指定した CRTMQMQ コマンドを使用して、応答先キュー定義をセットアップします。

QNAME PAYROLL.REPLY
QTYPE *LCL
TEXT 'Reply queue for replies to query messages sent to QM2'
PUTENBL *YES
GETENBL *YES

応答先キューは PUT(ENABLED) として定義されます。この定義によって、応答メッセージが確実にキューに書き込まれるようになります。応答先キューに応答を入れることができない場合、応答は、QM1 の送達不能キューに送信されるか、あるいは、このキューが使用できない場合は、キュー・マネージャー QM2 の伝送キュー QM1 に残されます。このキューは GET(ENABLED) として定義され、応答メッセージを検索することができます。

- キュー・マネージャー QM2 で以下のコマンドを実行します。

- a) 以下の属性を指定した CRTMQMQ コマンドを使用して、ローカル・キュー定義をセットアップします。

QNAME	PAYROLL
QTYPE	*LCL
TEXT	'Local queue for QM1 payroll details'
PUTENBL	*YES
GETENBL	*YES

このキューは、キュー・マネージャー QM1 における応答先キュー定義の場合と同じ理由で、PUT(ENABLED) および GET(ENABLED) として定義されます。

- b) 以下の属性を指定した CRTMQMQ コマンドを使用して、伝送キュー定義をセットアップします。

QNAME	QM1
QTYPE	*LCL
TEXT	'Transmission queue to QM1'
USAGE	*TMQ
PUTENBL	*YES
GETENBL	*YES
TRGENBL	*YES
TRGTYPE	*FIRST
INITQNAME	SYSTEM.CHANNEL.INITQ
TRIGDATA	QM2.TO.QM1

最初のメッセージがこの伝送キューに書き込まれると、開始キュー SYSTEM.CHANNEL.INITQ にトリガー・メッセージが送信されます。チャンネル・イニシエーターは、開始キューからメッセージを入手して、トリガー・データで示されているチャンネルを始動させます。

- c) 以下の属性を指定した CRTMQMCHL コマンドを使用して、送信側チャンネル定義をセットアップします。

CHLNAME	QM2.TO.QM1
CHLTYPE	*SDR
TRPTYPE	*TCP
TEXT	'Sender channel to QM1'
TMQNAME	QM1
CONNNAME	'192.0.2.0(1411)'

- d) 以下の属性を指定した CRTMQMCHL コマンドを使用して、受信側チャンネル定義をセットアップします。

CHLNAME	QM1.TO.QM2
CHLTYPE	*RCVR

TRPTYPE	*TCP
TEXT	'Receiver channel from QM1'

IBM i IBM i の例の実行および展開

チャンネル・イニシエーターおよびリスナーの開始に関する情報と、このシナリオの展開についての提案を示します。

このタスクについて

これらの定義が作成されたら、以下に示すことを行う必要があります。


- 各キュー・マネージャーに対してチャンネル・イニシエーターを開始させます。
- 各キュー・マネージャーに対してリスナーを開始させます。

これで、アプリケーションは相互にメッセージを送信できるようになります。いずれかの伝送キューに最初のメッセージが到達すると、チャンネルがトリガーされて始動するので、STRMQMCHL コマンドを出す必要はありません。

例を展開することもできます。

手順

1. チャンネル・イニシエーターおよびリスナーを開始します。
チャンネル・イニシエーターとリスナーの開始について詳しくは、[IBM i でのチャンネルのモニターと制御を参照してください](#)。
2. この例を展開するには、以下のようになります。
 - キュー定義およびチャンネル定義をさらに追加して、他のアプリケーションが2つのキュー・マネージャー間でメッセージを送信できるようにします。
 - チャンネルにユーザー出口プログラムを追加して、リンク暗号化、セキュリティー検査、または追加のメッセージ処理を行うことができますようにします。
 - キュー・マネージャーの別称および応答先キューの別称を使用して、使用されているキュー・マネージャーのネットワーク構成において、これらのオブジェクトをどのように使用できるかを理解します。

 MQSC コマンドを使用するこの例の詳細については、[159 ページの『Example: planning a message channel on z/OS』](#)を参照してください。



Example: planning a message channel on z/OS

How to connect z/OS or MVS queue managers together so that messages can be sent between them. This example involves a payroll query application connected to queue manager QM1 that sends payroll query messages to a payroll processing application running on queue manager QM2. The payroll query application needs the replies to its queries sent back to QM1.

About this task

The example illustrates the preparations needed to allow an application using queue manager QM1 to put messages on a queue at queue manager QM2. An application running on QM2 can retrieve these messages, and send responses to a reply queue on QM1.

The example illustrates the use of both TCP/IP and LU 6.2 connections. The example assumes that channels are to be triggered to start when the first message arrives on the transmission queue they are servicing.

Note:   TCP/IP を使用するメッセージ・チャンネルは、ネットワーク・スループットを大幅に向上させることができる高速 TCP/IP トンネルを提供する IBM Aspera faspio Gateway を指すことができます。 [Linux または Windows での Aspera gateway 接続の定義](#)を参照してください。

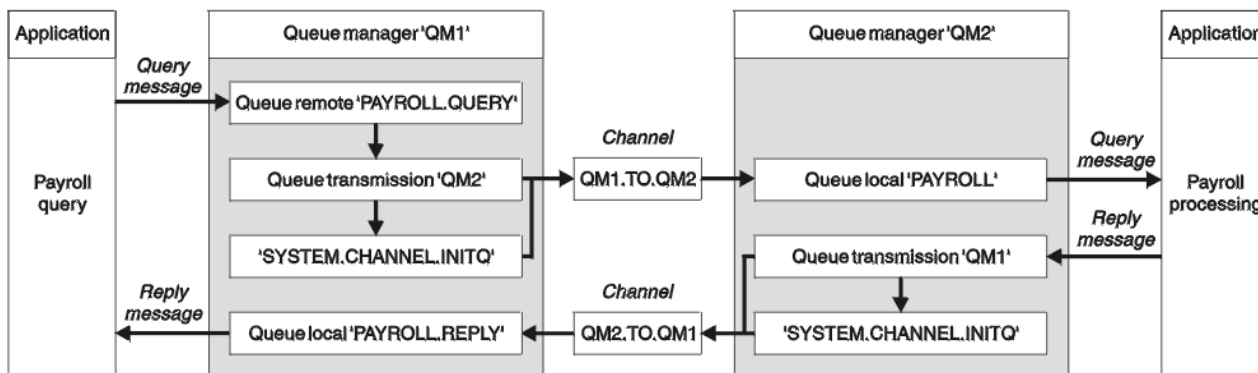


Figure 9. The first example for IBM MQ for z/OS

The payroll query messages are sent from QM1 to QM2 on a sender-receiver channel called QM1.TO.QM2, and the reply messages are sent back from QM2 to QM1 on another sender-receiver channel called QM2.TO.QM1. Both of these channels are triggered to start as soon as they have a message to send to the other queue manager.

The payroll query application puts a query message to the remote queue "PAYROLL.QUERY" defined on QM1. This remote queue definition resolves to the local queue "PAYROLL" on QM2. In addition, the payroll query application specifies that the reply to the query is sent to the local queue "PAYROLL.REPLY" on QM1. The payroll processing application gets messages from the local queue "PAYROLL" on QM2, and sends the replies to wherever they are required; in this case, local queue "PAYROLL.REPLY" on QM1.

Both queue managers are assumed to be running on z/OS. In the example definitions for TCP/IP, QM1 has a host address of 192.0.2.0 and is listening on port 1411, and QM2 has a host address of 192.0.2.1 and is listening on port 1412. In the definitions for LU 6.2, QM1 is listening on a symbolic luname called LUNAME1 and QM2 is listening on a symbolic luname called LUNAME2. The example assumes that these lunames are already defined on your z/OS system and available for use. To define them, see [“Example: setting up IBM MQ cross-platform communication on z/OS”](#) on page 44.

The object definitions that need to be created on QM1 are:

- Remote queue definition, PAYROLL.QUERY
- Transmission queue definition, QM2 (default=remote queue manager name)
- Sender channel definition, QM1.TO.QM2
- Receiver channel definition, QM2.TO.QM1
- Reply-to queue definition, PAYROLL.REPLY

The object definitions that need to be created on QM2 are:

- Local queue definition, PAYROLL
- Transmission queue definition, QM1 (default=remote queue manager name)
- Sender channel definition, QM2.TO.QM1
- Receiver channel definition, QM1.TO.QM2

The example assumes that all the SYSTEM.COMMAND.* and SYSTEM.CHANNEL.* queues required to run DQM have been defined as shown in the supplied sample definitions, **CSQ4INSG** and **CSQ4INSX**.

The connection details are supplied in the CONNAME attribute of the sender channel definitions.

You can see a diagram of the arrangement in [Figure 9 on page 160](#).

Procedure

See:

- [“Setting up the message channel agent on z/OS”](#) on page 161 for details on setting up the message channels

- “[Running and expanding the example for z/OS](#)” on page 163 for suggestions on how you can connect more applications and user exits.

z/OS

Setting up the message channel agent on z/OS

The following object definitions allow applications connected to queue manager QM1 to send request messages to a queue called PAYROLL on QM2 and also allows applications to receive replies on a queue called PAYROLL.REPLY on QM1. The definitions also allow applications connected to queue manager QM2 to retrieve request messages from a local queue called PAYROLL, and to put replies to these request messages to a queue called PAYROLL.REPLY on queue manager QM1.

About this task

All the object definitions have been provided with the DESCR and REPLACE attributes and are the minimum required to make the example work. The attributes that are not supplied take the default values for queue managers QM1 and QM2.

You do not need to provide a remote queue definition to enable the replies to be returned to QM1. The message descriptor of the message retrieved from local queue PAYROLL contains both the reply-to queue and the reply-to queue manager names. Therefore, as long as QM2 can resolve the reply-to queue manager name to that of a transmission queue on queue manager QM2, the reply message can be sent. In this example, the reply-to queue manager name is QM1 and so queue manager QM2 requires a transmission queue of the same name.

Procedure

- Run the following commands on queue manager QM1:
 - a) Setup the remote queue definition:

```
DEFINE QREMOTE(PAYROLL.QUERY) DESCR('Remote queue for QM2') REPLACE +
PUT(ENABLED) XMITQ(QM2) RNAME(PAYROLL) RQMNAME(QM2)
```

Note: The remote queue definition is not a physical queue, but a means of directing messages to the transmission queue, QM2, so that they can be sent to queue manager QM2.

- b) Setup the transmission queue definition:

```
DEFINE QLOCAL(QM2) DESCR('Transmission queue to QM2') REPLACE +
USAGE(XMITQ) PUT(ENABLED) GET(ENABLED) TRIGGER TRIGTYPE(FIRST) +
TRIGDATA(QM1.TO.QM2) INITQ(SYSTEM.CHANNEL.INITQ)
```

When the first message is put on this transmission queue, a trigger message is sent to the initiation queue, SYSTEM.CHANNEL.INITQ. The channel initiator gets the message from the initiation queue and starts the channel identified in the trigger data. The channel initiator can only get trigger messages from the SYSTEM.CHANNEL.INITQ queue, so do not use any other queue as the initiation queue.

- c) Setup the sender channel definition:

For a TCP/IP connection:

```
DEFINE CHANNEL(QM1.TO.QM2) CHLTYPE(SDR) TRPTYPE(TCP) +
REPLACE DESCR('Sender channel to QM2') XMITQ(QM2) +
CONNAME('192.0.2.1(1412)')
```

For an LU 6.2 connection:

```
DEFINE CHANNEL(QM1.TO.QM2) CHLTYPE(SDR) TRPTYPE(LU62) +
REPLACE DESCR('Sender channel to QM2') XMITQ(QM2) +
CONNAME('LUNAME2')
```

d) Setup the receiver channel definition:

For a TCP/IP connection:

```
DEFINE CHANNEL(QM2.TO.QM1) CHLTYPE(RCVR) TRPTYPE(TCP) +  
REPLACE DESCR('Receiver channel from QM2')
```

For an LU 6.2 connection:

```
DEFINE CHANNEL(QM2.TO.QM1) CHLTYPE(RCVR) TRPTYPE(LU62) +  
REPLACE DESCR('Receiver channel from QM2')
```

e) Setup the reply-to queue definition:

```
DEFINE QLOCAL(PAYROLL.REPLY) REPLACE PUT(ENABLED) GET(ENABLED) +  
DESCR('Reply queue for replies to query messages sent to QM2')
```

The reply-to queue is defined as PUT(ENABLED) which ensures that reply messages can be put to the queue. If the replies cannot be put to the reply-to queue, they are sent to the dead-letter queue on QM1 or, if this queue is not available, remain on transmission queue QM1 on queue manager QM2. The queue has been defined as GET(ENABLED) to allow the reply messages to be retrieved.

- Run the following commands on queue manager QM2:

a) Setup the local queue definition:

```
DEFINE QLOCAL(PAYROLL) REPLACE PUT(ENABLED) GET(ENABLED) +  
DESCR('Local queue for QM1 payroll details')
```

This queue is defined as PUT(ENABLED) and GET(ENABLED) for the same reason as the reply-to queue definition on queue manager QM1.

b) Setup the transmission queue definition:

```
DEFINE QLOCAL(QM1) DESCR('Transmission queue to QM1') REPLACE +  
USAGE(XMITQ) PUT(ENABLED) GET(ENABLED) TRIGGER TRIGTYPE(FIRST) +  
TRIGDATA(QM2.TO.QM1) INITQ(SYSTEM.CHANNEL.INITQ)
```

When the first message is put on this transmission queue, a trigger message is sent to the initiation queue, SYSTEM.CHANNEL.INITQ. The channel initiator gets the message from the initiation queue and starts the channel identified in the trigger data. The channel initiator can only get trigger messages from SYSTEM.CHANNEL.INITQ so do not use any other queue as the initiation queue.

c) Setup the sender channel definition:

For a TCP/IP connection:

```
DEFINE CHANNEL(QM2.TO.QM1) CHLTYPE(SDR) TRPTYPE(TCP) +  
REPLACE DESCR('Sender channel to QM1') XMITQ(QM1) +  
CONNNAME('192.0.2.0(1411)')
```

For an LU 6.2 connection:

```
DEFINE CHANNEL(QM2.TO.QM1) CHLTYPE(SDR) TRPTYPE(LU62) +  
REPLACE DESCR('Sender channel to QM1') XMITQ(QM1) +  
CONNNAME('LUNAME1')
```

d) Setup the receiver channel definition:

For a TCP/IP connection:

```
DEFINE CHANNEL(QM1.TO.QM2) CHLTYPE(RCVR) TRPTYPE(TCP) +  
REPLACE DESCR('Receiver channel from QM1')
```

For an LU 6.2 connection:

```
DEFINE CHANNEL(QM1.TO.QM2) CHLTYPE(RCVR) TRPTYPE(LU62) +  
REPLACE DESCR('Receiver channel from QM1')
```

Running and expanding the example for z/OS

Information about starting the channel initiator and listener and suggestions for expanding on this example.

About this task

Once these definitions have been created, you need to:

- Start the channel initiator on each queue manager.
- Start the listener for each queue manager.

The applications can then send messages to each other. Because the channels are triggered to start by the arrival of the first message on each transmission queue, you do not need to issue the START CHANNEL MQSC command.

You can also expand the example.

Procedure

1. Start the channel initiator and listener.

See [Starting a channel initiator](#), and [Starting a channel listener](#) for details on how to start a channel initiator and listener.

2. You can expand this example by:

- Adding more queues, and channel definitions to allow other applications to send messages between the two queue managers.
- Adding user exit programs on the channels to allow for link encryption, security checking, or additional message processing.
- Using queue manager aliases and reply-to queue aliases to understand more about how these aliases can be used in the organization of your queue manager network.

Example: planning a message channel for z/OS using queue sharing groups

This example illustrates the preparations needed to allow an application using queue manager QM3 to put a message on a queue in a queue sharing group that has queue members QM4 and QM5, and also shows the IBM MQ commands (MQSC) that you can use in IBM MQ for z/OS for distributed queuing with queue sharing groups.

About this task

Ensure you are familiar with the example in [“Example: planning a message channel on z/OS” on page 159](#) before trying this one. This example expands the payroll query scenario of that example, to show how to add higher availability of query processing by adding more serving applications to serve a shared queue.

The payroll query application is now connected to queue manager QM3 and puts a query to the remote queue 'PAYROLL QUERY' defined on QM3. This remote queue definition resolves to the shared queue

'PAYROLL' hosted by the queue managers in the queue sharing group QSG1. The payroll processing application now has two instances running, one connected to QM4 and one connected to QM5.

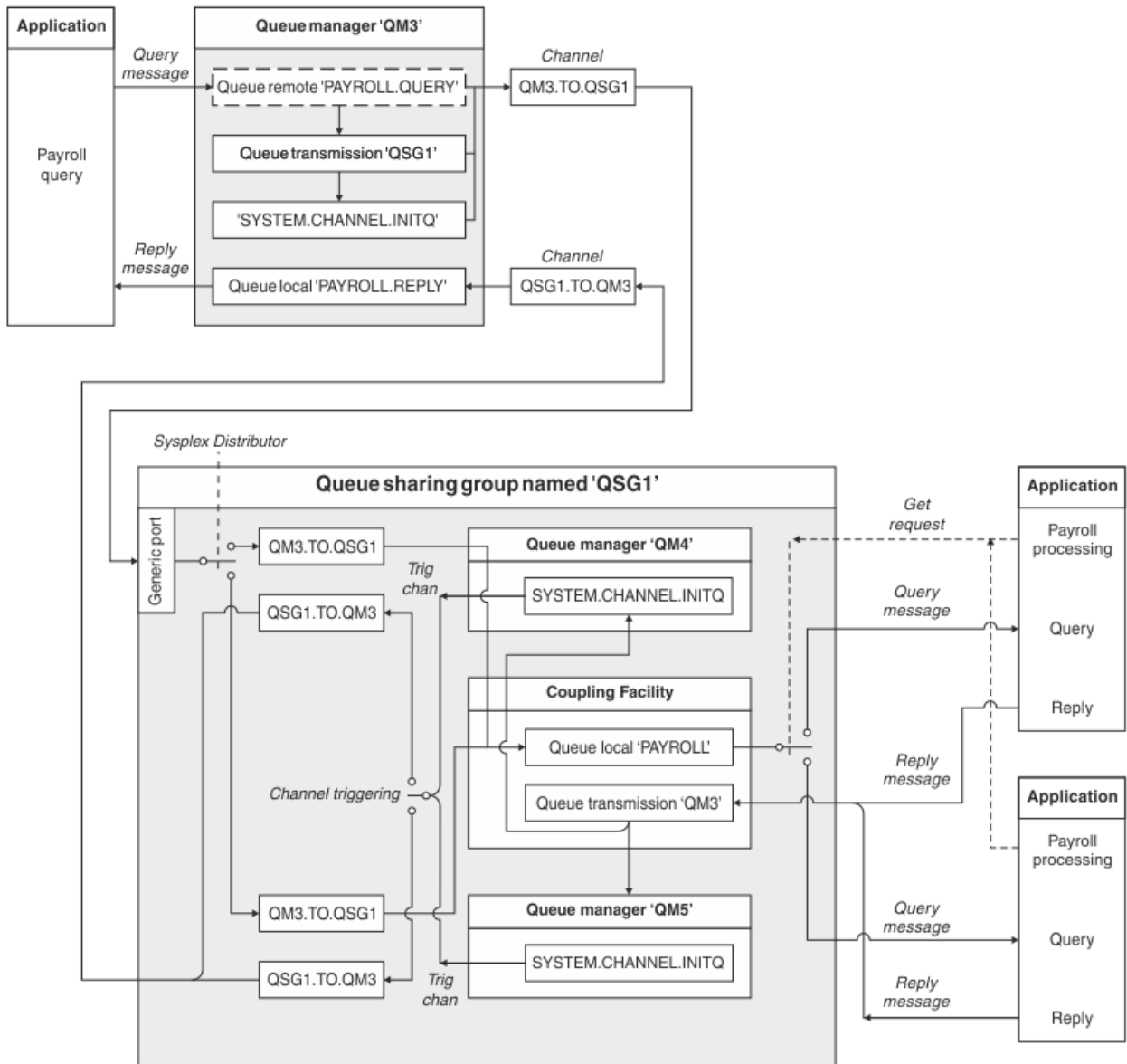


Figure 10. Message channel planning example for IBM MQ for z/OS using queue sharing groups

All three queue managers are assumed to be running on z/OS. In the example definitions for TCP/IP, QM4 has a VIPA address of MVSIP01 and QM5 has a VIPA address of MVSIP02. Both queue managers are listening on port 1414. The generic address that Sysplex Distributor provides for this group is QSG1.MVSIP. QM3 has a host address of 192.0.2.0 and is listening on port 1411.

In the example definitions for LU6.2, QM3 is listening on a symbolic luname called LUNAME1. The name of the generic resource defined for VTAM for the lunames listened on by QM4 and QM5 is LUQSG1. The example assumes that they are already defined on your z/OS system and are available for use. To define them see [“Establishing an LU 6.2 connection into a queue sharing group”](#) on page 49.

In this example QSG1 is the name of a queue sharing group, and queue managers QM4 and QM5 are the names of members of the group.

Procedure

See:

- [“Setting up the queue sharing group definitions and a queue manager QM3 not in the queue sharing group” on page 165](#) for details on setting up the definitions.
- [“Running the queue sharing group example for z/OS” on page 166](#) for details on starting the channel initiators and listeners for each queue manager.

Setting up the queue sharing group definitions and a queue manager QM3 not in the queue sharing group

Producing the following object definitions for one member of the queue sharing group makes them available to all the other members. QM3 is not a member of the queue sharing group.

About this task

Queue managers QM4 and QM5 are members of the queue sharing group. The definitions produced for QM4 are also available for QM5.

The coupling facility list structure is assumed to be called 'APPLICATION1'. If it is not called 'APPLICATION1', you must use your own coupling facility list structure name for the example.

As QM3 is not a member of the queue sharing group you need the object definitions for that queue manager to allow it to put messages to a queue in the queue sharing group.

Procedure

- Setup the shared objects for the queue sharing group definition:
 - a) Use the following commands to setup the shared object definitions that are stored in Db2, and their associated messages that are stored within the coupling facility.

```
DEFINE QLOCAL(PAYROLL) QSGDISP(SHARED) REPLACE PUT(ENABLED) GET(ENABLED) +
CFSTRUCT(APPLICATION1) +
DESCR('Shared queue for payroll details')

DEFINE QLOCAL(QM3) QSGDISP(SHARED) REPLACE USAGE(XMITQ) PUT(ENABLED) +
CFSTRUCT(APPLICATION1) +
DESCR('Transmission queue to QM3') TRIGGER TRIGTYPE(FIRST) +
TRIGDATA(QSG1.TO.QM3) GET(ENABLED) INITQ(SYSTEM.CHANNEL.INITQ)
```

- Use the following commands to setup the group object definitions that are stored in Db2®. Each queue manager in the queue sharing group creates a local copy of the defined object.
 - a) Setup the sender channel:

Sender channel definition for a TCP/IP connection:

```
DEFINE CHANNEL(QSG1.TO.QM3) CHLTYPE(SDR) QSGDISP(GROUP) TRPTYPE(TCP) +
REPLACE DESCR('Sender channel to QM3') XMITQ(QM3) +
CONNNAME('192.0.2.0(1411)')
```

Sender channel definition for an LU 6.2 connection:

```
DEFINE CHANNEL(QSG1.TO.QM3) CHLTYPE(SDR) QSGDISP(GROUP) TRPTYPE(LU62) +
REPLACE DESCR('Sender channel to QM3') XMITQ(QM3) +
CONNNAME('LUNAME1')
```

- b) Setup the receiver channel:

Receiver channel definition for a TCP/IP connection:

```
DEFINE CHANNEL(QM3.TO.QSG1) CHLTYPE(RCVR) TRPTYPE(TCP) +
REPLACE DESCR('Receiver channel from QM3') QSGDISP(GROUP)
```

Receiver channel definition for an LU 6.2 connection:

```
DEFINE CHANNEL(QM3.TO.QSG1) CHLTYPE(RCVR) TRPTYPE(LU62) +
REPLACE DESCR('Receiver channel from QM3') QSGDISP(GROUP)
```

- Setup queue manager QM3 object definitions.

a) Setup the CONNAME

The CONNAME for this channel is the generic address of the queue sharing group, which varies according to transport type.

For a TCP/IP connection:

```
DEFINE CHANNEL(QM3.TO.QSG1) CHLTYPE(SDR) TRPTYPE(TCP) +
REPLACE DESCR('Sender channel to QSG1') XMITQ(QSG1) +
CONNAME('QSG1.MVSIP(1414)')
```

For an LU 6.2 connection:

```
DEFINE CHANNEL(QM3.TO.QSG1) CHLTYPE(SDR) TRPTYPE(LU62) +
REPLACE DESCR('Sender channel to QSG1') XMITQ(QSG1) +
CONNAME('LUQSG1') TPNAME('MQSERIES') MODENAME('#INTER')
```

b) Setup the other definitions.

These definitions are required for the same purposes as those used in the sub topics for [“Example: planning a message channel on z/OS”](#) on page 159.

```
DEFINE QREMOTE(PAYROLL.QUERY) DESCR('Remote queue for QSG1') REPLACE +
PUT(ENABLED) XMITQ(QSG1) RNAME(APPL) RQMNAME(QSG1)

DEFINE QLOCAL(QSG1) DESCR('Transmission queue to QSG1') REPLACE +
USAGE(XMITQ) PUT(ENABLED) GET(ENABLED) TRIGGER TRIGTYPE(FIRST) +
TRIGDATA(QM3.TO.QSG1) INITQ(SYSTEM.CHANNEL.INITQ)

DEFINE CHANNEL(QSG1.TO.QM3) CHLTYPE(RCVR) TRPTYPE(TCP) +
REPLACE DESCR('Receiver channel from QSG1')

DEFINE CHANNEL(QSG1.TO.QM3) CHLTYPE(RCVR) TRPTYPE(LU62) +
REPLACE DESCR('Receiver channel from QSG1')

DEFINE QLOCAL(PAYROLL.REPLY) REPLACE PUT(ENABLED) GET(ENABLED) +
DESCR('Reply queue for replies to query messages sent to QSG1')
```

z/OS

Running the queue sharing group example for z/OS

Information about starting the channel initiators and listeners.

About this task

After you have created the required objects, you need to:

- Start the channel initiator for all three queue managers.
- Start the listeners for both queue managers.

Procedure

1. Start the channel initiators.

See [Starting a channel initiator](#) for details on how to start a channel initiator.

2. Start the listeners.

See [Starting a channel listener](#) for details on how to start a listener.

For a TCP/IP connection, each member of the group must have a group listener started that is listening on port 1414.

```
STA LSTR PORT(1414) IPADDR(MVSIP01) INDISP(GROUP)
```

The previous entry starts the listener on QM4, for example.

For an LU6.2 connection, each member of the group must have a group listener started that is listening on a symbolic luname. This luname must correspond to the generic resource LUQSG1.

```
STA LSTR PORT(1411)
```

The previous entry starts the listener on QM3.

別名を使用して MQ ライブラリーを参照する

MQ ライブラリーの名前を直接使用する代わりに、別名を定義して JCL で MQ ライブラリーを参照することができます。その後、MQ ライブラリーの名前が変更された場合には、単に別名を削除して再定義することができます。

例

以下の例では、MQ ライブラリー MQM.V600.SCSQANLE を参照するための別名 MQM.SCSQANLE を定義します。

```
//STEP1 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
DELETE (MQM.SCSQANLE)
DEFINE ALIAS (NAME(MQM.SCSQANLE) RELATE(MQM.V600.SCSQANLE))
/*
```

その後、JCL で MQM.V600.SCSQANLE ライブラリーを参照するには、別名 MQM.SCSQANLE を使用します。

注：ライブラリー名と別名は同じカタログに含まれる必要があるため、両方に同じ高位修飾子を使用してください（この例で高位修飾子は MQM です）。

Managed File Transfer 構成リファレンス

Managed File Transfer の構成に役立つ参照情報。

MFT プロパティーでの環境変数の使用

ファイルまたはディレクトリーの場所を表す Managed File Transfer プロパティーで環境変数を使用することができます。これにより、製品の一部を実行するときに使用されるファイルまたはディレクトリーの場所を、現在の環境（例えば、コマンドを実行するユーザー）に応じて変えることができます。

以下のプロパティーは、ファイルまたはディレクトリーの場所を受け入れるため、環境変数を含めることができます。

- agentQMGrAuthenticationCredentialsFile
- agentSslKeyStore
- agentSslKeyStoreCredentialsFile
- agentSslTrustStore
- agentSslTrustStoreCredentialsFile
- cdNodeKeystoreCredentialsFile
- cdNodeTruststoreCredentialsFile
- cdTmpDir
- cdNodeKeystore

- cdNodeTruststore
- commandPath
- connectionQMgrAuthenticationCredentialsFile
- connectionSslKeyStore
- connectionSslKeyStoreCredentialsFile
- connectionSslTrustStore
- connectionSslTrustStoreCredentialsFile
- coordinationSslKeyStore
- coordinationSslKeyStoreCredentialsFile
- coordinationQMgrAuthenticationCredentialsFile
- coordinationSslTrustStore
- coordinationSslTrustStoreCredentialsFile
- exitClassPath
- exitNativeLibraryPath
- javaCoreTriggerFile
- loggerQMgrAuthenticationCredentialsFile
- sandboxRoot
- transferRoot
- wmqfte.database.credentials.file

Windows の例

Windows Windows システムで、ユーザー fteuser が次のように環境変数 USERPROFILE を使用したとします。

```
wmqfte.database.credentials.file=%USERPROFILE%\logger\mqmftcredentials.xml
```

これは次のファイル・パスに解決されます。

```
C:\Users\fteuser\logger\mqmftcredentials.xml
```

AIX and Linux の例

Linux **AIX** UNIX システムで、ユーザー fteuser が次のように環境変数 HOME を使用したとします。

```
transferRoot=$HOME/fte/
```

これは次のファイル・パスに解決されます。

```
/home/fteuser/fte/
```

関連資料

196 ページの『[MFT coordination.properties ファイル](#)』

coordination.properties ファイルは、調整キュー・マネージャーとの接続の詳細を指定します。複数の Managed File Transfer インストール済み環境で同じ調整キュー・マネージャーが共用される場合があるため、共用ドライブ上の共通の coordination.properties ファイルへのシンボリック・リンクを使用できます。

[200 ページの『MFT command.properties ファイル』](#)

`command.properties` ファイルには、ユーザーがコマンドを発行するときに接続先となるコマンド・キュー・マネージャーと、Managed File Transfer がそのキュー・マネージャーと情報のやり取りをするのに必要な情報を指定します。

[173 ページの『MFT agent.properties ファイル』](#)

各 Managed File Transfer Agent には独自のプロパティ・ファイル `agent.properties` があります。このファイルには、エージェントがキュー・マネージャーに接続するために使用する情報が含まれている必要があります。`agent.properties` ファイルには、エージェントの動作を変更するプロパティを含めることもできます。

[MFT 用の SSL/TLS プロパティ](#)

[204 ページの『MFT logger.properties ファイル』](#)

Managed File Transfer ロガーには、一連の構成プロパティがあります。これらのプロパティを `logger.properties` ファイルで指定します。このファイルは、`MQ_DATA_PATH/mqft/config/coordination_qmgr_name/loggers/logger_name` ディレクトリーにあります。

[ユーザー出口用の MFT エージェント・プロパティ](#)

[プロトコル・ブリッジ・プロパティ・ファイルのフォーマット](#)

[Connect:Direct プロセスの定義ファイルのフォーマット](#)

[Connect:Direct ノードのプロパティ・ファイルのフォーマット](#)

MFT installation.properties ファイル

`installation.properties` ファイルは、構成オプションのデフォルト・セットの名前を指定します。この項目は、Managed File Transfer を、使用する構成が含まれる一連の構造化されたディレクトリーおよびプロパティ・ファイルに向けます。一般に、構成オプションのセット名は、関連付けられている調整キュー・マネージャーの名前です。

このファイルはインストーラーによって作成され、**`fteChangeDefaultConfigurationOptions`** コマンドを使用して変更できます。

`installation.properties` ファイルは `MQ_DATA_PATH` ディレクトリーにあります。例えば、Windows の場合、デフォルトのファイルの場所は

`MQ_DATA_PATH\mqft\installations\installation_name` で、AIX and Linux システムの場合、デフォルトのファイルの場所は `/var/mqm/mqft/installations/installation_name` です。

Redistributable Managed File Transfer Agent については、データ・パスは **`fteCreateEnvironment`** コマンドの実行時に設定されます。このコマンドを実行するときに、選択した場所を **`-d`** パラメーターで指定すると、その場所がデータ・パスに設定されます。**`fteCreateEnvironment`** コマンドで場所を指定しない場合、Redistributable Managed File Transfer Agent が解凍されたルート・ディレクトリーの下にディレクトリー `mftdata` が作成されます。Redistributable Managed File Transfer Agent の

`installation.properties` ファイルは、`MQ_DATA_PATH\mqft\installations\MFTZipInstall` ディレクトリーにあります。

`installation.properties` ファイルには以下の値が含まれます。

表 64. 基本プロパティ

プロパティ名	説明	デフォルト値
commandMessagePriority	<p>fteStopAgent、fteCancelTransfer、およびftePingAgent コマンドに関する内部メッセージとコマンド・メッセージの両方の優先順位を設定します。</p> <p>例えば、多数の小さいファイルを転送するための大量の転送要求を立て続けに送信する場合、新しい転送要求をソース・エージェントのコマンド・キューで待機状態にすることができます。外部メッセージと内部メッセージにはデフォルトの IBM MQ メッセージ優先順位が設定されていて、内部メッセージは新しい転送要求によってブロックされるようになっています。これにより、転送ネゴシエーション時間を超過して、転送がリカバリー状態になることがあります。</p> <p>また、commandMessagePriority プロパティを使用すると、内部確認応答メッセージと確認応答が必要なメッセージの優先順位を設定することもできます。</p> <p>内部 Managed File Transfer メッセージに新しい転送要求より高い優先順位を設定するには、このプロパティを 1 (最低の優先順位) から 9 (最高の優先順位) までの値に設定します。</p> <p>commandMessagePriority プロパティのデフォルト値は 8 です。これは、エージェント・コマンド・キューの IBM MQ 属性 DEFPRTY (デフォルト優先順位) が 7 以下の場合の場合に、内部のネゴシエーション・メッセージが新しい転送要求よりも優先されるという意味です。DEFPRTY 属性の値が 8 または 9 のいずれかに設定されている場合、commandMessagePriority プロパティの有効性を維持するには、DEFPRTY プロパティまたは commandMessagePriority プロパティのいずれかを変更する必要があります。</p>	デフォルト値は 8 です。

表 64. 基本プロパティ (続き)		
プロパティ名	説明	デフォルト値
commonCredentialsKeyFile	<p>資格情報の暗号化時に使用される資格情報鍵を含むファイルの完全修飾パス名。MFT 資格情報ファイルの最も一般的な名前は <code>MQMFTCredentials.xml</code> です。</p> <p><code>commonCredentialsKeyFile</code> プロパティの使用について詳しくは、資格情報の暗号化解除を参照してください。</p>	キー・ファイルの完全修飾パス
defaultProperties	<p>構成オプションのデフォルト・セットの名前。この値は、構成ディレクトリーにあるディレクトリーの名前です。このディレクトリーには複数のディレクトリーと、構成情報を指定するプロパティ・ファイルが含まれています。</p>	デフォルトなし
enableFunctionalFixPack	<p>使用可能にするフィックスパック機能のレベル。デフォルトでは、フィックスパックに組み込まれている新しい機能は使用可能になっていません。このプロパティをバージョン ID に設定すると、そのバージョンで使用できる新しいフィーチャーを使用可能にすることができます。</p> <p>バージョン ID は、ピリオド文字 (.) を使用しても使用しなくても指定できます。例えば、IBM MQ 8.0.0 Fix Pack 2 で使用できる機能を使用するには、このプロパティを <code>8002</code> または <code>8.0.0.2</code> に設定してください。</p>	デフォルトなし

表 64. 基本プロパティ (続き)		
プロパティ名	説明	デフォルト値
messagePublicationFormat	<p>MFT エージェントがその状況 XML メッセージ用に使用するメッセージ・パブリケーション形式を指定できます。このプロパティは、以下の値に設定できます。</p> <p>messagePublicationFormat=mixed メッセージは MQMD FORMAT なしでパブリッシュされます (MQFMT_NONE)。ただし、/LOG トピック・ツリーの下で発行されるメッセージは例外で、MQFMT_STRING の MQMD 形式でパブリッシュされます。</p> <p>messagePublicationFormat=MQFMT_NONE メッセージは MQMD FORMAT なしでパブリッシュされます。</p> <p>messagePublicationFormat=MQFMT_STRING メッセージはストリング形式でパブリッシュされます。</p>	messagePublicationFormat=mixed
<p>z/OS z/OS 固有:</p>		
productID	<p>MFT 使用率の記録の対象となる製品タイプ。</p> <ul style="list-style-type: none"> • スタンドアロン Managed File Transfer 製品。(MFT が productID)。 • IBM MQ Advanced 製品の一部。(ADVANCED が productID)。 • z/OS IBM MQ Advanced for z/OS Value Unit Edition 製品の一部。(ADVANCEDVUE が productID)。 <p>製品使用記録の詳細については、製品情報のレポートを参照してください。</p> <p>Multi マルチプラットフォームでは、このプロパティは無視されます。</p>	MFT

以下のテキストは、installation.properties ファイルの内容の例です。

```
defaultProperties=ERIS
```

ERIS は、`installation.properties` ファイルと同じディレクトリーにあるディレクトリーの名前で、ディレクトリー ERIS には複数のディレクトリーと一連の構成オプションを記述するプロパティー・ファイルが含まれています。

関連概念

[MFT 構成オプション \(Multiplatforms\)](#)

関連資料

[fteChangeDefaultConfigurationOptions](#)

MFT agent.properties ファイル

各 Managed File Transfer Agent には独自のプロパティー・ファイル `agent.properties` があります。このファイルには、エージェントがキュー・マネージャーに接続するために使用する情報が含まれている必要があります。`agent.properties` ファイルには、エージェントの動作を変更するプロパティーを含めることもできます。

`agent.properties` ファイルは、インストーラー、**`fteCreateAgent`**、**`fteCreateBridgeAgent`**、または **`fteCreateCDAgent`** コマンドによって作成されます。これらのコマンドのいずれかを **`-f`** フラグと共に使用して、基本エージェント・キュー・マネージャー・プロパティーおよび、作成しているエージェントのタイプに関連付けられている拡張エージェント・プロパティーを変更できます。拡張エージェント・プロパティーを変更または追加するには、テキスト・エディターでファイルを編集する必要があります。

Multi Multiplatforms では、エージェントの `agent.properties` ファイルは `MQ_DATA_PATH/mqft/config/coordination_qmgr_name/agents/agent_name` ディレクトリーにあります。

z/OS z/OS では、`agent.properties` ファイルの場所は `$BFG_CONFIG variable/mqft/config/coordination_qmgr_name/agents/agent_name` です。

`agent.properties` ファイルを変更した場合は、エージェントを再起動してその変更が反映されるようにする必要があります。

ファイルまたはディレクトリーの場所を表すいくつかの Managed File Transfer プロパティーでは、環境変数を使用できます。これにより、製品の一部の実行時に使用されるファイルまたはディレクトリーの場所を、環境の変更 (プロセスを実行しているユーザーなど) に合わせて変えることができます。詳しくは、[167 ページの『MFT プロパティーでの環境変数の使用』](#)を参照してください。

Windows

注: Windows では、次の 2 つのプロパティーがあります。

- `windowsService`
- `windowsServiceVersion`

これらは Windows サービスとして実行するエージェントをセットアップするために使用される MFT コマンドによって、`agent.properties` ファイルに追加されます。

プロパティーを手動で追加したり変更したりしないでください。手動で追加すると、エージェントが正しく動作しなくなります。

Windows サービスとして開始するようにエージェントをセットアップする方法について詳しくは、[Windows サービスとしての MFT エージェントの開始](#)を参照してください。

エージェントの基本プロパティー


各 MFT `agent.properties` ファイルには、以下のエージェント基本プロパティーが含まれています。

表 65. エージェントの基本プロパティ		
プロパティ名	説明	デフォルト値
agentName	エージェントの名前。 エージェントの名前は IBM MQ オブジェクト命名規則に準拠していなければなりません。 詳しくは、 231 ページの『MFT オブジェクト命名規則』 を参照してください。	デフォルトなし
agentDesc	エージェントの説明 (説明の作成を選択した場合)。	デフォルトなし
agentQMGr	エージェントのキュー・マネージャーの名前。	デフォルトなし
agentQMGrHost	エージェントのキュー・マネージャーのホスト名または IP アドレス。	デフォルトなし
agentQMGrPort	エージェントのキュー・マネージャーとのクライアント接続で使用されるポート番号。	1414
agentQMGrChannel	エージェントのキュー・マネージャーとの接続に使用される SVRCONN チャンネル名。	SYSTEM.DEF.SVRCONN
agentType	エージェントのタイプ。 <ul style="list-style-type: none"> • 標準非ブリッジ・エージェント (STANDARD) • プロトコル・ブリッジ・エージェント (BRIDGE) • Connect:Direct® ブリッジ・エージェント (CD_BRIDGE) • IBM Integration Bus によって使用される組み込みエージェント (EMBEDDED) • Sterling File Gateway 組み込みエージェント (SFG) 	STANDARD


agentQMGrHost プロパティの値を指定しない場合は、デフォルトでバインディング・モードが使用されます。

agentQMGrHost プロパティに値を指定し、agentQMGrPort および agentQMGrChannel プロパティには値を指定しない場合、ポート番号 1414 およびチャンネル SYSTEM.DEF.SVRCONN がデフォルトで使用されます。

拡張エージェント・プロパティ

Managed File Transfer は、エージェントの構成に役立つ拡張エージェント・プロパティも備えています。以下のプロパティのいずれかを使用する場合、手動で agent.properties ファイルを編集し、必要な拡張プロパティを追加します。括弧、コンマ (,)、および円記号 (¥) は MFT コマンドの特殊文字であり、円記号 (¥) 文字でエスケープする必要があります。  Windows 上のファイル・パスは、分離文字として二重の円記号 (\\) を使用するか、単一のスラッシュ (/) を使用して指定できます。Java プロパティ・ファイル内の文字エスケープについては、Oracle 資料 [Javadoc](#) を参照してください。

- [エージェント・サイズ・プロパティ](#)
- [コード・ページ・プロパティ](#)
- [コマンド・プロパティ](#)
- [接続プロパティ](#)
- [Connect:Direct ブリッジ・プロパティ](#)
- [ファイルからメッセージへの転送とメッセージからファイルへの転送に関するエージェント・プロパティ](#)
- [汎用エージェント・プロパティ](#)
- [高可用性プロパティ](#)
- [入出力プロパティ](#)
- [転送ログ・プロパティ](#)
- [マルチチャンネル・サポート・プロパティ](#)
- [マルチインスタンス・プロパティ](#)
- [プロセス・コントローラー・プロパティ](#)
- [プロトコル・ブリッジ・プロパティ](#)

- プロトコル・ブリッジ・エージェントのログ・プロパティ
- キュー・プロパティ
- リソース・モニター・プロパティ
- ルート・ディレクトリー・プロパティ
- スケジューラー・プロパティ
- セキュリティー・プロパティ
- SSL/TLS プロパティ
- タイムアウト・プロパティ
- 転送リカバリー・タイムアウト・プロパティ
- トレースおよびロギングのプロパティ
- 転送制限プロパティ
- ユーザー出口ルーチン・プロパティ
- IBM MQ クライアント圧縮プロパティ
-  z/OS 固有のプロパティ
- その他のプロパティ

プロパティ名	説明	デフォルト値
agentCheckpointInterval	<p>チェックポイントがリカバリー目的で設定される完全なデータ・フレーム単位での間隔。これは拡張プロパティであり、ほとんどの Managed File Transfer 構成では、このプロパティの値を変更する必要はありません。</p> <p>何らかの問題によって転送がリカバリー状態になる場合、転送をリカバリーできるのは、チェックポイントの境界までだけです。したがって、この値を大きくすると (agentChunkSize、agentWindowSize、および agentFrameSize にも大きな値を設定)、エージェントが転送をリカバリーするのに必要な時間が長くなります。転送がリカバリー状態になることがほとんどない信頼性のある Managed File Transfer ネットワークでは、この値を増やすと、全体的なパフォーマンスの改善に役立つ可能性があります。</p>	1
agentChunkSize	<p>ファイル・データのトランスポートの各転送チャンクのサイズしたがって、ソース・エージェントと宛先エージェントとの間で転送される IBM MQ メッセージの最大サイズを意味します。これは拡張プロパティであり、ほとんどの Managed File Transfer 構成では、このプロパティの値を変更する必要はありません。</p> <p>この値は、ソース・エージェントと宛先エージェントの間でネゴシエーションされ、両者の値のうち大きい方が使用されます。このプロパティの値を変更する場合は、ソース・エージェント側と宛先エージェント側の両方で値を変更してください。</p> <p>agentChunkSize は整数値です。例えば、agentChunkSize=10240 とすると、チャンク・サイズは 10 KB に設定されます。</p>	262144 バイト (256 KB 相当)
agentFrameSize	<p>転送フレームのウィンドウの数。これは拡張プロパティであり、ほとんどの Managed File Transfer 構成では、このプロパティの値を変更する必要はありません。</p> <p>待ち時間の長いネットワークでは、この値を増やすとエージェントが同時にアクティブにできるメッセージ・チャンクが増えるため、全体的なパフォーマンスの改善に役立つ可能性があります。</p> <p>このプロパティの値に agentWindowSize を乗算し、さらに agentChunkSize を乗算した値は、転送ごとのエージェントのメモリー消費量の上限を意味します。例えば、262144 バイトのチャンク $\times 10 \times 5 = 12.5$ MB (転送ごと) となります。</p> <p>注: 単一の転送で転送されるファイルのサイズが 12.5 MB より小さい場合、このプロパティの値を増やしても、転送のパフォーマンスに与える効果はありません。</p>	5

プロパティ名	説明	デフォルト値
agentWindowSize	<p>各ウィンドウのチャンクの数。これは拡張プロパティであり、ほとんどの Managed File Transfer 構成では、このプロパティの値を変更する必要はありません。</p> <p>待ち時間の長いネットワークでは、この値を増やすと、全体的なパフォーマンスの改善に役立つ可能性があります。その理由は、エージェントが同時にアクティブにできるメッセージ・チャンクが増えて、確認応答メッセージがソース・エージェントに返信される頻度が減るためです。</p> <p>このプロパティの値に agentFrameSize を乗算し、さらに agentChunkSize を乗算した値は、転送ごとのエージェントのメモリー消費量の上限、および宛先エージェントのデータ・キューでの IBM MQ メッセージ・データの上限を意味します。例えば、262144 バイトのチャンク x 10 x 5 = 上限 12.5 MB (転送ごと) となります。</p> <p>注: 単一の転送で転送されるファイルのサイズが 12.5 MB より小さい場合、このプロパティの値を増やしても、転送のパフォーマンスに与える効果はありません。</p>	10

プロパティ名	説明	デフォルト値
agentCcsid	<p>エージェントがそのエージェント・キュー・マネージャーに接続するときに使用するコード・ページ。agentCcsid の値を指定する場合は、agentCcsidName の値も指定する必要があります。JVM の既知のコード・ページを表示する方法については、fteCreateBridgeAgent コマンドの -hsc パラメーターを参照してください。</p>	1208
agentCcsidName	<p>agentCcsid の Java 表記。agentCcsidName の値を指定する場合は、agentCcsid の値も指定する必要があります。</p>	UTF8

プロパティ名	説明	デフォルト値
maxCommandHandlerThreads	<p>初期構文解析および転送コマンド・メッセージの処理に使用可能なスレッド数を制御します。アクティブである場合には、スレッドはキュー・マネージャーへの接続を必要としますが、アイドルである場合には、スレッドは接続を解放します。</p>	5
maxCommandOutput	<p>コマンド出力用に保管される最大バイト数。このプロパティは、管理対象呼び出し用に指定されたコマンド、および、管理対象転送用の preSource、postSource、preDestination、および postDestination コマンドに適用されます。この値によって、SYSTEM.FTE トピックの転送ログに書き込まれるコマンド出力の長さが制限されます。</p>	10240
maxCommandRetries	<p>エージェントが許可するコマンド再試行の最大回数。このプロパティは、管理対象呼び出し用に指定されたコマンド、および、管理対象転送用の preSource、postSource、preDestination、および postDestination コマンドに適用されます。</p>	9
maxCommandWait	<p>エージェントが許可する各再試行間の最大待ち時間 (秒単位)。このプロパティは、管理対象呼び出し用に指定されたコマンド、および、管理対象転送用の preSource、postSource、preDestination、および postDestination コマンドに適用されます。</p>	60
immediateShutdownTimeout	<p>エージェントの即時シャットダウンの場合は、このプロパティを使用して、シャットダウンを強制する前にエージェントがその転送の完了を待つ最大時間を秒単位で指定することができます。</p> <p>注: このプロパティの値をデフォルト (10 秒) 未満に変更してはなりません。エージェントの即時シャットダウンには、外部プロセスをすべて終了するための十分な時間が必要です。このプロパティの値が低すぎる場合は、プロセスが実行されたままになる可能性があります。</p> <p>このプロパティに 0 の値が指定されると、エージェントは、未解決の転送がすべて停止するのを待ちます。このプロパティに無効値が指定されると、デフォルト値が使用されます。</p>	10

表 69. 拡張エージェント・プロパティ: 接続		
プロパティ名	説明	デフォルト値
javaLibraryPath	Managed File Transfer では、バインディング・モードでキュー・マネージャーに接続するときに、IBM MQ Java バインディング・ライブラリーへのアクセス権限が必要になります。Managed File Transfer は、デフォルトでは、IBM MQ で定義されているデフォルトの場所でバインディング・ライブラリーを検索します。バインディング・ライブラリーが別の場所にある場合は、このプロパティを使用して、バインディング・ライブラリーの場所を指定してください。	なし

表 70. 拡張エージェント・プロパティ: Connect:Direct ブリッジ		
プロパティ名	説明	デフォルト値
cdNode	Connect:Direct ブリッジを使用する場合に必要なプロパティ。 Connect:Direct ブリッジ・エージェントから宛先の Connect:Direct ノードにメッセージを転送するために使用する Connect:Direct ノードの名前。このノードは、Connect:Direct ブリッジの一部であって、転送のソースまたは宛先であるリモート・ノードではありません。詳しくは、 Connect:Direct ブリッジ を参照してください。	デフォルトなし
cdNodeHost	Connect:Direct ブリッジ・エージェントから宛先ノードにファイルを転送するときに使用する Connect:Direct ノード (Connect:Direct ブリッジ・ノード) のホスト名または IP アドレス。 ほとんどの場合、Connect:Direct ブリッジ・ノードは、Connect:Direct ブリッジ・エージェントと同じシステム上にあります。その場合、このプロパティのデフォルト値 (ローカル・システムの IP アドレス) は適正です。システムに複数の IP アドレスがある場合、または Connect:Direct ブリッジ・ノードが Connect:Direct ブリッジ・エージェントと異なるシステム上にあり、両者のシステムが同一のファイル・システムを共有する場合は、このプロパティを使用して Connect:Direct ブリッジ・ノード用の適切なホスト名を指定します。 cdNode プロパティを設定しなかった場合、このプロパティは無視されます。	ローカル・システムのホスト名または IP アドレス
cdNodePort	クライアント・アプリケーションがノードとの通信に使用する Connect:Direct ブリッジ・ノードのポート番号。Connect:Direct の製品資料で、このポートは API ポートと呼ばれています。 cdNode プロパティを設定しなかった場合、このプロパティは無視されます。	1363
cimpDir	ファイルが宛先の Connect:Direct ノードに転送される前に、一時的に保管される、Connect:Direct ブリッジ・エージェントが実行されているシステム上の場所。 このプロパティは、ファイルが一時的に保管されるディレクトリーの絶対パスを指定します。例えば、cdTmpDir を /tmp に設定すると、ファイルは /tmp ディレクトリーに一時的に置かれます。 Connect:Direct ブリッジ・エージェントと Connect:Direct ブリッジ・ノードは、このパラメーターによって指定されたディレクトリーに同じパス名を使用してアクセスできる必要があります。Connect:Direct ブリッジのインストールを計画する際には、この点を考慮してください。可能な場合は、Connect:Direct ブリッジの一部である Connect:Direct ノードが配置されているシステム上にエージェントを作成します。エージェントとノードが別個のシステム上にある場合は、ディレクトリーが共有ファイル・システム上にあり、両方のシステムから同じパス名を使用してアクセスできる必要があります。サポートされる構成について詳しくは、 Connect:Direct ブリッジ を参照してください。 cdNode プロパティを設定しなかった場合、このプロパティは無視されます。 このプロパティの値には、環境変数を含めることができます。 詳しくは、 167 ページ の『MFT プロパティでの環境変数の使用』を参照してください。	<code>value_of_java.io.tmpdir</code> <code>/cdbridge-agentName</code>
cdTrace	Connect:Direct ブリッジ・エージェントと Connect:Direct ノード間で送信されるデータをエージェントがトレースするかどうかを指定します。このプロパティの値は、true または false のいずれかになります。	false
cdMaxConnectionRetries	正常な接続がまだ確立されていない場合のファイル転送で、Connect:Direct 接続を試行できる最大回数。この回数を超えると転送は失敗します。	-1 (試行回数は無制限)
cdMaxPartialWorkConnectionRetries	前回の接続試行が正常に終了し、転送作業が完了した場合に、ファイル転送で Connect:Direct 接続を試行できる最大回数。この回数を超えると転送は失敗します。	-1 (試行回数は無制限)


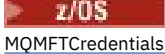
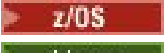
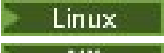
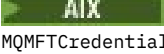

表 70. 拡張エージェント・プロパティ: Connect:Direct ブリッジ (続き)		
プロパティ名	説明	デフォルト値
cdMaxWaitForProcessEndStats	プロセス終了後、Connect:Direct ノードの統計情報内で Connect:Direct プロセス完了情報が使用できるようになるまでの最大待機時間(ミリ秒)。この時間を超えるとファイル転送が失敗と判定されます。一般に、この情報は即時に使用可能になりますが、特定の失敗条件下においては情報がパブリッシュされません。そうした条件において、このプロパティで指定した待機時間後にファイル転送は失敗します。	60000
cdAppName	Connect:Direct ブリッジ・エージェントがブリッジの一部である Connect:Direct ノードに接続するとき使用するアプリケーション名。	Managed File Transfer <i>current version</i> (<i>current version</i> は、製品のバージョン番号です)。
cdNodeLocalPortRange	Connect:Direct ブリッジ・エージェントと、ブリッジの一部である Connect:Direct ノードとの間のソケット接続で使用するローカル・ポートの範囲。この値の形式は、値または範囲のコンマ区切りリストです。 デフォルトでは、オペレーティング・システムによってローカル・ポート番号が選択されます。	なし
cdNodeProtocol	Connect:Direct ブリッジ・エージェントがブリッジの一部である Connect:Direct ノードに接続するとき使用するプロトコル。有効な値は以下のとおりです。 <ul style="list-style-type: none"> • TCPIP • SSL • TLS 	TCPIP
cdNodeKeystore	Connect:Direct ブリッジ・エージェントと、ブリッジの一部である Connect:Direct ノードとの間のセキュア通信で使用する鍵ストアのパス。 SSL または TLS に cdNodeProtocol プロパティを設定しなかった場合、このプロパティは無視されます。 このプロパティの値には、環境変数を含めることができます。 詳しくは、167 ページの『MFT プロパティでの環境変数の使用』を参照してください。	なし
cdNodeKeystoreType	cdNodeKeystore プロパティで指定する鍵ストアのファイル・フォーマット。有効な値は、jks と pkcs12 です。 SSL または TLS に cdNodeProtocol プロパティを設定しなかった場合、このプロパティは無視されます。	jks
cdNodeKeystoreCredentialsFile	cdNodeKeystore 資格情報が含まれているファイルのパス。 このプロパティの値には、環境変数を含めることができます。詳しくは、167 ページの『MFT プロパティでの環境変数の使用』を参照してください。  資格情報ファイルの作成について詳しくは、MFT と IBM MQ の接続認証を参照してください。  資格情報ファイルの作成について詳しくは、z/OS での MQMFTCredentials.xml の構成を参照してください。	このプロパティのデフォルト値は以下のとおりです。    \$HOME/ MQMFTCredentials.xml  %USERPROFILE%/ MQMFTCredentials.xml
cdNodeTruststore	Connect:Direct ブリッジ・エージェントと、ブリッジの一部である Connect:Direct ノードとの間のセキュア通信で使用するトラストストアのパス。 SSL または TLS に cdNodeProtocol プロパティを設定しなかった場合、このプロパティは無視されます。 このプロパティの値には、環境変数を含めることができます。 詳しくは、167 ページの『MFT プロパティでの環境変数の使用』を参照してください。	なし
cdNodeTruststoreType	cdNodeTruststore プロパティで指定するトラストストアのファイル・フォーマット。有効な値は、jks と pkcs12 です。 SSL または TLS に cdNodeProtocol プロパティを設定しなかった場合、このプロパティは無視されます。	jks







表 70. 拡張エージェント・プロパティ: Connect:Direct ブリッジ (続き)		
プロパティ名	説明	デフォルト値
cdNodeTruststoreCredentialsFile	<p>cdNodeTruststore 資格情報が含まれているファイルのパス。</p> <p>このプロパティの値には、環境変数を含めることができます。詳しくは、167 ページの『MFT プロパティでの環境変数の使用』を参照してください。</p> <p> 資格情報ファイルの作成については、MFT と IBM MQ の接続認証を参照してください。</p> <p> 資格情報ファイルの作成については、z/OS での MQMFTCredentials.xml の構成を参照してください。</p>	<p>このプロパティのデフォルト値は以下のとおりです。</p> <p> z/OS</p> <p> Linux</p> <p> AIX \$HOME/MQMFTCredentials.xml</p> <p> Windows %USERPROFILE%/MQMFTCredentials.xml</p>
logCDProcess	output0.log ファイルで、エージェントのイベント・ログに記録する Connect:Direct プロセス・ロギングのレベル。このプロパティがとることのできる値は None、Failures、または All です。	なし

表 71. 拡張エージェント・プロパティ: ファイルからメッセージへの転送とメッセージからファイルへの転送		
プロパティ名	説明	デフォルト値
deleteTmpFileAfterRenameFailure	このプロパティを値 false に設定すると、名前変更操作が失敗した場合に一時ファイルが宛先から削除されません。この場合、転送されたデータは、宛先の一時ファイル (.part) に残ります。このファイルは、後から手動で名前変更できます。デフォルトでは、このプロパティの値は true です。このプロパティは、メッセージからファイルへの転送とファイルからファイルへの転送の両方に適用されます。	true
enableQueueInputOutput	デフォルトでは、エージェントが転送の一環としてソース・キューからデータを読み取ったり、宛先キューにデータを書き込んだりすることはできません。この値を true に設定すると、エージェントは、ファイルからメッセージへの転送とメッセージからファイルへの転送を実行できるようになります。このプロパティの値は、true または false のいずれかになります。	false
enableSystemQueueInputOutput	エージェントが IBM MQ のシステム・キューとの間で読み取り/書き込みを実行できるかどうかを指定します。システム・キューには、SYSTEM 修飾子が接頭部として付いています。 注: システム・キューは、IBM MQ、Managed File Transfer、さらには重要な情報を送信する他のアプリケーションが使用するキューです。このプロパティを変更すると、エージェントがそれらのキューにアクセスできるようになります。このプロパティを有効にする場合は、ユーザー・サンドボックス機能を使用して、エージェントがアクセスできるキューを制限してください。	false
enableClusterQueueInputOutput	エージェントが IBM MQ クラスター・キューとの間で読み取り/書き込みを実行できるかどうかを指定します。 注: enableQueueInputOutput プロパティへの追加として、enableClusterQueueInputOutput エージェント・プロパティの指定が必要です。	false
maxDelimiterMatchLength	ファイルからメッセージへの転送の一環として 1 つのテキスト・ファイルを複数のメッセージに分割するときに使用する Java 正規表現のマッチング項目と見なせる文字の最大数。	5
maxInputOutputMessageLength	エージェントによってソース・キューから読み取られるメッセージ、または宛先キューに書き込まれるメッセージの最大長 (バイト単位)。転送のソース・エージェントの maxInputOutputMessageLength プロパティは、ソース・キューのメッセージから読み取ることができるバイト数を決定します。転送の宛先エージェントの maxInputOutputMessageLength プロパティは、宛先キューのメッセージに書き込むバイト数を決定します。メッセージの長さがこのプロパティの値を超えた場合、転送はエラーを出して失敗します。このプロパティは、Managed File Transfer の内部キューに影響を及ぼしません。このプロパティの変更については、 メッセージ・サイズに関連する MQ 属性および MFT プロパティを設定する際のガイダンス を参照してください。	1048576
monitorGroupRetryLimit	<p>メッセージ・グループがまだキューに存在している場合に、モニターがメッセージからファイルへの転送を再度起動する最大回数。メッセージからファイルへの転送が起動された回数は、グループの最初のメッセージの MQMD バックアウト・カウントで判別されます。</p> <p>エージェントが再始動した場合、転送の起動回数が monitorGroupRetryLimit の値を超えている場合でも、モニターは再度転送を起動します。この動作により、転送の起動回数が monitorGroupRetryLimit の値を超えてしまった場合、エージェントはイベント・ログにエラーを書き込みます。</p> <p>このプロパティに値 -1 が指定された場合、モニターは、起動条件が満たされなくなるまで何度でも転送を再度起動します。</p>	10


表 72. 拡張エージェント・プロパティ: 汎用		
プロパティ名	説明	デフォルト値
agentStatusPublishRateLimit	ファイル転送状況に変更があった場合に、エージェントがその状況をリパブリッシュする最大速度 (秒)。このプロパティに小さすぎる値を設定すると、IBM MQ ネットワークのパフォーマンスにマイナスの影響を与える可能性があります。	30
agentStatusPublishRateMin	エージェントが自身の状況をパブリッシュする最小速度 (秒単位)。この値は、 agentStatusPublishRateLimit プロパティの値以上でなければなりません。	300
enableMemoryAllocationChecking	このプロパティの値は、true または false のいずれかになります。転送を受け入れる前に、転送の実行に使用可能なメモリーが十分あることを Managed File Transfer Agent が検査するかどうかを決定します。この検査は、ソース・エージェントと宛先エージェントの両方で行われます。使用可能なメモリーが不足している場合は、転送はリジェクトされます。 転送に必要なメモリーの計算時には、転送に必要な最大メモリーが使用されます。そのため、値は、転送で使用される実際のメモリーより大きくなる可能性があります。このため、 enableMemoryAllocationChecking プロパティが true に設定すると、実行できる同時転送数が減少する可能性があります。メモリー不足エラーのために Managed File Transfer で障害が起こる問題が発生している場合のみ、このプロパティを true に設定することをお勧めします。大量のメモリーを消費する可能性のある転送には、メッセージのサイズが大きい場合、ファイルからメッセージへの転送とメッセージからファイルへの転送があります。	false
enableDetailedReplyMessages	このプロパティの値は、true または false のいずれかになります。このプロパティを true に設定すると、管理対象の転送要求の応答に、転送されたファイルに関する詳細な情報を含めることができます。詳細な情報とその形式は、転送ログの進行メッセージ (つまり <transferSet> エレメント) に公開されるものと同じです。詳しくは、 ファイル転送ログ・メッセージ・フォーマット を参照してください。 詳細な応答情報は、管理対象転送要求で詳細な応答情報が必要であると指定した場合にのみ含められます。この要件を指定するには、ソース・エージェントに送信される managedTransfer XML 要求メッセージの <reply> エレメントの detailed 属性を設定します。詳しくは、 ファイル転送要求メッセージ・フォーマット を参照してください。 転送要求ごとに、複数の応答メッセージを生成できます。その数は、転送の転送ログ進行メッセージの数値に 1 を加えたものと同じになります (最初の応答メッセージは単純な ACK 応答です)。詳細情報は ACK 応答メッセージを除くすべてのメッセージに含まれますが、全体の転送結果は最後の詳細な応答メッセージのみに含まれます。	true

表 72. 拡張エージェント・プロパティ: 汎用 (続き)

プロパティ名	説明	デフォルト値
enableUserMetadataOptions	<p>このプロパティの値は、true または false のいずれかになります。新しい転送要求でユーザー定義メタデータの既知のキーを使用して追加の転送オプションを指定できるかどうかを決定します。これらの既知のキーは、常に接頭部 <code>com.ibm.wmqfte.</code> で始まります。 enableUserMetadataOptions プロパティが true に設定されると、この接頭部を使用するキーは、ユーザー定義の用途ではサポートされなくなります。 enableUserMetadataOptions プロパティが true に設定されている場合、現在サポートされているキーは以下のとおりです。</p> <p>com.ibm.wmqfte.insertRecordLineSeparator</p> <p>テキスト転送用。このキーを true に設定すると、レコード単位ファイル(z/OS データ・セットなど)の読み取り時に、レコード間に行分離文字を挿入することを指定します。</p> <p>このキーが false に設定されている場合、レコード単位のファイルを読み取る際に、レコード間に行分離文字は挿入されません。</p> <p>com.ibm.wmqfte.newRecordOnLineSeparator</p> <p>テキスト転送用。このキーが true に設定されている場合、レコード単位ファイル(z/OS データ・セットなど)に書き込むときに、行分離文字が新規レコードを示し、データの一部として書き込まれないことを指定します。</p> <p>このキーが false に設定されている場合、レコード単位のファイルに書き込む際に、行分離文字は他の文字と同じように扱われます(つまり、レコードは分離されません)。</p> <p>com.ibm.wmqfte.convertLineSeparators</p> <p>テキスト転送用。行分離文字シーケンスの CRLF および LF を宛先に必要な行分離文字シーケンスに変換するかどうかを指定します。現在のところ、この変換は以下の場合にのみ有効です。</p> <ul style="list-style-type: none"> ユーザー定義のメタデータ・キー com.ibm.wmqfte.newRecordOnLineSeparator が false に設定され、転送先がレコード単位のファイルである場合。 ユーザー定義のメタデータ・キー com.ibm.wmqfte.com.ibm.wmqfte.insertRecordLineSeparator が false に設定されていて、転送がレコード単位ファイルからのものである場合。 <p>fteCreateTransfer: 新規ファイル転送の開始も参照してください。</p>	false
failTransferOnFirstFailure	<p>このプロパティの値は、true または false のいずれかになります。管理対象転送内にある転送項目が失敗するとすぐにその管理対象転送を失敗させるようにエージェントを構成できるようになります。</p> <p>この機能を有効にするには、ソース・エージェントと宛先エージェントの両方に APAR IT03450 を適用し、ソース・エージェントの <code>agent.properties</code> ファイルで failTransferOnFirstFailure プロパティを true に設定する必要があります。宛先エージェントでプロパティを true に設定するのは任意です。</p> <p>failTransferOnFirstFailure プロパティが true に設定されている場合、エージェントは管理対象転送要求の処理を通常どおり開始します。ただし、転送項目が失敗するとすぐに、管理対象転送は失敗としてマークされ、それ以降の転送項目は処理されません。管理対象転送が失敗する前に正常に処理された転送項目は、次のように処理されます。</p> <ul style="list-style-type: none"> これらの転送項目に対してはソースの後処理が優先されます。例えば、転送項目のソース後処理が <code>delete</code> に設定されていた場合、ソース・ファイルは削除されます。 書き込まれた宛先ファイルは宛先ファイル・システムに残り、削除されません。 <p>failTransferOnFirstFailure プロパティが true に設定されておらず、管理対象ファイル転送に複数のファイルが含まれていて、これらのファイルの 1 つが転送に失敗した場合(例えば、宛先ファイルが既に存在し、<code>overwrite</code> プロパティが <code>error</code> に設定されている場合)、ソース・エージェントは続行し、要求内の残りのファイルを転送しようとします。</p>	false
itemsPerProgressMessage	<p>エージェントが次の進行状況ログ・メッセージをパブリッシュするまでに転送されるファイルの数。このプロパティは、転送中に進行状況ログ・メッセージが調整キュー・マネージャーにパブリッシュされる速度を制御します。</p> <p>このプロパティに設定できる最大値は 1000 です。</p> <p>注: 進行メッセージには、前回の進行メッセージのパブリッシュ以降に転送されたすべてのファイルに関する情報が含まれます。この値を増やすと進行メッセージのサイズが大きくなるため、パフォーマンスに影響を与える可能性があります。</p>	50

プロパティ名	説明	デフォルト値
maxInlineFileSize	<p>単一のファイル間転送またはファイルからメッセージへの転送で、初期転送要求メッセージに自動的に組み込むことのできる最大ファイル・サイズ (バイト単位)。</p> <p>このプロパティを使用すると、転送の速度を改善できますが、ファイル・サイズの設定値が大きすぎると、このプロパティによってパフォーマンスが低下する可能性があります。このプロパティに推奨される初期サイズは 100 KB ですが、ご使用のシステムに最適なファイル・サイズを見つけるまでさまざまな値を十分にテストすることをお勧めします。</p> <p>この機能はデフォルトでオフになっています。また、maxInlineFileSize プロパティを 0 に設定するとオフになります。</p>	0

プロパティ名	説明	デフォルト値
highlyAvailable	このプロパティはエージェントの始動時に読み取られます。値 true に設定すると、エージェントは高可用性モードで開始されます。このプロパティを指定しない場合、または値を false に設定する場合、エージェントは高可用性を持たないエージェントとして開始されます。	false
standbyPoll インターバル	<p>このプロパティは、指定された間隔で共有キューを開こうとするスタンバイ・インスタンスによって使用されます。</p> <p>IBM MQ 9.3.0 以降、このプロパティはさらにすべてのインスタンスで使用されるようになり、インスタンスがエージェント・キュー・マネージャーから切断された場合に、再接続の試行と試行の間にインスタンスが待機する時間が、これによって決定されます。</p> <p>この試行は、インスタンスがエージェント・キュー・マネージャーに再接続するか、SYSTEM.FTE.HA.<agent name> キューを開くか (既にスタンバイ・インスタンスとして登録されている場合)、または fteStopAgent コマンドによって停止されるまで繰り返されます。</p>	5 秒
standbyStatusDiscardTime	<p>このプロパティは、アクティブ・インスタンスがスタンバイ・インスタンスからの状況パブリケーションを待機する期間を設定します。</p> <p>この時間が経過してもスタンバイ・インスタンスからパブリケーションを受信しない場合、アクティブ・インスタンスはスタンバイ・インスタンスのリストからスタンバイ・インスタンス情報を削除します。</p> <p>スタンバイ・インスタンスをリストから削除する前に、アクティブ・インスタンスが長時間待機するようにするため、デフォルト値は standbyStatusPublishInterval プロパティの 2 倍になっています。</p>	600 秒
standbyStatus 有効期限	このプロパティは、エージェントのコマンド・キューに書き込むスタンバイ状況メッセージの有効期限時刻を設定します。エージェントのアクティブ・インスタンスがこのメッセージを処理しない場合、メッセージの有効期限が切れず。	30 秒
standbyStatusPublishInterval	このプロパティは、スタンバイ・インスタンスがその状態をパブリッシュする頻度を設定するために使用します。	300 秒

プロパティ名	説明	デフォルト値
doNotUseTempOutputFile	<p>デフォルトでは、エージェントは宛先の一時ファイルに書き込み、ファイル転送が完了してから、この一時ファイルを必要なファイル名に変更します。この値を TRUE に設定すると、最終宛先ファイルに、エージェントが直接書き込むこととなります。</p> <p> z/OS システムでは、この動作は順次 データ・セットではなく PDS データ・セット・メンバーに適用されます。</p> <p>転送の場合、このプロパティの値は、宛先エージェントによって定義されます。</p>	false

プロパティ名	説明	デフォルト値
enableMandatoryLocking	<p>通常のファイルにアクセスするときに、Managed File Transfer は読み取りには共有ロックを、書き込みには排他ロックを使用します。</p> <p>Windows Windows では、ファイル・ロックは勧告のみです。このプロパティが TRUE に設定されると、Managed File Transfer はファイル・ロックを施行します。Windows の場合、これは、別のアプリケーションによりファイルが開かれていると、そのファイルのモニターはファイルが閉じられるまで起動しないことを意味します。そのファイルを含んでいる Managed File Transfer 転送は失敗します。</p> <p>UNIX UNIX タイプのプラットフォームでは、ファイル・ロックはプロセス全体を通して施行されます。UNIX タイプのプラットフォームの場合、このプロパティの設定は影響を与えません。</p> <p>このプロパティは、通常の Managed File Transfer エージェントだけに適用されます。Managed File Transfer は、ファイル・サーバーのファイル・ロック・メカニズムをサポートしていません。したがって、このプロパティはプロトコル・ブリッジ・エージェントでは無効です。プロトコル・ブリッジ・エージェントは、ファイル転送時にファイル・サーバーのファイルをロックしないからです。</p> <p>このプロパティの値は、true または false のいずれかになります。</p>	false
ioIdleThreadTimeout	<p>スレッドがシャットダウンするまで、ファイル・システムの入出力スレッドがアイドルとなる時間 (ミリ秒単位)。</p> <p>z/OS IBM MQ 9.3.0 以降、このプロパティは IBM MQ for z/OS で実行されているエージェントには適用されません。</p>	10000
ioQueueDepth	キューに入れられる入出力要求の最大数。	10
ioThreadPoolSize	<p>有効なファイル・システムの入出力スレッドの最大数。通常は、個々の転送は各自のファイル・システム入出力スレッドを使用しますが、同時転送の数がこの制限を超過すると、ファイル・システムの入出力スレッドは、この転送間で共有されます。</p> <p>進行中の同時転送の数が、ioThreadPoolSize 値より多くなることがたびたび起きるようであれば、個々の転送が各自のファイル・システム入出力スレッドを持てるようにこの値を大きくすると改善が見られます。</p>	10
textReplacementCharacterSequence	<p>テキスト・モード転送で、データ・バイトのいずれかが、ソース・コード・ページから 宛先コード・ページに変換されない場合、ファイル転送の失敗がデフォルトの動作となります。</p> <p>転送を正常に完了できるようにこのプロパティを設定するには、特定の文字値を挿入します。このプロパティ値は、単一文字です。通常、マップ不可能な文字には疑問符 (?) が使用されます。例えば、textReplacementCharacterSequence=? という形式を使用します。ここで、疑問符 (?) は置換文字です。置換文字に空白文字を使用することはできません。</p>	なし

プロパティ名	説明	デフォルト値
logTransfer	<p>転送のロギングをオンまたはオフにします。</p> <p>指定できる値は以下のとおりです。</p> <p>info 転送の高位レベルのログ情報を使用可能にします。これがデフォルト値です。</p> <p>moderate 転送の中間レベルのログ情報を使用可能にします。</p> <p>verbose 転送の詳細なログ情報を使用可能にします。</p> <p>off 転送のロギングをオフにします</p>	info
logTransferFileSize	転送ログ・ファイルの最大サイズをメガバイト単位で定義します。	20
logTransfer ファイル	最も古いファイルが廃棄される前に保持される転送ファイルの最大数を定義します。	5

表 76. 拡張エージェント・プロパティ: マルチチャンネル・サポート		
プロパティ名	説明	デフォルト値
agentMultipleChannelsEnabled	<p>このプロパティを true に設定すると、Managed File Transfer Agent が複数の IBM MQ チャネルを介して転送データ・メッセージを送信できるようになります。シナリオによっては、このプロパティを設定するとパフォーマンスが改善されることがあります。ただし、マルチチャンネル・サポートを使用可能にするのは、パフォーマンス上の明白な利点がある場合のみにしてください。複数のチャネルを介して送信されるのは、SYSTEM.FTE.DATA.destinationAgentName キューに入れられるメッセージのみです。他のすべてのメッセージの動作は変更されません。</p> <p>このプロパティを true に設定した場合は、以下のいずれかのトピック中の IBM MQ 構成ステップをすべて行って、マルチチャンネル・サポートを使用可能にする必要もあります。</p> <ul style="list-style-type: none"> 複数のチャネルに対応した MFT エージェントの構成(クラスター) 複数のチャネルに対応した MFT エージェントの構成(非クラスター) <p>さらに、Managed File Transfer エージェントに必要な標準の IBM MQ 構成ステップも実行する必要があります。これについては、初めて使用するための MFT の構成で詳しく説明されています。</p> <p>このプロパティの値は、true または false のいずれかになります。</p>	false
agentMessageBatchSize	<p>複数のチャネルを指定して構成すると、ソース・エージェントはラウンドロビン・ベースで各チャネルを介して転送用のデータ・メッセージを送信します。このプロパティによって、各チャネルで一度に送信できるメッセージの数を制御します。</p>	5

表 77. 拡張エージェント・プロパティ: マルチインスタンス・キュー・マネージャー		
プロパティ名	説明	デフォルト値
agentQMGrStandby	<p>agentQMGr によって定義されている、複数インスタンスのエージェント・キュー・マネージャーのスタンバイ・インスタンスに、クライアント接続するために使用するホスト名およびポート番号 (IBM MQ の CONNAME 形式で指定)。例えば、host_name(port_number)</p> <p>エージェントは、接続切断エラー (MQRC 2009 など) を検出すると、スタンバイ・キュー・マネージャーへの接続を試行します。エージェントがスタンバイ・キュー・マネージャーに接続されると、スタンバイ・キュー・マネージャーが使用不可になるまでエージェントは接続されたままになります。</p>	デフォルトなし

表 78. 拡張エージェント・プロパティ: プロセス・コントローラー		
プロパティ名	説明	デフォルト値
agentQMGrRetryInterval	<p>エージェントのプロセス・コントローラーがキュー・マネージャーの使用可否を検査する間隔 (秒単位)。</p>	30
maxRestartCount	<p>maxRestartInterval プロパティの値によって指定された時間間隔内に行える再開の最大回数。この値を超えると、エージェントのプロセス・コントローラーはエージェントの再開を中止し、その代わりに maxRestartDelay プロパティの値に基づいたアクションを実行します。</p>	4
maxRestartInterval	<p>エージェントのプロセス・コントローラーがエージェントの再開数を測定する間隔 (秒単位)。この間隔内に行われた再開の回数が maxRestartCount プロパティの値を超過すると、エージェントのプロセス・コントローラーはそのエージェントの再開を中止します。エージェントのプロセス・コントローラーは、その代わりに maxRestartDelay プロパティの値に基づいたアクションを実行します。</p>	120
maxRestartDelay	<p>エージェントの再開率が maxRestartCount プロパティおよび maxRestartInterval プロパティの値を超過した場合にエージェントのプロセス・コントローラーがとる動作を決定します。ゼロ以下の値を指定すると、エージェントのプロセス・コントローラーは停止します。ゼロより大きな値を指定した場合、その秒数だけ待機した後、エージェントのプロセス・コントローラーが保持する再開履歴情報がリセットされ、エージェントが再開されます。</p>	-1

表 79. 拡張エージェント・プロパティ: プロトコル・ブリッジ		
プロパティ名	説明	デフォルト値
protocolBridgeCredentialConfiguration	<p>このプロパティの値は、protocolBridgeCredentialExitClasses によって指定された出口クラスの initialize() メソッドに文字列として渡されます。</p>	null
protocolBridgeCredentialExitClasses	<p>プロトコル・ブリッジ資格情報ユーザー出口ルーチンを実装するクラスのコンマ区切りリストを指定します。詳しくは、出口クラスを使用したファイル・サーバーの資格情報のマップを参照してください。</p>	デフォルトなし

プロパティ名	説明	デフォルト値
protocolBridgeDataTimeout	プロトコル・ブリッジ・エージェントが、FTP サーバーへのデータ接続が確立されるのを待機する場合のタイムアウト、または既に確立されている接続上で FTP サーバーからデータを受信するのを待機する場合のタイムアウト (ミリ秒)。このプロパティを値 0 に設定すると、プロトコル・ブリッジ・エージェントは無期限に待機します。タイムアウト時間に達すると、プロトコル・ブリッジ・エージェントは FTP サーバーへの既存のすべてのデータ接続を閉じて、現在の転送が再開される前に新しいデータ接続の確立を試行します。新しいデータ接続を確立しようとしたときに失敗すると、現在の転送も失敗します。	0
protocolBridgeLogoutBeforeDisconnect	FTP セッションをクローズし、切断する前に、プロトコル・ブリッジ・エージェントがファイル・サーバーからユーザーをログアウトするかどうかを指定します。このプロパティを true に設定すると、プロトコル・ブリッジ・エージェントはファイル・サーバーに FTP QUIT コマンドを発行します。	false
protocolBridgePropertiesConfiguration	protocolBridgeServerPropertiesExitClasses プロパティによって指定された出口クラスの initialize() メソッドに、ブリッジ・プロパティの 1 つとして渡されます。	デフォルトなし
protocolBridgePropertiesExitClasses	プロトコル・ブリッジ・サーバー・プロパティ・ユーザー・出口ルーチンを実装するクラスのコンマ区切りリストを指定します。詳しくは、 ProtocolBridgePropertiesExit2: プロトコル・ファイル・サーバー・プロパティ の検索を参照してください。	デフォルトなし

プロパティ名	説明	デフォルト値
agentLog	プロトコル・ブリッジ・エージェントと FTP/SFTP/FTPS ファイル・サーバー間の FTP コマンドと応答のロギングを有効または無効にするキー値ペア・コンポーネントと操作。 以下に例を示します。 agentLog=on すべてのコンポーネントのロギングをオンにします agentLog=off すべてのコンポーネントのロギングをオフにします agentLog=ftp=on, sftp=on, ftps=off ロギングを FTP と SFTP ではオンにして、FTPS ではオフにします	デフォルトなし
agentLogFileSize	キャプチャー・ファイルの最大サイズを定義します (メガバイト)。正規のトレース・デフォルト・ファイル・サイズのデフォルトと同じです。	20
agentLogFiles	保持されるキャプチャー・ファイルの最大数を定義します。この数を超えると、最も古いファイルが破棄されます。 <ul style="list-style-type: none"> agentLogFiles エージェント・プロパティのデフォルト値が 10 から 5 に変更されました。つまり、IBM MQ 9.3.0 以降では、デフォルトが設定されている場合、agentevent0.log から agentevent4.log までのプロトコル・ブリッジ・エージェントのイベント・ログ・ファイルを最大 5 つまで作成できます。ただし、必要に応じてこの値を変更できます。 エージェントが IBM MQ 9.3.0 より前のバージョンからマイグレーションされている場合は、agentevent5.log から agentevent9.log までのファイルが存在しているのであれば手動で削除する必要があります。 ただし、各ログ・ファイルのサイズは 20 MB のままです。 	IBM MQ 9.3.0 以降、デフォルト値は 5 になりました。
agentLogFilter	デフォルトで、エージェントが接続しているすべての FTP サーバーとの通信をキャプチャーします。 以下に例を示します。 <ul style="list-style-type: none"> ホスト/IP アドレスでのフィルター <pre>host=ftpprod.ibm.com, ftp2.ibm.com host=9.182.*</pre> メタデータに基づくフィルター <pre>metadata="outbound files to xyz corp"</pre> 	*

プロパティ名	説明	デフォルト値
dynamicQueuePrefix	このプロパティは、一時動的キューを作成するときに使用する接頭部を定義します。	WMQFTE.*
modelQueueName	このプロパティは、一時動的キューを作成するときに使用するモジュール・キューの名前を定義します。	SYSTEM.DEFAULT.MODEL.QUEUE
publicationMDUser	調整キュー・マネージャーによってパブリッシュされるために送信されたメッセージと関連付ける MQMD ユーザー ID。このプロパティを設定しない場合は、MQMD ユーザー ID の設定に関する IBM MQ 規則に基づいて MQMD ユーザー ID が設定されます。	デフォルトなし

プロパティ名	説明	デフォルト値
monitorFilepathPlatformSeparator	\$FILEPATH 変数内で、プラットフォーム固有のパス区切り文字を使用するかどうかを指定します。値が true の場合、プラットフォーム固有のパス区切り文字が使用されます。値が false の場合、UNIX スタイルのスラッシュ (/) パス区切り文字がすべてのプラットフォームで使用されます。	true
monitorMaxResourcesInPoll	各ポーリング間隔で起動されるモニター対象リソースの最大数を指定します。例えば、モニター・パターンに *.txt、ポーリング間隔に 10 秒を指定し、 monitorMaxResourcesInPoll プロパティを 10 に設定すると、 monitorMaxResourcesInPoll プロパティによりエージェントはポーリング間隔ごとに最大値の 10 個のマッチングで起動するように制限されます。マッチング・リソースが 10 個の限度を超えると、その後のポーリング間隔で起動されます。 また、 monitorMaxResourcesInPoll プロパティと fteCreateMonitor コマンド上の -bs マッチング・パラメーターを併用すると、例えば各ポーリング間隔で 1 回のみ転送が起動されるよう制限することができます。 ゼロ以下の値を指定すると、ポーリング間隔で起動されるモニター・リソースの数は無制限になります。	-1
monitorReportTriggerFail	モニターで検出される環境および構成で発生した障害状態をログ・メッセージとして SYSTEM.FTE トピックに報告するかどうかを指定します。値を true にするとメッセージをログに記録します。値を false にするとメッセージをログに記録しません。	true
monitorReportTriggerNotSatisfied	トリガーが条件を満たさなかった場合、詳細情報を組み込んだログ・メッセージを SYSTEM.FTE トピックに送信するかどうかを指定します。値を true にするとメッセージをログに記録します。値を false にするとメッセージをログに記録しません。	false
monitorReportTriggerSatisfied	トリガーが条件を満たした場合、詳細情報を組み込んだログ・メッセージを SYSTEM.FTE トピックに送信するかどうかを指定します。値を true にするとメッセージをログに記録します。値を false にするとメッセージをログに記録しません。	false
monitorSilenceOnTriggerFailure	リソース・モニター・トリガーが連続してこの回数だけ失敗すると、障害がレポートされなくなる許容回数。	5
monitorStopOnInternalFailure	リソース・モニターの内部 FFDC 条件が連続してこの回数だけ発生すると、モニターの状態が停止に変更される許容回数。	10

表 83. 拡張エージェント・プロパティ: ルート・ディレクトリー

プロパティ名	説明	デフォルト値
commandPath	<p>以下のいずれかのメソッドを使用してコマンドを呼び出すことができるパスのセットを指定します。</p> <ul style="list-style-type: none"> エージェント Ant fte:call Ant タスク、fte:filecopy タスク、または fte:filemove タスク エージェントに渡された XML メッセージでは、サポートされるいずれかの Managed File Transfer Agent コマンド XML スキーマ (例えば、<code>managedCall</code>、<code>managedTransfer</code>)。 <p><code>commandPath</code> プロパティの値の有効な構文については、commandPath MFT プロパティを参照してください。</p> <p>重要: 指定された <code>commandPath</code> のいずれかのコマンドは、エージェントにコマンドを送信できるリモート・クライアント・システムから効果的に呼び出すことができるため、このプロパティを設定する際には細心の注意を払ってください。このため、デフォルトでは、<code>commandPath</code> を指定すると、以下のようになります。</p> <ul style="list-style-type: none"> 既存のエージェント・サンドボックスは、エージェントの開始時にエージェントによって構成されます。これにより、転送のアクセスを拒否したディレクトリーのリストに、すべての <code>commandPath</code> ディレクトリーが自動的に追加されます。 エージェントの始動時に既存のユーザー・サンドボックスが更新され、すべての <code>commandPath</code> ディレクトリー (およびそのサブディレクトリー) が <code><exclude></code> エレメントとして <code><read></code> エレメントと <code><write></code> エレメントに追加されます。 エージェントがエージェント・サンドボックスもユーザー・サンドボックスも使用するように構成されていない場合は、エージェントの開始時に、拒否されたディレクトリーとして指定された <code>commandPath</code> ディレクトリーを持つ新しいエージェント・サンドボックスが作成されます。 <p>このプロパティの値には、環境変数を含めることができます。</p> <p>詳しくは、167 ページの『MFT プロパティでの環境変数の使用』を参照してください。</p> <p><code>addCommandPathToSandbox</code> プロパティを <code>false</code> に設定すると、このデフォルトの動作をオーバーライドできます。</p> <p>重要: このオーバーライドによって事実上、クライアントはどのようなコマンドでもエージェント・システムに転送して、そのコマンドを呼び出すことができるようになります。したがって、このオーバーライドは、細心の注意を払って使用する必要があります。</p>	なし - どのコマンドも呼び出すことはできません。
addCommandPathToSandbox	<p><code>commandPath</code> プロパティで指定されたディレクトリー (およびそのすべてのサブディレクトリー) を以下に追加するかどうかを指定します。</p> <ul style="list-style-type: none"> 既存のエージェント・サンドボックスの拒否ディレクトリー。 定義されているすべてのユーザー・サンドボックスの <code><read></code> エレメントおよび <code><write></code> エレメントの <code><exclude></code> エレメント。 新しいエージェント・サンドボックス (エージェント・サンドボックスまたは 1 つ以上のユーザー・サンドボックスを使用するようにエージェントが構成されていない場合)。 <p><code>commandPath</code> プロパティの値の有効な構文については、commandPath MFT プロパティを参照してください。</p>	True
additionalWildcardSandboxChecking	<p>エージェントがファイルの転送先/転送元として使用できる場所を制限するために、ユーザー・サンドボックスまたはエージェント・サンドボックスが構成されているエージェントで、ワイルドカード転送に対する追加検査を行うかどうかを指定します。</p> <p>このプロパティが <code>true</code> に設定されている場合、追加の検査が有効になります。転送要求でワイルドカードのファイル・マッチング用に定義されたサンドボックスの外側にある場所の読み取りが試行されると、転送は失敗します。1 つの転送要求内に複数の転送があり、サンドボックスの外側にある場所を読み取ろうとしたためにこれらの要求のいずれかが失敗した場合、転送全体が失敗します。検査が失敗した場合は、失敗の理由がエラー・メッセージに示されます (ワイルドカード転送の追加検査を参照)。</p> <p>このプロパティが省略されるか、<code>false</code> に設定されると、ワイルドカード転送で追加の検査は行われません。</p>	なし

プロパティ名	説明	デフォルト値
sandboxRoot	<p>サンドボックスを使用する際に組み込むルート・パス・セットおよび除外するルート・パス・セットを指定します。この機能については、MFT エージェント・サンドボックスの処理を参照してください。</p> <p>パスは、プラットフォーム固有の分離文字で区切ります。パスの接頭部に感嘆符文字 (!) 文字を付けて、パスをサンドボックスから除外することを示してください。この機能は、組み込んだルート・パスの下の1つのサブディレクトリーを除外する場合に便利です。</p> <p>sandboxRoot プロパティは、プロトコル・ブリッジ・エージェントではサポートされません。</p> <p>sandboxRoot プロパティと userSandboxes プロパティを一緒に指定することはできません。</p> <p>このプロパティの値には、環境変数を含めることができます。</p> <p>詳しくは、167 ページの『MFT プロパティでの環境変数の使用』を参照してください。</p>	なし - サンドボックスなし
transferRoot	<p>エージェントに指定された相対パスのデフォルトのルート・ディレクトリー。</p> <p>このプロパティの値には、環境変数を含めることができます。</p> <p>詳しくは、167 ページの『MFT プロパティでの環境変数の使用』を参照してください。</p>	エージェント・プロセスを開始したユーザーのホーム・ディレクトリー。
transferRootHLQ	<p>エージェントに指定された非完全修飾データ・セットのデフォルトの HLQ (ユーザー ID)</p>	エージェント・プロセスを開始したユーザーのユーザー名。
userSandboxes	<p>転送を要求するユーザーの MQMD ユーザー名に基づいて、ファイルの転送元や転送先として使用できるファイル・システムの領域を制限します。詳しくは、MFT ユーザー・サンドボックスの処理を参照してください。</p> <p>userSandboxes プロパティは、プロトコル・ブリッジ・エージェントではサポートされません。</p> <p>sandboxRoot プロパティと userSandboxes プロパティを一緒に指定することはできません。</p>	false

プロパティ名	説明	デフォルト値
maxSchedulerRunDelay	<p>エージェントがスケジュール済み転送の検査を待機する最大間隔 (分単位)。</p> <p>このプロパティを使用可能にするには、正整数を指定します。このプロパティの使用理由について詳しくは、スケジュール済みファイル転送が実行されなかったり遅延したりする場合の対策を参照してください。</p> <p>スケジュール済み転送の実行予定時刻にエージェントがコマンド・キューからコマンドを読み取っている場合があるので、スケジュール済み転送の開始前に追加の遅延が発生する可能性があります。この場合、スケジューラーはそのコマンドが完了した直後に実行します。</p>	-1


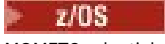

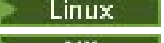
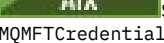


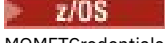



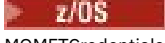
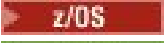



プロパティ名	説明	デフォルト値
agentCredentialsKeyFile	<p>資格情報の暗号化の際に使用される資格情報キーが含まれるファイルの名前。</p>	<p>ストリング・プロパティにデフォルト値はありません。</p>
agentQMGrAuthenticationCredentials File	<p>エージェント・キュー・マネージャーへの接続時に使用する必要がある資格情報が含まれているファイルへのパス。</p> <p>このプロパティの値には、環境変数を含めることができます。詳しくは、167 ページの『MFT プロパティでの環境変数の使用』を参照してください。</p> <p> 資格情報ファイルの作成について詳しくは、MFT および IBM MQ の接続認証を参照してください。</p> <p> 認証資格情報ファイルの作成について詳しくは、z/OS での MQMFTCredentials.xml の構成を参照してください。</p>	<p>このプロパティのデフォルト値は以下のとおりです。</p> <p> z/OS</p> <p> Linux</p> <p> AIX \$HOME/MQMFTCredentials.xml</p> <p> Windows</p> <p>%USERPROFILE%/MQMFTCredentials.xml</p>
authorityChecking	<p>『MFT エージェント・アクションのユーザー権限の制限』に記載されているセキュリティ機能を有効にするかどうかを指定します。</p> <p>inquire 許可は、すべてのエージェント権限キューに対する必須権限です。</p>	false

表 85. 拡張エージェント・プロパティ: セキュリティー (続き)		
プロパティ名	説明	デフォルト値
logAuthorityChecks	output0.log ファイルで、エージェントのイベント・ログに記録する権限検査ロギングのレベル。このプロパティがとることのできる値は None、Failures、または All です。	なし
userIdForClientConnect	IBM MQ へのクライアント接続を介してフローされるユーザー ID。java が指定されている場合、JVM によって報告されるユーザー名が、IBM MQ 接続要求の一部として送られます。このプロパティがとることのできる値は None または java です。	なし

表 86. 拡張エージェント・プロパティ: SSL/TLS		
プロパティ名	説明	デフォルト値
agentSslCipherSpec	エージェントとエージェント・キュー・マネージャーとの間でデータが交換されるときに使用されるプロトコル、ハッシュ・アルゴリズム、および暗号化アルゴリズム、および暗号鍵で使用されるビット数を指定します。 agentSslCipherSpec の値は、CipherSpec 名です。この CipherSpec 名は、エージェント・キュー・マネージャー・チャンネルで使用されている名前と同じです。有効な CipherSpec 名のリストは、 IBM MQ classes for Java の SSL/TLS CipherSpec および CipherSuite および IBM MQ classes for JMS の SSL/TLS CipherSpec および CipherSuite に記載されています。 agentSslCipherSpec は、agentSslCipherSuite とよく似ています。agentSslCipherSuite と agentSslCipherSpec の両方を指定した場合は、agentSslCipherSpec の値が使用されます。	なし
agentSslCipherSuite	エージェントとエージェント・キュー・マネージャーのデータ交換方法に関する SSL の動作を指定します。 agentSslCipherSuite の値は、CipherSuite 名です。この CipherSuite 名は、エージェント・キュー・マネージャー・チャンネルで使用される CipherSpec 名にマップされます。詳しくは、 CipherSuite と CipherSpec の名前マッピング を参照してください。 agentSslCipherSuite は、agentSslCipherSpec とよく似ています。agentSslCipherSuite と agentSslCipherSpec の両方を指定した場合は、agentSslCipherSpec の値が使用されます。	なし
agentSslPeerName	エージェント・キュー・マネージャーによって提供される名前と合致しなければならない識別名の骨組みを指定します。その識別名に基づいて、接続時にキュー・マネージャーによって提示される識別用証明書が検査されます。	なし
agentSslTrustStore	エージェントが信頼する証明書の場所を指定します。agentSslTrustStore の値は、ファイル・パスです。括弧、コンマ (,)、および円記号 (¥) は MFT コマンドの特殊文字であり、円記号 (¥) 文字でエスケープする必要があります。 Windows Windows 上のファイル・パスは、分離文字として二重の円記号 (\\) を使用するか、単一のスラッシュ (/) を使用して指定できます。 このプロパティの値には、環境変数を含めることができます。	なし
agentSslKeyStore	エージェントの秘密鍵の場所を指定します。agentSslKeyStore の値は、ファイル・パスです。括弧、コンマ (,)、および円記号 (¥) は MFT コマンドの特殊文字であり、円記号 (¥) 文字でエスケープする必要があります。 Windows Windows 上のファイル・パスは、分離文字として二重の円記号 (\\) を使用するか、単一のスラッシュ (/) を使用して指定できます。このプロパティが必要なのは、エージェント・キュー・マネージャーでクライアント認証が必要な場合に限られます。 このプロパティの値には、環境変数を含めることができます。	なし
agentSslFipsRequired	エージェントのレベルで FIPS サポートを使用可能にすることを指定します。このプロパティの値は、true または false のいずれかになります。詳しくは、 MFT の FIPS サポート を参照してください。	false
agentSslKeyStoreType	使用する SSL 鍵ストアのタイプ。JKS および PKCS#12 の鍵ストアがサポートされています。このプロパティの値は、jks または pkcs12 のいずれかになります。	jks

プロパティ名	説明	デフォルト値
agentSslKeyStoreCredentialsFile	<p>エージェントの鍵ストアにアクセスするための資格情報を含むファイルへのパス。</p> <p>このプロパティの値には、環境変数を含めることができます。詳しくは、167 ページの『MFT プロパティでの環境変数の使用』を参照してください。</p> <p> 資格情報ファイルの作成について詳しくは、MFT および IBM MQ の接続認証を参照してください。</p> <p> 認証資格情報ファイルの作成について詳しくは、z/OS での MQMFTCredentials.xml の構成を参照してください。</p>	<p>このプロパティのデフォルト値は以下のとおりです。</p> <p> z/OS</p> <p> Linux</p> <p> AIX \$HOME/MQMFTCredentials.xml</p> <p> Windows</p> <p>%USERPROFILE%/MQMFTCredentials.xml</p>
agentSslTrustStoreType	<p>使用する SSL 鍵ストアのタイプ。JKS および PKCS#12 の鍵ストアがサポートされています。このプロパティの値は、jks または pkcs12 のいずれかになります。</p>	jks
agentSslTrustStoreCredentialsFile	<p>エージェントのトラストストアにアクセスするための資格情報を含むファイルへのパス。</p> <p>このプロパティの値には、環境変数を含めることができます。詳しくは、167 ページの『MFT プロパティでの環境変数の使用』を参照してください。</p> <p> 資格情報ファイルの作成について詳しくは、MFT および IBM MQ の接続認証を参照してください。</p> <p> 認証資格情報ファイルの作成について詳しくは、z/OS での MQMFTCredentials.xml の構成を参照してください。</p>	<p>このプロパティのデフォルト値は以下のとおりです。</p> <p> z/OS</p> <p> Linux</p> <p> AIX \$HOME/MQMFTCredentials.xml</p> <p> Windows</p> <p>%USERPROFILE%/MQMFTCredentials.xml</p>

プロパティ名	説明	デフォルト値
maxTransferNegotiationTime	<p>宛先エージェントがネゴシエーションを完了するまで転送が待機する最大時間 (ミリ秒)。この時間内にネゴシエーションが完了しない場合、転送は再同期状態に置かれ、別の転送が使用可能な場合はその転送を実行できます。</p> <p>ソース・エージェントまたは宛先エージェントに大きな負荷がかかるシナリオの場合、エージェントが時間内でネゴシエーション要求に対応するには、デフォルト値では小さすぎる可能性があります。その可能性が特に高いのは、ソース・エージェントに多数のリソース・モニターが定義されている場合、あるいはソース・エージェントのリソース・モニターが大量のファイルを含むディレクトリーをモニターしている場合です。その一方、数多くの転送要求がエージェントに送信される場合にも、この事態が起こる可能性があります。そのようなシナリオでは、このプロパティの値を 200,000 以上に設定しなければならない場合があります。</p>	30 000
recoverableTransferRetryInterval	<p>リカバリー可能な転送エラーを検出してから転送の再開を試みるまでの待機時間 (ミリ秒)。</p>	60 000
senderTransferRetryInterval	<p>宛先で既に最大転送数が実行されているために、拒否された転送が再試行されるまで待機する時間 (ミリ秒)。最小値は 1000 です。</p>	30 000
transferAckTimeout	<p>相手側からの確認応答またはデータを転送が待機して再試行を実行するまでのタイムアウト (ミリ秒単位)。これは拡張プロパティであり、ほとんどの Managed File Transfer 構成では、このプロパティの値を変更する必要はありません。</p> <p>確認応答は、完全なデータ・ウィンドウが受信されると、常に受信側エージェントから送信側エージェントへ送信されます。処理能力が制限されている、または信頼性のないネットワークの場合、および agentWindowSize と agentChunkSize の設定値が大きい場合、デフォルト値では十分な長さでない可能性があります。この場合、エージェント間でデータが不要に再転送されることがあります。したがって、この値を大きくすることが有効で、ネットワーク速度が遅いために転送がリカバリー・モードになる可能性を減らすことができます。</p>	60 000
transferAckTimeoutRetries	<p>エージェントが転送を中止してリカバリー状態に移行するまで行う、応答のない転送に対する確認応答の最大試行回数。</p>	5

プロパティ名	説明	デフォルト値
xmlConfigReloadInterval	<p>ランタイム中にエージェントがXML 構成ファイルを再ロードする間隔(秒単位)。ランタイム中にエージェントがXML 構成ファイルを再ロードしないようにするには、このプロパティを -1 に設定してください。このプロパティによって、以下のXML 構成ファイルが影響を受けます。</p> <ul style="list-style-type: none"> ConnectDirectCredentials.xml ConnectDirectNodeProperties.xml ConnectDirectProcessDefinitions.xml ProtocolBridgeCredentials.xml ProtocolBridgeProperties.xml UserSandboxes.xml 	30

プロパティ名	説明	デフォルト値
javaCoreTriggerFile	<p>エージェントがモニターするファイルの場所への絶対パス。指定された場所にファイルが存在する場合、エージェントの開始によって Javacore が起動されます。エージェントが開始した後、この場所でファイルを更新すると、エージェントは再度 Javacore ファイルを起動します。</p> <p>別スレッドがこのファイルを 30 秒ごとにポーリングし、ファイルの作成や更新が行われたかどうかを検査します。前回のポーリング以降にファイルが作成または更新された場合、エージェントは MQ_DATA_PATH/mqft/logs/coordination_qmgr_name/agents/agent_name ディレクトリーに Javacore ファイルを生成します。</p> <p>このプロパティを指定すると、エージェントは始動時に以下のメッセージを出力します。</p> <pre>BFGAG0092I The <insert_0> file will be used to request JVM diagnostic information.</pre> <p>このプロパティの値には、環境変数を含めることができます。 詳しくは、167 ページの『MFT プロパティでの環境変数の使用』を参照してください。</p>	なし
トレース	<p>エージェントの開始時に使用するトレース仕様。これは、クラスまたはパッケージ(あるいはその両方)、等号文字、およびトレース・レベルのコンマ区切りリストです。</p> <p>例えば、エージェントの始動時に com.ibm.wmqfte.agent.Agent クラスと com.ibm.wmqfte.commandhandler パッケージ内のクラスをトレースするには、以下の項目を agent.properties ファイルに追加します。</p> <pre>trace=com.ibm.wmqfte.agent.Agent,com.ibm.wmqfte.commandhandler=all</pre> <p>コロン区切りのリストで、複数のトレース仕様を指定できます。例:</p> <pre>trace=com.ibm.wmqfte.agent.Agent=all:com.ibm.wmqfte.commandhandler=moderate</pre> <p>特殊トレース仕様 =all は、エージェントと、エージェント・キュー・マネージャーとのすべての通信を処理する Java Message Queuing Interface (JMQUI) をトレースするために使用されます。これを有効にするには、以下のエントリーを agent.properties ファイルに追加します。</p> <pre>trace==all</pre> <p>IBM サポート担当員が特に指定しない限り、トレース仕様 com.ibm.wmqfte=all を以下のように使用します。</p> <pre>trace=com.ibm.wmqfte=all</pre>	なし
outputLogFiles	保持する output.log ファイルの合計数。この値は、エージェント自体に加えて、エージェントのプロセス・コントローラーにも適用されます。	5
outputLogSize	出力が次のファイルで折り返す前の各 output.log ファイルの最大サイズ (MB 単位)。この値は、エージェント自体に加えて、エージェントのプロセス・コントローラーにも適用されます。	1


表 88. 拡張エージェント・プロパティ: トレースおよびロギング (続き)		
プロパティ名	説明	デフォルト値
outputLogEncoding	エージェントが output.log ファイルを書き込む際に使用する文字エンコード。	エージェントが実行されているプラットフォームのデフォルトの文字エンコード。
traceFiles	保持するトレース・ファイルの合計数。この値は、エージェント自体に加えて、エージェントのプロセス・コントローラーにも適用されます。	5
traceSize	トレースが次のファイルで折り返す前の各トレース・ファイルの最大サイズ (MB 単位)。この値は、エージェント自体に加えて、エージェントのプロセス・コントローラーにも適用されます。	20
traceMaxBytes	トレース・ファイルに出力されるメッセージ・データ量の制限。	4096 バイト
logTransferRecovery	このプロパティの値を true に設定すると、転送がリカバリー状態になるたびに、診断イベントがエージェントのイベント・ログである output0.log ファイルに報告されます。	デフォルト値は true です。
logCapture	このエージェントに送信された転送要求メッセージと、エージェントが調整キュー・マネージャーにバブリッシュしたログ・メッセージをキャプチャーします。キャプチャーされるこれらのメッセージは、転送の問題をデバッグする際に役に立ちます。キャプチャーされたメッセージは、capture?.log という名前のエージェント・ログ・ディレクトリーにあるファイルに保管されます。? は数値です。数値 0 を含むファイルには、キャプチャーされた最新のメッセージが保持されます。	false
logCaptureFileSize	キャプチャー・ファイルの最大サイズを定義します (メガバイト)。	10
logCaptureFiles	保持されるキャプチャー・ファイルの最大数を定義します。この数を超えると、最も古いファイルが破棄されます。	10
logCaptureFilter	エージェントがメッセージのトピック名の突き合わせに使用する Java 正規表現。この正規表現に一致したメッセージのみキャプチャーされます。	.* (match all)
resourceMonitorLog	<p>リソース・モニターと、ロギングをオンまたはオフにする操作のキーと値のペア。</p> <p>指定できる値は以下のとおりです。</p> <ul style="list-style-type: none"> • info • moderate • verbose • off <p>以下に例を示します。</p> <ul style="list-style-type: none"> • resourceMonitorLog=MON1,MON2=info:MON3=off MON1 および MON2 のロギングをオンにし、MON3 のロギングをオフにします。 • resourceMonitorLog=info すべてのリソース・モニターの info レベルのロギングをオンにします。 <p>リソース・モニター・ログは、resmoneventN.log という名前のファイルに書き込まれます。ここで、N は数値を表します。例えば、resmonevent0.log です。</p> <p> 重要: 1つのエージェントのリソース・モニターはすべて同じログ・ファイルに書き込まれます。</p> <p>詳しくは、MFT リソース・モニターのロギングを参照してください。</p>	info
resourceMonitorLogFileSize	キャプチャー・ファイルの最大サイズを定義します (メガバイト)。	20
resourceMonitorLogFiles	<p>保持されるキャプチャー・ファイルの最大数を定義します。この数を超えると、最も古いファイルが破棄されます。</p> <ul style="list-style-type: none"> • resourceMonitorLogFiles エージェント・プロパティのデフォルト値が 10 から 5 に変更されました。つまり、IBM MQ 9.3.0 以降では、デフォルトが設定されている場合、resmonevent0.log から resmonevent4.log までの最大 5 つのリソース・モニター・イベント・ログ・ファイルを使用できます。ただし、必要に応じてこの値を変更できます。 • エージェントが IBM MQ 9.3.0 より前のバージョンからマイグレーションされている場合は、resmonevent5.log から resmonevent9.log までのファイルが存在しているのであれば手動で削除する必要があります。 • ただし、各ログ・ファイルのサイズは 20 MB のままです。 	IBM MQ 9.3.0 以降、デフォルト値は 5 になりました。

表 89. 拡張エージェント・プロパティ: 転送制限		
プロパティ名	説明	デフォルト値
maxDestinationTransfers	宛先エージェントが時点を問わず常に一度に処理できる並行転送の最大数。ここで指定する合計値でカウントされるのは、エージェントに実行依頼されるそれぞれの転送要求です。要求を満たすために転送されるファイルの数は無関係です。つまり、1個のファイルを転送する転送要求も、10個のファイルを転送する転送要求も、同じようにカウントされる、ということです。 宛先エージェントが maxDestinationTransfers プロパティで指定された制限に達すると、エージェント・キューが転送します。 エージェント・プロパティ値の合計 (maxSourceTransfers + maxDestinationTransfers + maxQueuedTransfers) が状態格納キュー (SYSTEM.FTE.STATE.agent name) の MAXDEPTH 設定の値を超えると、エージェントは開始しません。	25 (Connect:Direct 以外のすべてのエージェントの場合) 5 (Connect:Direct ブリッジ・エージェントの場合)
maxFilesForTransfer	1つの管理対象転送に許可される転送項目の最大数。管理対象転送に含まれる項目数が maxFilesForTransfer の値を上回る場合、管理対象転送は失敗し、それ以降の転送項目は処理されません。 このプロパティを設定すると、転送要求が正しくないことが原因で誤って多数のファイルが転送されることを防ぐことができます。例えば、ユーザーが誤ってキーワード conref=" ../common/mqent.dita#mqent/unixlinuxbix " /> システムのルート・ディレクトリー/を指定した場合などです。	5000
maxSourceTransfers	ソース・エージェントが時点を問わず常に一度に処理できる並行転送の最大数。ここで指定する合計値でカウントされるのは、エージェントに実行依頼されるそれぞれの転送要求です。要求を満たすために転送されるファイルの数は無関係です。つまり、1個のファイルを転送する転送要求も、10個のファイルを転送する転送要求も、同じようにカウントされる、ということです。 宛先エージェントが maxSourceTransfers プロパティで指定された制限に達すると、ソース・エージェント・キューが転送します。 エージェント・プロパティ値の合計 (maxSourceTransfers + maxDestinationTransfers + maxQueuedTransfers) が状態格納キュー (SYSTEM.FTE.STATE.agent name) の MAXDEPTH 設定の値を超えると、エージェントは開始しません。	25 (Connect:Direct ブリッジ・エージェント以外のすべてのエージェントの場合) 5 (Connect:Direct ブリッジ・エージェントの場合)
maxQueuedTransfers	エージェントが新しい転送要求を拒否するまでに、ソース・エージェントによってキューに入れることができる保留中の転送の最大数。このプロパティを設定して、maxDestinationTransfers や maxSourceTransfers が制限に達した場合、あるいは制限を超えた場合でも、その時点で作成した新規の転送要求が受け入れられ、キューに入れられ、後で実行されるようにすることができます。 キューに入れられた転送要求が処理される順序は、その優先順位とキューに入れられている期間の係数です。古くて、優先順位が高い保留中の転送が最初に選択されます。長期間キューに入れられている、優先順位が低い転送は、より新しく優先順位が高い転送に優先して選択されます。 エージェント・プロパティ値の合計 (maxSourceTransfers + maxDestinationTransfers + maxQueuedTransfers) が状態格納キュー (SYSTEM.FTE.STATE.agent name) の MAXDEPTH 設定の値を超えると、エージェントは開始しません。	1000

表 90. 拡張エージェント・プロパティ: 転送リカバリー・タイムアウト		
プロパティ名	説明	デフォルト値
transferRecoveryTimeout	停止したファイル転送のリカバリーをソース・エージェントが試行し続ける時間 (秒単位) を設定します。 このプロパティを設定しない場合のエージェントのデフォルトの動作は、転送が正常にリカバリーされるまで試行し続ける動作です。転送リカバリー・タイムアウト・プロパティには、以下の値を設定できます。 -1 エージェントは、停止した転送のリカバリーを、転送が完了するまで試行し続けます。このオプションを使用すると、このプロパティを設定しない場合のエージェントのデフォルトの動作と同じになります。 0 エージェントは、リカバリーに入るとすぐにファイル転送を停止します。 >0 エージェントは、指定された正整数値で設定された時間 (秒単位) だけ、停止した転送のリカバリーを試行し続けます。例えば、transferRecoveryTimeout=21600 は、エージェントがリカバリーに入ってから6時間にわたって転送のリカバリーを試行し続けることを示しています。このパラメーターの最大値は999999999です。	-1

プロパティ名	説明	デフォルト値
agentForceConsistentPathDelimiters	転送出口に対して指定するソース・ファイル情報と宛先ファイル情報のパス区切り文字を強制的に UNIX スタイルのスラッシュ (/) にします。有効なオプションは、true と false です。	false
destinationTransferEndExitClasses	宛先転送ユーザー出口ルーチンを実装するクラスのコンマ区切りリストを指定します。	デフォルトなし
destinationTransferStartExitClasses	宛先転送開始ユーザー出口ルーチンを実装するクラスのコンマ区切りリストを指定します。	デフォルトなし
exitClassPath	ユーザー出口ルーチンのクラスパスの役割を果たす、プラットフォーム固有のディレクトリーの文字区切りリストを指定します。 エージェントの exits ディレクトリーは、このクラス・パス内の項目より前に検索されます。	エージェントの exits ディレクトリー
exitNativeLibraryPath	ユーザー出口ルーチンのネイティブ・ライブラリー・パスの役割を果たす、プラットフォーム固有のディレクトリーの文字区切りリストを指定します。	エージェントの exits ディレクトリー
ioMaxRecordLength	レコード単位ファイルのサポート可能な最大レコード長 (バイト単位)。Managed File Transfer は、レコード単位ファイルへの書き込みをいかなるレコード長でもサポートできます。しかし、レコード長が大きいとメモリー不足エラーになる可能性があるため、このエラーが発生しないように、最大レコード長はデフォルトで 64 K に制限されます。レコード単位ファイルから読み取る場合は、レコード全体が単一の転送チャンクに収まらなければならないので、レコード長は転送チャンク・サイズによりさらに制限されます。このプロパティは入出力ユーザー出口のレコード単位ファイル専用です。	64 KB
monitorExitClasses	モニター出口ルーチンを実装するクラスのコンマ区切りリストを指定します。詳しくは、 MFT リソース・モニター・ユーザー出口 を参照してください。	デフォルトなし
protocolBridgeCredentialExitClasses	プロトコル・ブリッジ資格情報ユーザー出口ルーチンを実装するクラスのコンマ区切りリストを指定します。詳しくは、 出口クラスを使用したファイル・サーバーの資格情報のマップ を参照してください。	デフォルトなし
sourceTransferEndExitClasses	ソース転送終了出口ルーチンを実装するクラスのコンマ区切りリストを指定します。	デフォルトなし
sourceTransferStartExitClasses	ソース転送開始出口ルーチンを実装するクラスのコンマ区切りリストを指定します。	デフォルトなし
IOExitClasses	入出力ユーザー出口ルーチンを実装するクラスのコンマ区切りリストを指定します。IOExit インターフェースを実装するクラスのみリストします。つまり、IOExitResourcePath や IOExitChannel などの他の入出力ユーザー出口インターフェースを実装するクラスはリストしないでください。詳しくは、 MFT 転送入出力ユーザー出口の使用 を参照してください。	デフォルトなし

プロパティ名	説明	デフォルト値
agentDataCompression	このプロパティは、クライアント接続でのみサポートされます。 リモート IBM MQ サーバーとネゴシエーションするための、ファイル・データ転送用の圧縮タイプのコンマ区切りリスト。これらの圧縮タイプについては、 メッセージ・データ圧縮リストのトピック を参照してください。 これらの値は、妥当性検査後に、エージェントのクライアント・チャンネルに (指定されている順序で) プロパティとしてパススルーされます。次いで IBM MQ クライアントは、このクライアント・チャンネルとリモート・サーバー・チャンネルの間のネゴシエーションを処理して、その 2 つのチャンネルの圧縮プロパティ間でマッチングに関する最低限の共通基準を検出します。一致項目が見つからない場合は、MQCOMPRESS_NONE が常に選択されます。	MQCOMPRESS_NONE
agentHeaderCompression	このプロパティは、クライアント接続でのみサポートされます。 リモート IBM MQ サーバーとネゴシエーションするための、ヘッダー・データ転送用の圧縮タイプのコンマ区切りリスト。受け入れられる値は MQCOMPRESS_NONE または MQCOMPRESS_SYSTEM です。これらの圧縮タイプについては、 HdrCompList [2] (MQLONG) のトピックを参照してください。 これらの値は、妥当性検査後に、エージェントのクライアント・チャンネルに (指定されている順序で) プロパティとしてパススルーされます。次いで IBM MQ クライアントは、このクライアント・チャンネルとリモート・サーバー・チャンネルの間のネゴシエーションを処理して、その 2 つのチャンネルの圧縮プロパティ間でマッチングに関する最低限の共通基準を検出します。一致項目が見つからない場合は、MQCOMPRESS_NONE が常に選択されます。	MQCOMPRESS_NONE


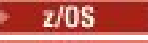
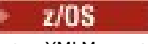
表 93. 拡張エージェント・プロパティ: z/OS 固有		
プロパティ名	説明	デフォルト値
adminGroup	<p>セキュリティ・マネージャー・グループ。このグループのメンバーは以下を実行できます。</p> <ul style="list-style-type: none"> • fteStartAgent コマンドを使用してエージェントを開始します。 • fteStopAgent コマンドを使用してエージェントを停止します。 • fteSetAgentTraceLevel コマンドを使用してエージェントのトレースをオンまたはオフにします。 • fteSetAgentLogLevel コマンドを使用してエージェントのログをオンまたはオフにします。 • -d パラメーターを指定して fteShowAgentDetails コマンドを実行し、ローカル・エージェントの詳細を表示します。 <p>セキュリティ・マネージャー・グループ (例えば MFTADMIN) を定義し、開始済みタスクのユーザー ID と管理者 TSO ID をこのグループに追加します。エージェント・プロパティ・ファイルを編集し、adminGroup プロパティの値をこのセキュリティ・マネージャー・グループの名前に設定します。</p> <pre>adminGroup=MFTADMIN</pre>	なし
 bpxwdynAllocAdditionalOptions	<p>Managed File Transfer は、BPXWDYN テキスト・インターフェースを使用して、z/OS データ・セットの作成とオープンを行います。データ・セットの割り振りに BPXWDYN を使用すると、Managed File Transfer はデフォルトで、可能であれば、データ・デバイスがマウントされるようにします (ディスク・ベアスのデータ・セットでは不要ですが、テープ・データ・セットの場合は必要です)。このオプションは特定の環境でサポートされない場合があるので、このプロパティを使用してこの動作を変更します。また、データ・セットへの転送時に、コマンド行で BPXWDYN のオプションを指定することもできます。これらのオプションは、このプロパティで指定したオプションに追加されません。</p> <p>agent.properties ファイルで bpxwdynAllocAdditionalOptions プロパティを使用する場合は、一部の BPXWDYN オプションを指定してはなりません。これらのプロパティのリストについては、MFT で使用できない BPXWDYN のプロパティを参照してください。</p>	<p>デフォルトは以下のとおりです。</p> <ul style="list-style-type: none"> • MOUNT (z/OS V1R8 以降の場合)
armELEMTYPE	<p>オプションのプロパティ。エージェントを自動リスタート・マネージャー (ARM) で再始動するように構成する場合は、関連付けられた ARM ポリシー内で指定された ARM ELEMTYPE パラメーター値をこのプロパティに設定します。エージェントの場合、ELEMTYPE は SYSBFGAG に設定します。</p>	設定なし
armELEMENT	<p>オプションのプロパティ。エージェントを自動リスタート・マネージャー (ARM) で再始動するように構成する場合は、関連付けられた ARM ポリシー内で指定された ARM ELEMENT パラメーター値をこのプロパティに設定します。ELEMENT 値には、エージェント名に対応する値を設定できます。</p>	設定なし

表 94. 拡張エージェント・プロパティ: その他のプロパティ		
プロパティ名	説明	デフォルト値
  legacyXMLMessageMQMDFormat	<p>エージェントにより生成される Managed File Transfer XML メッセージ (例えば、ログおよび転送の進行状況メッセージ) は、ブランクの MQMD 形式フィールドと共に、キューに送信されるようになりました。以前のバージョンの製品では、MQMD 形式フィールドを MQSTR (テキスト・メッセージ・ストリング) に設定していました。このプロパティを true に設定すると、エージェントにより生成される Managed File Transfer XML メッセージを、MQSTR の MQMD 形式フィールドと共にキューに送信することが可能になります。</p> <p>注: コマンドへのエージェント応答メッセージは、対応するコマンド要求に一致するメッセージ形式で送信されます。</p> <p>MQMD 形式フィールドが MQSTR に設定されている場合、データ変換が使用可能になっている MQ ネットワーク内のチャネルが存在すると、Managed File Transfer コマンドの XML メッセージが壊れる可能性があります。</p>	false
adjustScheduleTimeForDaylightSavings	<p>企業が以下のようにスケジュール転送を作成して、毎日スケジュールされた転送を実行しているとする。</p> <ul style="list-style-type: none"> • -oi パラメーターを日にち • -tb パラメーターをソース <p>に設定した fteCreateTransfer コマンドで毎日実行している場合、このプロパティを true に設定すると、時計が 1 時間進んだらスケジュールされた転送の時間を 1 時間進め、時計が 1 時間戻ったら 1 時間戻します。</p> <p>例えば、スケジュールされた転送が 1:00 AM に設定されている場合、時計が進むと転送は 2:00 AM に実行され、時計が戻ると転送は 1:00 AM に戻ります。</p>	true

関連概念

[MFT 構成オプション \(Multiplatforms\)](#)

[ファイル転送のリカバリーのタイムアウト・オプション](#)

[MFT のサンドボックス](#)

関連タスク

[複数のチャンネルに対応した MFT エージェントの構成 \(クラスター\)](#)

[複数のチャンネルに対応した MFT エージェントの構成 \(非クラスター\)](#)

関連資料

[217 ページの『MFT Java システム・プロパティー』](#)

いくつかの Managed File Transfer コマンド・プロパティーとエージェント・プロパティーは Java システム・プロパティーとして定義する必要があります。これは、コマンド・プロパティーおよびエージェント・プロパティーのメカニズムを使用できない初期の関数用の構成を定義するためです。

[MFT 用の SSL/TLS プロパティー](#)

[200 ページの『MFT command.properties ファイル』](#)

command.properties ファイルには、ユーザーがコマンドを発行するときに接続先となるコマンド・キュー・マネージャーと、Managed File Transfer がそのキュー・マネージャーと情報のやり取りをするのに必要な情報を指定します。

[196 ページの『MFT coordination.properties ファイル』](#)

coordination.properties ファイルは、調整キュー・マネージャーとの接続の詳細を指定します。複数の Managed File Transfer インストール済み環境で同じ調整キュー・マネージャーが共用される場合があるため、共用ドライブ上の共通の coordination.properties ファイルへのシンボリック・リンクを使用できます。

[204 ページの『MFT logger.properties ファイル』](#)

Managed File Transfer ロガーには、一連の構成プロパティーがあります。これらのプロパティーを logger.properties ファイルで指定します。このファイルは、MQ_DATA_PATH/mqft/config/coordination_qmgr_name/loggers/logger_name ディレクトリーにあります。

[fteCreateAgent](#)

[fteCreateBridgeAgent](#)

[fteCreateCDAgent](#)

[167 ページの『MFT プロパティーでの環境変数の使用』](#)

ファイルまたはディレクトリーの場所を表す Managed File Transfer プロパティーで環境変数を使用することができます。これにより、製品の一部を実行するときに使用されるファイルまたはディレクトリーの場所を、現在の環境 (例えば、コマンドを実行するユーザー) に応じて変更することができます。

MFT coordination.properties ファイル

coordination.properties ファイルは、調整キュー・マネージャーとの接続の詳細を指定します。複数の Managed File Transfer インストール済み環境で同じ調整キュー・マネージャーが共用される場合があるため、共用ドライブ上の共通の coordination.properties ファイルへのシンボリック・リンクを使用できます。

coordination.properties ファイルは、インストーラーまたは **fteSetupCoordination** コマンドにより作成されます。**fteSetupCoordination** コマンドを、**-f** フラグと共に使用して、このファイルの基本調整キュー・マネージャー・プロパティーを変更できます。拡張調整キュー・マネージャー・プロパティーを変更または追加するには、テキスト・エディターでファイルを編集する必要があります。

coordination.properties ファイルは MQ_DATA_PATH/mqft/config/coordination_qmgr_name ディレクトリーにあります。

MFT coordination.properties ファイルには以下の値が含まれます。

プロパティー名	説明	デフォルト値
coordinationCredentialsKeyFile	資格情報の暗号化の際に使用される資格情報キーが含まれるファイルの名前。	ストリング・プロパティーにデフォルト値はありません。

表 95. 基本調整キュー・マネージャー・プロパティ (続き)		
プロパティ名	説明	デフォルト値
coordinationQMGr	調整キュー・マネージャーの名前。	デフォルトなし
coordinationQMGrHost	調整キュー・マネージャーのホスト名または IP アドレス。	デフォルトなし
coordinationQMGrPort	調整キュー・マネージャーとのクライアント接続で使用されるポート番号。	1414
coordinationQMGrChannel	調整キュー・マネージャーとの接続に使用される SVRCONN チャネル名。	SYSTEM.DEF.SVRCONN

coordinationQMGrHost プロパティの値を指定しない場合は、デフォルトでバインディング・モードが使用されます。

coordinationQMGrHost プロパティに値を指定し、coordinationQMGrPort および coordinationQMGrChannel プロパティには値を指定しない場合、ポート番号 1414 およびチャネル SYSTEM.DEF.SVRCONN がデフォルトで使用されます。

以下は、coordination.properties ファイルの内容の例です。

```
coordinationQMGr=ERIS
coordinationQMGrHost=kuiper.example.com
coordinationQMGrPort=2005
coordinationQMGrChannel=SYSTEM.DEF.SVRCONN
```

この例での ERIS は、システム kuiper.example.com にある IBM MQ キュー・マネージャーの名前です。キュー・マネージャー ERIS は、Managed File Transfer のログ情報の送信先であるキュー・マネージャーです。

拡張調整プロパティ

Managed File Transfer は、さらに拡張調整プロパティも提供します。以下のいずれかのプロパティを使用する場合は、coordination.properties ファイルを手動で編集して、必要な拡張プロパティを追加します。括弧、コンマ (,)、および円記号 (¥) は MFT コマンドの特殊文字であり、円記号 (¥) 文字でエスケープする必要があります。 **Windows** Windows 上のファイル・パスは、分離文字として二重の円記号 (\\) を使用するか、単一のスラッシュ (/) を使用して指定できます。Java プロパティ・ファイル内の文字エスケープについては詳しくは、Oracle 資料 [Properties クラスの Javadoc](#) を参照してください。

- [エージェント・プロパティ](#)
- [コード・ページ・プロパティ](#)
- [接続プロパティ](#)
- [マルチインスタンス・キュー・マネージャー・プロパティ](#)
- [キュー・プロパティ](#)
- [セキュリティ・プロパティ](#)
- [SSL プロパティ](#)
- [サブスクリプション・プロパティ](#)

表 96. 拡張調整プロパティ: エージェント		
プロパティ名	説明	デフォルト値
agentStatusJitterTolerance	<p>エージェントの状況メッセージのバブリッシュが遅れてもよい最大時間。その時間を超えると、メッセージは延滞と見なされます。この値は、ミリ秒単位で指定します。</p> <p>状況メッセージの経過時間は、調整キュー・マネージャーでバブリッシュされた時点が基点になります。ただし、メッセージがエージェントから送出されるのは、調整キュー・マネージャーでこのメッセージが受信される少し前です。これは、メッセージが IBM MQ ネットワークを移動するために必要な時間を許容するためです。この移動にかかる時間が常に同じであれば、実際の移動時間に関わらず、60 秒間隔で作成されるメッセージは、60 秒間隔でバブリッシュされます。ところが、その移動時間がメッセージごとに異なる場合は、60 秒間隔で作成されたメッセージが、例えば 61 秒、59 秒、58 秒、62 秒などの間隔でバブリッシュされる可能性があります。この場合に発生する 60 秒からの最大偏差のことをジッターといいます (この例では 2 秒です)。このプロパティでは、ジッターによる遅れの最大時間を指定します (その時間を超えると、メッセージは延滞と見なされます)。</p>	3000

表 97. 拡張調整プロパティ: コード・ページ		
プロパティ名	説明	デフォルト値
coordinationCcsid	コマンドが、調整キュー・マネージャーに接続するとき使用するコード・ページ。エージェントによる調整キュー・マネージャーへのすべてのパブリケーションも、このコード・ページを使用して実行されます。coordinationCcsid の値を指定する場合は、coordinationCcsidName の値も指定する必要があります。	1208
coordinationCcsidName	coordinationCcsid の Java 表記。coordinationCcsidName の値を指定する場合は、coordinationCcsid の値も指定する必要があります。	UTF8


表 98. 拡張調整プロパティ: 接続		
プロパティ名	説明	デフォルト値
javaLibraryPath	Managed File Transfer では、バインディング・モードでキュー・マネージャーに接続するとき、IBM MQ Java バインディング・ライブラリーへのアクセス権限が必要になります。Managed File Transfer は、デフォルトでは、IBM MQ で定義されているデフォルトの場所でバインディング・ライブラリーを検索します。バインディング・ライブラリーが別の場所にある場合は、このプロパティを使用して、バインディング・ライブラリーの場所を指定してください。	MQ_INSTALLATION_PATH/ java/lib

表 99. 拡張調整プロパティ: マルチインスタンス・キュー・マネージャー		
プロパティ名	説明	デフォルト値
coordinationQMGrStandby	coordinationQMGr プロパティによって定義されている、複数インスタンスの調整キュー・マネージャーのスタンバイ・インスタンスに、クライアント接続するために使用する ホスト名およびポート番号 (IBM MQ の CONNAME 形式で指定)。例えば、host_name(port_number)	デフォルトなし

表 100. 拡張調整プロパティ: キュー		
プロパティ名	説明	デフォルト値
dynamicQueuePrefix	<p>このプロパティは、一時キュー名を生成するために使用する IBM MQ 接頭辞を定義します。</p> <p>dynamicQueue 接頭部プロパティの形式は、IBM MQ MQOD 構造の DynamicQName フィールドの形式に従います。詳細については、『動的キューの作成』を参照してください。</p> <p>エージェントからの応答を必要とするコマンドにより生成される一時応答キューに、特定の IBM MQ 接頭辞を使用する場合は、command.properties ファイルでこのプロパティを定義することもできます。</p>	WMQFTE.*
modelQueueName	<p>このプロパティは、一時キューを生成するために使用する IBM MQ MQ モデル・キューを定義します。</p> <p>エージェントからの応答を必要とするコマンドにより生成される一時応答キューに、特定の IBM MQ モデル・キューを使用する場合は、command.properties ファイルでこのプロパティを定義することもできます。詳しくは、200 ページの『MFT command.properties ファイル』を参照してください。</p>	SYSTEM.DEFAULT.MODEL.QUEUE

プロパティ名	説明	デフォルト値
userIdForClientConnect	IBM MQ へのクライアント接続を介してフローされるユーザー ID。java が指定されている場合、JVM によって報告されるユーザー名が、IBM MQ 接続要求の一部として送られます。このプロパティの値は None または java のいずれかになります。	なし
coordinationQMGrAuthenticationCredentialsFile	調整キュー・マネージャーに接続するための MQ 接続資格情報が入ったファイルへのパス。	<p>z/OS 認証資格情報ファイルの作成について詳しくは、z/OS での MQMFTCredentials.xml の構成を参照してください。</p> <p>ALW このファイルの場所およびアクセス権については、MQMFTCredentials.xml の構成を参照してください。</p> <p>ALW 認証資格情報ファイルの作成について詳しくは、MFT と IBM MQ の接続認証を参照してください。</p>

プロパティ名	説明	デフォルト値
coordinationSslCipherSpec	コマンドと調整キュー・マネージャー間でのデータの交換時に使用されるプロトコル、ハッシュ・アルゴリズム、および暗号化アルゴリズムと、そのときに暗号鍵で使用されるビット数を指定します。 coordinationSslCipherSpec の値は、CipherSpec 名です。この CipherSpec 名は、調整キュー・マネージャー・チャンネルで使用されている CipherSpec 名と同じです。有効な CipherSpec 名のリストは、IBM MQ classes for Java の SSL/TLS CipherSpec および CipherSuite および IBM MQ classes for JMS の SSL/TLS CipherSpec および CipherSuite に記載されています。 coordinationSslCipherSpec は、coordinationSslCipherSuite とよく似ています。coordinationSslCipherSuite と coordinationSslCipherSpec の両方を指定した場合は、coordinationSslCipherSpec の値が使用されます。	なし
coordinationSslCipherSuite	コマンドと調整キュー・マネージャーのデータ交換方法に関する SSL の動作を指定します。 coordinationSslCipherSuite の値は、CipherSuite 名です。この CipherSuite 名は、エージェント・キュー・マネージャー・チャンネルで使用される CipherSpec 名にマップされます。詳しくは、CipherSuite と CipherSpec の名前マッピングを参照してください。 coordinationSslCipherSuite は、coordinationSslCipherSpec とよく似ています。coordinationSslCipherSuite と coordinationSslCipherSpec の両方を指定した場合は、coordinationSslCipherSpec の値が使用されます。	なし
coordinationSslPeerName	調整キュー・マネージャーによって提供される名前と合致しなければならない識別名の骨組みを指定します。その識別名に基づいて、接続時に調整キュー・マネージャーによって提示される識別用証明書が検査されます。	なし
coordinationSslTrustStore	コマンドが信頼する証明書の場所を指定します。coordinationSslTrustStore の値は、ファイル・パスです。括弧、コンマ (,)、および円記号 (¥) は MFT コマンドの特殊文字であり、円記号 (¥) 文字でエスケープする必要があります。 Windows Windows 上のファイル・パスは、分離文字として二重の円記号 (\\) を使用するか、単一のスラッシュ (/) を使用して指定できます。 IBM WebSphere MQ 7.5 以降、このプロパティの値に環境変数を含めることができます。	なし
coordinationSslTrustStoreType	使用する SSL 鍵ストアのタイプ。JKS および PKCS#12 の鍵ストアがサポートされています。このプロパティの値は、jks または pkcs12 のいずれかになります。	jks
coordinationSslTrustStoreCredentialsFile	coordinationSslTrustStore 資格情報が含まれているファイルのパス。 このプロパティの値には、環境変数を含めることができます。	このプロパティのデフォルト値は、Windows の場合は %USERPROFILE%/MQMFTCredentials.xml、その他のプラットフォームの場合は \$HOME/MQMFTCredentials.xml です。

プロパティ名	説明	デフォルト値
coordinationSslKeyStore	コマンドの秘密鍵の場所を指定します。 coordinationSslKeyStore の値は、ファイル・パスです。括弧 (), および円記号 (¥) は MFT コマンドの特殊文字であり、円記号 (¥) 文字でエスケープする必要があります。  Windows 上のファイル・パスは、分離文字として二重の円記号 (\\) を使用するか、単一のスラッシュ (/) を使用して指定できます。このプロパティが必要なのは、調整キュー・マネージャーでクライアント認証が必要な場合に限られます。 このプロパティの値には、環境変数を含めることができます。	なし
coordinationSslKeyStoreType	使用する SSL 鍵ストアのタイプ。 JKS および PKCS#12 の鍵ストアがサポートされています。このプロパティの値は、 jks または pkcs12 のいずれかになります。	jks
coordinationSslKeyStoreCredentialsFile	coordinationSslKeyStore 資格情報が含まれているファイルのパス。 このプロパティの値には、環境変数を含めることができます。	このプロパティのデフォルト値は、Windows の場合は %USERPROFILE%/MQMFTCredentials.xml、その他のプラットフォームの場合は \$HOME/MQMFTCredentials.xml です。
coordinationSslFipsRequired	調整キュー・マネージャーのレベルで FIPS サポートを使用可能にすることを指定します。このプロパティの値は、 true または false のいずれかになります。詳しくは、MFT の FIPS サポートを参照してください。	false

プロパティ名	説明	デフォルト値
coordinationSubscriptionTopic	このプロパティを使用して、IBM MQ ネットワークの状況に関するパブリケーションを取得するためにサブスクライブする SYSTEM.FTE 以外のトピックを指定します。すべてのツールは、引き続き SYSTEM.FTE トピック。ただし、IBM MQ トポロジーを変更して、これらのパブリケーションをその内容に基づいてさまざまなトピックに配布することができます。それから、この機能を使用して、他のトピックのいずれかにサブスクライブするよう、ツールを強制することができます。	SYSTEM.FTE

関連概念

[MFT 構成オプション \(Multiplatforms\)](#)

関連資料

[fteSetupCoordination](#)

MFT 用の SSL/TLS プロパティ

173 ページの『MFT agent.properties ファイル』

各 Managed File Transfer Agent には独自のプロパティ・ファイル agent.properties があります。このファイルには、エージェントがキュー・マネージャーに接続するために使用する情報が含まれている必要があります。agent.properties ファイルには、エージェントの動作を変更するプロパティを含めることもできます。

200 ページの『MFT command.properties ファイル』

command.properties ファイルには、ユーザーがコマンドを発行するときに接続先となるコマンド・キュー・マネージャーと、Managed File Transfer がそのキュー・マネージャーと情報のやり取りをするのに必要な情報を指定します。

204 ページの『MFT logger.properties ファイル』

Managed File Transfer ロガーには、一連の構成プロパティがあります。これらのプロパティを logger.properties ファイルで指定します。このファイルは、MQ_DATA_PATH/mqft/config/coordination_qmgr_name/loggers/logger_name ディレクトリーにあります。

MFT command.properties ファイル

command.properties ファイルには、ユーザーがコマンドを発行するときに接続先となるコマンド・キュー・マネージャーと、Managed File Transfer がそのキュー・マネージャーと情報のやり取りをするのに必要な情報を指定します。

command.properties ファイルは、インストーラーまたは **fteSetupCommands** コマンドにより作成されます。**fteSetupCommands** コマンドを **-f** フラグと共に使用して、このファイルで基本コマンド・キュー

ー・マネージャー・プロパティを変更できます。拡張コマンド・キュー・マネージャー・プロパティを変更または追加するには、テキスト・エディターでファイルを編集する必要があります。

一部の Managed File Transfer のコマンドは、コマンド・キュー・マネージャーの代わりにエージェント・キュー・マネージャーまたは調整キュー・マネージャーに接続します。どのコマンドがどのキュー・マネージャーに接続するかについては、[MFT コマンドとその接続先のキュー・マネージャー](#)を参照してください。

command.properties ファイルは `MQ_DATA_PATH/mqft/config/coordination_qmgr_name` ディレクトリにあります。

MFT command.properties ファイルには以下の値が含まれます。

プロパティ名	説明	デフォルト値
connectionCredentialsKeyFile	資格情報の暗号化の際に使用される資格情報キーが含まれるファイルの名前。	ストリング・プロパティにデフォルト値はありません。
connectionQMGr	IBM MQ ネットワークとの接続に使用されるキュー・マネージャーの名前。	デフォルトなし
connectionQMGrHost	接続キュー・マネージャーのホスト名または IP アドレス。	デフォルトなし
connectionQMGrPort	クライアント・モードの接続キュー・マネージャーとの接続に使用されるポート番号。	1414
connectionQMGrChannel	接続キュー・マネージャーとの接続に使用される SVRCONN チャンネル名。	SYSTEM.DEF.SVRCONN

connectionQMGrHost プロパティの値を指定しない場合は、デフォルトでバインディング・モードが使用されます。

connectionQMGrHost プロパティに値を指定し、connectionQMGrPort および connectionQMGrChannel プロパティには値を指定しない場合、ポート番号 1414 およびチャンネル SYSTEM.DEF.SVRCONN がデフォルトで使用されます。

以下は、command.properties ファイルの内容の例です。

```
connectionQMGr=PLUTO
connectionQMGrHost=kuiper.example.com
connectionQMGrPort=1930
connectionQMGrChannel=SYSTEM.DEF.SVRCONN
```

この例での PLUTO は、システム `kuiper.example.com` にある IBM MQ キュー・マネージャーの名前です。キュー・マネージャー PLUTO は、Managed File Transfer のコマンドの接続先のキュー・マネージャーです。

拡張コマンド・プロパティ

Managed File Transfer は、さらに拡張コマンド・プロパティも提供します。以下のいずれかのプロパティを使用する場合は、command.properties ファイルを手動で編集して、必要な拡張プロパティを追加します。括弧 ()、および円記号 (¥) は MFT コマンドの特殊文字であり、円記号 (¥) 文字でエスケープする必要があります。Windows Windows 上のファイル・パスは、分離文字として二重の円記号 (\\) を使用するか、単一のスラッシュ (/) を使用して指定できます。Java プロパティ・ファイル内の文字エスケープについては、Oracle 資料 [Properties クラスの Javadoc](#) を参照してください。

- [エージェント・プロパティ](#)
- [コード・ページ・プロパティ](#)
- [マルチインスタンス・キュー・マネージャー・プロパティ](#)
- [キュー・プロパティ](#)
- [セキュリティ・プロパティ](#)
- [SSL プロパティ](#)

表 105. 拡張コマンド・プロパティ: エージェント		
プロパティ名	説明	デフォルト値
failCleanAgentWithNoArguments	デフォルトでは、このプロパティの値は true に設定されているため、エージェント名のパラメーターのみが指定されている場合、 fteCleanAgent コマンドの実行が失敗します。このプロパティを false に設定すると、エージェント名のパラメーターのみが設定されている場合に、 fteCleanAgent コマンドの動作は -all パラメーターを指定したときの動作と同じになります。	true

表 106. 拡張コマンド・プロパティ: コード・ページ		
プロパティ名	説明	デフォルト値
connectionCcsid	コマンドが、コマンド・キュー・マネージャーに接続するときに使用するコード・ページ。connectionCcsid の値を指定する場合は、connectionCcsidName の値も指定する必要があります。	1208
connectionCcsidName	connectionCcsid の Java 表記。connectionCcsidName の値を指定する場合は、connectionCcsid の値も指定する必要があります。	UTF8

表 107. 拡張接続プロパティ: マルチインスタンス・キュー・マネージャー		
プロパティ名	説明	デフォルト値
connectionQMGrStandby	connectionQMGr プロパティによって定義されている、複数インスタンスのコマンド・キュー・マネージャーのスタンバイ・インスタンスに、クライアント接続するために使用するホスト名およびポート番号 (IBM MQ の CONNAME 形式で指定)。例えば、host_name(port_number)	デフォルトなし

表 108. 拡張コマンド・プロパティ: キュー		
プロパティ名	説明	デフォルト値
dynamicQueuePrefix	エージェントからの応答を必要とするコマンドの場合、このプロパティは、一時応答キュー名生成のために使用する IBM MQ 接頭部を定義します。 dynamicQueue 接頭部プロパティの形式は、IBM MQ MQOD 構造の DynamicQName フィールドの形式に従います。詳細については、『 動的キューの作成 』を参照してください。 WMQFTE によって生成される一時キューに特定の IBM MQ 接頭部を使用する場合は、coordination.properties ファイルでこのプロパティを定義することもできます。	WMQFTE.*
modelQueueName	エージェントからの応答を必要とするコマンドの場合、このプロパティは、一時応答キュー生成のために使用する IBM MQ モデル・キューを定義します。 WMQFTE により生成される一時キューに特定の IBM MQ モデル・キューを使用する場合は、coordination.properties ファイルでこのプロパティを定義することもできます。詳しくは、 196 ページ の『 MFT coordination.properties ファイル 』を参照してください。	SYSTEM.DEFAULT.MODEL.QUEUE


接続プロパティ:		
javaLibraryPath	Managed File Transfer では、バインディング・モードでキュー・マネージャーに接続するときに、IBM MQ Java バインディング・ライブラリーへのアクセス権限が必要になります。Managed File Transfer は、デフォルトでは、IBM MQ で定義されているデフォルトの場所でバインディング・ライブラリーを検索します。バインディング・ライブラリーが別の場所にある場合は、このプロパティを使用して、バインディング・ライブラリーの場所を指定してください。	/opt/mqm/java/lib
 legacyXMLMessageMQMDFormat	Managed File Transfer コマンドの XML メッセージは、ブランクの MQMD 形式フィールドと共に、キューに送信されるようになりました。以前のバージョンの製品では、MQMD 形式フィールドを MQSTR (テキスト・メッセージ・ストリング) に設定していました。このプロパティを true に設定すると、Managed File Transfer コマンドの XML メッセージを、MQSTR の MQMD 形式フィールドと共にキューに送信することが可能になります。 MQMD 形式フィールドが MQSTR に設定されている場合、データ変換が使用可能になっている MQ ネットワーク内のチャネルが存在すると、Managed File Transfer コマンドの XML メッセージが壊れる可能性があります。	false

表 109. 拡張コマンド・プロパティ: セキュリティー		
プロパティ名	説明	デフォルト値
userIdForClientConnect	IBM MQ へのクライアント接続を介してフローされるユーザー ID。java が指定されている場合、JVM によって報告されるユーザー名が、IBM MQ 接続要求の一部として送られます。このプロパティの値は None または java のいずれかになります。	なし

表 109. 拡張コマンド・プロパティ: セキュリティー (続き)		
プロパティ名	説明	デフォルト値
connectionQMGrAuthenticationCredentialsFile	コマンド・キュー・マネージャーに接続するための MQ 接続資格情報が入ったファイルへのパス。	MFT および IBM MQ の接続認証とその子トピックを参照してください。

表 110. 拡張コマンド・プロパティ: SSL/TLS		
プロパティ名	説明	デフォルト値
connectionSslCipherSpec	<p>コマンドとコマンド・キュー・マネージャー間でのデータの交換時に使用されるプロトコル、ハッシュ・アルゴリズム、および暗号化アルゴリズムと、そのときに暗号鍵で使用されるビット数を指定します。</p> <p>connectionSslCipherSpec の値は、CipherSpec 名です。この CipherSpec 名は、コマンド・キュー・マネージャー・チャンネルで使用されている名前と同じです。有効な CipherSpec 名のリストは、IBM MQ classes for Java の SSL/TLS CipherSpec および CipherSuite および IBM MQ classes for JMS の SSL/TLS CipherSpec および CipherSuite に記載されています。</p> <p>connectionSslCipherSpec は、connectionSslCipherSuite とよく似ています。connectionSslCipherSuite と connectionSslCipherSpec の両方を指定した場合は、connectionSslCipherSpec の値が使用されます。</p>	なし
connectionSslCipherSuite	<p>コマンドとコマンド・キュー・マネージャーのデータ交換方法に関する SSL の動作を指定します。</p> <p>connectionSslCipherSuite の値は、CipherSuite 名です。この CipherSuite 名は、エージェント・キュー・マネージャー・チャンネルで使用される CipherSpec 名にマップされます。詳しくは、CipherSuite と CipherSpec の名前マッピングを参照してください。</p> <p>connectionSslCipherSuite は、connectionSslCipherSpec とよく似ています。connectionSslCipherSuite と connectionSslCipherSpec の両方を指定した場合は、connectionSslCipherSpec の値が使用されます。</p>	なし
connectionSslPeerName	コマンド・キュー・マネージャーによって提供される名前と合致しなければならない識別名の骨組みを指定します。その識別名に基づいて、接続時にコマンド・キュー・マネージャーによって提示される識別用証明書が検査されます。	なし
connectionSslTrustStore	<p>コマンドが信頼する証明書の場所を指定します。connectionSslTrustStore の値は、ファイル・パスです。括弧、コンマ (,)、および円記号 (¥) は MFT コマンドの特殊文字であり、円記号 (¥) 文字でエスケープする必要があります。</p> <p>Windows Windows 上のファイル・パスは、分離文字として二重の円記号 (\\) を使用するか、単一のスラッシュ (/) を使用して指定できます。</p> <p>このプロパティの値には、環境変数を含めることができます。</p>	なし
connectionSslTrustStoreType	使用する SSL トラストストアのタイプ。JKS および PKCS#12 の鍵ストアがサポートされています。このプロパティの値は、jks または pkcs12 のいずれかになります。	jks
connectionSslTrustStoreCredentialsFile	connectionSslTrustStore 資格情報が含まれているファイルのパス。このプロパティの値には、環境変数を含めることができます。	このプロパティのデフォルト値は、Windows の場合は %USERPROFILE%/MQMFTCredentials.xml、その他のプラットフォームの場合は \$HOME/MQMFTCredentials.xml です。
connectionSslKeyStore	<p>コマンドの秘密鍵の場所を指定します。connectionSslKeyStore の値は、ファイル・パスです。括弧、コンマ (,)、および円記号 (¥) は MFT コマンドの特殊文字であり、円記号 (¥) 文字でエスケープする必要があります。</p> <p>Windows Windows 上のファイル・パスは、分離文字として二重の円記号 (\\) を使用するか、単一のスラッシュ (/) を使用して指定できます。このプロパティが必要なのは、コマンド・キュー・マネージャーでクライアント認証が必要な場合に限られます。</p> <p>このプロパティの値には、環境変数を含めることができます。</p>	なし
connectionSslKeyStoreType	使用する SSL 鍵ストアのタイプ。JKS および PKCS#12 の鍵ストアがサポートされています。このプロパティの値は、jks または pkcs12 のいずれかになります。このプロパティの値には、環境変数を含めることができます。	jks
connectionSslKeyStoreCredentialsFile	connectionSslKeyStore 資格情報が含まれているファイルのパス。このプロパティの値には、環境変数を含めることができます。	このプロパティのデフォルト値は、Windows の場合は %USERPROFILE%/MQMFTCredentials.xml、その他のプラットフォームの場合は \$HOME/MQMFTCredentials.xml です。

表 110. 拡張コマンド・プロパティ: SSL/TLS (続き)		
プロパティ名	説明	デフォルト値
connectionSslFipsRequired	コマンド・キュー・マネージャーのレベルで FIPS サポートを使用可能にすることを指定します。このプロパティの値は、true または false のいずれかになります。詳しくは、 MFT の FIPS サポート を参照してください。	false

関連概念

[MFT 構成オプション \(Multiplatforms\)](#)

関連資料

[217 ページの『MFT Java システム・プロパティ』](#)

いくつかの Managed File Transfer コマンド・プロパティとエージェント・プロパティは Java システム・プロパティとして定義する必要があります。これは、コマンド・プロパティおよびエージェント・プロパティのメカニズムを使用できない初期の関数用の構成を定義するためです。

[MFT 用の SSL/TLS プロパティ](#)

[173 ページの『MFT agent.properties ファイル』](#)

各 Managed File Transfer Agent には独自のプロパティ・ファイル `agent.properties` があります。このファイルには、エージェントがキュー・マネージャーに接続するために使用する情報が含まれている必要があります。`agent.properties` ファイルには、エージェントの動作を変更するプロパティを含めることもできます。

[196 ページの『MFT coordination.properties ファイル』](#)

`coordination.properties` ファイルは、調整キュー・マネージャーとの接続の詳細を指定します。複数の Managed File Transfer インストール済み環境で同じ調整キュー・マネージャーが共用される場合があるため、共用ドライブ上の共通の `coordination.properties` ファイルへのシンボリック・リンクを使用できます。

[204 ページの『MFT logger.properties ファイル』](#)

Managed File Transfer ロガーには、一連の構成プロパティがあります。これらのプロパティを `logger.properties` ファイルで指定します。このファイルは、`MQ_DATA_PATH/mqft/config/coordination_qmgr_name/loggers/logger_name` ディレクトリーにあります。

[fteSetupCommands: MFT command.properties ファイルの作成](#)

[fteCleanAgent: MFT エージェントのクリーンアップ](#)

MFT logger.properties ファイル

Managed File Transfer ロガーには、一連の構成プロパティがあります。これらのプロパティを `logger.properties` ファイルで指定します。このファイルは、`MQ_DATA_PATH/mqft/config/coordination_qmgr_name/loggers/logger_name` ディレクトリーにあります。

ファイルまたはディレクトリーの場所を表すいくつかの Managed File Transfer プロパティでは、環境変数を使用できます。これにより、製品の一部の実行時に使用されるファイルまたはディレクトリーの場所を、環境の変更(プロセスを実行しているユーザーなど)に合わせて変えることができます。詳しくは、[167 ページの『MFT プロパティでの環境変数の使用』](#) を参照してください。

注: 括弧、コンマ (,)、および円記号 (¥) は MFT コマンドの特殊文字であり、円記号 (¥) 文字でエスケープする必要があります。 **Windows** Windows 上のファイル・パスは、分離文字として二重の円記号 (\\) を使用するか、単一のスラッシュ (/) を使用して指定できます。Oracle の Java プロパティ・ファイルでの文字エスケープについて詳しくは、[Properties クラスの場合は Javadoc](#) を参照してください。

MFT `logger.properties` ファイルには以下の値が含まれます。

- [205 ページの『バインディング・モード接続のプロパティ』](#)
- [212 ページの『クライアント・モード SSL/TLS 接続のプロパティ』](#)

バイディング・モード接続のプロパティ

表 111. <i>logger.properties</i> ファイルのバイディング・モードの接続プロパティ		
プロパティ名	説明	デフォルト値
<code>wmqfte.logger.type</code>	使用しているローガー・タイプ (file または database)。この値は FILE または DATABASE に設定してください。	デフォルト値なし
<code>wmqfte.max.transaction.messages</code>	トランザクションがコミットされる前にトランザクションで処理されるメッセージの最大数。循環ロギング・モードでは、未完了データに使用可能な固定スペース量はキュー・マネージャーが持っています。使用可能なスペースを使い尽くさないよう十分に低い値をこのプロパティに設定するようにしてください。	50
<code>wmqfte.max.transaction.time</code>	トランザクションのコミット間に経過する時間の最大長 (ミリ秒)。	5000
<code>wmqfte.max.consecutive.reject</code>	連続して (つまり、有効メッセージが出現することなく) リジェクト可能なメッセージの最大数。 この数値を超えると、ローガーは問題がメッセージ自体にあるのではなく構成にあると見なします。例えば、使用しているすべてのエージェント名よりデータベース内のエージェント名を狭く作ると、エージェントを参照しているすべてのメッセージがリジェクトされます。	50
<code>wmqfte.reject.queue.name</code>	ローガーがメッセージを処理できない場合にローガーがメッセージを置くキューの名前。データベース・ローガーを使用している場合、このキューに置かれる可能性のあるメッセージについて詳しくは、 MFT ローガーのエラー処理とリジェクト を参照してください。	SYSTEM.FTE.LOG.RJCT. <i>logger_name</i>
<code>wmqfte.command.queue.name</code>	ローガーがその動作を制御するコマンド・メッセージを読み取る際に、その読み取り元となるキューの名前。	SYSTEM.FTE.LOG.CMD. <i>logger_name</i>
<code>wmqfte.queue.manager</code>	ローガーの接続先キュー・マネージャー。このパラメーターは必須で、キュー・マネージャーへのバイディング・モード接続に必要なものはこのパラメーターのみです。(リモート・キュー・マネージャーに接続するためのプロパティについては、 212 ページの表 112 を参照してください。)	デフォルト値なし
<code>wmqfte.message.source.type</code>	以下のいずれかの値になります。 automatic subscription デフォルト値。ローガーは、SYSTEM.FTE/Log/# で定義される独自の永続的な管理対象サブスクリプションをキュー・マネージャー上に作成して使用します。これは、ほとんどのシナリオで適切な値となります。 administrative subscription 自動サブスクリプションが適切でない場合は、(例えば、IBM MQ Explorer、MQSC、または PCF を使用して) 異なるサブスクリプションを定義し、ローガーにそのサブスクリプションを使用するよう命令できます。例えば、この値を使用してログ・スペースを分割し、1 番目のローガーは A から H のエージェントを、2 番目のローガーは I から P のエージェントを、3 番目のローガーは Q から Z のエージェントを処理するようにします。 queue IBM MQ トポロジーを使用するとローガーのサブスクリプションを作成することが不都合であるとわかった場合は、代わりにキューを使用できます。通常は調整キュー・マネージャー上の SYSTEM.FTE/Log/# へのサブスクリプションによって受信されるメッセージをキューが受信するように、IBM MQ を構成します。	automatic subscription

表 111. logger.properties ファイルのバインディング・モードの接続プロパティ (続き)

プロパティ名	説明	デフォルト値
wmqfte.message.source.name	<p>メッセージ送信元タイプが administrative subscription または queue である場合に使用するサブスクリプションまたはキューの名前。このプロパティは、送信元タイプが automatic subscription である場合は無視されます。</p>	デフォルト値なし
wmqfte.database.credentials.file	<p>データベースに接続するためのユーザー名とパスワードを格納するファイル。</p> <p>このプロパティの値には、環境変数を含めることができます。</p> <p>詳しくは、MFT の資格情報ファイルのフォーマットを参照してください。</p>	<p>z/OS 認証資格情報ファイルの作成については、z/OS での MQMFTCredentials.xml の構成を参照してください。</p> <p>ALW このファイルの場所およびアクセス権については、MQMFTCredentials.xml の構成を参照してください。</p> <p>ALW MFT および IBM MQ 接続認証も参照してください。</p>
wmqfte.database.driver	<p>データベースの JDBC ドライバー・クラスの場所。これは通常、JAR ファイルのパスおよびファイル名です。</p> <p>AIX 例えば、AIX システム上の Db2 用の Type 2 ドライバーは、ファイル /opt/IBM/db2/V9.5/java/db2jcc.jar を必要とします。</p> <p>Windows Windows システムの場合、パス分離文字をスラッシュ文字 (/) で指定します。例: C:/Program Files/IBM/SQLLIB/java/db2jcc.jar。</p> <p>z/OS z/OS の場合、db2jcc.jar ファイルの絶対パスを指定します。例: wmqfte.database.driver=/db2/db2v10/jdbc/classes/db2jcc.jar。</p> <p>z/OS z/OS システムの場合、以下のすべての JAR ファイルを参照する必要があります。</p> <ul style="list-style-type: none"> • db2jcc.jar • db2jcc_license_cisuz.jar • db2jcc_javax.jar <p>データベース・ドライバーが複数の JAR ファイルで構成されている場合は (例えば、Db2 V9.1 にはドライバー JAR ファイルとライセンス JAR ファイルが必要です)、これらの JAR ファイルをすべてこのプロパティに含めます。複数のファイル名は、ご使用のプラットフォームのクラスパス分離文字、つまり、Windows システムの場合はセミコロン文字 (;)、他のプラットフォームの場合はコロン文字 (:) を使用して分離します。</p>	デフォルト値なし

表 111. `logger.properties` ファイルのバインディング・モードの接続プロパティ (続き)

プロパティ名	説明	デフォルト値
<code>wmqfte.database.exclude.metadata</code>	<p>データベース・ロガー・スキーマ内の他の表にある情報を含むメタデータ表にエントリを保管するかどうかを制御します。この値は <code>true</code> または <code>false</code> に設定してください。これらのメタデータ・エントリは、既存のデータと重複しており、データベース・ストレージ容量を無駄に占有することになるため、デフォルトでは保管されません。同じデータが出現するプロパティ・エントリおよび表は以下のとおりです。</p> <ul style="list-style-type: none"> • <code>com.ibm.wmqfte.SourceAgent TRANSFER_EVENT</code> または <code>CALL_REQUEST</code> • <code>com.ibm.wmqfte.DestinationAgent TRANSFER_EVENT</code> • <code>com.ibm.wmqfte.MqmdUser TRANSFER_EVENT</code> または <code>CALL_REQUEST</code> • <code>com.ibm.wmqfte.OriginatingUser TRANSFER_EVENT</code> または <code>CALL_REQUEST</code> • <code>com.ibm.wmqfte.OriginatingHost TRANSFER_EVENT</code> または <code>CALL_REQUEST</code> • <code>com.ibm.wmqfte.TransferId TRANSFER</code> または <code>CALL_REQUEST</code> • <code>com.ibm.wmqfte.JobName TRANSFER</code> または <code>CALL_REQUEST</code> <p>このプロパティの値を <code>false</code> に設定すると、これらのメタデータ・エントリがメタデータ表に保管されます。</p>	<code>true</code>
<code>wmqfte.database.host</code>	<p>Db2 のみ:</p> <p>タイプ 4 JDBC ドライバーを使用して接続するデータベース・サーバーのホスト名。このプロパティの値を指定する場合は、<code>wmqfte.database.port</code> の値も指定する必要があります。両方のプロパティが定義されていない場合は、データベース・ロガーがデフォルトの Type 2 JDBC ドライバーを使用して接続されます。</p> <p>このプロパティに値を指定する場合は、データベースに接続するユーザー名とパスワードを定義するために、このロガー用の資格情報ファイル (<code>wmqfte.database.credentials.file</code> プロパティでファイル・パスを定義) が存在しており、アクセス可能であることが必要です。資格情報ファイルは、データベースがローカル・システム上にある場合でも必要です。</p>	デフォルト値なし
<code>wmqfte.database.name</code>	<p>Managed File Transfer ログ表を含むデータベース・インスタンス (または、Db2 for z/OS を使用する場合はサブシステム) の名前。</p>	デフォルト値なし
<code>wmqfte.database.type</code>	<p>使用中のデータベース管理システム。Db2 または Oracle。この値を <code>db2</code> または <code>oracle</code> に設定してください。</p>	<code>db2</code>

表 111. `logger.properties` ファイルのバインディング・モードの接続プロパティ (続き)

プロパティ名	説明	デフォルト値
<code>wmqfte.database.port</code>	<p>Db2 のみ:</p> <p>タイプ 4 JDBC ドライバーを使用して接続するデータベース・サーバーのポート番号。このプロパティの値を指定する場合は、<code>wmqfte.database.host</code> の値も指定する必要があります。両方のプロパティが定義されていない場合は、データベース・ロガーがデフォルトの Type 2 JDBC ドライバーを使用して接続されます。</p> <p>このプロパティに値を指定する場合は、データベースに接続するユーザー名とパスワードを定義するために、このロガー用の資格情報ファイル (<code>wmqfte.database.credentials.file</code> プロパティでファイル・パスを定義) が存在しており、アクセス可能である必要があります。資格情報ファイルは、データベースがローカル・システム上にある場合でも必要です。</p>	デフォルト値なし
<code>wmqfte.database.schema</code>	<p>Db2 のみ:</p> <p>Managed File Transfer ログイン表を含むデータベース・スキーマ。ほとんどの場合、デフォルト値が適切ですが、サイト固有のデータベースに関する考慮事項に応じて代替値を指定する必要がある場合があります。</p>	FTELOG
<code>wmqfte.database.native.library.path</code>	<p>選択したデータベース・ドライバーが必要とするネイティブ・ライブラリーを含むパス (該当する場合)。</p> <p>AIX 例: <code>/opt/IBM/db2/V9.5/lib32/</code> からライブラリーを必要とします。このプロパティの代わりに、他の方法を使用して <code>java.library.path</code> システム・プロパティを設定できます。</p>	デフォルト値なし
<code>wmqfte.file.logger.fileDirectory</code>	ファイル・ロガーのログ・ファイルを配置するディレクトリー。	<code>mqft/logs/coordination_dir/loggers/logger_name/logs</code>
<code>wmqfte.file.logger.fileSize</code>	<p>ログ・ファイルが大きくなるのが許可される最大サイズ。サイズの値は、ゼロより大きい正整数で、単位 KB、MB、GB、m (分)、h (時)、d (日)、w (週) のいずれか 1 つを後に付けます。例えば、<code>wmqfte.file.logger.fileSize=5MB</code> は最大ファイル・サイズとして 5MB を指定し、<code>wmqfte.file.logger.fileSize=2d</code> は最大ファイル・サイズとして 2 日分のデータを指定します。</p>	10MB
<code>wmqfte.file.logger.fileCount</code>	作成するログ・ファイルの最大数。データ量がこの数のファイルに保管できる最大量を超えた場合、最も古いファイルが削除されます。そのため、ファイル数が指定された値より大きくなることはありません。	3

表 111. *logger.properties* ファイルのバインディング・モードの接続プロパティ (続き)

プロパティ名	説明	デフォルト値
<p><code>wmqfte.file.logger.mode</code></p>	<p>使用しているローガー・モード (circular または linear)。この値は CIRCULAR または LINEAR に設定してください。</p> <p>CIRCULAR - ファイル・ローガーは、wmqfte.file.logger.fileSize プロパティを使用して定義される最大サイズにファイルが到達するまで、そのファイルに情報を書き込みます。最大サイズに到達すると、ファイル・ローガーは新規ファイルを開始します。このモードで書き込まれるファイルの最大数は、wmqfte.file.logger.fileCount プロパティを使用して定義される値によって制御されます。このファイル最大数に到達すると、ファイル・ローガーは最初のファイルを削除して再作成し、それを現行のアクティブ・ファイルとして使用します。wmqfte.file.logger.fileSize プロパティに定義されている値が固定サイズのバイト単位 (例えば、KB、MB、または GB) である場合、このモードで使用されるディスク・スペースの上限は、<code>fileSize</code> に <code>fileCount</code> を乗算した値になります。</p> <p>wmqfte.file.logger.fileSize プロパティで定義された値が時間単位 (例えば、m、h、d、w) の場合、最大サイズは、ご使用のシステムにおけるその時間内のログ・メッセージのスループットによって決まります。このモードで実行される場合に使用されるログ・ファイル命名規則は、<code>logger_namenumber-timestamp.log</code> です。説明:</p> <ul style="list-style-type: none"> • <code>logger_name</code> は、fteCreateLogger コマンドでローガーに付けられた名前です。 • <code>number</code> は、セット内のファイルの番号です。 • <code>timestamp</code> は、ファイルが作成されたときのタイム・スタンプです。 <p>例えば、<code>LOGGER1-20111216123430147.log</code></p> <p>LINEAR - ファイル・ローガーは、wmqfte.file.logger.fileSize プロパティを使用して定義される最大サイズにファイルが到達するまで、そのファイルに情報を書き込みます。最大サイズに到達すると、ファイル・ローガーは新規ファイルを開始します。以前に書き込まれたファイルは削除されないため、それらをログ・メッセージの履歴レコードとして保持できます。linear モードで実行している場合、ファイルは削除されず、作成可能なファイル数の上限もないため、wmqfte.file.logger.fileCount プロパティは無視されます。このモードで実行している場合には上限がないため、ディスク・スペースが少なくならないように、ログ・ファイルによって使用されているディスク・スペースの量を追跡する必要があります。このモードで実行される場合に使用されるログ・ファイル命名規則は、<code>logger_name-timestamp.log</code> です。説明:</p> <ul style="list-style-type: none"> • <code>logger_name</code> は、fteCreateLogger コマンドでローガーに付けられた名前です。 • <code>timestamp</code> は、ファイルが作成されたときのタイム・スタンプです。 <p>例えば、<code>LOGGER-20111216123430147.log</code></p>	<p>デフォルト値なし</p>

表 111. logger.properties ファイルのバインディング・モードの接続プロパティ (続き)

プロパティ名	説明	デフォルト値
wmqfte.max.retry.interval	<p>ロガーが永続エラーを検出した場合の、再試行間の最大時間 (ミリ秒)。</p> <p>エラー条件によっては (例えば、データベース接続が失われた場合)、ロガーが続行できなくなります。このタイプの条件が発生した場合、ロガーは現行トランザクションをロールバックし、ある一定の時間待機した後、再試行します。ロガーの待機時間は最初非常に短いため、一時的なエラーであればすぐに克服できます。しかし、ロガーの再試行のたびに、待機時間は増大します。これにより、エラー条件がより長く続く場合、例えば、データベースが保守のためのサービス停止になっている場合に、あまりに多くの不要な作業が発生しないようになっています。</p> <p>このプロパティを使用して待機時間の長さに制限を設けると、エラー条件が解決される妥当な時間で再試行が発生します。</p>	600
immediateShutdownTimeout	<p>未処理の操作が完了して、適切にシャットダウンするのをロガーが待機する時間 (秒数)。デフォルトでは、ロガーは操作が完了するまで 10 秒間待機します。タイムアウトまでに操作が完了しない場合、ロガーは以下のイベント・メッセージを <code>output0.log</code> に書き込んでから終了します。</p> <p><code>BFGDB0082I: The logger is ending immediately.</code></p> <p>ゼロの値を指定すると、ロガーは現在の操作が完了するまで待機します。</p> <p><code>immediateShutdownTimeout</code> の値がゼロより小さい値に設定されている場合は、デフォルト値が使用されます。</p> <p>このプロパティは、スタンドアロン・データベース・ロガーとファイル・タイプ・ロガーの両方に適用されます。</p>	10
loggerCredentialsKeyFile	資格情報の暗号化の際に使用される資格情報キーが含まれるファイルの名前。	ストリング・プロパティにデフォルト値はありません。
loggerQMGrRetryInterval	ロガーのプロセス・コントローラーがキュー・マネージャーの使用可否を検査する間隔 (秒単位)。	30
maxRestartCount	<code>maxRestartInterval</code> プロパティの値によって指定された時間間隔内に行える再開の最大回数。この値を超えると、ロガーのプロセス・コントローラーはロガーの再開を中止し、その代わりに <code>maxRestartDelay</code> プロパティの値に基づいたアクションを実行します。	4
maxRestartInterval	ロガーのプロセス・コントローラーがロガーの再開数を測定する間隔 (秒単位)。この間隔内に行われた再開の回数が <code>maxRestartCount</code> プロパティの値を超えると、ロガーのプロセス・コントローラーはそのロガーの再開を中止します。ロガーのプロセス・コントローラーは、その代わりに <code>maxRestartDelay</code> プロパティの値に基づいたアクションを実行します。	120
maxRestartDelay	ロガーの再開率が <code>maxRestartCount</code> プロパティおよび <code>maxRestartInterval</code> プロパティの値を超過した場合にロガーのプロセス・コントローラーがとる動作を決定します。ゼロ以下の値を指定すると、ロガーのプロセス・コントローラーは停止します。ゼロより大きな値を指定した場合、この秒数だけ待機した後、ロガーのプロセス・コントローラーが保持する再開履歴情報がリセットされ、ロガーが再開されます。	-1
wmqfte.oracle.port	ロガーが Oracle インスタンスへの接続に使用するポート。このポートを TNS リスナーとも言います。	1521

表 111. logger.properties ファイルのバインディング・モードの接続プロパティ (続き)		
プロパティ名	説明	デフォルト値
wmqfte.oracle.host	ロガーが Oracle インスタンスへの接続に使用するホスト。	localhost
armELEMTYPE	オプションのプロパティ。ロガーを自動リスタート・マネージャー (ARM) で再始動するように構成する場合は、関連付けられた ARM ポリシー内で指定された ARM ELEMTYPE パラメーター値をこのプロパティに設定します。ロガーの場合、ELEMTYPE は SYSBFGLG に設定します。	設定なし
armELEMENT	オプションのプロパティ。ロガーを自動リスタート・マネージャー (ARM) で再始動するように構成する場合は、関連付けられた ARM ポリシー内で指定された ARM ELEMENT パラメーター値をこのプロパティに設定します。ELEMENT 値には、ロガー名に対応する値を設定できます。	設定なし
loggerQMGrAuthenticationCredentials File	ロガーの調整キュー・マネージャーに接続するための MQ 接続資格情報が入ったファイルへのパス。	<p>z/OS 認証資格情報ファイルの作成については、z/OS での MQMFTCredentials.xml の構成を参照してください。</p> <p>ALW このファイルの場所およびアクセス権については、MQMFTCredentials.xml の構成を参照してください。</p> <p>ALW MFT および IBM MQ 接続認証も参照してください。</p>
トレース	<p>オプションのプロパティ。ロガーの開始時にトレースを有効にしてロガーを実行するトレース仕様。トレース仕様は、クラス、等号、およびトレース・レベルのコンマ区切りのリストです。</p> <p>例えば、com.ibm.wmqfte.databaselogger、および com.ibm.wmqfte.databaselogger.operation=all です。</p> <p>コロン区切りのリストで、複数のトレース仕様を指定できます。例えば、com.ibm.wmqfte.databaselogger=moderate:com.ibm.wmqfte.databaselogger.operation=all</p>	なし
traceFiles	オプションのプロパティ。保持するトレース・ファイルの合計数。この値は、ロガーのプロセス・コントローラーに加え、ロガー自体にも適用されます。	5
traceSize	オプションのプロパティ。トレースが次のファイルで折り返す前の、各トレース・ファイルの最大サイズ (MB 単位)。この値は、ロガーのプロセス・コントローラーとロガー自体に適用されます。	20

表 111. *logger.properties* ファイルのバインディング・モードの接続プロパティ (続き)

プロパティ名	説明	デフォルト値
<code>wmqfte.file.logger.filePermissions</code>	<p>オプションのプロパティ。 ロガーのログ・ファイルに必要な権限の種類を指定するために使用します。</p> <p>このプロパティは、リニア・ログと循環ログの両方に適用され、値として <code>UserReadWriteOnly</code> または <code>UserReadWriteAllRead</code> を取ることができます。</p> <p><code>UserReadWriteOnly</code> 値には既存の同等の権限 600 があり、<code>UserReadWriteAllRead</code> 値には同等の権限 644 があります。</p> <p>権限の変更は、新しく作成されたロガー・ファイルに適用されます。</p> <p>無効なプロパティに値を入力すると、ロガーはデフォルト値を使用し、メッセージ BFGDB0083W を出力ログに発行します。</p>	UserReadWriteOnly

クライアント・モード SSL/TLS 接続のプロパティ

SSL/TLS を使用したロガー・キュー・マネージャーへのクライアント・モード接続をサポートするために必要なプロパティ。

表 112. *logger.properties* ファイルのクライアント・モード SSL/TLS 接続のプロパティ


プロパティ名	説明	デフォルト値
<code>wmqfte.queue.manager.host</code>	ロガー・キュー・マネージャーのホスト名または IP アドレス。	デフォルト値なし
<code>wmqfte.queue.manager.port</code>	ロガー・キュー・マネージャーが listen するポート。	1414
<code>wmqfte.queue.manager.channel</code>	ロガー・キュー・マネージャーのサーバー接続チャンネルの名前。	SYSTEM.DEF.SVRCONN
<code>wmqfte.Ssl.CipherSuite</code>	<p>ロガーとロガー・キュー・マネージャーのデータ交換方法に関する TLS の動作を指定します。</p> <p>wmqfte.Ssl.CipherSuite の値は CipherSuite 名です。この CipherSuite 名は、ロガー・キュー・マネージャー・チャンネルで使用される CipherSpec 名にマップされます。</p> <p>詳しくは、CipherSuite と CipherSpec の名前のマッピングを参照してください。</p>	デフォルト値なし
<code>wmqfte.Ssl.PeerName</code>	ロガー・キュー・マネージャーによって提供される名前と合致しなければならない識別名の骨組みを指定します。その識別名に基づいて、接続時にキュー・マネージャーによって提示される識別用証明書が検査されます。	デフォルト値なし
<code>wmqfte.Ssl.TrustStore</code>	<p>ロガーが信頼する証明書の場所を指定します。wmqfte.Ssl.TrustStore の値はファイル・パスです。</p> <p>括弧、コンマ (,)、および円記号 (¥) は MFT コマンドの特殊文字であり、円記号 (¥) 文字でエスケープする必要があります。</p> <p> Windows 上のファイル・パスは、分離文字として二重の円記号 (\\) を使用するか、単一のスラッシュ (/) を使用して指定できます。</p> <p>このプロパティの値には環境変数を含めることができる点にご注意ください。</p>	デフォルト値なし

表 112. <code>logger.properties</code> ファイルのクライアント・モード SSL/TLS 接続のプロパティ (続き)		
プロパティ名	説明	デフォルト値
<code>wmqfte.Ssl.TrustStoreCredentialsFile</code>	wmqfte.Ssl.TrustStore 資格情報が含まれているファイルのパス。 このプロパティの値には環境変数を含めることができる点にご注意ください。	デフォルト値なし
<code>wmqfte.Ssl.TrustStoreType</code>	使用する SSL 鍵ストアのタイプ。JKS および PKCS#12 の鍵ストアがサポートされています。このプロパティの値は、jks または pkcs12 のいずれかになります。	jks
<code>wmqfte.Ssl.KeyStore</code>	ローガの秘密鍵の場所を指定します。 wmqfte.Ssl.KeyStore の値はファイル・パスです。 括弧、コンマ (,)、および円記号 (¥) は MFT コマンドの特殊文字であり、円記号 (¥) 文字でエスケープする必要があります。 Windows Windows 上のファイル・パスは、分離文字として二重の円記号 (\\) を使用するか、単一のスラッシュ (/) を使用して指定できます。 このプロパティの値には環境変数を含めることができる点にご注意ください。	デフォルト値なし
<code>wmqfte.Ssl.KeyStore.CredentialsFile</code>	wmqfte.Ssl.KeyStore 資格情報が含まれているファイルのパス。 このプロパティの値には環境変数を含めることができる点にご注意ください。	デフォルト値なし
<code>wmqfte.Ssl.KeyStoreType</code>	使用する SSL 鍵ストアのタイプ。JKS および PKCS#12 の鍵ストアがサポートされています。このプロパティの値は、jks または pkcs12 のいずれかになります。	jks
<code>wmqfte.Ssl.FipsRequired</code>	ローガのレベルで FIPS サポートを使用可能にすることを指定します。このプロパティの値は、true または false のいずれかになります。詳しくは、 MFT の FIPS サポート を参照してください。	false

関連概念

[MFT 用の SSL/TLS プロパティ](#)

関連資料

[167 ページの『MFT プロパティでの環境変数の使用』](#)

ファイルまたはディレクトリの場所を表す Managed File Transfer プロパティで環境変数を使用することができます。これにより、製品の一部を実行するときに使用されるファイルまたはディレクトリの場所を、現在の環境 (例えば、コマンドを実行するユーザー) に応じて変えることができます。

[173 ページの『MFT agent.properties ファイル』](#)

各 Managed File Transfer Agent には独自のプロパティ・ファイル `agent.properties` があります。このファイルには、エージェントがキュー・マネージャーに接続するために使用する情報が含まれている必要があります。`agent.properties` ファイルには、エージェントの動作を変更するプロパティを含めることもできます。

[200 ページの『MFT command.properties ファイル』](#)

`command.properties` ファイルには、ユーザーがコマンドを発行するときに接続先となるコマンド・キュー・マネージャーと、Managed File Transfer がそのキュー・マネージャーと情報のやり取りをするのに必要な情報を指定します。

[196 ページの『MFT coordination.properties ファイル』](#)

`coordination.properties` ファイルは、調整キュー・マネージャーとの接続の詳細を指定します。複数の Managed File Transfer インストール済み環境で同じ調整キュー・マネージャーが共用される場合があ

るため、共用ドライブ上の共通の `coordination.properties` ファイルへのシンボリック・リンクを使用できます。

LogTransfer 関数によって生成される出力

転送ログ・イベントは、転送が実行依頼されてから完了するまでの転送の進行状況の詳細をキャプチャします。再同期化に入る転送に関する情報もキャプチャされ、転送の進行状況を把握するのに役立ちます。

転送イベント形式

転送イベントは JSON 形式で、`transferlogN.json` ファイルに書き込まれます。このファイルは、エージェントのログ・ディレクトリーに作成されます。ここで、`N` は数値で、`0` がデフォルトです。すべてのイベントに以下の共通属性が含まれます。

- 日付および時刻 (UTC)
- 固有 ID

イベントのタイプおよび転送ログのレベルに応じて、書き込まれるイベント情報には追加の属性が含まれます。転送ログ・レベルの `info` では最小限の情報が書き込まれますが、`verbose` レベルではより詳細な情報が含まれます。以下の [214 ページの『サンプル・イベント』](#) のセクションで、エージェントによってログに記録される転送イベントの例をいくつか示します。

固有 ID

転送の進行に応じてさまざまなフェーズを簡単に識別できるように、固有 ID が含まれています (例: `BFGTL0001`)。固有 ID は `eventDescription` 属性の一部であり、以下の 2 つの部分で構成されます。

BFGTL

すべての ID に使用される接頭部。ここで、`BFG` は Managed File Transfer で使用される標準接尾部であり、`TL` はこれが転送ログであることを示します。

数値

1 から始まる固有の番号。以下に例を示します。

```
{
  "eventDescription": "BFGTL0001: New transfer request submitted"
}
```

サンプル・イベント

以下の表で、追加の関数によってログに記録される情報の例として、いくつかのイベントを説明します。表の 2 列目のログ・レベルは、イベントがログに記録されるときのレベルを示しています。

重要: `logTransfer` レベルが詳細または中程度に設定されている場合、以下の属性がイベント情報に組み込まれます。

- `sourceAgent`
- `destinationAgent`
- `threadId`

イベント	ログ・レベル	説明
転送する項目のリスト	verbose	<pre> { "dateTime": "<Data time in UTC>", "eventDescription": "BFGTL0002I: Generated detailed transfer item list.", "destinationAgent": "<Name of destination agent>", "sourceAgent": "<Name of source agent>", "threadId": "0000001d", "totalItemsInTransfer": <Number of items in the transfer>, "transferId": "<Transfer Identifier>", "transferItemsList": [{ "source": "source item name", "destination": "destination item name" }] } Example: { "dateTime": "2022-01-14T12:56:54.219Z UTC", "eventDescription": "BFGTL0002I: Generated detailed transfer item list.", "destinationAgent": "QMBAGQ", "sourceAgent": "QMBAG1", "threadId": "0000001d", "totalItems": 1, "transferId": "414d5120514d4120202020202020202063bd17610a390040", "transferItems": [{ "destination": "/results/rts/target/destFile.txt", "source": "DESTINATIONQ@QMB" }] } </pre>

イベント	ログ・レベル	説明
エージェントの開始時にリカバリーされる転送のリスト	verbose	<pre>{ "dateTime": "<Date and time in UTC>", "eventDescription": "The list of transfers being recovered as part of agent recovery process.", "agentName": "<Agent name>", "transfers": [{ "transferId": "<transfer state>" }] }</pre> <p>Example:</p> <pre>{ "dateTime": "2022-01-14T14:42:24.902Z UTC", "eventDescription": "The list of transfers being recovered as part of agent recovery process.", "agentName": "CQMXX01AG1", "transfers": [{ "414D512043514D4858303120202020B0D4176101370040": "completeReceived" }, { "414D512043514D4858303120202020B0D4176101370050": "resynchronizing" }] "threadId": "0000001c", }</pre>

関連資料

217 ページの『MFT Java システム・プロパティー』

いくつかの Managed File Transfer コマンド・プロパティーとエージェント・プロパティーは Java システム・プロパティーとして定義する必要があります。これは、コマンド・プロパティーおよびエージェント・プロパティーのメカニズムを使用できない初期の関数用の構成を定義するためです。

fteCreateAgent

167 ページの『MFT プロパティーでの環境変数の使用』

ファイルまたはディレクトリーの場所を表す Managed File Transfer プロパティーで環境変数を使用することができます。これにより、製品の一部を実行するときを使用されるファイルまたはディレクトリーの場所を、現在の環境 (例えば、コマンドを実行するユーザー) に応じて変えることができます。

MFT Java システム・プロパティー

いくつかの Managed File Transfer コマンド・プロパティーとエージェント・プロパティーは Java システム・プロパティーとして定義する必要があります。これは、コマンド・プロパティーおよびエージェント・プロパティーのメカニズムを使用できない初期の関数用の構成を定義するためです。

システム・プロパティーおよび Managed File Transfer コマンドを実行する JVM のその他の JVM オプションは、環境変数 `BFG_JVM_PROPERTIES` を定義することで定義します。例えば、UNIX-type タイプのプラットフォームで `com.ibm.wmqfte.maxConsoleLineLength` プロパティーを設定するには、次のように変数を定義します。

```
export BFG_JVM_PROPERTIES="-Dcom.ibm.wmqfte.maxConsoleLineLength=132"
```

エージェントを Windows サービスとして実行している場合は、**fteModifyAgent** コマンドで `-sj` パラメーターを指定することにより、エージェントの Java システム・プロパティーを変更できます。




プロパティー名	説明	値
<code>com.ibm.wmqfte.maxConsoleLineLength</code>	コンソールに書き込むことができる行の最大長。この長さを超える行は折り返されます。この値は、文字数ではなくバイト数で表します。	 IBM i のデフォルト長は 132 バイトです。  AIX, Linux, and Windows の場合、長さは無制限です。  z/OS の場合、長さは無制限です。

表 113. Java システム・プロパティ (続き)

プロパティ名	説明	値
com.ibm.wmqfte.daemon.windows.windowsServiceLogFilesm	(Windows のみ。) 保持する Windows サービス・ログ・ファイルの最大数を指定します。エージェントおよびデータベース・ロガーが Windows サービスとして稼働している場合、これらのアプリケーションのログ・ディレクトリーに Windows サービス・ログ・ファイルが作成されます。Windows サービス・ログ・ファイルは、名前に接頭部 <i>service</i> が付いており、サービスの開始および停止に関するメッセージを含んでいます。	5

関連概念

[MFT 構成オプション \(Multiplatforms\)](#)

[MFT の使用のヒント](#)

MFT で使用する SHA-2 の CipherSpec と CipherSuite

Managed File Transfer は、SHA-2 CipherSpec および CipherSuite をサポートします。

エージェントと IBM MQ キュー・マネージャーの間の接続に使用可能な CipherSpecs および CipherSuites について詳しくは、[IBM MQ classes for Java](#) での [TLS CipherSpecs](#) および [CipherSuites](#) および [IBM MQ classes for JMS](#) の [SSL/TLS CipherSpecs](#) および [CipherSuites](#) を参照してください。

プロトコル・ブリッジ・エージェント (PBA) および FTPS サーバーで使用するための CipherSpecs および CipherSuites の構成について詳しくは、[プロトコル・ブリッジによる FTPS サーバー・サポート](#) および [プロトコル・ブリッジ・プロパティ・ファイル・フォーマット](#) を参照してください。

SP 800-131A に準拠する必要がある場合は、以下の要件を満たす必要があります。

- 適切な方法で構成した FTPS を使用する必要があります。SFTP はサポートされていません。
- リモート・サーバーは、SP 800-131A に準拠した暗号スイートのみを送信しなければなりません。

関連概念

[MFT 用の SSL/TLS プロパティ](#)

MFT ファイル・ロガーの構成ファイル

logger.properties ファイルに加えて、Managed File Transfer スタンドアロン・ファイル・ロガーの構成ディレクトリーにも XML 構成ファイルがあります。この構成ファイルは、FileLoggerFormat.xml という名前であり、ファイル・ロガーがログ・ファイルにメッセージを書き込むために使用する形式を定義します。このファイルの内容は、FileLoggerFormat.xsd ファイルで定義されている XML スキーマに準拠する必要があります。

関連概念

[MFT スタンドアロン・ファイル・ロガー・フォーマット](#)

関連資料

204 ページの『[MFT logger.properties ファイル](#)』

Managed File Transfer ロガーには、一連の構成プロパティがあります。これらのプロパティを logger.properties ファイルで指定します。このファイルは、MQ_DATA_PATH/mqft/config/coordination_qmgr_name/loggers/logger_name ディレクトリーにあります。

219 ページの『[MFT スタンドアロン・ファイル・ロガーのデフォルトのログ・フォーマット](#)』

Managed File Transfer スタンドアロン・ファイル・ロガーのデフォルトのログ・ファイル・フォーマット定義です。

223 ページの『[スタンドアロン・ファイル・ロガーのフォーマット XSD](#)』

スタンドアロン・ファイル・フォーマットのスキーマです。

MFT スタンドアロン・ファイル・ロガーのデフォルトのログ・フォーマット

Managed File Transfer スタンドアロン・ファイル・ロガーのデフォルトのログ・ファイル・フォーマット定義です。

```
<?xml version="1.0" encoding="UTF-8"?>
<logFormatDefinition xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  version="1.00" xsi:noNamespaceSchemaLocation="FileLoggerFormat.xsd">
  <messageTypes>
    <callCompleted>
      <format>
        <inserts>
          <insert type="user" width="19" ignoreNull="false"/>/transaction/action/@time</insert>
          <insert type="user" width="48" ignoreNull="false"/>/transaction/@ID</insert>
          <insert type="system" width="6" ignoreNull="false">type</insert>
          <insert type="user" width="3" ignoreNull="false">/transaction/status/@resultCode</insert>
          <insert type="user" width="0" ignoreNull="false">/transaction/agent/@agent</insert>
          <insert type="user" width="0" ignoreNull="false">/transaction/agent/@QMgr</insert>
          <insert type="user" width="0" ignoreNull="false">/transaction/job/name</insert>
          <insert type="user" width="0" ignoreNull="true">/transaction/transferSet/call/command/
@type</insert>
          <insert type="user" width="0" ignoreNull="true">/transaction/transferSet/call/command/
@name</insert>
          <insert type="system" width="0" ignoreNull="true">callArguments</insert>
          <insert type="user" width="0" ignoreNull="true">/transaction/transferSet/call/callResult/
@outcome</insert>
          <insert type="user" width="0" ignoreNull="true">/transaction/transferSet/call/callResult/
result/error</insert>
        </inserts>
        <separator>;</separator>
      </format>
    </callCompleted>
    <callStarted>
      <format>
        <inserts>
          <insert type="user" width="19" ignoreNull="false"/>/transaction/action/@time</insert>
          <insert type="user" width="48" ignoreNull="false"/>/transaction/@ID</insert>
          <insert type="system" width="6" ignoreNull="false">type</insert>
          <insert type="user" width="0" ignoreNull="false">/transaction/agent/@agent</insert>
          <insert type="user" width="0" ignoreNull="false">/transaction/agent/@QMgr</insert>
          <insert type="user" width="0" ignoreNull="false">/transaction/job/name</insert>
          <insert type="user" width="0" ignoreNull="true">/transaction/transferSet/call/command/
@type</insert>
          <insert type="user" width="0" ignoreNull="true">/transaction/transferSet/call/command/
@name</insert>
          <insert type="system" width="0" ignoreNull="true">callArguments</insert>
        </inserts>
        <separator>;</separator>
      </format>
    </callStarted>
    <monitorAction>
      <format>
        <inserts>
          <insert type="user" width="19" ignoreNull="false"/>/monitorLog/action/@time</insert>
          <insert type="user" width="48" ignoreNull="false"/>/monitorLog/@referenceId</insert>
          <insert type="system" width="6" ignoreNull="false">type</insert>
          <insert type="user" width="3" ignoreNull="false">/monitorLog/status/@resultCode</insert>
          <insert type="user" width="0" ignoreNull="false">/monitorLog/@monitorName</insert>
          <insert type="user" width="0" ignoreNull="false">/monitorLog/monitorAgent/@agent</insert>
          <insert type="user" width="0" ignoreNull="false">/monitorLog/monitorAgent/@QMgr</insert>
          <insert type="user" width="0" ignoreNull="false">/monitorLog/action</insert>
        </inserts>
        <separator>;</separator>
      </format>
    </monitorAction>
    <monitorCreate>
      <format>
        <inserts>
          <insert type="user" width="19" ignoreNull="false"/>/monitorLog/action/@time</insert>
          <insert type="user" width="48" ignoreNull="false"/>/monitorLog/@referenceId</insert>
          <insert type="system" width="6" ignoreNull="false">type</insert>
          <insert type="user" width="0" ignoreNull="false">/monitorLog/@monitorName</insert>
          <insert type="user" width="0" ignoreNull="false">/monitorLog/monitorAgent/@agent</insert>
          <insert type="user" width="0" ignoreNull="false">/monitorLog/monitorAgent/@QMgr</insert>
          <insert type="user" width="0" ignoreNull="false">/monitorLog/action</insert>
        </inserts>
        <separator>;</separator>
      </format>
    </monitorCreate>
    <monitorFired>
```

```

<format>
  <inserts>
    <insert type="user" width="19" ignoreNull="false">/monitorLog/action/@time</insert>
    <insert type="user" width="48" ignoreNull="false">/monitorLog/@referenceId</insert>
    <insert type="system" width="6" ignoreNull="false">type</insert>
    <insert type="user" width="3" ignoreNull="false">/monitorLog/status/@resultCode</insert>
    <insert type="user" width="0" ignoreNull="false">/monitorLog/@monitorName</insert>
    <insert type="user" width="0" ignoreNull="false">/monitorLog/monitorAgent/@agent</insert>
    <insert type="user" width="0" ignoreNull="false">/monitorLog/monitorAgent/@QMgr</insert>
    <insert type="user" width="0" ignoreNull="false">/monitorLog/action</insert>
    <insert type="user" width="48" ignoreNull="false">/monitorLog/references/taskRequest</insert>
  </inserts>
  <separator>;</separator>
</format>
</monitorFired>
<notAuthorized>
  <format>
    <inserts>
      <insert type="user" width="19" ignoreNull="false">/notAuthorized/action/@time</insert>
      <insert type="user" width="48" ignoreNull="false">/notAuthorized/@ID</insert>
      <insert type="system" width="6" ignoreNull="false">type</insert>
      <insert type="user" width="3" ignoreNull="false">/notAuthorized/status/@resultCode</insert>
      <insert type="user" width="12" ignoreNull="false">/notAuthorized/authority</insert>
      <insert type="user" width="0" ignoreNull="false">/notAuthorized/originator/userID</insert>
      <insert type="user" width="0" ignoreNull="false">/notAuthorized/status/supplement</insert>
    </inserts>
    <separator>;</separator>
  </format>
</notAuthorized>
<scheduleDelete>
  <format>
    <inserts>
      <insert type="user" width="19" ignoreNull="false">/schedulelog/action/@time</insert>
      <insert type="user" width="48" ignoreNull="false">/schedulelog/@ID</insert>
      <insert type="system" width="6" ignoreNull="false">type</insert>
      <insert type="user" width="3" ignoreNull="false">/schedulelog/status/@resultCode</insert>
      <insert type="user" width="0" ignoreNull="false">/schedulelog/sourceAgent/@agent</insert>
      <insert type="user" width="12" ignoreNull="false">/schedulelog/action</insert>
      <insert type="user" width="0" ignoreNull="false">/schedulelog/originator/userID</insert>
      <insert type="user" width="0" ignoreNull="true">/schedulelog/status/supplement</insert>
    </inserts>
    <separator>;</separator>
  </format>
</scheduleDelete>
<scheduleExpire>
  <format>
    <inserts>
      <insert type="user" width="19" ignoreNull="false">/schedulelog/action/@time</insert>
      <insert type="user" width="48" ignoreNull="false">/schedulelog/@ID</insert>
      <insert type="system" width="6" ignoreNull="false">type</insert>
      <insert type="user" width="3" ignoreNull="false">/schedulelog/status/@resultCode</insert>
      <insert type="user" width="0" ignoreNull="false">/schedulelog/sourceAgent/@agent</insert>
      <insert type="user" width="12" ignoreNull="false">/schedulelog/action</insert>
      <insert type="user" width="0" ignoreNull="false">/schedulelog/originator/userID</insert>
      <insert type="user" width="0" ignoreNull="true">/schedulelog/status/supplement</insert>
    </inserts>
    <separator>;</separator>
  </format>
</scheduleExpire>
<scheduleSkipped>
  <format>
    <inserts>
      <insert type="user" width="19" ignoreNull="false">/schedulelog/action/@time</insert>
      <insert type="user" width="48" ignoreNull="false">/schedulelog/@ID</insert>
      <insert type="system" width="6" ignoreNull="false">type</insert>
      <insert type="user" width="3" ignoreNull="false">/schedulelog/status/@resultCode</insert>
      <insert type="user" width="0" ignoreNull="false">/schedulelog/sourceAgent/@agent</insert>
      <insert type="user" width="12" ignoreNull="false">/schedulelog/action</insert>
      <insert type="user" width="0" ignoreNull="false">/schedulelog/originator/userID</insert>
      <insert type="user" width="0" ignoreNull="true">/schedulelog/status/supplement</insert>
    </inserts>
    <separator>;</separator>
  </format>
</scheduleSkipped>
<scheduleSubmitInfo>
  <format>
    <inserts>
      <insert type="user" width="19" ignoreNull="false">/schedulelog/action/@time</insert>
      <insert type="user" width="48" ignoreNull="false">/schedulelog/@ID</insert>
      <insert type="system" width="6" ignoreNull="false">type</insert>
      <insert type="user" width="3" ignoreNull="false">/schedulelog/status/@resultCode</insert>
    </inserts>
  </format>
</scheduleSubmitInfo>

```

```

        <insert type="user" width="0" ignoreNull="false">/schedulelog/sourceAgent/@agent</insert>
        <insert type="user" width="12" ignoreNull="false">/schedulelog/action</insert>
        <insert type="user" width="0" ignoreNull="false">/schedulelog/originator/userID</insert>
        <insert type="user" width="0" ignoreNull="true">/schedulelog/schedule/submit</insert>
        <insert type="user" width="0" ignoreNull="true">/schedulelog/schedule/submit/@timezone</
insert>
        <insert type="user" width="3" ignoreNull="true">/schedulelog/schedule/repeat/frequency</
insert>
        <insert type="user" width="12" ignoreNull="true">/schedulelog/schedule/repeat/frequency/
@interval</insert>
        <insert type="user" width="3" ignoreNull="true">/schedulelog/schedule/repeat/expireCount</
insert>
        <insert type="user" width="0" ignoreNull="true">/schedulelog/status/supplement</insert>
    </inserts>
    <separator>;</separator>
</format>
</scheduleSubmitInfo>
<scheduleSubmitTransfer>
    <format>
        <inserts>
            <insert type="user" width="19" ignoreNull="false">/schedulelog/action/@time</insert>
            <insert type="user" width="48" ignoreNull="false">/schedulelog/@ID</insert>
            <insert type="system" width="10" ignoreNull="false">type</insert>
            <insert type="user" width="0" ignoreNull="false">/transaction/sourceAgent/@agent |
            /transaction/sourceWebUser/@webGatewayAgentName |
            /transaction/sourceWebGateway/@webGatewayAgentName</insert>
            <insert type="user" width="0" ignoreNull="false">/transaction/sourceAgent/@QMgr |
            /transaction/sourceWebUser/@webGatewayAgentQMgr |
            /transaction/sourceWebGateway/@webGatewayAgentQMgr</insert>
            <insert type="user" width="0" ignoreNull="false">/transaction/destinationAgent/@agent |
            /transaction/destinationWebUser/@webGatewayAgentName |
            /transaction/destinationWebGateway/@webGatewayAgentName</insert>
            <insert type="user" width="0" ignoreNull="false">/transaction/destinationAgent/@QMgr |
            /transaction/destinationWebUser/@webGatewayAgentQMgr |
            /transaction/destinationWebGateway/@webGatewayAgentQMgr</insert>
        </inserts>
        <separator>;</separator>
    </format>
</scheduleSubmitTransfer>
<scheduleSubmitTransferSet>
    <format>
        <inserts>
            <insert type="user" width="19" ignoreNull="false">/schedulelog/action/@time</insert>
            <insert type="user" width="48" ignoreNull="false">/schedulelog/@ID</insert>
            <insert type="system" width="10" ignoreNull="false">type</insert>
            <insert type="user" width="0" ignoreNull="false">source/file | source/queue</insert>
            <insert type="user" width="5" ignoreNull="true">source/@type</insert>
            <insert type="user" width="6" ignoreNull="true">source/@disposition</insert>
            <insert type="user" width="0" ignoreNull="false">destination/file | destination/queue</
insert>
            <insert type="user" width="5" ignoreNull="true">destination/@type</insert>
            <insert type="user" width="9" ignoreNull="true">destination/@exist</insert>
        </inserts>
        <separator>;</separator>
    </format>
</scheduleSubmitTransferSet>
<transferStarted>
    <format>
        <inserts>
            <insert type="user" width="19" ignoreNull="false">/transaction/action/@time</insert>
            <insert type="user" width="48" ignoreNull="false">/transaction/@ID</insert>
            <insert type="system" width="6" ignoreNull="false">type</insert>
            <insert type="user" width="3" ignoreNull="true">/transaction/status/@resultCode</insert>
            <insert type="user" width="0" ignoreNull="false">/transaction/sourceAgent/@agent |
            /transaction/sourceWebUser/@webGatewayAgentName |
            /transaction/sourceWebGateway/@webGatewayAgentName</insert>
            <insert type="user" width="0" ignoreNull="true">/transaction/sourceAgent/@QMgr |
            /transaction/sourceWebUser/@webGatewayAgentQMgr |
            /transaction/sourceWebGateway/@webGatewayAgentQMgr</insert>
            <insert type="user" width="0" ignoreNull="true">/transaction/sourceAgent/@agentType |
            /transaction/sourceWebUser/@webGatewayAgentType |
            /transaction/sourceWebGateway/@webGatewayAgentType</insert>
            <insert type="user" width="0" ignoreNull="false">/transaction/destinationAgent/@agent |
            /transaction/destinationWebUser/@webGatewayAgentName |
            /transaction/destinationWebGateway/@webGatewayAgentName</insert>
            <insert type="user" width="0" ignoreNull="true">/transaction/destinationAgent/@QMgr |
            /transaction/destinationWebUser/@webGatewayAgentQMgr |
            /transaction/destinationWebGateway/@webGatewayAgentQMgr</insert>
            <insert type="user" width="0" ignoreNull="true">/transaction/originator/userID</insert>
            <insert type="user" width="0" ignoreNull="true">/transaction/job/name</insert>
            <insert type="user" width="0" ignoreNull="true">/transaction/scheduleLog/@ID</insert>
        </inserts>

```



```

/transaction/sourceWebGateway/@webGatewayAgentType</insert>
<insert type="user" width="0" ignoreNull="false">/transaction/destinationAgent/@agent |
/transaction/destinationWebUser/@webGatewayAgentName |
/transaction/destinationWebGateway/@webGatewayAgentName</insert>
<insert type="user" width="0" ignoreNull="true">/transaction/destinationAgent/@QMgr |
/transaction/destinationWebUser/@webGatewayAgentQMgr |
/transaction/destinationWebGateway/@webGatewayAgentQMgr</insert>
<insert type="user" width="0" ignoreNull="true">/transaction/destinationAgent/@agentType |
/transaction/destinationWebUser/@webGatewayAgentType |
/transaction/destinationWebGateway/@webGatewayAgentType</insert>
<insert type="user" width="0" ignoreNull="true">/transaction/originator/userID</insert>
<insert type="user" width="0" ignoreNull="true">/transaction/job/name</insert>
<insert type="user" width="0" ignoreNull="true">/transaction/status/supplement</insert>
</inserts>
<separator>;</separator>
</format>
</transferDelete>
<transferProgress>
<format>
<inserts>
<insert type="user" width="19" ignoreNull="false">/transaction/action/@time</insert>
<insert type="user" width="48" ignoreNull="false">/transaction/@ID</insert>
<insert type="system" width="6" ignoreNull="false">type</insert>
<insert type="user" width="3" ignoreNull="true">status/@resultCode</insert>
<insert type="user" width="0" ignoreNull="false">source/file | source/queue</insert>
<insert type="user" width="0" ignoreNull="false">source/file/@size | source/queue/@size</
insert>
<insert type="user" width="5" ignoreNull="true">source/@type</insert>
<insert type="user" width="6" ignoreNull="true">source/@disposition</insert>
<insert type="user" width="0" ignoreNull="true">source/file/@alias | source/queue/@alias</
insert>
<insert type="user" width="0" ignoreNull="true">source/file/@filesystem | source/queue/
@filesystem</insert>
<insert type="user" width="0" ignoreNull="true">source/@correlationBoolean1</insert>
<insert type="user" width="0" ignoreNull="true">source/@correlationNum1</insert>
<insert type="user" width="0" ignoreNull="true">source/@correlationString1</insert>
<insert type="user" width="0" ignoreNull="false">destination/file | destination/queue</
insert>
<insert type="user" width="0" ignoreNull="false">destination/file/@size | destination/queue/
@size</insert>
<insert type="user" width="5" ignoreNull="true">destination/@type</insert>
<insert type="user" width="9" ignoreNull="true">destination/@exist</insert>
<insert type="user" width="0" ignoreNull="true">destination/file/@alias | destination/queue/
@alias</insert>
<insert type="user" width="0" ignoreNull="true">destination/file/@filesystem | destination/
queue/@filesystem</insert>
<insert type="user" width="0" ignoreNull="true">destination/file/@truncateRecords</insert>
<insert type="user" width="0" ignoreNull="true">destination/@correlationBoolean1</insert>
<insert type="user" width="0" ignoreNull="true">destination/@correlationNum1</insert>
<insert type="user" width="0" ignoreNull="true">destination/@correlationString1</insert>
<insert type="user" width="0" ignoreNull="true">status/supplement</insert>
</inserts>
<separator>;</separator>
</format>
</transferProgress>
</messageTypes>
</logFormatDefinition>

```

関連資料

[MFT スタンドアロン・ファイル・ロガー・フォーマット](#)

223 ページの『[スタンドアロン・ファイル・ロガーのフォーマット XSD](#)』

スタンドアロン・ファイル・フォーマットのスキーマです。

スタンドアロン・ファイル・ロガーのフォーマット XSD

スタンドアロン・ファイル・フォーマットのスキーマです。

スキーマ

```

<?xml version="1.0" encoding="UTF-8"?>
<!--
@start_non_restricted_prolog@
Version: %Z% %I% %W% %E% %U% [%H% %T%]

Licensed Materials - Property of IBM

```

Copyright IBM Corp. 2011, 2024. All Rights Reserved.

US Government Users Restricted Rights - Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

@end_non_restricted_prolog@

-->

<!--

This schema defines the format of the FileLoggerFormat XML file that contains the definition of the format to use when logging FTE log messages to a file. When an XML file that conforms to this schema is processed by a file logger it can contain definitions for one or more message type(s) that define how log messages of those types are output to the file log.

-->

<xsd:schema xmlns:xsd="https://www.w3.org/2001/XMLSchema">

<xsd:include schemaLocation="fteutils.xsd"/>

<!--

Defines the logFileDefinition and version number
<logFileDefinition version="1.00" ...
 <messageTypes>

 ...

 </messageTypes>

</logFileDefinition>

-->

<xsd:element name="logFileDefinition">

 <xsd:complexType>

 <xsd:sequence>

 <xsd:element name="messageTypes" type="messageTypesType" maxOccurs="1" minOccurs="1"/>

 </xsd:sequence>

 <xsd:attribute name="version" type="versionType" use="required"/>

 </xsd:complexType>

</xsd:element>

<!--

Defines the set of accepted message types. The definition of individual message types is optional. If a particular types element is present but empty then no line will be output for messages of that type. If a particular types element is not present then the default format will be used to format messages of that type.

-->

<xsd:complexType name="messageTypesType">

 <xsd:sequence>

 <xsd:element name="callCompleted" type="messageType" maxOccurs="1" minOccurs="0"/>

 <xsd:element name="callStarted" type="messageType" maxOccurs="1" minOccurs="0"/>

 <xsd:element name="monitorAction" type="messageType" maxOccurs="1" minOccurs="0"/>

 <xsd:element name="monitorCreate" type="messageType" maxOccurs="1" minOccurs="0"/>

 <xsd:element name="monitorFired" type="messageType" maxOccurs="1" minOccurs="0"/>

 <xsd:element name="notAuthorized" type="messageType" maxOccurs="1" minOccurs="0"/>

 <xsd:element name="scheduleDelete" type="messageType" maxOccurs="1" minOccurs="0"/>

 <xsd:element name="scheduleExpire" type="messageType" maxOccurs="1" minOccurs="0"/>

 <xsd:element name="scheduleSkipped" type="messageType" maxOccurs="1" minOccurs="0"/>

 <xsd:element name="scheduleSubmitInfo" type="messageType" maxOccurs="1" minOccurs="0"/>

 <xsd:element name="scheduleSubmitTransfer" type="messageType" maxOccurs="1" minOccurs="0"/>

 <xsd:element name="scheduleSubmitTransferSet" type="messageType" maxOccurs="1" minOccurs="0"/>

 <xsd:element name="transferStarted" type="messageType" maxOccurs="1" minOccurs="0"/>

 <xsd:element name="transferCancelled" type="messageType" maxOccurs="1" minOccurs="0"/>

 <xsd:element name="transferComplete" type="messageType" maxOccurs="1" minOccurs="0"/>

 <xsd:element name="transferDelete" type="messageType" maxOccurs="1" minOccurs="0"/>

 <xsd:element name="transferProgress" type="messageType" maxOccurs="1" minOccurs="0"/>

 </xsd:sequence>

</xsd:complexType>


```

<!--
  Defines the content of a message type definition e.g.

  <callStarted>
  <format>
  ...
  </format>
  <callStarted>
-->
<xsd:complexType name="messageType">
  <xsd:sequence>
    <xsd:element name="format" type="messageFormatType" maxOccurs="1" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<!--
  Defines the content of a message format definition e.g.

  <format>
  <inserts>
  ...
  </inserts>
  <separator>;</separator>
  </format>
-->
<xsd:complexType name="messageFormatType">
  <xsd:sequence>
    <xsd:element name="inserts" type="insertsType" maxOccurs="1" minOccurs="1"/>
    <xsd:element name="separator" type="scheduleType" maxOccurs="1" minOccurs="1"/>
  </xsd:sequence>
</xsd:complexType>

<!--
  Defines the content of the inserts element e.g.

  <inserts>
  <insert ...>
  <insert ...>
  ...
  </inserts>
-->
<xsd:complexType name="insertsType">
  <xsd:sequence>
    <xsd:element name="insert" type="insertType" maxOccurs="unbounded" minOccurs="1"/>
  </xsd:sequence>
</xsd:complexType>

<!--
  Defines the content of an insert definition e.g.

  <insert type="user" width="0" ignoreNull="true">/transaction/@ID</insert>
-->
<xsd:complexType name="insertType">
  <xsd:attribute name="type" type="insertTypeType" use="required"/>
  <xsd:attribute name="width" type="xsd:nonNegativeInteger" use="required"/>
  <xsd:attribute name="ignoreNull" type="xsd:boolean" use="required"/>
</xsd:complexType>

<!--
  Defines the accepted choices for the insert type attribute.
-->
<xsd:simpleType name="insertTypeType">
  <xsd:restriction base="xsd:token">
    <xsd:enumeration value="user"/>
    <xsd:enumeration value="system"/>
  </xsd:restriction>
</xsd:simpleType>
</xsd:schema>

```

関連資料

[MFT スタンドアロン・ファイル・ロガー・フォーマット](#)

219 ページの『[MFT スタンドアロン・ファイル・ロガーのデフォルトのログ・フォーマット](#)』

Managed File Transfer スタンドアロン・ファイル・ロガーのデフォルトのログ・ファイル・フォーマット定義です。

IBM MQ Managed File Transfer for z/OS 提供の SCSQFCMD ライブラリーには、Managed File Transfer 構成の作成、およびエージェントまたはロガーの作成と管理に使用できるジョブのテンプレートとして機能するメンバーが含まれています。

このライブラリーの内容を以下の表に示します。

メンバー	説明
BFGCOPY	SCSQFCMD ライブラリーのコピーを作成するために使用されるジョブ
BFGCUSTM	エージェントまたはロガー用にライブラリーのコピーをカスタマイズするときに使用するジョブ
BFGXCROB	fteObfuscate サンプル・テンプレート。
BFGXLGCR	fteCreateLogger テンプレート。
BFGXMNCR	fteCreateMonitor サンプル・テンプレート。
BFGXMNDE	fteDeleteMonitor サンプル・テンプレート。
BFGXPRAN	fteAnt サンプル・テンプレート
BFGXSTDE	fteDeleteScheduledTransfer サンプル・テンプレート
BFGXTMCR	fteCreateTemplate サンプル・テンプレート
BFGXTMDE	fteDeleteTemplate サンプル・テンプレート
BFGXTRCA	fteCancelTransfer サンプル・テンプレート
BFGXTRCR	fteCreateTransfer サンプル・テンプレート
BFGYAGST	エージェントを開始する開始タスク・プロシージャ用テンプレート
BFGYLGST	ロガーを開始する開始タスク・プロシージャ用テンプレート
BFGZAGCL	fteCleanAgent サンプル・テンプレート
BFGZAGCR	fteCreateAgent サンプル・テンプレート
BFGZAGDE	fteDeleteAgent サンプル・テンプレート
BFGZAGLG	fteSetAgentLogLevel サンプル・テンプレート
BFGZAGLI	fteListAgents サンプル・テンプレート
BFGZAGPI	ftePingAgent サンプル・テンプレート
BFGZAGSH	fteShowAgentDetails サンプル・テンプレート
BFGZAGSP	fteStopAgent サンプル・テンプレート
BFGZAGST	fteStartAgent サンプル・テンプレート
BFGZAGTC	fteSetAgentTraceLevel サンプル・テンプレート
BFGZCFCR	fteSetupCoordination サンプル・テンプレート
BFGZCFDF	fteChangeDefaultConfigurationOptions サンプル・テンプレート
BFGZCMCR	fteSetupCommands サンプル・テンプレート
BFGZCMD	データ・セット内の他のメンバーによって使用される REXX スクリプトのテンプレート

メンバー	説明
BFGZLGDE	fteDeleteLogger サンプル・テンプレート
BFGZLGSH	fteShowLoggerDetails サンプル・テンプレート
BFGZLGSP	fteStopLogger サンプル・テンプレート
BFGZLGST	fteStartLogger サンプル・テンプレート
BFGZLGTC	fteSetLoggerTraceLevel サンプル・テンプレート
BFGZMNL	fteListMonitors サンプル・テンプレート
BFGZPID	fteSetProductId サンプル・テンプレート
BFGZPROF	データ・セット内の他のメンバーによって使用されるシェル・スクリプトのテンプレート
BFGZPRSH	fteDisplayVersion サンプル・テンプレート
BFGZRAS	fteRas サンプル・テンプレート
BFGZSTLI	fteListScheduledTransfers サンプル・テンプレート
BFGZTMLI	fteListTemplates サンプル・テンプレート

SCSQFCMD ライブラリーを使用して Managed File Transfer 構成を作成し、エージェントまたはロガーを作成および管理するための新規ライブラリーを生成する方法については、[Managed File Transfer for z/OS の構成](#)を参照してください。

関連資料

167 ページの『MFT プロパティーでの環境変数の使用』

ファイルまたはディレクトリーの場所を表す Managed File Transfer プロパティーで環境変数を使用することができます。これにより、製品の一部を実行するとき使用されるファイルまたはディレクトリーの場所を、現在の環境 (例えば、コマンドを実行するユーザー) に応じて変更することができます。

SYSTEM.FTE トピック

SYSTEM.FTE トピックは、Managed File Transfer が転送をログに記録し、エージェント、モニター、スケジュール、およびテンプレートに関する情報を保管するために使用する調整キュー・マネージャー上のトピックです。

トピックの構造

```
SYSTEM.FTE
  /Agents
    /agent_name
  /monitors
    /agent_name
  /Scheduler
    /agent_name
  /Templates
    /template_ID
  /Transfers
    /agent_name
    /transfer_ID
  /Log
    /agent_name
    /Monitors
    /schedule_ID
    /transfer_ID
```

SYSTEM.FTE/Agents/agent_name

このトピックには Managed File Transfer ネットワークのエージェント、およびそのプロパティーを記述した保存パブリケーションが含まれます。このトピックに関するメッセージは、エージェント状況

によって定期的に更新されます。詳しくは、[MFT エージェント状況メッセージ・フォーマット](#)を参照してください。

SYSTEM.FTE/monitors/agent_name

このトピックにはエージェント *agent_name* と関連付けられたリソース・モニターを記述した保存パブリケーションが含まれます。保存パブリケーションのXMLは、スキーマ *MonitorList.xsd* に準拠します。詳しくは、[MFT モニター・リスト・メッセージ・フォーマット](#)を参照してください。

SYSTEM.FTE/Scheduler/agent_name

このトピックには、エージェント *agent_name* と関連付けられたすべてのアクティブ・スケジュールを記述する保存パブリケーションが含まれます。保存パブリケーションのXMLは、スキーマ *ScheduleList.xsd* に準拠します。詳しくは、[MFT スケジュール・リスト・メッセージ・フォーマット](#)を参照してください。

SYSTEM.FTE/Templates

このトピックには Managed File Transfer トポロジーに定義されたすべてのテンプレートを記述した保存パブリケーションが含まれます。

- 各テンプレートと関連付けられたパブリケーションは、*SYSTEM.FTE/Templates/template_ID* という名前でサブトピックにパブリッシュされます。

この保存パブリケーションの内容例については、[MFT テンプレート XML メッセージの例](#)を参照してください。

SYSTEM.FTE/Transfers/agent_name

このトピックにはエージェント *agent_name* で発生した転送の状況を記述したパブリケーションが含まれます。各転送と関連付けられたパブリケーションは、*SYSTEM.FTE/Transfers/agent_name/transfer_ID* という名前でサブトピックにパブリッシュされます。これらのパブリケーションは、個々の転送に関する進捗情報を提供するために、IBM MQ Explorer ・ プラグインによって使用されます。パブリケーションのXMLはスキーマ *TransferStatus.xsd* に準拠します。詳しくは、[ファイル転送状況メッセージ・フォーマット](#)を参照してください。

SYSTEM.FTE/Log/agent_name

このトピックには、エージェント *agent_name* で発生した転送、モニター、およびスケジュールに関する情報を記録したパブリケーションが含まれます。これらのパブリケーションは、使用中の Managed File Transfer ネットワークで発生したイベントの監査レコードを提供するために、データベース・ロガーによって記録することができます。

- 各転送と関連付けられたパブリケーションは、*SYSTEM.FTE/Log/agent_name/transfer_ID* という名前でサブトピックにパブリッシュされ、パブリケーションのXMLはスキーマ *TransferLog.xsd* に準拠します。詳しくは、[ファイル転送ログ・メッセージ・フォーマット](#)を参照してください。
- 各スケジュール済み転送と関連付けられたパブリケーションは、*SYSTEM.FTE/Log/agent_name/schedule_ID* という名前でサブトピックにパブリッシュされ、パブリケーションのXMLはスキーマ *ScheduleLog.xsd* に準拠します。詳しくは、[スケジュール済みファイル転送ログ・メッセージ・フォーマット](#)を参照してください。
- 各モニターと関連付けられたパブリケーションは、*SYSTEM.FTE/Log/agent_name/Monitors/monitor_name/monitor_ID* という名前でサブトピックにパブリッシュされ、パブリケーションのXMLはスキーマ *MonitorLog.xsd* に準拠します。詳しくは、[MFT モニター・ログ・メッセージ・フォーマット](#)を参照してください。

MFT エージェント・キュー設定

fteCreateAgent コマンドで生成される MQSC コマンド・スクリプトは、次の値に設定されたパラメーターでエージェント・キューを作成します。キューを作成するために指定された MQSC スクリプトを使用せず、キューを手動で作成する場合、指定された値に次のパラメーターを設定する必要があります。

エージェント操作キュー

エージェントの操作キューの名前は、以下のとおりです。

- SYSTEM.FTE.COMMAND.agent_name*

- SYSTEM.FTE.DATA.agent_name
- SYSTEM.FTE.EVENT.agent_name
- SYSTEM.FTE.REPLY.agent_name
- SYSTEM.FTE.STATE.agent_name

表 114. エージェント操作キューのパラメーター	
パラメーター	値 (該当する場合)
DEFPRTY	0
DEFSOPT	SHARED
GET	ENABLED
MAXDEPTH	5000
MAXMSGL	4194304
MSGDLVSQ	PRIORITY
PUT	ENABLED
RETINTVL	999999999
SHARE	
NOTRIGGER	
USAGE	NORMAL
REPLACE	

エージェント権限キュー

エージェントの権限キューの名前は、以下のとおりです。

- SYSTEM.FTE.AUTHADM1.agent_name
- SYSTEM.FTE.AUTHAGT1.agent_name
- SYSTEM.FTE.AUTHMON1.agent_name
- SYSTEM.FTE.AUTHOPS1.agent_name
- SYSTEM.FTE.AUTHSCH1.agent_name
- SYSTEM.FTE.AUTHTRN1.agent_name

表 115. エージェント権限キューのパラメーター	
パラメーター	値 (該当する場合)
DEFPRTY	0
DEFSOPT	SHARED
GET	ENABLED
MAXDEPTH	0
MAXMSGL	0
MSGDLVSQ	PRIORITY
PUT	ENABLED
RETINTVL	999999999
SHARE	

表 115. エージェント権限キューのパラメーター (続き)	
パラメーター	値 (該当する場合)
NOTRIGGER	
USAGE	NORMAL
REPLACE	

関連資料

[fteCreateAgent \(MFT エージェントの作成\)](#)

MFT システム・キューおよびシステム・トピック

Managed File Transfer には、内部使用専用のいくつかのシステム・キューおよび 1 つのシステム・トピックがあります。

名前が SYSTEM.FTE で始まるキューは、Managed File Transfer (MFT) の内部システム・キューです。これらのキューは削除しないでください。削除すると、IBM MQ MFT が正しく動作しなくなります。230 ページの表 116 は、各キューにあるメッセージのタイプを示しています。

表 116. キュー名、タイプ、および使用法		
キュー名	キュー・タイプ	使用法
SYSTEM.FTE.AUTHAGT1.agent_name	権限	転送要求を送受信する権限を構成するためのキュー。
SYSTEM.FTE.AUTHTRN1.agent_name	権限	管理対象転送を開始およびキャンセルする権限を構成するためのキュー。また、管理対象呼び出しを開始します。
SYSTEM.FTE.AUTHMON1.agent_name	権限	同じユーザーによって作成されたリソース・モニターをユーザーが作成または削除できるようにする権限を構成するためのキュー。
SYSTEM.FTE.AUTHOPS1.agent_name	権限	別のユーザーによって作成されたリソース・モニターおよびスケジュールされた転送を削除する権限を構成するためのキュー。
SYSTEM.FTE.AUTHSCH1.agent_name	権限	同じユーザーによって作成されたスケジュール済み転送を作成または削除する権限を構成するためのキュー。
SYSTEM.FTE.AUTHADM1.agent_name	権限	fteStopAgent コマンドの -m オプションを使用してエージェントをシャットダウンする権限を構成するためのキュー。
SYSTEM.FTE.COMMAND.agent_name	Operation	コマンド要求をエージェントに送信するためのキュー。
SYSTEM.FTE.DATA.agent_name	Operation	ソース・エージェントによって送信されたデータを保持するために宛先エージェントによって使用されるキュー。

表 116. キュー名、タイプ、および使用法 (続き)

キュー名	キュー・タイプ	使用法
SYSTEM.FTE.REPLY.agent_name	Operation	宛先エージェントから応答を受信するためのキュー。
SYSTEM.FTE.STATE.agent_name	Operation	転送要求の状況を保持するためのキュー。
SYSTEM.FTE.EVENT.agent_name	Operation	リソース・モニター・ヒストリーを保持するためのキュー。
SYSTEM.FTE.HA.agent_name	Operation	高可用性エージェント・インスタンスによってロックとして使用されるキュー。

エージェントがメッセージからファイルへの転送またはファイルからメッセージへの転送に関与している場合は、SYSTEM.FTE.STATE.agent_name キューの定義を変更して、これらの管理対象転送を実行できるようにする必要がある場合があります。この設定について詳しくは、[メッセージ・サイズに関連した MQ 属性および MFT プロパティを設定するためのガイダンス](#)を参照してください。



重要: 他のシステム・キューの定義を変更してはなりません。

また、SYSTEM.FTE トピックは内部使用専用であるため、変更したり削除したりしないでください。

一時キュー

Managed File Transfer は、いくつかの目的で一時キューを作成します。各キューの名前は WMQFTE で始まります。このグループにデフォルトで追加されます (ピリオドはデフォルトの接頭部の一部です。) この接頭部を変更する場合は、command.properties ファイルまたは coordination.properties ファイル、あるいはその両方の **dynamicQueuePrefix** プロパティを使用します。command.properties ファイルのプロパティは、エージェントからの応答が必要なコマンドに対する応答のために作成される一時キューの接頭部を設定するために使用されます。coordination.properties ファイルのプロパティは、その他の目的のために作成される一時キューの接頭部を設定するために使用されます。例えば、WMQFTE.FTE.TIMECHCK.QUEUE という一時キューの場合、WMQFTE が **dynamicQueuePrefix** プロパティによって定義された値です。

関連資料

[MFT エージェント・アクションのユーザー権限の制限](#)

MFT オブジェクト命名規則

Managed File Transfer オブジェクトに以下の命名規則を使用してください。

- エージェントおよびログナーの名前:
 - 最大 28 文字の長さにするのができ、大/小文字の区別はありません。
 - 小文字または大/小文字混合で入力された名前は、大文字に変換されます。
 - 標準 IBM MQ オブジェクト命名規則に準拠していなければなりません。

これらの規則の詳細については、[IBM MQ オブジェクトの命名規則](#)を参照してください。
- IBM MQ オブジェクト命名規則に加えて、以下の規則が適用されます。
 - スラッシュ (/) 文字をエージェント名またはログナー名に使用することはできません。
 - パーセント (%) 文字をエージェント名またはログナー名に使用することはできません。
- プロパティ・ファイルのプロパティ名は大/小文字を区別します。
- キュー・マネージャー名は、大/小文字を区別します。
- 一部のプラットフォームでは、ファイル名で大/小文字を区別します。
- リソース・モニターおよび転送テンプレートの名前:

- 大/小文字の区別はありません。
 - 小文字または大/小文字混合で入力された名前は、大文字に変換されます。
 - アスタリスク (*)、パーセント (%)、および疑問符 (?) を含めることはできません。
 - プロトコル・ファイル・サーバー名:
 - 2 文字以上の長さでなければならず、最大長の制限はありません。
 - 大/小文字の区別はありません。
 - 標準 IBM MQ オブジェクト命名規則に準拠していなければなりません。
- これらの規則の詳細については、[IBM MQ オブジェクトの命名規則](#)を参照してください。

IBM i 統合ファイル・システム (IFS) のファイル

IBM i IFS のファイル名には次の文字を使用できません。

- 円記号 (¥)
- スラッシュ (/)
- コロン (:)
- アスタリスク (*)
- 疑問符 (?)
- 引用符 (")
- より小記号 (<)
- より大記号 (>)
- 垂直バー (|)

名前にこれらの文字のいずれかを含むファイルを IBM i IFS に転送しようとする、これらのファイルの転送は失敗します。

データ・セット名

z/OS データ・セットには、データ・セット名で名前の最大長と使用できる文字に影響する命名上の制約があります。PDS データ・セット・メンバー名は、最大 8 文字で、ドット (.) 文字を使用することはできません。データ・セットに転送する場合、明示的に名前を指定する必要があります。つまり、これらの命名上の制約によって問題が発生しないことを意味します。ただしファイルから PDS メンバーに転送する場合には、ファイル・パスが PDS メンバー名にマップされない可能性があります。PDS データ・セットに転送する場合、個々のソース・ファイルが PDS メンバーとなり、各メンバー名がソースの名前から生成されます。

PDS メンバー名は z/OS の非修飾名で、以下の正規表現で定義されます。

```
[a-zA-Z$#@][a-zA-Z0-9$#@]{0-7}
```

次の方式が、ソース・データ・セットまたはソース・ファイル名を有効な PDS メンバー名に変換するために使用されます。以下の順に考慮事項が適用されます。

1. 名前のうち、最後のスラッシュ (/)、最後のバックスラッシュ (¥)、最後のコロン (:) 以降の文字のみが使用されます。つまり、ファイル・パスの名前部分のみが使用されます。
2. ソース・ファイル (データ・セット、PDS メンバー以外) の場合、最後のドット (.) 文字以降 (ドットを含む) が無視されます。
3. 8 文字より長い名前については、最後の 8 文字のみが使用されます。
4. ドット文字はアットマーク (@) 文字に置き換えられます。
5. 無効文字はアットマーク (@) 文字に置き換えられます。
6. この変換により文字が生成されない場合、PDS メンバー名は @ となります。

MFT エージェントの状況メッセージ

高可用性エージェントは、状況情報を XML 形式でパブリッシュします。

3つのスタンバイ・インスタンスに関する情報を示すサンプル XML

```
<?xml version="1.0" encoding="UTF-8"?>
<AgentStandbyStatus version="6.00" xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="AgentStandbyStatus.xsd">
  <instance host="9.122.123.124" agentVersion="9.1.4.0" />
  <instance host="agenthost.ibm.com" agentVersion="9.1.4.0" />
  <instance host="10.11.12.14" agentVersion="9.1.4.0" />
</AgentStandby>
```

スタンバイ状況 XML が埋め込まれたエージェント状況のパブリケーション。

スタンバイ状況 XML は、太字で示されています。

```
<?xml version="1.0" encoding="UTF-8"?>
<properties version="1.0">
  <entry key="SourceTransferStates"/>
  <entry key="queueManagerPort">1414</entry>
  <entry key="agentStandbyInstances">&lt;?xml version="1.0" encoding="UTF-8"?&gt;&lt;AgentStandbyStatus
version="6.00"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="AgentStandbyStatus.xsd"&gt;&lt;Instances&gt;&lt;instance
host="9.122.123.124"
  agentVersion="9.1.4.0" /&gt;&lt;instance host="agenthost.ibm.com" agentVersion="9.1.4.0" /
&gt;&lt;instance host="10.11.12.14"
  agentVersion="9.1.4.0" /&gt;&lt;/Instances&gt;&lt;/AgentStandbyStatus&gt;</entry>
  <entry key="agentType">STANDARD</entry>
  <entry key="agentDeclaredHostName">MFTHA1</entry>
  <entry key="agentDescription"/>
  <entry key="maxQueuedTransfers">1000</entry>
  <entry key="agentTimeZone">America/Los_Angeles</entry>
  <entry key="agentOsName">Windows Server 2012 R2</entry>
  <entry key="PublishTimeUTC">2019-05-22T06:02:50Z</entry>
  <entry key="queueManagerHost">localhost</entry>
  <entry key="AgentStartTimeUTC">2019-05-22T04:13:02Z</entry>
  <entry key="agentTraceLevel">&lt;?xml version="1.0" encoding="UTF-8"?&gt;&lt;
agentTraceStatus version="6.00" xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="AgentTraceStatus.xsd"&gt;&lt;trace
level="all"&gt;&lt;com.ibm.wmqfte&lt;/trace&gt;&lt;/agentTraceStatus&gt;</entry>
  <entry key="DestinationTransferStates"/>
  <entry key="queueManager">MFTHAQM</entry>
  <entry key="agentProductVersion">9.1.4.0</entry>
  <entry key="AgentStatusPublishRate">300</entry>
  <entry key="maxSourceTransfers">25</entry>
  <entry key="AgentStatus">STARTED</entry>
  <entry key="maxDestinationTransfers">25</entry>
  <entry key="agentName">SRC</entry>
  <entry key="CommandTimeUTC">2019-05-22T06:02:50Z</entry>
  <entry key="queueManagerChannel">MFT_HA_CHN</entry>
  <entry key="agentInterfaceVersion">6.00</entry>
  <entry key="agentVersion">p914-L191119</entry>
</properties>
```

関連資料

[fteCreateAgent](#)

[エージェント GET](#)

IBM MQ Internet Pass-Thru 構成リファレンス

IBM MQ Internet Pass-Thru (MQIPT) では、mqipt.conf という構成ファイルを使用して経路を定義し、MQIPT サーバーの動作を制御します。IBM MQ 9.2 以降では、mqiptAdmin コマンドの構成プロパティを、プロパティ・ファイルで指定することもできます。

MQIPT 構成ファイル

MQIPT 構成ファイルは、複数のセクションで構成されます。[global] セクションが1つと、定義されている MQIPT を通る経路ごとに追加の [route] セクションがあります。

各セクションには、名前/値のプロパティ・ペアが含まれます。プロパティには、[global] セクションのみに出現可能なもの、[route] セクションのみに出現可能なもの、また [route] と [global] の両方のセクションに出現可能なものがあります。プロパティが route セクションと [global] セクションの両方に表示される場合、[route] セクションのプロパティの値はグローバル値をオーバーライドしますが、これは該当する経路についてのみです。このようにして、[global] セクションを使用して、個々の [route] セクションに設定されていないプロパティに使用するデフォルト値を設定できます。

[global] セクションは、[global] 文字を含む行で開始し、最初の [route] セクションの開始で終了します。[global] セクションは、ファイル内のすべての [route] セクションに先行する必要があります。

各 [route] セクションは、[route] 文字を含む行で開始し、次の [route] セクションが開始した場合、または構成ファイルの終端に達した場合に終了します。

認識されないプロパティ名はすべて無視されます。[route] セクション内のプロパティの名前が認識されても値が無効である場合 (例えば、MinConnectionThreads=x または HTTP=unsure など)、その経路は使用不可になります (つまり、着信接続を何も listen しません)。



重要: mqipt.conf ファイルに追加できる経路の数の最大制限は 100 です。

[global] セクションのプロパティに無効な値があると、MQIPT、またはコマンド・サーバーが起動しない可能性があります。コマンド・サーバーが始動しない場合、MQIPT は、影響を受けるコマンド・ポートに **mqiptAdmin** コマンドが送信する管理コマンドを listen しません。MQIPT のリフレッシュ時に [global] セクションに無効な値を持つプロパティが存在する場合、警告メッセージが発行され、そのプロパティの有効な値は変更されません。したがって、無効なプロパティ値のために、MQIPT のアクティブなインスタンスがリフレッシュ時にシャットダウンせずに済みます。

プロパティが true または false の値を取るものとしてリストされている場合は、プロパティの値に大文字と小文字を任意に混ぜて使用できます。

プロパティ値の変更は、mqipt.conf ファイルを編集して行うことができます。変更を適用するには、**-refresh** キーワードを指定した **mqiptAdmin** コマンドを使用して、MQIPT をリフレッシュします。

構成ファイルにコメントを含めるには、「#」文字で行を開始します。

他のプロパティが既に使用可能である場合のみ、特定のプロパティに対する変更により経路の再始動が生じます。例えば、HTTP プロパティに対して何らかの変更が行われた場合、その HTTP プロパティが使用可能である場合のみ、変更が有効になります。

経路が再始動すると、既存の接続は終了します。この動作を指定変更するには、**RouteRestart** プロパティを false に設定します。これにより、経路は再始動しなくなり、**RouteRestart** プロパティが再び使用可能になるまで既存の接続をアクティブな状態に維持することができます。

単純な構成のセットアップ方法については、[MQIPT の概要](#)を参照してください。サンプル構成については、MQIPT インストール・ディレクトリーにある **mqiptSample.conf** ファイルを参照してください。

mqiptAdmin プロパティ・ファイル

mqiptAdmin コマンドの構成プロパティは、別個のプロパティ・ファイルで指定することができます。これらの構成プロパティは、**mqiptAdmin** が MQIPT TLS コマンド・ポートに接続する時に必要になります。

mqiptAdmin プロパティ・ファイルで指定可能なプロパティのリストに関しては、[263 ページの『mqiptAdmin プロパティ』](#)を参照してください。プロパティ名では大/小文字が区別されます。認識されないプロパティは無視されます。


プロパティ・ファイルにコメントを含めるには、行頭に「#」文字を付けてコメントを指定します。

MQIPT プロパティの要約

この表は、MQIPT 構成プロパティの要約を示し、以下の情報が含まれます。

- MQIPT プロパティのアルファベット順リストおよび [route] セクション内または [global] セクション内 ([route] セクションが適用されない場合) の詳細情報へのリンク。
- 値が有効になるために true に設定する必要があるプロパティ。
- プロパティが [global] セクション、[route] セクション、または両方に適用されるかどうか。
- プロパティが [route] セクションと [global] セクションのどちらにも存在しない場合に使用されるデフォルト値。値 true および false を指定する場合、大文字と小文字は任意に混在させて使用することができます。



プロパティの名前	true に設定するプロパティ	グローバル	ルート	デフォルト
AccessPW		yes	no	null
アクティブ		yes	yes	true
V 9.4.0 V 9.4.0 244 ページの『[MQ 9.4.0 2024 年 6 月][MQ 9.4.0 2024 年 6 月]AllowedProtocols』		yes	yes	mq
ClientAccess		yes	yes	false
CommandPort		yes	no	null
CommandPortListenerAddress		yes	no	null
ConnectionLog		yes	no	true
Destination		no	yes	null
DestinationPort		no	yes	1414
241 ページの『EnableAdvancedCapabilities』		yes	no	false
HTTP		yes	yes	false
V 9.4.0 V 9.4.0 244 ページの『[MQ 9.4.0 2024 年 6 月][MQ 9.4.0 2024 年 6 月]HTTPConnectionTimeout』		yes	yes	5000
HTTPProxy	HTTP	yes	yes	null
HTTPProxyPort	HTTP	yes	yes	8080
HTTPS	HTTP	yes	yes	false
HTTPServer	HTTP	yes	yes	null
HTTPServerPort	HTTP	yes	yes	null
IdleTimeout		yes	yes	0
IgnoreExpiredCRLs		yes	yes	false
エルダップ		yes	yes	false
LDAPIgnoreErrors	LDAP	yes	yes	false
LDAPCacheTimeout	LDAP	yes	yes	24

プロパティの名前	true に設定する プロパティ	グローバル	ルート	デフォルト
<u>LDAPServer1</u>	LDAP	yes	yes	null
<u>LDAPServer1Port</u>	LDAP	yes	yes	389
<u>LDAPServer1Userid</u>	LDAP	yes	yes	null
<u>LDAPServer1Password</u>	LDAP	yes	yes	null
<u>LDAPServer1Timeout</u>	LDAP	yes	yes	0
<u>LDAPServer2</u>	LDAP	yes	yes	null
<u>LDAPServer2Port</u>	LDAP	yes	yes	389
<u>LDAPServer2Userid</u>	LDAP	yes	yes	null
<u>LDAPServer2Password</u>	LDAP	yes	yes	null
<u>LDAPServer2Timeout</u>	LDAP	yes	yes	0
<u>ListenerAddress</u>		yes	yes	null
<u>ListenerPort</u>		no	yes	null
<u>LocalAddress</u>		yes	yes	null
<u>LocalAdmin</u>		yes	no	true
<u>MaxConnectionThreads</u>		yes	yes	100
<u>MaxLogFileSize</u>		yes	no	50
<u>MinConnectionThreads</u>		yes	yes	5
名前		no	yes	null
<u>OutgoingPort</u>		no	yes	0
 <u>PasswordProtection</u>		yes	yes	REQUIRED
<u>QMGrAccess</u>		yes	yes	true
<u>RemoteCommandAuthentication</u>		yes	no	無し
<u>RemoteShutdown</u>		yes	no	false
<u>RouteRestart</u>		yes	yes	true
<u>SecurityExit</u>		yes	yes	false
<u>SecurityExitName</u>	SecurityExit	yes	yes	null
<u>SecurityExitPath</u>	SecurityExit	yes	yes	<i>mqi</i> pt_home \\exits
<u>SecurityExitTimeout</u>	SecurityExit	yes	yes	30
<u>SecurityManager (注 3)</u>		yes	no	false
<u>SecurityManager ポリシー (注 3)</u>		yes	no	null
<u>SocksClient</u>		yes	yes	false
<u>SocksProxyHost</u>	SocksClient	yes	yes	null
<u>SocksProxyPort</u>	SocksClient	yes	yes	1080
<u>SocksServer</u>		yes	yes	false

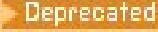
プロパティの名前	true に設定する プロパティ	グローバル	ルート	デフォルト
SSLClient		yes	yes	false
SSLClientCAKeyRing	SSLClient	yes	yes	null
SSLClientCAKeyRingPW	SSLClient	yes	yes	null
250 ページの『SSLClientCAKeyRingUseCryptoHardware』	SSLClient	yes	yes	false
SSLClientCipherSuites	SSLClient	yes	yes	null
SSLClientConnectTimeout	SSLClient	yes	yes	30
SSLClientCustomOutboundSNI	SSLClient	yes	yes	null
SSLClientDN_C	SSLClient	yes	yes	* (注 1)
SSLClientDN_CN	SSLClient	yes	yes	* (注 1)
SSLClientDN_DC	SSLClient	yes	yes	* (注 1)
SSLClientDN_DNQ	SSLClient	yes	yes	* (注 1)
SSLClientDN_L	SSLClient	yes	yes	* (注 1)
SSLClientDN_O	SSLClient	yes	yes	* (注 1)
SSLClientDN_OU	SSLClient	yes	yes	* (注 1)
SSLClientDN_PC	SSLClient	yes	yes	* (注 1)
SSLClientDN_ST	SSLClient	yes	yes	* (注 1)
SSLClientDN_Street	SSLClient	yes	yes	* (注 1)
SSLClientDN_T	SSLClient	yes	yes	* (注 1)
SSLClientDN_UID	SSLClient	yes	yes	* (注 1)
SSLClientExit		yes	yes	false
SSLClientKeyRing	SSLClient	yes	yes	null
SSLClientKeyRingPW	SSLClient	yes	yes	null
253 ページの『SSLClientKeyRingUseCryptoHardware』	SSLClient	yes	yes	false
253 ページの『SSLClientOutboundSNI』	SSLClient	yes	yes	hostname
SSLClientProtocols	SSLClient	yes	yes	TLSv1.2 TLSv1.3
SSLClientSiteDN_C	SSLClient	yes	yes	* (注 1)
SSLClientSiteDN_CN	SSLClient	yes	yes	* (注 1)
SSLClientSiteDN_DC	SSLClient	yes	yes	* (注 1)
SSLClientSiteDN_DNQ	SSLClient	yes	yes	* (注 1)
SSLClientSiteDN_L	SSLClient	yes	yes	* (注 1)

プロパティの名前	true に設定する プロパティ	グローバル	ルート	デフォルト
SSLClientSiteDN_O	SSLClient	yes	yes	* (注 1)
SSLClientSiteDN_OU	SSLClient	yes	yes	* (注 1)
SSLClientSiteDN_PC	SSLClient	yes	yes	* (注 1)
SSLClientSiteDN_ST	SSLClient	yes	yes	* (注 1)
SSLClientSiteDN_Street	SSLClient	yes	yes	* (注 1)
SSLClientSiteDN_T	SSLClient	yes	yes	* (注 1)
SSLClientSiteDN_UID	SSLClient	yes	yes	* (注 1)
SSLClientSiteLabel	SSLClient	yes	yes	null
SSLCommandPort		yes	no	null
SSLCommandPortCipherSuites		yes	no	null
SSLCommandPortListenerAddress		yes	no	null
SSLCommandPortKeyRing		yes	no	null
SSLCommandPortKeyRingPW		yes	no	null
SSLCommandPortKeyRingUseCryptoHardware		yes	no	false
SSLCommandPortProtocols		yes	no	TLSv1.2 TLSv1.3
SSLCommandPortSiteLabel		yes	no	null
SSLExitData	SSLServerExit	yes	yes	null
SSLExitName	SSLServerExit	yes	yes	null
SSLExitPath	SSLServerExit	yes	yes	<i>mqi</i> pt_home \ exits
SSLExitTimeout	SSLServerExit	yes	yes	30
SSLProxyMode		yes	yes	false
SSLPlainConnections	SSLServer または SSLProxyMode	yes	yes	false
SSLServer		yes	yes	false
SSLServerAskClientAuth	SSLServer	yes	yes	false
SSLServerCAKeyRing	SSLServer	yes	yes	null
SSLServerCAKeyRingPW	SSLServer	yes	yes	null
257 ページの『SSLServerCAKeyRingUseCryptoHardware』	SSLServer	yes	yes	false
SSLServerCipherSuites	SSLServer	yes	yes	null
SSLServerDN_C	SSLServer	yes	yes	* (注 1)
SSLServerDN_CN	SSLServer	yes	yes	* (注 1)

プロパティの名前	true に設定する プロパティ	グローバル	ルート	デフォルト
SSLServerDN_DC	SSLServer	yes	yes	* (注 1)
SSLServerDN_DNQ	SSLServer	yes	yes	* (注 1)
SSLServerDN_L	SSLServer	yes	yes	* (注 1)
SSLServerDN_O	SSLServer	yes	yes	* (注 1)
SSLServerDN_OU	SSLServer	yes	yes	* (注 1)
SSLServerDN_PC	SSLServer	yes	yes	* (注 1)
SSLServerDN_ST	SSLServer	yes	yes	* (注 1)
SSLServerDN_Street	SSLServer	yes	yes	* (注 1)
SSLServerDN_T	SSLServer	yes	yes	* (注 1)
SSLServerDN_UID	SSLServer	yes	yes	* (注 1)
SSLServerExit		yes	yes	false
SSLServerKeyRing	SSLServer	yes	yes	null
SSLServerKeyRingPW	SSLServer	yes	yes	null
260 ページの『SSLServerKeyRingUseCryptoHardware』	SSLServer	yes	yes	false
SSLServerProtocols	SSLServer	yes	yes	TLSv1.2 TLSv1.3
SSLServerSiteDN_C	SSLServer	yes	yes	* (注 1)
SSLServerSiteDN_CN	SSLServer	yes	yes	* (注 1)
SSLServerSiteDN_DC	SSLServer	yes	yes	* (注 1)
SSLServerSiteDN_DNQ	SSLServer	yes	yes	* (注 1)
SSLServerSiteDN_L	SSLServer	yes	yes	* (注 1)
SSLServerSiteDN_O	SSLServer	yes	yes	* (注 1)
SSLServerSiteDN_OU	SSLServer	yes	yes	* (注 1)
SSLServerSiteDN_PC	SSLServer	yes	yes	* (注 1)
SSLServerSiteDN_ST	SSLServer	yes	yes	* (注 1)
SSLServerSiteDN_Street	SSLServer	yes	yes	* (注 1)
SSLServerSiteDN_T	SSLServer	yes	yes	* (注 1)
SSLServerSiteDN_UID	SSLServer	yes	yes	* (注 1)
SSLServerSiteLabel	SSLServer	yes	yes	null
StoredCredentialsFormat		yes	yes	null
TCPKeepAlive		yes	yes	false
Trace		yes	yes	0

プロパティの名前	true に設定する プロパティ	グローバル	ルート	デフォルト
 traceFileCount		yes	no	25
 traceFileSize		yes	no	200
263 ページの『TraceUser データ』		yes	yes	64
UriName	HTTP	yes	yes	(注 2)

注:

1. アスタリスク (*) はワイルドカードを表します。
2. デフォルト設定の詳細については、[243 ページの『MQIPT 経路のプロパティ』](#)の UriName を参照してください。
3.  このプロパティは、将来のリリースで削除される予定です。

関連資料

[233 ページの『IBM MQ Internet Pass-Thru 構成リファレンス』](#)

IBM MQ Internet Pass-Thru (MQIPT) では、mqipt.conf という構成ファイルを使用して経路を定義し、MQIPT サーバーの動作を制御します。IBM MQ 9.2 以降では、mqiptAdmin コマンドの構成プロパティを、プロパティ・ファイルで指定することもできます。

[240 ページの『MQIPT グローバル・プロパティ』](#)

mqipt.conf 構成ファイルには、複数のグローバル・プロパティを含めることができます。

[243 ページの『MQIPT 経路のプロパティ』](#)

mqipt.conf 構成ファイルには、個々の経路のプロパティを含めることができます。

MQIPT グローバル・プロパティ

mqipt.conf 構成ファイルには、複数のグローバル・プロパティを含めることができます。

以下のプロパティは、mqipt.conf の [global] セクションにのみ表示されます。ListenerPort、Destination、DestinationPort、Name、および OutgoingPort を除くすべての経路プロパティが [global] セクションに表示されることもあります。プロパティが route セクションと [global] セクションの両方に表示される場合、[route] セクションのプロパティの値はグローバル値をオーバーライドしますが、これは該当する経路についてのみです。このようにして、[global] セクションを使用して、個々の [route] セクションに設定されていないプロパティに使用するデフォルト値を設定できます。

AccessPW

mqiptAdmin コマンドを使用して MQIPT コマンド・ポートに送信されるコマンドを認証するために使用されるパスワード。

この値は、mqiptPW コマンドを使用して暗号化されたパスワードか、平文パスワードにできます。平文パスワードに含めることができるのは、英数字のみです。MQIPT 構成に格納されるパスワードを暗号化することをぜひお勧めします。MQIPT 構成のパスワードの暗号化方法については、[保管されるパスワードの暗号化](#)を参照してください。

次の両方の条件に当てはまる場合、コマンド・ポートで受信された管理コマンドに対して認証が実行されます。

- **AccessPW** プロパティが指定されており、ブランクではない値に設定されている。
- **RemoteCommandAuthentication** プロパティが指定されており、none 以外の値に設定されている。

CommandPort

非セキュアのコマンド・ポートの TCP/IP ポート番号。MQIPT は、mqiptAdmin コマンドがこのコマンド・ポートに送信した管理コマンドを受け入れます。

非セキュアのコマンド・ポートへの接続は TLS で保護されません。コマンド・ポートに送信されたデータ (アクセス・パスワードを含む) に、ネットワークの他のユーザーがアクセスする可能性があります。TLS で保護されたコマンド・ポートを構成するには、代わりに **SSLCommandPort** プロパティを設定します。

CommandPort プロパティが未指定の場合、MQIPT 非セキュアのコマンド・ポートで管理コマンドを listen しません。**mqiptAdmin** コマンドでデフォルトで使用されるデフォルトのポート番号 1881 を使用するには、**CommandPort** を 1881 に設定します。

CommandPortListenerAddress

非セキュアのコマンド・ポートが使用するローカル・リスナー・アドレス。ローカル・リスナー・アドレスを設定することにより、非セキュアのコマンド・ポートへのインバウンド接続を、特定のネットワーク・インターフェースからの接続に制限できます。デフォルトでは、すべてのネットワーク・インターフェースで listen します。

ConnectionLog

true または false のいずれか。true の場合、MQIPT は、すべての接続試行 (成功かそうでないかによらず) を logs サブディレクトリーに、また、切断イベントをファイル **mqiptYYYYMMDDHHmmSS.log** (YYYYMMDDHHmmSS は現在の日付と時刻を表す文字) にログ記録します。**ConnectionLog** のデフォルト値は true です。このプロパティが true から false に変更されると、MQIPT は既存の接続ログを閉じて、新しい接続ログを作成します。新しいログは、このプロパティが true に再設定されたときに使用されます。

EnableAdvancedCapabilities

このプロパティを true に設定すると、IBM MQ Advanced、IBM MQ Advanced for z/OS、IBM MQ Advanced for z/OS VUE、または IBM MQ Appliance の資格を必要とする拡張機能を MQIPT で使用できるようになります。適切なライセンスがある場合は、MQIPT の拡張機能を使用できます。経路で拡張機能が有効になっている場合、MQIPT 経路を使用して接続されているローカル・キュー・マネージャーにも、IBM MQ Advanced、IBM MQ Advanced for z/OS、IBM MQ Advanced for z/OS VUE、または IBM MQ Appliance のライセンスが必要です。拡張機能を使用する経路は、このプロパティが true に設定されていないと開始できません。このプロパティを true から false に変更すると、拡張機能を使用する経路は停止します。

LocalAdmin

コマンド・ポートを使用しないローカル管理を許可するかどうかを指定します。このプロパティが false に設定されている場合、コマンド・ポートの代わりにローカル管理を使用して **mqiptAdmin** コマンドにより送信された管理コマンドは受け入れられません。

このプロパティの有効値は、true および false です。デフォルト値は true です。

MaxLogFileSize

接続ログ・ファイルの最大サイズ (KB 単位で指定)。ファイル・サイズがこの最大値を超えて増大すると、バックアップ・コピー (**mqipt001.log**) が作成され、新規ファイルが開始されます。2 つのバックアップ・ファイル (**mqipt001.log** と **mqipt002.log**) のみが保持されます。メイン・ログ・ファイルがいっぱいになるごとに、古いバックアップが消去されます。**MaxLogFileSize** のデフォルト値は 50 です。許容最小値は 5 です。

RemoteCommandAuthentication

非セキュアのコマンド・ポートまたは TLS コマンド・ポートで受信した管理コマンドを認証するかどうかを指定します。コマンドの認証は、提供されたパスワードが **AccessPW** プロパティで指定されたパスワードと一致するかをチェックすることによって行われます。値は、次の値のうちのいずれかです。

なし

どちらのコマンド・ポートに発行されたコマンドに対しても、認証は実行されません。

mqiptAdmin コマンドのユーザーは、パスワードを入力する必要がありません。これがデフォルト値です。

オプション

mqiptAdmin コマンドのユーザーは、パスワードを提供する必要がありません。ただし、パスワードを提供すると、そのパスワードが有効になります。

required

mqiptAdmin コマンドのユーザーは、コマンド・ポートへのコマンドの発行ごとに、有効なパスワードを提供する必要があります。

コマンド・ポートの認証を有効にするには、**AccessPW** プロパティーも指定する必要があります。

RemoteShutDown

mqiptAdmin コマンドによって非セキュア・コマンド・ポートまたは TLS コマンド・ポートに送信された停止コマンドによって MQIPT をシャットダウンできるかどうかを指定します。どちらでもこれらのコマンド・ポートで受信された停止コマンドを処理するためには、このプロパティーが **true** に設定されている必要があります。

このプロパティーの有効値は、**true** および **false** です。デフォルト値は **false** です。


SecurityManager

MQIPT のこのインスタンスに対して Java security manager を有効にするには、このプロパティーを **true** に設定します。正しい許可が付与されていることを確認する必要があります。詳しくは、[Java security manager](#) を参照してください。このプロパティーのデフォルト値は **false** です。

このプロパティーは、将来のリリースで削除される予定です。

SecurityManagerPolicy

Java security manager ポリシー・ファイルの完全修飾ファイル名です。このプロパティーが設定されていない場合、デフォルトのシステムおよびユーザー・ポリシー・ファイルのみが使用されます。Java security manager が既に使用可能になっている場合、このプロパティーに対する変更は Java security manager が使用不可にされて再び使用可能にされるまで有効になりません。

 **Deprecated** このプロパティーは、将来のリリースで削除される予定です。

SSLCommandPort

TLS コマンド・ポートの TCP/IP ポート番号。MQIPT は、**mqiptAdmin** コマンドがこのコマンド・ポートに送信した管理コマンドを受け入れます。このポートは TLS 接続のみ受け入れます。TLS コマンド・ポートを有効にするためには、このプロパティーを指定する必要があります。

SSLCommandPortCipherSuites

TLS コマンド・ポートで有効にする暗号スイートの名前。値をコンマで区切ることによって、複数の暗号スイートを指定することができます。MQIPT で提供されている Java runtime environment (JRE) でデフォルトで有効になっている TLS 1.2 および TLS 1.3 暗号スイートのみを指定できます。このプロパティーが指定されていない場合は、JRE で有効になっているすべての暗号スイートが TLS コマンド・ポートで有効になります。

SSLCommandPortListenerAddress

TLS コマンド・ポートが使用するローカル・リスナー・アドレス。ローカル・リスナー・アドレスを設定することにより、TLS コマンド・ポートへのインバウンド接続を、特定のネットワーク・インターフェースからの接続に制限できます。デフォルトでは、すべてのネットワーク・インターフェースで **listen** します。

SSLCommandPortKeyRing

TLS コマンド・ポート・サーバー証明書が含まれる PKCS#12 鍵リング・ファイルの名前。

Windows プラットフォームでは、ファイル分離文字として二重円記号 (¥¥) を使用する必要があります。

SSLCommandPortKeyRingPW

TLS コマンド・ポート鍵リング・ファイルまたは PKCS #11 鍵ストアにアクセスするための暗号化されたパスワード。パスワードは、**mqiptPW** コマンドを使用して暗号化する必要があります。したがってこのプロパティーの値は、**mqiptPW** によって出力された文字列に設定されている必要があります。

SSLCommandPortKeyRingUseCryptoHardware

PKCS #11 インターフェースをサポートする暗号化ハードウェアを TLS コマンド・サーバー証明書の鍵ストアとして使用するかどうかを指定します。このプロパティーの有効値は、**true** および **false** です。このプロパティーが **true** に設定されている場合、**SSLCommandPortKeyRing** を同時に指定することはできません。

MQIPT での暗号化ハードウェアの使用は、IBM MQ Advanced の機能です。IBM MQ Advanced 資格があることを確認するには、EnableAdvancedCapabilities プロパティを true に設定する必要があります。

SSLCommandPortProtocols

TLS コマンド・ポートで有効にされるプロトコルのコマンド区切りリスト。以下の 1 つ以上の値を指定できます。

表 117. コマンド・ポート TLS プロトコルに許可される値	
値	プロトコル
TLSv1.2	TLS 1.2
TLSv1.3	TLS 1.3

このプロパティを指定しない場合、デフォルトで TLS 1.2 および TLS 1.3 が有効になります。

SSLCommandPortSiteLabel

TLS コマンド・ポートが使用するサーバー証明書のラベル名。このプロパティが指定されていない場合は、暗号スイートと互換性のある TLS コマンド・ポート鍵ストアの任意の証明書が選択されます。

トレース

経路に関連付けられていないグローバル MQIPT スレッド、および **Trace** プロパティが設定されていない経路のトレース・レベル。例えば、メインの MQIPT 制御スレッドおよびコマンド・サーバー・スレッドは経路に関連付けられておらず、[global] セクションでトレースが有効になっている場合にのみトレースされます。[route] セクション内の **Trace** プロパティの値によって、該当のグローバル **Trace** プロパティが指定変更されます。経路に関連付けられているトレース・スレッドの詳細については、[route] セクションの **Trace** を参照してください。

このプロパティの値は、以下のいずれかです。

0

トレースが有効になっていません

任意の正整数

トレースは有効です

デフォルト値は 0 です。

V 9.4.0 traceFileCount

トレース・データを書き込むために MQIPT によって使用される一連のローテーション・ファイル内のトレース・ファイルの数。

許可される最小値は 3 です。デフォルト値は 25 です。

このプロパティの値を変更すると、現在のトレース・ファイルが閉じ、トレース・ファイルのローテーション・セット内の次のファイルが開きます。

V 9.4.0 traceFileSize

MQIPT によって生成されるトレース・ファイルの最大サイズ (MB 単位で指定)。

許可される最小値は 1 です。デフォルト値は 200 です。

このプロパティの値を変更すると、現在のトレース・ファイルが閉じ、トレース・ファイルのローテーション・セット内の次のファイルが開きます。

MQIPT 経路のプロパティ

mqipt.conf 構成ファイルには、個々の経路のプロパティを含めることができます。

mqipt.conf 構成ファイルの [route] セクションには、以下のプロパティーを含めることができます。

アクティブ

経路で着信接続が受け入れられるのは、**Active** の値が **true** に設定されている場合のみですつまり、この値を **false** に設定すれば、構成ファイルから [route] セクションを削除しなくても宛先へのアクセスを一時的にシャットオフできます。このプロパティーを **false** に変更すると、リフレッシュ・コマンドの発行時に経路が停止します。経路への接続はすべて停止します。

V9.4.0 V9.4.0 AllowedProtocols

この経路で受け入れられるプロトコルを指定します。このリストに指定されていないプロトコルを使用する接続は拒否されます。複数のプロトコルをコンマ区切りリストとして指定できます。以下のプロトコルを指定できます。

mq

経路は、IBM MQ プロトコルを使用する接続を受け入れます。

http

この経路は、MQIPT の別のインスタンスからの HTTP 接続を受け入れます。

このプロパティーのデフォルト値は **mq** です。

このプロパティーの値が変更された場合、リフレッシュ・コマンドが発行されると、経路は停止され、再始動されます。経路への接続はすべて停止します。

ClientAccess

経路で着信クライアント・チャンネル接続が許可されるのは、**ClientAccess** の値が **true** に設定されている場合のみです。クライアント要求のみ、キュー・マネージャー要求のみ、または両方のタイプの要求を受け入れるように MQIPT を構成できることに注意してください。このプロパティーは、**QMGrAccess** プロパティーと組み合わせて使用します。このプロパティーを **false** に変更すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

Destination

この経路の接続先キュー・マネージャー、または後続の MQIPT インスタンスのホスト名 (またはドット 10 進 IP アドレス)。各 [route] セクションには明示的な **Destination** 値が含まれている必要がありますが、複数の [route] セクションが同じ宛先を指すことができます。このプロパティーの変更が経路に影響する場合、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。**SocksProxyHost** プロパティーを使用する場合、**Destination** プロパティーでドット 10 進 IPv4 アドレス・フォーマットを使用する必要があります。

DestinationPort

この経路の接続先の宛先ホスト上のポート。各 [route] セクションには明示的な **DestinationPort** 値が含まれている必要がありますが、複数の経路が **Destination** 値と **DestinationPort** 値の同じ組み合わせを指すことができます。このプロパティーの変更が経路に影響する場合、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

HTTP

アウトバウンド HTTP トンネリング要求を行う経路の場合は、**HTTP** を **true** に設定します。経路の **Destination** プロパティーは、**HTTP true** に設定されている場合、MQIPT の別のインスタンスのホスト名でなければなりません。IBM MQ キュー・マネージャーに接続されている経路の場合は、**HTTP** を **false** に設定します。このプロパティーを変更すると、経路は停止します。**HTTP** が **true** に設定されている場合は、少なくとも 1 つの **HTTPProxy** プロパティーまたは **HTTPServer** プロパティーも指定する必要があります。このプロパティーを **SocksClient** プロパティーと組み合わせて使用することはできません。

V9.4.0 V9.4.0 HTTPConnectionTimeout

このプロパティーは、受信した HTTP 接続が正常に確立されるのを MQIPT が待機する時間 (ミリ秒) を指定します。この時間が経過すると、接続は拒否されます。

デフォルト値は 5000 です。

HTTPProxy

この経路のすべての接続で使用される HTTP プロキシのホスト名 (またはドット 10 進 IP アドレス)。HTTP プロキシが構成されていない場合、通常使用される **POST** 要求ではなく **CONNECT** 要求が HTTP プロキシに発行されます。このプロパティを変更 (および **HTTP** を **true** に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

HTTPProxyPort

HTTP プロキシで使用するポート・アドレス。デフォルト値は **8080** です。このプロパティを変更 (および **HTTP** を **true** に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

HTTPServer

この経路のすべての接続で使用される HTTP サーバーのホスト名 (またはドット 10 進 IP アドレス)。通常、これは別の MQIPT のホスト名です。

HTTPProxy が指定されていない場合、MQIPT は **HTTPServer** で指定されたホストに接続し、経路の **Destination** プロパティで指定されたホストに HTTP **POST** 要求を発行します。**HTTPProxy** が指定された場合、MQIPT は代わりに **HTTPProxy** で指定されたホストに接続し、プロキシが **HTTPServer** で指定されたホストへのトンネルを確立することを要求します。

HTTPProxy が指定されている場合、デフォルト値は経路 **Destination** です。

このプロパティを変更 (および **HTTP** を **true** に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

HTTPS

HTTPS 要求を行うには、**HTTPS** を **true** に設定します。SSL/TLS 操作の場合と同様に、**HTTP** プロパティと **SSLClient** プロパティ、および **SSLClientKeyRing** プロパティまたは **SSLClientKeyRingUseCryptoHardware** プロパティを使用して構成されたクライアント鍵リングも使用可能にする必要があります。**HTTPS** プロパティを変更 (および **HTTP** を **true** に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

HTTPServerPort

HTTP サーバーで使用するポート・アドレス。デフォルト値は **8080** です。ただし、**HTTPProxy** を指定した場合のデフォルト値は経路の **DestinationPort** です。

このプロパティを変更 (および **HTTP** を **true** に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

IdleTimeout

アイドル接続を閉じるまでの時間 (分単位)。キュー・マネージャー間チャンネルには、**DISCONT** プロパティも使用することに注意してください。**IdleTimeout** パラメーターを設定した場合は、**DISCONT** をメモしておいてください。**IdleTimeout** が **0** に設定されている場合、アイドル・タイムアウトはありません。このプロパティの変更が有効になるのは、経路の再始動時のみです。

IgnoreExpiredCRLs

IgnoreExpiredCRLs を **true** に設定すると、有効期限切れの CRL が無視されます。デフォルト値は **false** です。**IgnoreExpiredCRLs** を **true** に設定すると、取り消された証明書が SSL/TLS 接続で使用される可能性があります。

LDAP

LDAP を **true** に設定すると、SSL/TLS 接続使用時に LDAP サーバーを使用できます。MQIPT では LDAP サーバーを使用して、CRL と ARL を取得します。このプロパティを有効にするには、**SSLClient** プロパティまたは **SSLServer** プロパティも **true** に設定する必要があります。

LDAPCacheTimeout

LDAP サーバーから取得された CRL が保管される一時キャッシュの有効期限時刻 (時間単位)。この時刻を過ぎると、CRL キャッシュ全体が空になります。例えば、値を 1 時間に指定すると、キャッシュは 1 時間に 1 回空になります。デフォルト値は **24** です。タイムアウト値を **0** に指定すると、経路が再始動するまでキャッシュ内の項目の有効期限は切れません。プロパティを変更 (および **LDAP** を **true** に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

LDAPIgnoreErrors

LDAPIgnoreErrors を true に設定すると、LDAP 検索の実行時に接続またはタイムアウトのエラーが無視されます。MQIPT で検索を正常に実行できない場合は、このプロパティーが使用可能でない限り、クライアント接続を完了できません。正常な検索は、CRL が取得されたか、指定された CA で使用可能な CRL がないことを意味します。プロパティーを変更(および **LDAP** を true に設定)すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

注: このプロパティーを使用可能にすると、取り消された証明書が SSL/TLS 接続で使用される可能性があります。

LDAPServer1

メイン LDAP サーバーのホスト名または IP アドレス。LDAP が true に設定されている場合は、このプロパティーを設定する必要があります。プロパティーを変更(および **LDAP** を true に設定)すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

LDAPServer1Port

メイン LDAP サーバーの listen ポート番号。デフォルト値は 389 です。プロパティーを変更(および **LDAP** を true に設定)すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

LDAPServer1Userid

メイン LDAP サーバーへのアクセスに必要なユーザー ID。メイン LDAP サーバーへのアクセス許可が必要な場合は、このプロパティーを設定する必要があります。プロパティーを変更(および **LDAP** を true に設定)すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

LDAPServer1Password

メイン LDAP サーバーへのアクセスに必要なパスワード。**LDAPServer1Userid** が true に設定されている場合は、このプロパティーを設定する必要があります。プロパティーを変更(および **LDAP** を true に設定)すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

この値は、**mqiptPW** コマンドを使用して暗号化されたパスワードか、平文パスワードにできます。平文パスワードに含めることができるのは、英数字のみです。MQIPT 構成に格納されるパスワードを暗号化することをぜひお勧めします。MQIPT 構成のパスワードの暗号化方法については、[保管されるパスワードの暗号化](#)を参照してください。

LDAPServer1Timeout

MQIPT がメイン LDAP サーバーからの応答を待機する時間(秒単位)。デフォルト値は 0 です。これは、接続がタイムアウトにならないことを意味します。プロパティーを変更(および **LDAP** を true に設定)すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

LDAPServer2

バックアップ LDAP サーバーのホスト名または IP アドレス。このプロパティーはオプションです。プロパティーを変更(および **LDAP** を true に設定)すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

LDAPServer2Port

バックアップ LDAP サーバーの listen ポート番号。デフォルト値は 389 です。プロパティーを変更(および **LDAP** を true に設定)すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

LDAPServer2Userid

バックアップ LDAP サーバーへのアクセスに必要なユーザー ID。バックアップ LDAP サーバーへのアクセス許可が必要な場合は、このプロパティーを設定する必要があります。プロパティーを変更(および **LDAP** を true に設定)すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

LDAPServer2Password

バックアップ LDAP サーバーへのアクセスに必要なパスワード。**LDAPServer2** が true に設定されている場合は、このプロパティーを設定する必要があります。プロパティーを変更(および **LDAP** を

true に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

この値は、**mqiptPW** コマンドを使用して暗号化されたパスワードか、平文パスワードにできます。平文パスワードに含めることができるのは、英数字のみです。MQIPT 構成に格納されるパスワードを暗号化することをぜひお勧めします。MQIPT 構成のパスワードの暗号化方法については、[保管されるパスワードの暗号化](#)を参照してください。

LDAPServer2Timeout

MQIPT がバックアップ LDAP サーバーからの応答を待機する時間 (秒単位)。デフォルト値は 0 です。これは、接続がタイムアウトにならないことを意味します。プロパティを変更 (および **LDAP** を true に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

ListenerAddress

MQIPT システムに複数の IP アドレスがあり、経路のリスナー・ポートを特定のアドレスにバインドする必要がある場合は、このプロパティを使用します。これは、インバウンド接続を特定のネットワーク・インターフェースからの接続に制限する場合に役立ちます。このプロパティの値は、MQIPT が実行されているシステム上のネットワーク・インターフェースのいずれかに属している IP アドレスでなければなりません。デフォルトでは、すべてのネットワーク・インターフェースからの接続が受け入れられます。

ListenerPort

経路が着信要求を listen する必要があるポート番号。各 [route] セクションには、明示的な **ListenerPort** 値を含める必要があります。各セクションに設定する **ListenerPort** 値は異ならなければなりません。ポート 80 や 443 など、任意の有効なポート番号を使用できます。ただし、選択したポートが、同じホスト上で実行されている他の TCP/IP リスナーでまだ使用されていない場合に限りです。

LocalAddress

このコンピューター上のこの経路用の、すべての接続をバインドするための IP アドレス。選択したアドレスは、MQIPT が実行されているコンピューター上のネットワーク・インターフェースのいずれかに関連付けられている IP アドレスでなければなりません。このプロパティを変更すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

MaxConnectionThreads

この経路で処理できる接続スレッドの最大数 (つまり、同時接続の最大数)。この限度に達した場合、**MaxConnectionThreads** 値で、すべてのスレッドが使用されているときにキューに入れられる接続の数も示されます。その数を超えると、後続の接続要求は拒否されます。

最小許容値は 1 と **MinConnectionThreads** の値のどちらか大きいほうになります。

値を大きくすると、リフレッシュ・コマンドの発行時に新しい値が使用されます。すべての接続で新しい値が即時に使用されます。経路は停止しません。

値を小さくすると、新しい値は経路の再始動時のみ有効になります。

MinConnectionThreads

経路の開始時に経路で着信接続を処理するために割り振られる接続スレッドの数。経路がアクティブになっている間は、割り振られたスレッドの数がこの値を下回ることはありません。

値は、0 から **MaxConnectionThreads** の値までの範囲でなければなりません。

このプロパティの変更が有効になるのは、経路の再始動時のみです。

名前

経路を識別しやすい名前。このプロパティはオプションです。値は、コンソール・メッセージとトレース情報で示されます。このプロパティの変更が有効になるのは、経路の再始動時のみです。

OutgoingPort

発信接続で使用する開始ポート番号。ポート番号の範囲は、この経路の **MaxConnectionThread** 値と一致します。デフォルト値 0 の場合は、システム定義のポート番号が使用されます。このプロパティを変更すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。この経路への接続はすべて停止します。HTTP が使用されている場合は、各チャンネル接続で 2 つの発信ポートが必要になります。詳しくは、[ポート番号の制御](#)を参照してください。

V 9.4.0 PasswordProtection

TLS 暗号化を追加または削除するように構成されている MQIPT 経路について、クライアントとキュー・マネージャーの間の互換性を維持するために、IBM MQ clients によって MQCSP 構造で送信される資格情報の保護を MQIPT が追加または削除できるかどうかを指定します。

MQCSP 構造の資格情報は、IBM MQ MQCSP パスワード保護機能を使用して保護することも、TLS 暗号化を使用して暗号化することもできます。MQCSP パスワード保護は、TLS 暗号化をセットアップするよりも簡単ですが、それほどセキュアではないため、テストおよび開発の目的で役立ちます。

MQCSP パスワード保護について詳しくは、[MQCSP パスワード保護](#)を参照してください。

TLS 暗号化を追加または削除するように MQIPT 経路が構成されている場合、MQIPT は、接続を成功させるために、MQCSP 構造内の資格情報を保護するか、MQCSP パスワード保護を削除する必要がある場合があります。

このプロパティの値には、以下のいずれかの値を指定できます。

required

MQIPT は、MQCSP 構造内の資格情報が TLS を使用して暗号化されているか、MQCSP パスワード保護によって保護されていることを確認します。

MQCSP 構造の資格情報が TLS 暗号化を使用してクライアントによって暗号化されて送信され、MQIPT 経路が TLS 暗号化を除去する場合、MQIPT は、資格情報を経路宛先に転送する前に、MQCSP パスワード保護を使用して資格情報を保護します。これは、MQIPT 経路が `SSLServer=true` および `SSLClient=false` で構成され、選択された CipherSuite がヌル暗号を使用しない場合に発生します。

MQCSP パスワード保護でクライアントによって MQCSP 構造の資格情報が保護されている場合、MQIPT と経路宛先との間の接続で TLS 暗号化が使用されている場合でも、MQIPT は保護を解除しません。MQIPT と経路宛先との間の接続で TLS 暗号化が使用されている場合、接続は理由コード `MQRC_PASSWORD_PROTECTION_ERROR (2594)` で失敗する可能性があります。

これがデフォルト値です。

compatible

MQIPT は、接続が正常に行われるようにするために、必要に応じて MQCSP パスワード保護を適用または除去します。

MQCSP 構造の資格情報が TLS 暗号化を使用してクライアントによって暗号化されて送信され、MQIPT 経路が TLS 暗号化を除去する場合、MQIPT は、パスワードを経路宛先に転送する前に MQCSP パスワード保護を使用して資格情報を保護します。これは、MQIPT 経路が `SSLServer=true` および `SSLClient=false` で構成され、選択された CipherSuite がヌル暗号を使用しない場合に発生します。

MQCSP 構造内の資格情報が MQCSP パスワード保護でクライアントによって保護されており、MQIPT 経路に TLS 暗号化が追加されている場合、MQIPT は、資格情報を経路宛先に転送する前に MQCSP パスワード保護を除去します。これは、MQIPT 経路が `SSLServer=false` および `SSLClient=true` で構成され、選択された CipherSuite がヌル暗号を使用しない場合に発生します。

このオプションは、最良の互換性を提供します。ただし、これはトラステッド・ネットワークでのテストおよび開発の目的にのみ使用してください。これは、パスワードがネットワーク上で保護されていることが保証されていないためです。

passthru

MQCSP 構造内の資格情報は、MQCSP パスワード保護を追加または削除することなく、MQIPT によって経路宛先に転送されます。TLS 暗号化を追加または削除するように MQIPT 経路が構成されている場合、クライアント接続が理由コード `MQRC_PASSWORD_PROTECTION_ERROR (2594)` で失敗することがあります。

QMGrAccess

QMGrAccess を `true` に設定すると、着信キュー・マネージャー・チャネル接続 (送信側チャネルなど) が許可されます。このプロパティを `false` に変更すると、リフレッシュ・コマンドの発行時に経路が停止します。この経路への接続はすべて停止します。

RouteRestart

RouteRestart を `false` に設定すると、他の経路プロパティが変更され、リフレッシュ・コマンドが発行されている場合、経路は再始動しません。このプロパティのデフォルト値は `true` です。

SecurityExit

SecurityExit を `true` に設定すると、ユーザー定義のセキュリティー出口が有効になります。このプロパティのデフォルト値は `false` です。

SecurityExitName

ユーザー定義のセキュリティー出口のクラス名。**SecurityExit** が `true` に設定されている場合は、このプロパティを設定する必要があります。このプロパティを変更 (および **SecurityExit** を `true` に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。この経路への接続はすべて停止します。

SecurityExitPath

ユーザー定義のセキュリティー出口を含む完全修飾パス名。このプロパティが設定されていない場合は、デフォルトで出口サブディレクトリーに設定されます。このプロパティで、ユーザー定義のセキュリティー出口を含む Java アーカイブ (JAR) ファイルの名前を定義することもできます。このプロパティを変更 (および **SecurityExit** を `true` に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。この経路への接続はすべて停止します。

SecurityExitTimeout

接続要求の検証時に応答を待機する時間を決定するために、MQIPT で使用されるタイムアウト値 (秒単位)。デフォルト値は 30 です。このプロパティを変更 (および **SecurityExit** を `true` に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SocksClient

SocksClient を `true` に設定すると、経路は SOCKS クライアントとして機能し、**SocksProxyHost** プロパティと **SocksProxyPort** プロパティを使用して SOCKS プロキシを介するすべての接続が定義されます。このプロパティを変更すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。このプロパティを以下と一緒に使用することはできません。

- HTTP
- SocksServer
- SSLClient
- SSLProxyMode

SocksProxyHost

この経路のすべての接続で使用される SOCKS プロキシのホスト名 (またはドット 10 進 IPv4 アドレス)。このプロパティを変更 (および **SocksClient** を `true` に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。この経路への接続はすべて停止します。

SocksProxyHost プロパティを使用する場合、**Destination** プロパティでドット 10 進形式を使用する必要があります。

SocksProxyPort

SOCKS プロキシで使用するポート番号。デフォルト値は 1080 です。このプロパティを変更 (および **SocksClient** を `true` に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SocksServer

SocksServer を `true` に設定すると、経路は SOCKS プロキシとして機能し、SOCKS クライアント接続が受け入れられます。このプロパティを変更すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。このプロパティを次のプロパティと一緒に使用することはできません。

- SocksClient
- SSLProxyMode
- SSLServer

SSLClient

SSLClient を true に設定します。に設定すると、経路は SSL/TLS クライアントとして機能し、発信 SSL/TLS 接続が行われます。**SSLClient** を true に設定すると、宛先が SSL/TLS サーバーとして機能する MQIPT の別のインスタンスか、HTTP プロキシ/サーバーのいずれかになります。

SSLClient を true に設定する場合は、**SSLClientKeyRing** プロパティまたは **SSLClientCAKeyRing** プロパティを使用して SSL/TLS クライアント鍵リングを指定するか、**SSLClientKeyRingUseCryptoHardware** プロパティまたは **SSLClientCAKeyRingUseCryptoHardware** プロパティを設定して暗号ハードウェアを使用するように MQIPT を構成する必要があります。

SSLClient を変更すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。この経路への接続はすべて停止します。

このプロパティを次のプロパティと組み合わせて使用することはできません。

- **SSLProxyMode**

SSLClientCAKeyRing

SSL/TLS サーバーからの証明書の認証に使用される、CA 証明書を含む鍵リング・ファイルの完全修飾ファイル名。Windows プラットフォームでは、ファイル分離文字として二重円記号 (¥¥) を使用する必要があります。このプロパティを変更 (および **SSLClient** を true に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLClientCAKeyRingPW

SSLClientCAKeyRing プロパティで指定した SSL/TLS クライアント CA 鍵リング・ファイルを開くためのパスワードか、**SSLClientCAKeyRingUseCryptoHardware** プロパティが true に設定されている場合に暗号ハードウェア鍵ストアに接続するためのパスワードのいずれかです。

この値は、**mqiptPW** コマンドを使用して暗号化されたパスワードか、暗号化パスワードを含むファイルの完全修飾ファイル名にできます。Windows プラットフォーム上のファイル名を指定する場合は、ファイル分離文字として二重円記号 (¥¥) を使用する必要があります。現在ファイルに格納されている鍵リング・パスワードをすべてマイグレーションして、**mqiptPW** ユーティリティを使用してパスワードを再暗号化し、最新かつ最も安全性の高い保護方式を利用されることをお勧めします。MQIPT 構成のパスワードの暗号化方法について詳しくは、[保管されるパスワードの暗号化](#)を参照してください。

このプロパティを変更 (および **SSLClient** を true に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLClientCAKeyRingUseCryptoHardware

MQIPT が SSL/TLS クライアントとして動作している場合に、SSL/TLS サーバーからのサーバー証明書の認証に使用する CA 証明書の鍵ストアとして、PKCS#11 インターフェースをサポートする暗号ハードウェアを使用するかどうかを指定します。このプロパティが true に設定されている場合、**SSLClientCAKeyRing** を同じ経路に設定することはできません。

このプロパティを変更 (および **SSLClient** を true に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

MQIPT での暗号ハードウェアの使用は、IBM MQ Advanced の機能です。この機能を使用するには、MQIPT 経路を使用して接続されているローカル・キュー・マネージャーにも、IBM MQ Advanced、IBM MQ Appliance、IBM MQ Advanced for z/OS、または IBM MQ Advanced for z/OS VUE のライセンスが必要です。このプロパティが true に設定されている場合、IBM MQ Advanced 機能を使用できることを確認するために **EnableAdvancedCapabilities** グローバル・プロパティが設定されていない限り、経路は開始されません。

SSLClientCipherSuites

SSL/TLS クライアント・サイドで使用する SSL/TLS CipherSuite の名前。サポートされている CipherSuite を 1 つ以上指定することができます。このプロパティを空白のままにすると、SSL/TLS クライアントは鍵リングのサポートされている CipherSuite を使用します。このプロパティ

ーを変更(および **SSLClient** を true に設定)すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。この経路への接続はすべて停止します。

SSLClientConnectTimeout

SSL/TLS クライアントが SSL/TLS 接続の受け入れを待機する時間(秒単位)。このプロパティを変更(および **SSLClient** を true に設定)すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLClientCustomOutboundSNI

経路が custom に設定された **SSLClientOutboundSNI** で構成されている場合に、MQIPT が経路宛先への TLS 接続を開始するときの サーバー名表示 (SNI) の値を指定します。このプロパティを使用して、MQIPT によって自動的に設定できない特定の値に SNI を設定します。例えば、SNI をホスト名に設定したいが、経路の宛先が IP アドレスを使用して構成されている場合などです。

この値は、RFC 3490 仕様に準拠した有効な国際化ドメイン名 (IDN) でなければならず、末尾をドットにすることはできません。無効な値が指定された場合、経路は開始されません。

このプロパティの値を変更(および **SSLClientOutboundSNI** を custom に設定)すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。



重要: チャンネルの **CERTLABL** フィールドに証明書ラベルが構成されている IBM MQ チャンネルに接続を転送する場合は、この設定を使用しないでください。このような方法でクライアントを転送すると、MQRC_SSL_INITIALIZATION_ERROR 戻りコードと AMQ9673 エラーがリモート・キュー・マネージャーのエラー・ログに出力されて拒否されます。

SSLClientDN_C

この国名と一致する SSL/TLS サーバーから受信される証明書を受け入れる場合は、このプロパティを使用します。名前の先頭または末尾にアスタリスク (*) を付けて有効範囲を広げることができます。証明書の突き合わせでは大/小文字は区別されません。このプロパティを指定しないと、すべての国名が受け入れられます。このプロパティを変更(および **SSLClient** を true に設定)すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLClientDN_CN

この共通名と一致する SSL/TLS サーバーから受信される証明書を受け入れる場合は、このプロパティを使用します。名前の先頭または末尾にアスタリスク (*) を付けて有効範囲を広げることができます。証明書の突き合わせでは大/小文字は区別されません。このプロパティを指定しないと、すべての共通名が受け入れられます。このプロパティを変更(および **SSLClient** を true に設定)すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLClientDN_DC

このドメイン・コンポーネントと一致する SSL/TLS サーバーから受信される証明書を受け入れる場合は、このプロパティを使用します。名前の先頭または末尾にアスタリスク (*) を付けて有効範囲を広げることができます。証明書の突き合わせでは大/小文字は区別されません。コンマで区切って、複数の DC を指定することができます。各 DC はドメイン名のエレメントを示しています。例えば、ドメイン名が example.ibm.com の場合、example,ibm,com のように、コンマを使用して複数の値を区切ります。このプロパティを指定しないと、すべてのドメイン・コンポーネントが受け入れられます。このプロパティを変更(および **SSLClient** を true に設定)すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLClientDN_DNQ

このドメイン修飾子と一致する SSL/TLS サーバーから受信される証明書を受け入れる場合は、このプロパティを使用します。名前の先頭または末尾にアスタリスク (*) を付けて有効範囲を広げることができます。証明書の突き合わせでは大/小文字は区別されません。このプロパティを指定しないと、すべてのドメイン修飾子が受け入れられます。このプロパティを変更(および **SSLClient** を true に設定)すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLClientDN_L

このロケーションと一致する SSL/TLS サーバーから受信される証明書を受け入れる場合は、このプロパティを使用します。名前の先頭または末尾にアスタリスク (*) を付けて有効範囲を広げることができます。証明書の突き合わせでは大/小文字は区別されません。このプロパティを指定しないと、「すべてのロケーション」を暗黙指定したことになります。このプロパティを変更(および

SSLClient を true に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLClientDN_O

この組織と一致する SSL/TLS サーバーから受信される証明書を受け入れる場合は、このプロパティを使用します。名前の先頭または末尾にアスタリスク (*) を付けて有効範囲を広げることができます。証明書の突き合わせでは大/小文字は区別されません。このプロパティを指定しないと、すべての組織からの証明書が受け入れられます。このプロパティを変更 (および **SSLClient** を true に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLClientDN_OU

この組織単位 (OU) と一致する SSL/TLS サーバーから受信される証明書を受け入れる場合は、このプロパティを使用します。名前の先頭または末尾にアスタリスク (*) を付けて有効範囲を広げることができます。コンマで区切って、複数の OU を指定することができます。(リテラル・コンマは前に円記号 (¥) を付けて突き合わせます。) 証明書の突き合わせでは大/小文字は区別されません。このプロパティを指定しないと、すべての OU 名の証明書が受け入れられます。このプロパティを変更 (および **SSLClient** を true に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。この経路への接続はすべて停止します。

SSLClientDN_PC

この郵便番号と一致する SSL/TLS サーバーから受信される証明書を受け入れる場合は、このプロパティを使用します。名前の先頭または末尾にアスタリスク (*) を付けて有効範囲を広げることができます。証明書の突き合わせでは大/小文字は区別されません。このプロパティを指定しないと、すべての郵便番号が受け入れられます。このプロパティを変更 (および **SSLClient** を true に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLClientDN_ST

この都道府県と一致する SSL/TLS サーバーから受信される証明書を受け入れる場合は、このプロパティを使用します。名前の先頭または末尾にアスタリスク (*) を付けて有効範囲を広げることができます。証明書の突き合わせでは大/小文字は区別されません。このプロパティを指定しないと、すべての都道府県のサーバーからの証明書が受け入れられます。このプロパティを変更 (および **SSLClient** を true に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLClientDN_Street

この番地名と一致する SSL/TLS サーバーから受信される証明書を受け入れる場合は、このプロパティを使用します。名前の先頭または末尾にアスタリスク (*) を付けて有効範囲を広げることができます。証明書の突き合わせでは大/小文字は区別されません。このプロパティを指定しないと、すべての番地名が受け入れられます。このプロパティを変更 (および **SSLClient** を true に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLClientDN_T

この役職と一致する SSL/TLS サーバーから受信される証明書を受け入れる場合は、このプロパティを使用します。名前の先頭または末尾にアスタリスク (*) を付けて有効範囲を広げることができます。証明書の突き合わせでは大/小文字は区別されません。このプロパティを指定しないと、すべての役職が受け入れられます。このプロパティを変更 (および **SSLClient** を true に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLClientDN_UID

このユーザー ID と一致する SSL/TLS サーバーから受信される証明書を受け入れる場合は、このプロパティを使用します。名前の先頭または末尾にアスタリスク (*) を付けて有効範囲を広げることができます。証明書の突き合わせでは大/小文字は区別されません。このプロパティを指定しないと、すべてのユーザー ID が受け入れられます。このプロパティを変更 (および **SSLClient** を true に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLClientExit

経路が SSL/TLS クライアントとして機能している場合に、出口の使用を使用可能または使用不可にする場合は、このプロパティを使用します。これにより、構成ファイルで実際に使用しなくても出口の詳細を定義することができます。

SSLClientKeyRing

クライアント証明書を含む鍵リング・ファイルの完全修飾ファイル名。Windows プラットフォームでは、ファイル分離文字として二重円記号 (¥¥) を使用する必要があります。

SSLClientKeyRing を変更 (および **SSLClient** を true に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLClientKeyRingPW

SSLClientKeyRing プロパティで指定した SSL/TLS クライアント鍵リング・ファイルを開くためのパスワードか、**SSLClientKeyRingUseCryptoHardware** プロパティが true に設定されている場合に暗号ハードウェア鍵ストアに接続するためのパスワードのいずれかです。

この値は、**mqiptPW** コマンドを使用して暗号化されたパスワードか、暗号化パスワードを含むファイルの完全修飾ファイル名にできます。Windows プラットフォーム上のファイル名を指定する場合は、ファイル分離文字として二重円記号 (¥¥) を使用する必要があります。現在ファイルに格納されている鍵リング・パスワードをすべてマイグレーションして、**mqiptPW** ユーティリティーを使用してパスワードを再暗号化し、最新かつ最も安全性の高い保護方式を利用されることをお勧めします。MQIPT 構成のパスワードの暗号化方法については、[保管されるパスワードの暗号化](#)を参照してください。

SSLClientKeyRingPW を変更 (および **SSLClient** を true に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLClientKeyRingUseCryptoHardware

MQIPT が SSL/TLS クライアントとして動作している場合に、クライアント証明書を含む鍵ストアとして、PKCS#11 インターフェースをサポートする暗号ハードウェアを使用するかどうかを指定します。このプロパティが true に設定されている場合、**SSLClientKeyRing** を同じ経路に設定することはできません。

このプロパティを変更 (および **SSLClient** を true に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

MQIPT での暗号ハードウェアの使用は、IBM MQ Advanced の機能です。この機能を使用するには、MQIPT 経路を使用して接続されているローカル・キュー・マネージャーにも、IBM MQ Advanced、IBM MQ Appliance、IBM MQ Advanced for z/OS、または IBM MQ Advanced for z/OS VUE のライセンスが必要です。このプロパティが true に設定されている場合、IBM MQ Advanced 機能を使用できることを確認するために **EnableAdvancedCapabilities** グローバル・プロパティが設定されていない限り、経路は開始されません。

SSLClientOutboundSNI

MQIPT が経路宛先への TLS 接続を開始する際のサーバー名表示 (SNI) 拡張の値を指定します。SNI は、構成に応じて IBM MQ キュー・マネージャーが、TLS ハンドシェイク中に正しい証明書を提示するために、または宛先への接続を経路指定するために使用されます。

このプロパティは、**SSLClient=true** で定義された経路にのみ適用され、**HTTP=true** で定義された経路には指定できません。このプロパティの値を変更 (および **SSLClient** を true に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。



重要: 宛先チャンネルがチャンネル・オブジェクトの **CERTLABL** フィールドに証明書ラベルを指定して構成されている場合は、**CERTLABL** 設定をチャンネル値に設定する必要があります。クライアントがチャンネル SNI 設定なしで転送されると、MQRC_SSL_INITIALIZATION_ERROR 戻りコードと AMQ9673 メッセージがリモート・キュー・マネージャーのエラー・ログに出力されて拒否されます。

このプロパティの値には、以下のいずれかの値を指定できます。

ホスト名

SNI は、経路宛先のホスト名に設定されます。経路が、SNI を使用して要求を経路指定するロード・バランサーまたはルーターに接続する場合、このオプションを使用します。例えば Red Hat® OpenShift® Container Platform Router は、SNI を使用して要求を IBM MQ キュー・マネージャーに経路指定します。

経路宛先がキュー・マネージャーの場合、接続要求は、TLS ハンドシェイク中にリモート・キュー・マネージャーのデフォルト証明書を受信するため、チャンネルごとの証明書は使用できません。

経路宛先が IP アドレスを使用して指定されていて、逆引き DNS ルックアップを実行できない場合、SNI はブランクです。

これがデフォルト値です。

channel

SNI は、IBM MQ チャンネル名に設定されます。経路によって受信された接続が、以下のいずれかの理由で SNI にチャンネル名を含まない場合、このオプションを使用して、宛先キュー・マネージャーがチャンネルごとの証明書を使用できるようにします。

- 経路は、`SSLServer=false` または `SSLPlainConnections=true` のいずれかを使用して TLS で保護されていない接続を受け入れるように構成されています。
- 経路に接続するアプリケーションは、SNI を設定できないか、SNI を IBM MQ チャンネル名以外の値に設定するように構成されます。

passthru

経路が `SSLServer=true` で定義されている場合、アウトバウンド接続の SNI は、経路へのインバウンド接続で受信された SNI の値に設定されます。ルートが TLS 接続を受け入れるように構成されていない場合、SNI は宛先ホスト名に設定されます。

custom

SNI は、`SSLClientCustomOutboundSNI` プロパティで指定された値に設定されます。`SSLClientCustomOutboundSNI` プロパティが指定されていない場合、SNI は経路が `SSLClientOutboundSNI=hostname` で構成されている場合と同様に設定されます。

なし

SNI が設定されていません。

SSLClientProtocols

`SSLClient` が `true` に設定されている場合に、経路の宛先へのアウトバウンド接続を行うために使用される一連の有効なセキュア・ソケット・プロトコルを制限する際に使用します。

コマンドで区切って、複数の値を指定することができます。このプロパティを指定しない場合、デフォルトで TLS 1.2 および TLS 1.3 が有効になります。TLS 1.2 または TLS 1.3 以外のプロトコルを使用可能にするには、このプロパティで使用するプロトコルを指定する必要があります。また、非推奨のプロトコル および `CipherSuites` の手順に従って、Java runtime environment でプロトコルのサポートを追加する必要があります。次の 1 つ以上の値を指定することができます。

値	プロトコル
SSLv3	SSL 3.0
TLSv1	TLS 1.0
TLSv1.1	TLS 1.1
TLSv1.2	TLS 1.2
TLSv1.3	TLS 1.3

経路プロパティでは「値」列にリストされている項目を使用します。「プロトコル」列の対応する項目は参照用です。

SSLClientSiteDN_C

SSL/TLS サーバーに送信する証明書を選択するために国名を指定する場合は、このプロパティを使用します。証明書の突き合わせでは大/小文字は区別されません。このプロパティを指定しないと、すべての国名の証明書が受け入れられます。このプロパティを変更 (および `SSLClient` を `true` に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLClientSiteDN_CN

SSL/TLS サーバーに送信する証明書を選択するために共通名を指定する場合は、このプロパティを使用します。証明書の突き合わせでは大/小文字は区別されません。このプロパティを指定しないと、すべての共通名の証明書が受け入れられます。このプロパティを変更 (および **SSLClient** を true に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLClientSiteDN_DC

SSL/TLS サーバーに送信する証明書を選択するためにドメイン・コンポーネント名を指定する場合は、このプロパティを使用します。証明書の突き合わせでは大/小文字は区別されません。コンマで区切って、複数の DC を指定することができます。各 DC はドメイン名のエレメントを示しています。例えば、ドメイン名が `example.ibm.com` の場合、`example,ibm,com` のように、コンマを使用して複数の値を区切ります。このプロパティを指定しないと、すべてのドメイン・コンポーネント名の証明書が受け入れられます。このプロパティを変更 (および **SSLClient** を true に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLClientSiteDN_DNQ

SSL/TLS サーバーに送信する証明書を選択するためにドメイン修飾子を指定する場合は、このプロパティを使用します。証明書の突き合わせでは大/小文字は区別されません。このプロパティを指定しないと、すべてのドメイン修飾子の証明書が受け入れられます。このプロパティを変更 (および **SSLClient** を true に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLClientSiteDN_L

SSL/TLS サーバーに送信する証明書を選択するためにロケーション名を指定する場合は、このプロパティを使用します。証明書の突き合わせでは大/小文字は区別されません。このプロパティを指定しないと、すべてのロケーション名の証明書が受け入れられます。このプロパティを変更 (および **SSLClient** を true に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLClientSiteDN_O

SSL/TLS サーバーに送信する証明書を選択するために組織名を指定する場合は、このプロパティを使用します。証明書の突き合わせでは大/小文字は区別されません。このプロパティを指定しないと、すべての組織名の証明書が受け入れられます。このプロパティを変更 (および **SSLClient** を true に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLClientSiteDN_OU

SSL/TLS サーバーに送信する証明書を選択するために組織単位 (OU) 名を指定する場合は、このプロパティを使用します。コンマで区切って、複数の OU を指定することができます。(リテラル・コンマは前に円記号 (¥) を付けて突き合わせます。) 証明書の突き合わせでは大/小文字は区別されません。このプロパティを指定しないと、すべての OU 名の証明書が受け入れられます。このプロパティを変更 (および **SSLClient** を true に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。この経路への接続はすべて停止します。

SSLClientSiteDN_PC

SSL/TLS サーバーに送信する証明書を選択するために郵便番号を指定する場合は、このプロパティを使用します。証明書の突き合わせでは大/小文字は区別されません。このプロパティを指定しないと、すべての郵便番号の証明書が受け入れられます。このプロパティを変更 (および **SSLClient** を true に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLClientSiteDN_ST

SSL/TLS サーバーに送信する証明書を選択するために都道府県名を指定する場合は、このプロパティを使用します。証明書の突き合わせでは大/小文字は区別されません。このプロパティを指定しないと、すべての都道府県名の証明書が受け入れられます。このプロパティを変更 (および **SSLClient** を true に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLClientSiteDN_Street

SSL/TLS サーバーに送信する証明書を選択するために番地名を指定する場合は、このプロパティを使用します。証明書の突き合わせでは大/小文字は区別されません。このプロパティを指定しないと、

すべての番地名の証明書が受け入れられます。このプロパティを変更(および **SSLClient** を true に設定)すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLClientSiteDN_T

SSL/TLS サーバーに送信する証明書を選択するためにタイトルを指定する場合は、このプロパティを使用します。証明書の突き合わせでは大/小文字は区別されません。このプロパティを指定しないと、すべてのタイトルの証明書が受け入れられます。このプロパティを変更(および **SSLClient** を true に設定)すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLClientSiteDN_UID

SSL/TLS サーバーに送信する証明書を選択するためにユーザー ID を指定する場合は、このプロパティを使用します。証明書の突き合わせでは大/小文字は区別されません。このプロパティを指定しないと、すべてのユーザー ID の証明書が受け入れられます。このプロパティを変更(および **SSLClient** を true に設定)すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLClientSiteLabel

SSL/TLS サーバーに送信する証明書を選択するためにラベル名を指定する場合は、このプロパティを使用します。このプロパティを指定しないと、すべてのラベル名の証明書が受け入れられます。このプロパティを変更(および **SSLClient** を true に設定)すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLExitData

出口に渡されるユーザー定義のストリングを指定する場合は、このプロパティを使用します。

SSLExitName

経路が SSL/TLS クライアントまたは SSL/TLS サーバーとして機能している場合に、呼び出される出口のクラス名を定義する場合は、このプロパティを使用します。名前にはパッケージ名を含める必要があります(例えば、com.ibm.mq.ippt.exit.TestExit)。

SSLExitPath

出口のコピーをロードするために使用される出口のロケーションを定義する場合は、このプロパティを使用します。名前は、クラス・ファイルを見つけるために使用される完全修飾名か、クラス・ファイルを含む .jar ファイルの名前でなければなりません。(例えば、C:\mqipt\exits または C:\mqipt\exits\exits.jar)。

SSLExitTimeout

接続要求を終了する前に出口が完了するまで MQIPT が待機する時間を定義する場合は、このプロパティを使用します。値 0 は、MQIPT が無期限に待機することを意味します。

SSLPlainConnections

インバウンド SSL/TLS 接続を受け入れるように構成された経路の MQIPT リスナー・ポートへの接続で SSL/TLS を必須にするかどうかを指定する場合は、このプロパティを使用します。このプロパティは、**SSLServer** プロパティまたは **SSLProxyMode** プロパティのいずれかが true に設定されている経路に適用できます。このプロパティを使用可能にすると、非暗号化接続で経路リスナー・ポートに接続できます。これは、接続が暗号化されているかどうかに関係なく、MQIPT がすべての IBM MQ 接続をキュー・マネージャーのリスナー・ポートに転送できることを意味します。このパラメーターを設定しない場合や、false に設定した場合は、インバウンド SSL/TLS 接続のみが許可されます。このプロパティを変更すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLProxyMode

このプロパティを true に設定すると、経路で SSL/TLS クライアント接続要求のみが受け入れられ、要求は宛先に直接トンネリングされます。このプロパティを変更すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。この経路への接続はすべて停止します。このプロパティを次のプロパティと組み合わせて使用することはできません。

- **SocksClient**
- **SocksServer**
- **SSLClient**

- **SSLServer**

SSLServer

このプロパティを **true** に設定すると、経路は SSL/TLS サーバーとして機能し、着信 SSL/TLS 接続が受け入れられます。 **SSLServer** を **true** に設定すると、呼び出し元は SSL/TLS クライアントとして機能する別の MQIPT、または SSL/TLS が有効になっている IBM MQ クライアントまたはキュー・マネージャーになります。

SSLServer を **true** に設定した場合は、 **SSLServerKeyRing** プロパティを使用して SSL/TLS サーバー鍵リングを指定するか、 **SSLServerKeyRingUseCryptoHardware** プロパティを設定して暗号ハードウェアを使用するように MQIPT を構成する必要があります。

このプロパティを変更すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

このプロパティを次のプロパティと組み合わせて使用することはできません。

- **SocksServer**

- **SSLProxyMode**

SSLServerCAKeyRing

SSL/TLS クライアントからの証明書の認証に使用される、CA 証明書を含む鍵リング・ファイルの完全修飾ファイル名。Windows プラットフォームでは、ファイル分離文字として二重円記号 (¥¥) を使用する必要があります。このプロパティを変更 (および **SSLServer** を **true** に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。この経路への接続はすべて停止します。

SSLServerCAKeyRingPW

SSLServerCAKeyRing プロパティで指定した SSL/TLS サーバー CA 鍵リング・ファイルを開くためのパスワードか、 **SSLServerCAKeyRingUseCryptoHardware** プロパティが **true** に設定されている場合に暗号ハードウェア鍵ストアに接続するためのパスワードのいずれかです。

この値は、 **mqiptPW** コマンドを使用して暗号化されたパスワードか、暗号化パスワードを含むファイルの完全修飾ファイル名にできます。Windows プラットフォーム上のファイル名を指定する場合は、ファイル分離文字として二重円記号 (¥¥) を使用する必要があります。現在ファイルに格納されている鍵リング・パスワードをすべてマイグレーションして、 **mqiptPW** ユーティリティを使用してパスワードを再暗号化し、最新かつ最も安全性の高い保護方式を利用されることをお勧めします。MQIPT 構成のパスワードの暗号化方法については、 [保管されるパスワードの暗号化](#) を参照してください。

このプロパティを変更 (および **SSLServer** を **true** に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLServerCAKeyRingUseCryptoHardware

SSL/TLS クライアントからの証明書を認証するために、PKCS#11 インターフェースをサポートする暗号ハードウェアを CA 証明書の鍵ストアとして使用するかどうかを指定します。このプロパティが **true** に設定されている場合、 **SSLServerCAKeyRing** を同じ経路に設定することはできません。

このプロパティを変更 (および **SSLServer** を **true** に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

MQIPT での暗号ハードウェアの使用は、IBM MQ Advanced の機能です。この機能を使用するには、MQIPT 経路を使用して接続されているローカル・キュー・マネージャーにも、IBM MQ Advanced、IBM MQ Appliance、IBM MQ Advanced for z/OS、または IBM MQ Advanced for z/OS VUE のライセンスが必要です。このプロパティが **true** に設定されている場合、IBM MQ Advanced 機能を使用できることを確認するために **EnableAdvancedCapabilities** グローバル・プロパティが設定されていない限り、経路は開始されません。

SSLServerAskClientAuth

SSL/TLS サーバーによる SSL/TLS クライアント認証を要求するには、このプロパティを使用します。SSL/TLS クライアントは、SSL/TLS サーバーに送信する独自の証明書を持っていない限りなりません。

この証明書は鍵リング・ファイルから取得されます。このプロパティを変更(および **SSLServer** を **true** に設定)すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。この経路への接続はすべて停止します。

SSLServerCipherSuites

SSL/TLS サーバー・サイドで使用する SSL/TLS CipherSuite の名前。サポートされている CipherSuite を 1 つ以上指定することができます。これをブランクのままにすると、鍵リング内のサーバー証明書と互換性のある有効なプロトコルの CipherSuite を使用します。このプロパティを変更(および **SSLServer** を **true** に設定)すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。この経路への接続はすべて停止します。

SSLServerDN_C

この国名の SSL/TLS クライアントから受信される証明書を受け入れる場合は、このプロパティを使用します。名前の先頭または末尾にアスタリスク (*) を付けて有効範囲を広げることができます。証明書の突き合わせでは大/小文字は区別されません。このプロパティを指定しないと、すべての会社名の証明書が受け入れられます。このプロパティを変更(および **SSLServer** を **true** に設定)すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLServerDN_CN

この共通名の SSL/TLS クライアントから受信される証明書を受け入れる場合は、このプロパティを使用します。名前の先頭または末尾にアスタリスク (*) を付けて有効範囲を広げることができます。証明書の突き合わせでは大/小文字は区別されません。このプロパティを指定しないと、すべての共通名の証明書が受け入れられます。このプロパティを変更(および **SSLServer** を **true** に設定)すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLServerDN_DC

このドメイン・コンポーネント名の SSL/TLS クライアントから受信される証明書を受け入れる場合は、このプロパティを使用します。名前の先頭または末尾にアスタリスク (*) を付けて有効範囲を広げることができます。証明書の突き合わせでは大/小文字は区別されません。コンマで区切って、複数の DC を指定することができます。各 DC はドメイン名のエレメントを示しています。例えば、ドメイン名が `example.ibm.com` の場合、`example,ibm,com` のように、コンマを使用して複数の値を区切ります。このプロパティを指定しないと、すべてのドメイン・コンポーネント名の証明書が受け入れられます。このプロパティを変更(および **SSLServer** を **true** に設定)すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLServerDN_DNQ

このドメイン修飾子の SSL/TLS クライアントから受信される証明書を受け入れる場合は、このプロパティを使用します。名前の先頭または末尾にアスタリスク (*) を付けて有効範囲を広げることができます。証明書の突き合わせでは大/小文字は区別されません。このプロパティを指定しないと、すべてのドメイン修飾子の証明書が受け入れられます。このプロパティを変更(および **SSLServer** を **true** に設定)すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLServerDN_L

このロケーションの SSL/TLS クライアントから受信される証明書を受け入れる場合は、このプロパティを使用します。名前の先頭または末尾にアスタリスク (*) を付けて有効範囲を広げることができます。証明書の突き合わせでは大/小文字は区別されません。このプロパティを指定しないと、すべてのロケーションの証明書が受け入れられます。このプロパティを変更(および **SSLServer** を **true** に設定)すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLServerDN_O

この組織の SSL/TLS クライアントから受信される証明書を受け入れる場合は、このプロパティを使用します。名前の先頭または末尾にアスタリスク (*) を付けて有効範囲を広げることができます。証明書の突き合わせでは大/小文字は区別されません。このプロパティを指定しないと、すべての組織の証明書が受け入れられます。このプロパティを変更(および **SSLServer** を **true** に設定)すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLServerDN_OU

この組織単位 (OU) の SSL/TLS クライアントから受信される証明書を受け入れる場合は、このプロパティを使用します。名前の先頭または末尾にアスタリスク (*) を付けて有効範囲を広げることができます。コンマで区切って、複数の OU を指定することができます。(リテラル・コンマは前に円記号 (¥) を付けて突き合わせます。) 証明書の突き合わせでは大/小文字は区別されません。このプロパティを指定しないと、すべての OU 名の証明書が受け入れられます。このプロパティを変更 (および **SSLServer** を true に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。この経路への接続はすべて停止します。

SSLServerDN_PC

この郵便番号の SSL/TLS クライアントから受信される証明書を受け入れる場合は、このプロパティを使用します。名前の先頭または末尾にアスタリスク (*) を付けて有効範囲を広げることができます。証明書の突き合わせでは大/小文字は区別されません。このプロパティを指定しないと、すべての郵便番号の証明書が受け入れられます。このプロパティを変更 (および **SSLServer** を true に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLServerDN_ST

この都道府県の SSL/TLS クライアントから受信される証明書を受け入れる場合は、このプロパティを使用します。名前の先頭または末尾にアスタリスク (*) を付けて有効範囲を広げることができます。証明書の突き合わせでは大/小文字は区別されません。このプロパティを指定しないと、すべての都道府県の証明書が受け入れられます。このプロパティを変更 (および **SSLServer** を true に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLServerDN_Street

この番地名の SSL/TLS クライアントから受信される証明書を受け入れる場合は、このプロパティを使用します。名前の先頭または末尾にアスタリスク (*) を付けて有効範囲を広げることができます。証明書の突き合わせでは大/小文字は区別されません。このプロパティを指定しないと、すべての番地名の証明書が受け入れられます。このプロパティを変更 (および **SSLServer** を true に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLServerDN_T

このタイトルの SSL/TLS クライアントから受信される証明書を受け入れる場合は、このプロパティを使用します。名前の先頭または末尾にアスタリスク (*) を付けて有効範囲を広げることができます。証明書の突き合わせでは大/小文字は区別されません。このプロパティを指定しないと、すべてのタイトルの証明書が受け入れられます。このプロパティを変更 (および **SSLServer** を true に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLServerDN_UID

このユーザー ID の SSL/TLS クライアントから受信される証明書を受け入れる場合は、このプロパティを使用します。名前の先頭または末尾にアスタリスク (*) を付けて有効範囲を広げることができます。証明書の突き合わせでは大/小文字は区別されません。このプロパティを指定しないと、すべてのユーザー ID の証明書が受け入れられます。このプロパティを変更 (および **SSLServer** を true に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLServerExit

経路が SSL/TLS サーバーとして機能している場合に、出口の使用を使用可能または使用不可にする場合は、このプロパティを使用します。これにより、構成ファイルで実際に使用しなくても出口の詳細を定義することができます。

SSLServerKeyRing

サーバー証明書を含む鍵リング・ファイルの完全修飾ファイル名。Windows プラットフォームでは、ファイル分離文字として二重円記号 (¥¥) を使用する必要があります。このプロパティを変更 (および **SSLServer** を true に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLServerKeyRingPW

SSLServerKeyRing プロパティで指定した SSL/TLS サーバー鍵リング・ファイルを開くためのパスワードか、**SSLServerKeyRingUseCryptoHardware** プロパティが true に設定されている場合に暗号ハードウェア鍵ストアに接続するためのパスワードのいずれかです。

この値は、**mqiPTPW** コマンドを使用して暗号化されたパスワードか、暗号化パスワードを含むファイルの完全修飾ファイル名にできます。Windows プラットフォーム上のファイル名を指定する場合は、ファイル分離文字として二重円記号 (¥¥) を使用する必要があります。現在ファイルに格納されている鍵リング・パスワードをすべてマイグレーションして、**mqiPTPW** ユーティリティーを使用してパスワードを再暗号化し、最新かつ最も安全性の高い保護方式を利用されることをお勧めします。MQIPT 構成のパスワードの暗号化方法については、[保管されるパスワードの暗号化](#)を参照してください。

SSLServer を true に設定する場合は、**SSLServerKeyRingPW** を指定する必要があります。

このプロパティを変更(および **SSLServer** を true に設定)すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLServerKeyRingUseCryptoHardware

MQIPT が SSL/TLS サーバーとして動作している場合に、PKCS#11 インターフェースをサポートする暗号ハードウェアをサーバー証明書の鍵ストアとして使用するかどうかを指定します。このプロパティが true に設定されている場合、**SSLServerKeyRing** を同じ経路に設定することはできません。

このプロパティを変更(および **SSLServer** を true に設定)すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

MQIPT での暗号ハードウェアの使用は、IBM MQ Advanced の機能です。この機能を使用するには、MQIPT 経路を使用して接続されているローカル・キュー・マネージャーにも、IBM MQ Advanced、IBM MQ Appliance、IBM MQ Advanced for z/OS、または IBM MQ Advanced for z/OS VUE のライセンスが必要です。このプロパティが true に設定されている場合、IBM MQ Advanced 機能を使用できることを確認するために **EnableAdvancedCapabilities** グローバル・プロパティが設定されていない限り、経路は開始されません。

SSLServerProtocols

SSLServer が true に設定されている場合に、経路のリスナー・ポートへのインバウンド接続を受け入れるために使用される一連の有効なセキュア・ソケット・プロトコルを制限する際に使用します。

コマンドで区切って、複数の値を指定することができます。このプロパティを指定しない場合、デフォルトで TLS 1.2 および TLS 1.3 が有効になります。TLS 1.2 または TLS 1.3 以外のプロトコルを使用可能にするには、このプロパティで使用可能にするプロトコルを指定する必要があります。また、[非推奨のプロトコル](#) および [CipherSuites](#) の手順に従って、Java runtime environment でプロトコルのサポートを追加する必要があります。次の 1 つ以上の値を指定することができます。

値	プロトコル
SSLv3	SSL 3.0
TLSv1	TLS 1.0
TLSv1.1	TLS 1.1
TLSv1.2	TLS 1.2
TLSv1.3	TLS 1.3

経路プロパティでは「値」列にリストされている項目を使用します。「プロトコル」列の対応する項目は参照用です。

SSLServerSiteDN_C

SSL/TLS クライアントに送信する証明書を選択するために国名を指定する場合は、このプロパティを使用します。証明書の突き合わせでは大/小文字は区別されません。このプロパティを指定しない

と、すべての国名の証明書が受け入れられます。このプロパティを変更(および **SSLServer** を true に設定)すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLServerSiteDN_CN

SSL/TLS クライアントに送信する証明書を選択するために共通名を指定する場合は、このプロパティを使用します。証明書の突き合わせでは大/小文字は区別されません。このプロパティを指定しないと、すべての共通名の証明書が受け入れられます。このプロパティを変更(および **SSLServer** を true に設定)すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLServerSiteDN_DC

SSL/TLS クライアントに送信する証明書を選択するためにドメイン・コンポーネント名を指定する場合は、このプロパティを使用します。証明書の突き合わせでは大/小文字は区別されません。コマンドで区切って、複数の DC を指定することができます。各 DC はドメイン名のエレメントを示しています。例えば、ドメイン名が `example.ibm.com` の場合、`example,ibm,com` のように、コンマを使用して複数の値を区切ります。このプロパティを指定しないと、すべてのドメイン・コンポーネント名の証明書が受け入れられます。このプロパティを変更(および **SSLServer** を true に設定)すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLServerSiteDN_DNQ

SSL/TLS クライアントに送信する証明書を選択するためにドメイン修飾子を指定する場合は、このプロパティを使用します。証明書の突き合わせでは大/小文字は区別されません。このプロパティを指定しないと、すべてのドメイン修飾子の証明書が受け入れられます。このプロパティを変更(および **SSLServer** を true に設定)すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLServerSiteDN_L

SSL/TLS クライアントに送信する証明書を選択するためにロケーション名を指定する場合は、このプロパティを使用します。証明書の突き合わせでは大/小文字は区別されません。このプロパティを指定しないと、すべてのロケーション名の証明書が受け入れられます。このプロパティを変更(および **SSLServer** を true に設定)すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLServerSiteDN_O

SSL/TLS クライアントに送信する証明書を選択するために組織名を指定する場合は、このプロパティを使用します。証明書の突き合わせでは大/小文字は区別されません。このプロパティを指定しないと、すべての組織名の証明書が受け入れられます。このプロパティを変更(および **SSLServer** を true に設定)すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLServerSiteDN_OU

SSL/TLS クライアントに送信する証明書を選択するために組織単位 (OU) 名を指定する場合は、このプロパティを使用します。コマンドで区切って、複数の OU を指定することができます。(リテラル・コンマは前に円記号 (¥) を付けて突き合わせます。)証明書の突き合わせでは大/小文字は区別されません。このプロパティを指定しないと、すべての OU 名の証明書が受け入れられます。このプロパティを変更(および **SSLServer** を true に設定)すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。この経路への接続はすべて停止します。

SSLServerSiteDN_PC

SSL/TLS クライアントに送信する証明書を選択するために郵便番号を指定する場合は、このプロパティを使用します。証明書の突き合わせでは大/小文字は区別されません。このプロパティを指定しないと、すべての郵便番号の証明書が受け入れられます。このプロパティを変更(および **SSLServer** を true に設定)すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLServerSiteDN_ST

SSL/TLS クライアントに送信する証明書を選択するために都道府県名を指定する場合は、このプロパティを使用します。証明書の突き合わせでは大/小文字は区別されません。このプロパティを指定しないと、すべての都道府県名の証明書が受け入れられます。このプロパティを変更(および **SSLServer** を true に設定)すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLServerSiteDN_Street

SSL/TLS クライアントに送信する証明書を選択するために番地名を指定する場合は、このプロパティを使用します。証明書の突き合わせでは大/小文字は区別されません。このプロパティを指定しないと、すべての番地名の証明書が受け入れられます。このプロパティを変更(および **SSLServer** を true に設定)すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLServerSiteDN_T

SSL/TLS クライアントに送信する証明書を選択するためにタイトルを指定する場合は、このプロパティを使用します。証明書の突き合わせでは大/小文字は区別されません。このプロパティを指定しないと、すべてのタイトルの証明書が受け入れられます。このプロパティを変更(および **SSLServer** を true に設定)すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLServerSiteDN_UID

SSL/TLS クライアントに送信する証明書を選択するためにユーザー ID を指定する場合は、このプロパティを使用します。証明書の突き合わせでは大/小文字は区別されません。このプロパティを指定しないと、すべてのユーザー ID の証明書が受け入れられます。このプロパティを変更(および **SSLServer** を true に設定)すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

SSLServerSiteLabel

SSL/TLS クライアントに送信する証明書を選択するためにラベル名を指定する場合は、このプロパティを使用します。このプロパティを指定しないと、すべてのラベル名の証明書が受け入れられます。このプロパティを変更(および **SSLServer** を true に設定)すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。経路への接続はすべて停止します。

StoredCredentialsFormat

このプロパティを使用して、パスワード・プロパティの値が、IBM MQ 9.1.5 の MQIPT でサポートされている暗号化パスワード形式を使用するかどうかを指定します。MQIPT では、パスワードが暗号化パスワード形式で指定されているかどうかをほぼすべての場合に検出できます。このプロパティを設定する必要があるのは、MQIPT で暗号化パスワードと平文パスワードまたはファイル名を自動的に区別できないという稀にしか生じない状況においてのみです。

値は、次の値のうちのいずれかです。

encrypted

パスワード・プロパティには、IBM MQ 9.1.5 の MQIPT でサポートされている形式の暗号化されたパスワードが含まれています。

compat

パスワード・プロパティには、平文パスワードか、または鍵リング・パスワードの場合には暗号化パスワードが含まれるファイル名のいずれかが含まれます。

TCPKeepAlive

このプロパティを true に設定すると、TCP/IP キープアライブ・パケットを定期的送信して、この経路で接続がアイドル状態にならないようにすることができます。これにより、ファイアウォールまたはルーターによる MQIPT 接続の切断の可能性が低くなります。TCP/IP キープアライブ・パケットの送信は、オペレーティング・システムのチューニング・パラメーターで制御されます。キープアライブのチューニング方法について詳しくは、オペレーティング・システムの文書を参照してください。このパラメーターを設定しない場合や、false に設定した場合、キープアライブ・パケットは送信されません。

トレース

この経路に必要なトレースのレベル。1つの経路に対してトレースを使用可能にしても、他の経路のトレースは使用可能になりません。複数の経路をトレースする必要がある場合は、トレース対象の各経路の [route] セクションに **Trace** プロパティを追加する必要があります。

このプロパティの値は、以下のいずれかです。

0

トレースが有効になっていません

任意の正整数

トレースは有効です

デフォルト値は 0 です。

[route] セクションに **Trace** プロパティが含まれていない場合は、[global] セクションの **Trace** プロパティが使用されます。経路に関連付けられていないスレッドのトレースについては、[global] セクションの **Trace** を参照してください。このプロパティの変更が経路に影響する場合、リフレッシュ・コマンドの発行時に新しい値が使用されます。すべての接続で新しい値が即時に使用されます。経路は停止しません。

TraceUser データ

この経路でトレースが使用可能になっている場合に、トレースされる、この経路によって送受信されたネットワーク伝送のユーザー・データの量。値は、次の値のうちのいずれかです。

0

ユーザー・データはトレースされません。

すべて

すべてのユーザー・データがトレースされます。

numberOf バイト

指定されたバイト数のデータ (伝送セグメント・ヘッダー (TSH) を含む) がトレースされます。15 より大きい値を指定する必要があります。

UriName

このプロパティを使用すれば、HTTP プロキシの使用時にリソースの Uniform Resource Identifier の名前を変更することができます。ただし、ほとんどの構成には以下のデフォルト値で十分です。

```
HTTP://destination:destination_port/mqipt
```

このプロパティを変更 (および **HTTP** を **true** に設定) すると、リフレッシュ・コマンドの発行時に経路が停止し、再始動します。

mqiptAdmin プロパティ

mqiptAdmin コマンドは、このコマンドの始動時に指定されたプロパティ・ファイルから構成プロパティを読み取ります。

mqiptAdmin コマンドによって使用されるプロパティ・ファイルには、以下のプロパティを指定できます。プロパティ名では大/小文字が区別されます。

PasswordProtectionKeyFile

SSLClientCAKeyRingPW プロパティで指定されたトラストストアのパスワードを暗号化するために使用される暗号鍵を含むファイルの名前。このプロパティを指定しない場合、パスワードを復号するためにデフォルトの暗号鍵が使用されます。**mqiptAdmin** トラストストアのパスワードの暗号化に使用する暗号鍵を、mqipt.conf 構成ファイル内のパスワードの暗号化に使用する暗号鍵とは異なるものにすることができます。

SSLClientCAKeyRing

MQIPT TLS コマンド・ポートへの接続に使用する PKCS#12 トラストストアのファイル名。トラストストアには、MQIPT TLS コマンド・ポートが使用するサーバー証明書として構成されているサーバー証明書に署名した CA の CA 証明書が含まれている必要があります。ファイル名に含まれる円記号 (¥) はエスケープして、二重の円記号 (¥¥) として指定する必要があります。

SSLClientCAKeyRingPW

SSLClientCAKeyRing プロパティを使用して指定されたトラストストアにアクセスするための暗号化されたパスワード。パスワードは **mqiptPW** コマンドによって暗号化されている必要があります。したがって、このプロパティの値は **mqiptPW** によって出力された文字列に設定されている必要があります。

特記事項

本書は米国 IBM が提供する製品およびサービスについて作成したものです。

IBM 本書に記載の製品、サービス、または機能が日本においては提供されていない場合があります。日本で利用可能な製品、サービス、および機能については、日本 IBM の営業担当員にお尋ねください。本書で IBM 製品、プログラム、またはサービスに言及していても、その IBM 製品、プログラム、またはサービスのみが使用可能であることを意味するものではありません。これらに代えて、IBM の知的所有権を侵害することのない、機能的に同等の製品、プログラム、またはサービスを使用することができます。ただし、IBM 以外の製品とプログラムの操作またはサービスの評価および検証は、お客様の責任で行っていただきます。

IBM は、本書に記載されている内容に関して特許権 (特許出願中のものを含む) を保有している場合があります。本書の提供は、お客様にこれらの特許権について実施権を許諾することを意味するものではありません。実施権についてのお問い合わせは、書面にて下記宛先にお送りください。

〒 103-8510

東京都中央区日本橋箱崎町 19 番 21 号

日本アイ・ビー・エム株式会社

日本アイ・ビー・エム株式会社

法務・知的財産

U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing

Legal and Intellectual Property Law

〒 103-8510

19-21, Nihonbashi-Hakozakicho, Chuo-ku

Tokyo 103-8510, Japan

以下の保証は、国または地域の法律に沿わない場合は、適用されません。 INTERNATIONAL BUSINESS MACHINES CORPORATION は、法律上の瑕疵担保責任、商品性の保証、特定目的適合性の保証および法律上の瑕疵担保責任を含むすべての明示もしくは黙示の保証責任を負わないものとします。"" 国または地域によっては、法律の強行規定により、保証責任の制限が禁じられる場合、強行規定の制限を受けるものとします。

この情報には、技術的に不適切な記述や誤植を含む場合があります。本書は定期的に見直され、必要な変更は本書の次版に組み込まれます。IBM は予告なしに、随時、この文書に記載されている製品またはプログラムに対して、改良または変更を行うことがあります。

本書において IBM 以外の Web サイトに言及している場合がありますが、便宜のため記載しただけであり、決してそれらの Web サイトを推奨するものではありません。それらの Web サイトにある資料は、この IBM 製品の資料の一部ではありません。それらの Web サイトは、お客様の責任でご使用ください。

IBM は、お客様が提供するいかなる情報も、お客様に対してなんら義務も負うことのない、自ら適切と信ずる方法で、使用もしくは配布することができるものとします。

本プログラムのライセンス保持者で、(i) 独自に作成したプログラムとその他のプログラム (本プログラムを含む) との間での情報交換、および (ii) 交換された情報の相互利用を可能にすることを目的として、本プログラムに関する情報を必要とする方は、下記に連絡してください。

東京都中央区日本橋箱崎町 19 番 21 号

日本アイ・ビー・エム株式会社

Software Interoperability Coordinator, Department 49XA

3605 Highway 52 N

Rochester, MN 55901

U.S.A.

本プログラムに関する上記の情報は、適切な使用条件の下で使用することができますが、有償の場合もあります。

本書で説明されているライセンス・プログラムまたはその他のライセンス資料は、IBM 所定のプログラム契約の契約条項、IBM プログラムのご使用条件、またはそれと同等の条項に基づいて、IBM より提供されます。

この文書に含まれるいかなるパフォーマンス・データも、管理環境下で決定されたものです。そのため、他の操作環境で得られた結果は、異なる可能性があります。一部の測定が、開発レベルのシステムで行われた可能性があります。その測定値が、一般に利用可能なシステムのものと同じである保証はありません。さらに、一部の測定値が、推定値である可能性があります。実際の結果は、異なる可能性があります。お客様は、お客様の特定の環境に適したデータを確かめる必要があります。

IBM 以外の製品に関する情報は、その製品の供給者、出版物、もしくはその他の公に利用可能なソースから入手したものです。IBM は、それらの製品のテストは行っていません。したがって、他社製品に関する実行性、互換性、またはその他の要求については確認できません。IBM 以外の製品の性能に関する質問は、それらの製品の供給者をお願いします。

IBM の将来の方向または意向に関する記述については、予告なしに変更または撤回される場合があります、単に目標を示しているものです。

本書には、日常の業務処理で用いられるデータや報告書の例が含まれています。より具体性を与えるために、それらの例には、個人、企業、ブランド、あるいは製品などの名前が含まれている場合があります。これらの名称はすべて架空のものであり、名称や住所が類似する企業が実在しているとしても、それは偶然にすぎません。

著作権使用許諾:

本書には、様々なオペレーティング・プラットフォームでのプログラミング手法を例示するサンプル・アプリケーション・プログラムがソース言語で掲載されています。お客様は、サンプル・プログラムが書かれているオペレーティング・プラットフォームのアプリケーション・プログラミング・インターフェースに準拠したアプリケーション・プログラムの開発、使用、販売、配布を目的として、いかなる形式においても、IBM に対価を支払うことなくこれを複製し、改変し、配布することができます。このサンプル・プログラムは、あらゆる条件下における完全なテストを経ていません。従って IBM は、これらのサンプル・プログラムについて信頼性、利便性もしくは機能性があることをほのめかしたり、保証することはできません。

この情報をソフトコピーでご覧になっている場合は、写真やカラーの図表は表示されない場合があります。

プログラミング・インターフェース情報

プログラミング・インターフェース情報 (提供されている場合) は、このプログラムで使用するアプリケーション・ソフトウェアの作成を支援することを目的としています。

本書には、プログラムを作成するユーザーが IBM MQ のサービスを使用できるようにするためのプログラミング・インターフェースに関する情報が記載されています。

ただし、この情報には、診断、修正、および調整情報が含まれている場合があります。診断、修正、調整情報は、お客様のアプリケーション・ソフトウェアのデバッグ支援のために提供されています。

重要: この診断、修正、およびチューニング情報は、変更される可能性があるため、プログラミング・インターフェースとして使用しないでください。

商標

IBM、IBM ロゴ、ibm.com[®]は、世界の多くの国で登録された IBM Corporation の商標です。現時点での IBM の商標リストについては、"Copyright and trademark information" www.ibm.com/legal/copytrade.shtml をご覧ください。他の製品名およびサービス名等は、それぞれ IBM または各社の商標である場合があります。

Microsoft および Windows は、Microsoft Corporation の米国およびその他の国における商標です。

UNIX は The Open Group の米国およびその他の国における登録商標です。

Linux は、Linus Torvalds 氏の米国およびその他の国における登録商標です。

この製品には、Eclipse Project (<https://www.eclipse.org/>) により開発されたソフトウェアが含まれています。

Java およびすべての Java 関連の商標およびロゴは Oracle やその関連会社の米国およびその他の国における商標または登録商標です。



部品番号:

(1P) P/N: